



Developer Guide

# AWS RoboMaker



# AWS RoboMaker: Developer Guide

Copyright © 2024 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

<b>What Is AWS RoboMaker?</b> .....	<b>1</b>
Features .....	1
Pricing .....	2
<b>Getting started</b> .....	<b>3</b>
Concepts .....	3
Container .....	3
Simulation job .....	4
Simulation WorldForge .....	4
Environment .....	4
Application .....	4
Application configuration .....	5
Setting up .....	5
Sign up for an AWS account .....	5
Create a user with administrative access .....	6
Running your first simulation .....	7
<b>Development</b> .....	<b>8</b>
Building application containers .....	8
Prerequisites .....	8
Building application containers from a ROS workspace .....	9
Testing your containers .....	14
Publishing application containers .....	15
Application versioning .....	16
Working with robot applications .....	17
Creating a robot application .....	18
Creating a robot application version .....	18
Viewing a robot application .....	19
Updating a robot application .....	19
Deleting a robot application .....	20
Deleting a robot application version .....	20
Working with simulation applications .....	21
Creating a simulation application .....	21
Creating a simulation application version .....	22
Viewing a simulation application .....	23
Updating a simulation application .....	23

---

Deleting a simulation application .....	24
Deleting a simulation application version .....	24
Versioning applications .....	25
Versioning applications with images .....	26
The \$LATEST version .....	26
Updating an application version .....	27
Deleting an application version .....	27
Using images to develop applications .....	27
Migrating ROS applications to containers .....	28
ROS container FAQs .....	29
Requirements for AWS RoboMaker compatible containers .....	33
Creating images to run GPU applications .....	53
Creating images to run the hello world sample application .....	54
<b>Simulation .....</b>	<b>75</b>
Running a simulation .....	75
Configuring a simulation .....	79
Configuring a simulation job for Amazon VPC access .....	79
Internet access for simulation jobs .....	80
Configuring SimulationJob compute .....	81
Configuring custom simulation tools .....	82
Root access and system capabilities .....	82
Managing a simulation .....	84
Creating a simulation job .....	84
Viewing a simulation job .....	89
Cancelling a simulation job .....	90
Cloning a simulation job .....	91
Restarting a simulation job .....	91
Logging a simulation .....	92
Adding a custom upload configuration .....	93
Environment variables created by AWS RoboMaker .....	94
Batching simulations .....	95
Starting a simulation job batch .....	95
View a simulation job batch .....	97
Cancelling a simulation job batch .....	98
Cloning a simulation job batch .....	98
<b>Creating worlds .....</b>	<b>100</b>

Simulation WorldForge concepts .....	100
Understanding simulation world templates .....	101
Floor Plan .....	101
Interiors .....	103
Common tasks .....	105
Specifying a list of rooms for a floor .....	106
Requesting a long hallway .....	107
Requesting a doorway between rooms .....	108
Applying a configuration to all rooms .....	109
Requesting doors in doorways .....	110
Requesting no doors in doorways .....	112
Requesting a wide floor plan footprint .....	113
Requesting a custom ceiling height .....	114
Specifying the same material types for floors in different rooms .....	114
Specifying different material types for floors between rooms of the same type .....	115
Specifying more and less furniture in rooms .....	117
Adding specific furniture types to all bedrooms and a single shared living/dining room ....	118
Specifying a room without furniture .....	120
JSON schema for simulation world template body .....	121
Sample world templates in JSON .....	160
One-bedroom house .....	160
One room only .....	165
Two rooms .....	166
Managing simulation world templates .....	167
Creating a template .....	168
Viewing a template .....	186
Modifying a template .....	187
Deleting a template .....	188
Template releases .....	189
Managing world generation jobs .....	191
Creating a job .....	191
Viewing a job .....	192
Cancelling a job .....	193
Managing world export jobs .....	194
Creating an export job .....	194
Viewing an export job .....	196

Using exported worlds in simulation .....	196
Using an exported world as a data source .....	197
Using an exported world in ROS and Gazebo .....	199
Using an exported world with custom physics, lights, and models .....	201
<b>Security .....</b>	<b>202</b>
Data protection .....	202
Authentication and access control .....	203
Introduction to authorization and access control .....	204
Permissions required .....	204
How AWS RoboMaker works with IAM .....	212
Troubleshooting authentication and access control .....	213
What are policies? .....	213
AWS managed policies .....	216
Using service-linked roles .....	220
Getting started with IAM .....	224
Logging and monitoring .....	226
Monitoring AWS RoboMaker with Amazon CloudWatch .....	226
Logging calls with AWS CloudTrail .....	229
Tagging resources .....	232
Tag basics .....	232
Tag restrictions and limitations .....	233
Using tags with IAM policies .....	233
Security compliance .....	235
Resilience .....	236
Infrastructure security .....	236
VPC endpoints (AWS PrivateLink) .....	237
Considerations for AWS RoboMaker VPC endpoints .....	237
Creating an interface VPC endpoint for AWS RoboMaker .....	237
Creating a VPC endpoint policy for AWS RoboMaker .....	238
<b>API reference .....</b>	<b>239</b>
Actions .....	239
BatchDeleteWorlds .....	242
BatchDescribeSimulationJob .....	245
CancelDeploymentJob .....	251
CancelSimulationJob .....	254
CancelSimulationJobBatch .....	257

CancelWorldExportJob .....	260
CancelWorldGenerationJob .....	263
CreateDeploymentJob .....	266
CreateFleet .....	275
CreateRobot .....	279
CreateRobotApplication .....	284
CreateRobotApplicationVersion .....	291
CreateSimulationApplication .....	297
CreateSimulationApplicationVersion .....	305
CreateSimulationJob .....	311
CreateWorldExportJob .....	327
CreateWorldGenerationJob .....	334
CreateWorldTemplate .....	342
DeleteFleet .....	348
DeleteRobot .....	351
DeleteRobotApplication .....	354
DeleteSimulationApplication .....	357
DeleteWorldTemplate .....	360
DeregisterRobot .....	363
DescribeDeploymentJob .....	367
DescribeFleet .....	373
DescribeRobot .....	378
DescribeRobotApplication .....	383
DescribeSimulationApplication .....	388
DescribeSimulationJob .....	394
DescribeSimulationJobBatch .....	405
DescribeWorld .....	417
DescribeWorldExportJob .....	421
DescribeWorldGenerationJob .....	427
DescribeWorldTemplate .....	434
GetWorldTemplateBody .....	438
ListDeploymentJobs .....	441
ListFleets .....	446
ListRobotApplications .....	451
ListRobots .....	456
ListSimulationApplications .....	461

---

ListSimulationJobBatches .....	466
ListSimulationJobs .....	470
ListTagsForResource .....	474
ListWorldExportJobs .....	477
ListWorldGenerationJobs .....	481
ListWorlds .....	485
ListWorldTemplates .....	489
RegisterRobot .....	493
RestartSimulationJob .....	497
StartSimulationJobBatch .....	500
SyncDeploymentJob .....	516
TagResource .....	523
UntagResource .....	526
UpdateRobotApplication .....	529
UpdateSimulationApplication .....	535
UpdateWorldTemplate .....	542
Data Types .....	545
BatchPolicy .....	548
Compute .....	549
ComputeResponse .....	551
DataSource .....	553
DataSourceConfig .....	555
DeploymentApplicationConfig .....	558
DeploymentConfig .....	560
DeploymentJob .....	562
DeploymentLaunchConfig .....	565
Environment .....	567
FailedCreateSimulationJobRequest .....	568
FailureSummary .....	570
Filter .....	571
FinishedWorldsSummary .....	572
Fleet .....	574
LaunchConfig .....	576
LoggingConfig .....	579
NetworkInterface .....	580
OutputLocation .....	582



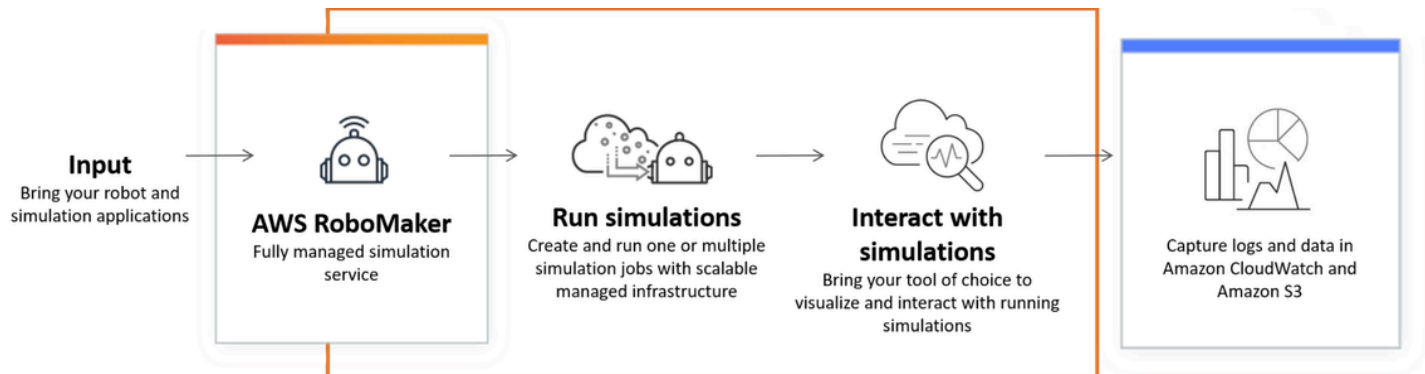
PortForwardingConfig .....	583
PortMapping .....	584
ProgressDetail .....	586
RenderingEngine .....	588
Robot .....	589
RobotApplicationConfig .....	592
RobotApplicationSummary .....	595
RobotDeployment .....	597
RobotSoftwareSuite .....	600
S3KeyOutput .....	601
S3Object .....	602
SimulationApplicationConfig .....	604
SimulationApplicationSummary .....	607
SimulationJob .....	609
SimulationJobBatchSummary .....	615
SimulationJobRequest .....	618
SimulationJobSummary .....	622
SimulationSoftwareSuite .....	625
Source .....	626
SourceConfig .....	628
TemplateLocation .....	630
TemplateSummary .....	631
Tool .....	633
UploadConfiguration .....	635
VPCConfig .....	637
VPCConfigResponse .....	639
WorldConfig .....	641
WorldCount .....	642
WorldExportJobSummary .....	643
WorldFailure .....	646
WorldGenerationJobSummary .....	648
WorldSummary .....	651
Common Errors .....	652
Common Parameters .....	653
<b>Endpoints and quotas .....</b>	<b>657</b>
Service endpoints .....	657

---

Service quotas .....	658
<b>Troubleshooting .....</b>	<b>663</b>
Simulation jobs .....	663
Simulation WorldForge .....	667
<b>Support policy .....</b>	<b>670</b>
Support changes: December 15, 2022 .....	670
Support changes: May 2, 2022 .....	670
Support changes: March 15, 2022 .....	671
Support ended: January 31, 2022 .....	672
Support ended: April 30, 2021 .....	673
<b>Document history .....</b>	<b>675</b>

# What Is AWS RoboMaker?

AWS RoboMaker is a cloud-based simulation service with which robotics developers can run, scale, and automate simulation without managing any infrastructure. With AWS RoboMaker, robotics developers can cost-effectively scale and automate simulation workloads, run large-scale and parallel simulations with a single API call, and create user-defined, randomized 3D virtual environments. Using the simulation service, you can speed application testing and create hundreds of new worlds from templates you define.



AWS RoboMaker is capable of automated testing within a continuous integration and continuous delivery (CI/CD) pipeline, training reinforcement models with high volumes of iterative trials and connecting multiple concurrent simulations to your fleet management software for testing. When combined with AWS machine learning, monitoring, and analytics services, robots can stream data, navigate, communicate, comprehend, and learn.

The [AWS RoboMaker resources](#) page contains links to simulation education resources, simulation world assets, sample applications, workshop and tutorial libraries, and hardware developer kits.

## AWS RoboMaker features

AWS RoboMaker includes the following features.

- [Simulation with AWS RoboMaker](#) – A fully managed simulation service with which you can run simulation jobs without provisioning or managing any infrastructure. The service supports large-scale and parallel simulations, and automatically scales based on the complexity of the scenarios being tested. The AWS RoboMaker simulation can be used to run the robot software and simulator of choice, including ROS, custom robot applications, Gazebo, Unity, Unreal, and Nvidia Isaac-based simulations.

- [Creating worlds with Simulation WorldForge](#) – Automatically create hundreds of pre-defined, randomized simulation worlds that mimic real-world conditions without engineering investment or managing world generation infrastructure. Today, Simulation WorldForge provides worlds for indoor home environments, complete with configurable floor plans and furnishings.

## AWS RoboMaker pricing

As with other AWS products, there are no contracts or minimum commitments for using AWS RoboMaker. For more information about the cost of using AWS RoboMaker, see [AWS RoboMaker Pricing](#).

To get started with AWS RoboMaker and learn more about the service, continue to [Getting started with AWS RoboMaker](#).

# Getting started with AWS RoboMaker

AWS RoboMaker runs robotic simulations in the cloud. To get started, you create an AWS account with an [IAM](#) role so you can see a simulated robot and environment in the console. Next, you build containers for both the environment and robot application to run simulation jobs. You then capture logs and data from the simulation job.

## Topics

- [AWS RoboMaker concepts](#)
- [Setting up AWS RoboMaker](#)
- [Running your first simulation](#)

## AWS RoboMaker concepts

This section describes key concepts and terminology you must understand to use AWS RoboMaker effectively. For more information, see [AWS RoboMaker FAQs](#).

### Concepts

- [Container](#)
- [Simulation job](#)
- [Simulation WorldForge](#)
- [Environment](#)
- [Application](#)
- [Application configuration](#)

## Container

Container images are stored in Amazon ECR. A container is created from an image when it is run by the service. A typical simulation might use Robot Operating System (ROS) with one container simulating the environment in Gazebo, and a second container simulating the robot. For more information, see [What Is Amazon ECR?](#) in the *Amazon ECR User Guide*.

## Simulation job

A single simulation job executes one or two applications. A typical simulation job includes a pairing of a robot application (custom logic that responds to environmental data) with an environment (models of the world that your robot inhabits). A simulation job provides results and metrics. For more information, see [Simulation with AWS RoboMaker](#).

## Simulation WorldForge

Simulation WorldForge makes it easier and faster to generate simulation worlds from the templates you define. In addition, it can help you manage simulation workloads that require a large number of simulation worlds with domain randomization. For more information, see [Creating worlds with Simulation WorldForge](#).

## Environment

An application specifies the configuration for, and the tools to run in, an environment. Tools running in an environment share the same file system, environment variables, and networking. The application and tools running in an environment should reflect changes to files in the environment and the environment should provide the available tools. You must provide a container image for the environment. For more information, see [Using images to develop AWS RoboMaker applications](#).

## Application

Before you can create a simulation job, you must create a robot application or simulation application in AWS RoboMaker. A robot application contains robot code for navigation and perception. A simulation application contains all of the assets and logic needed to simulate an environment. AWS RoboMaker supports creating more than one version of your robot application and simulation application. For more information, see [Versioning applications](#).

Our applications are made up of two (major) components:

- A *container*, which is a standard unit of software that packages up code and all its dependencies so the application runs quickly and reliably from one computing environment to another.
- A *software suite*, which indicates the environment from which the contents of the bundle can be extracted, sourced, validated, and run. Currently, the supported software suites are **General** (for robot applications) and **SimulationRuntime** (for simulation applications).

## Application configuration

When you supply a simulation or robot application in a [CreateSimulationJob](#), you actually specify a [RobotApplicationConfig](#) and a [SimulationApplicationConfig](#). In other words, specify the actual application ARN and version, plus the following launch configuration, upload configuration, and tools.

- [LaunchConfig](#) – Tells the simulation service how you want your application code run in the environment.
- [UploadConfiguration](#) – You can pass up to 10 upload configurations per application. AWS RoboMaker uploads files written to the upload configuration path to your output bucket.
- [Tool](#) – A list of the customized processes to run in the application container.

For more information, see [Simulation with AWS RoboMaker](#).

## Setting up AWS RoboMaker

To set up AWS RoboMaker, you must first create an AWS account and an IAM administrative user.

### Sign up for an AWS account

If you do not have an AWS account, complete the following steps to create one.

#### To sign up for an AWS account

1. Open <https://portal.aws.amazon.com/billing/signup>.
2. Follow the online instructions.

Part of the sign-up procedure involves receiving a phone call and entering a verification code on the phone keypad.

When you sign up for an AWS account, an *AWS account root user* is created. The root user has access to all AWS services and resources in the account. As a security best practice, assign administrative access to a user, and use only the root user to perform [tasks that require root user access](#).

AWS sends you a confirmation email after the sign-up process is complete. At any time, you can view your current account activity and manage your account by going to <https://aws.amazon.com/> and choosing **My Account**.

## Create a user with administrative access

After you sign up for an AWS account, secure your AWS account root user, enable AWS IAM Identity Center, and create an administrative user so that you don't use the root user for everyday tasks.

### Secure your AWS account root user

1. Sign in to the [AWS Management Console](#) as the account owner by choosing **Root user** and entering your AWS account email address. On the next page, enter your password.

For help signing in by using root user, see [Signing in as the root user](#) in the *AWS Sign-In User Guide*.

2. Turn on multi-factor authentication (MFA) for your root user.

For instructions, see [Enable a virtual MFA device for your AWS account root user \(console\)](#) in the *IAM User Guide*.

### Create a user with administrative access

1. Enable IAM Identity Center.

For instructions, see [Enabling AWS IAM Identity Center](#) in the *AWS IAM Identity Center User Guide*.

2. In IAM Identity Center, grant administrative access to a user.

For a tutorial about using the IAM Identity Center directory as your identity source, see [Configure user access with the default IAM Identity Center directory](#) in the *AWS IAM Identity Center User Guide*.

### Sign in as the user with administrative access

- To sign in with your IAM Identity Center user, use the sign-in URL that was sent to your email address when you created the IAM Identity Center user.



For help signing in using an IAM Identity Center user, see [Signing in to the AWS access portal](#) in the *AWS Sign-In User Guide*.

### Assign access to additional users

1. In IAM Identity Center, create a permission set that follows the best practice of applying least-privilege permissions.

For instructions, see [Create a permission set](#) in the *AWS IAM Identity Center User Guide*.

2. Assign users to a group, and then assign single sign-on access to the group.

For instructions, see [Add groups](#) in the *AWS IAM Identity Center User Guide*.

## Running your first simulation

The following sections of this guide explain how to run your first simulation and should be followed in order.

### To run your first simulation

1. [Build a containerized application](#)
2. [Publish to Amazon ECR](#)
3. [Run a simulation](#)

# Development with AWS RoboMaker

This section helps you get set up to develop with AWS RoboMaker. Learn how to build and publish images to Amazon ECR, and use images to develop applications.

## Topics

- [Building application containers](#)
- [Publishing application containers to Amazon ECR](#)
- [Working with robot applications](#)
- [Working with simulation applications](#)
- [Versioning applications](#)
- [Using images to develop AWS RoboMaker applications](#)

## Building application containers

There are three steps to submitting a simulation job in AWS RoboMaker: build the application containers, link the container to an AWS RoboMaker application, and use the containers to submit a simulation job. This section covers how to build application containers using Docker for AWS RoboMaker. We use the [hello-world sample application](#) to demonstrate the steps required to build sample robot and simulation application containers for a ROS-based example. This page also demonstrates how to test your container locally.

If you are not using ROS, see the blog post which describes [how to run any high-fidelity simulation in AWS RoboMaker with GPU and container support](#).

## Sections

- [Prerequisites](#)
- [Building application containers from a ROS workspace](#)
- [Testing your containers](#)

## Prerequisites

Before getting started, make sure your development environment has the necessary dependencies. You must have Docker, the AWS CLI, and the VCS Import Tool installed on your machine.

- Install the AWS CLI: <https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html>
- Install Docker: <https://docs.docker.com/get-docker/>
- Install the [VCS Import Tool](#) (if required for your workflow):

```
sudo pip3 install vcstool
```

You must also have an AWS account with an [IAM role containing the following permissions](#):

- Create an IAM role
- Create AWS RoboMaker resources (simulation job, robot, and simulation applications)
- Create and upload Amazon ECR repositories

Finally, you must know your account number and you must select a region in which to run the simulation. AWS RoboMaker is supported in the following Regions listed [AWS RoboMaker endpoints and quotas](#)

## Building application containers from a ROS workspace

AWS RoboMaker simulations are comprised of a simulation application and an optional robot application. Each of these applications is defined by a name and a container image. This section demonstrates how to build the container image for both a simulation application and a robot application. In the following example, both applications are built within a single workspace. The approach that follows is easily generalizable to any ROS project.

To start, clone the `hello world` repository and import the source.

```
git clone https://github.com/aws-robotics/aws-robomaker-sample-application-helloworld.git helloworld
cd helloworld
vcs import robot_ws < robot_ws/.rosinstall
vcs import simulation_ws < simulation_ws/.rosinstall
```

Next, create a new text file in the `helloworld` directory and name it `Dockerfile`. Copy and paste the following contents:

```
# ===== ROS/Colcon Dockerfile =====
# This sample Dockerfile will build a Docker image for AWS RoboMaker
```

```
# in any ROS workspace where all of the dependencies are managed by rosdep.
#
# Adapt the file below to include your additional dependencies/configuration
# outside of rosdep.
# =====

# ==== Arguments ====
# Override the below arguments to match your application configuration.
# =====

# ROS Distribution (ex: melodic, foxy, etc.)
ARG ROS_DISTRO=melodic
# Application Name (ex: helloworld)
ARG APP_NAME=robomaker_app
# Path to workspace directory on the host (ex: ./robot_ws)
ARG LOCAL_WS_DIR=workspace
# User to create and use (default: robomaker)
ARG USERNAME=robomaker
# The gazebo version to use if applicable (ex: gazebo-9, gazebo-11)
ARG GAZEBO_VERSION=gazebo-9
# Where to store the built application in the runtime image.
ARG IMAGE_WS_DIR=/home/$USERNAME/workspace

# ===== ROS Build Stages =====
# ${ROS_DISTRO}-ros-base
#   -> ros-robomaker-base
#     -> ros-robomaker-application-base
#       -> ros-robomaker-build-stage
#         -> ros-robomaker-app-runtime-image
# =====

# ==== ROS Base Image =====
# If running in production, you may choose to build the ROS base image
# from the source instruction-set to prevent impact from upstream changes.
# ARG UBUNTU_DISTRO=focal
# FROM public.ecr.aws/lts/ubuntu:${UBUNTU_DISTRO} as ros-base
# Instruction for each ROS release maintained by OSRF can be found here:
# https://github.com/osrf/docker\_images
# =====

# ==== Build Stage with AWS RoboMaker Dependencies ====
# This stage creates the robomaker user and installs dependencies required
# to run applications in RoboMaker.
# =====
```

```
FROM public.ecr.aws/docker/library/ros:${ROS_DISTRO}-ros-base AS ros-robomaker-base
ARG USERNAME
ARG IMAGE_WS_DIR

RUN apt-get clean
RUN apt-get update && apt-get install -y \
  lsb \
  unzip \
  wget \
  curl \
  xterm \
  python3-colcon-common-extensions \
  devilspie \
  xfce4-terminal

RUN groupadd $USERNAME && \
  useradd -ms /bin/bash -g $USERNAME $USERNAME && \
  sh -c 'echo "$USERNAME ALL=(root) NOPASSWD:ALL" >> /etc/sudoers'

USER $USERNAME
WORKDIR /home/$USERNAME

RUN mkdir -p $IMAGE_WS_DIR

# ==== ROS Application Base ====
# This section installs exec dependencies for your ROS application.
# Note: Make sure you have defined 'exec' and 'build' dependencies correctly
# in your package.xml files.
# =====
FROM ros-robomaker-base as ros-robomaker-application-base
ARG LOCAL_WS_DIR
ARG IMAGE_WS_DIR
ARG ROS_DISTRO
ARG USERNAME

WORKDIR $IMAGE_WS_DIR
COPY --chown=$USERNAME:$USERNAME $LOCAL_WS_DIR/src $IMAGE_WS_DIR/src

RUN sudo apt update && \
  rosdep update && \
  rosdep fix-permissions

# Note: This will install all dependencies.
```

```
# You could further optimize this by only defining the exec dependencies.
# Then, install the build dependencies in the build image.
RUN rosdep install --from-paths src --ignore-src -r -y

# ==== ROS Workspace Build Stage ====
# In this stage, we will install copy source files, install build dependencies
# and run a build.
# =====
FROM ros-robomaker-application-base AS ros-robomaker-build-stage
LABEL build_step="${APP_NAME}Workspace_Build"
ARG APP_NAME
ARG LOCAL_WS_DIR
ARG IMAGE_WS_DIR

RUN . /opt/ros/$ROS_DISTRO/setup.sh && \
    colcon build \
    --install-base $IMAGE_WS_DIR/$APP_NAME

# ==== ROS Robot Runtime Image ====
# In the final stage, we will copy the staged install directory to the runtime
# image.
# =====
FROM ros-robomaker-application-base AS ros-robomaker-app-runtime-image
ARG APP_NAME
ARG USERNAME
ARG GAZEBO_VERSION

ENV USERNAME=$USERNAME
ENV APP_NAME=$APP_NAME
ENV GAZEBO_VERSION=$GAZEBO_VERSION

RUN rm -rf $IMAGE_WS_DIR/src

COPY --from=ros-robomaker-build-stage $IMAGE_WS_DIR/$APP_NAME $IMAGE_WS_DIR/$APP_NAME

# Add the application source file to the entrypoint.
WORKDIR /
COPY entrypoint.sh /entrypoint.sh
RUN sudo chmod +x /entrypoint.sh && \
    sudo chown -R $USERNAME /entrypoint.sh && \
    sudo chown -R $USERNAME $IMAGE_WS_DIR/$APP_NAME

ENTRYPOINT ["/entrypoint.sh"]
```

The Dockerfile you just created is an instruction set used to build Docker images. Read through the comments in the Dockerfile to get a sense for what is being built and adapt as necessary for your need. For ease of development, the Dockerfile is based on the official ROS Docker images maintained by [Open Source Robotics Foundation \(OSRF\)](#). However, when running in production, you may choose to build the ROS base image with the [OSRF source instruction-set in GitHub](#) to prevent impact from upstream changes.

Next, create a new file called `entrypoint.sh`.

```
#!/bin/bash
set -e
source "/home/$USERNAME/workspace/$APP_NAME/setup.bash"
if [[ -f "/usr/share/$GAZEBO_VERSION/setup.sh" ]]
then
    source /usr/share/$GAZEBO_VERSION/setup.sh
fi
printenv
exec "${@:1}"
```

An ENTRYPOINT file is an executable that runs when the Docker container is spawned. We are using an entrypoint to source the ROS workspace, so we can easily run `roslaunch` commands in AWS RoboMaker. You may want to add your own environment configuration steps to this ENTRYPOINT file.

Our Dockerfile uses a multi-stage build and integrated caching with Docker BuildKit. Multi-stage builds allow workflows with separate build steps, so the build dependencies and source code are not copied into the runtime image. This reduces the size of the Docker image and improves performance. The caching operations speed up future builds by storing previously built files.

Build the robot application with the following command:

```
DOCKER_BUILDKIT=1 docker build . \
--build-arg ROS_DISTRO=melodic \
--build-arg LOCAL_WS_DIR=./robot_ws \
--build-arg APP_NAME=helloworld-robot-app \
-t robomaker-helloworld-robot-app
```

After the robot application has been built, you can build the simulation application as follows:

```
DOCKER_BUILDKIT=1 docker build . \
```

```
--build-arg GAZEBO_VERSION=gazebo-9 \  
--build-arg ROS_DISTRO=melodic \  
--build-arg LOCAL_WS_DIR=./simulation_ws \  
--build-arg APP_NAME=helloworld-sim-app \  
-t robomaker-helloworld-sim-app
```

Run the command `docker images` to confirm the Docker images have been successfully built. The output should resemble the following:

```
Administrator:~/environment/helloworld (ros1) $ docker images  
REPOSITORY              TAG          IMAGE ID          CREATED           SIZE  
robomaker-helloworld-sim-app  latest      5cb08816b6b3     6 minutes ago    2.8GB  
robomaker-helloworld-robot-app  latest      b5f6f755feec     10 minutes ago   2.79GB
```

At this point, you have successfully built your Docker images. It is a good idea to test these locally before uploading them for use with AWS RoboMaker. The next section describes how to do this.

## Testing your containers

The following commands give you the ability to run the application in your local development environment.

Launch the robot application:

```
docker run -it -v /tmp/.X11-unix/:/tmp/.X11-unix/ \  
-u robomaker -e ROBOMAKER_GAZEBO_MASTER_URI=http://localhost:5555 \  
-e ROBOMAKER_ROS_MASTER_URI=http://localhost:11311 \  
robomaker-helloworld-robot-app:latest roslaunch hello_world_robot rotate.launch
```

Launch the simulation application:

```
docker run -it -v /tmp/.X11-unix/:/tmp/.X11-unix/ \  
-u robomaker -e ROBOMAKER_GAZEBO_MASTER_URI=http://localhost:5555 \  
-e ROBOMAKER_ROS_MASTER_URI=http://localhost:11311 \  
robomaker-helloworld-sim-app:latest roslaunch hello_world_simulation empty_world.launch
```

Once you have confirmed that your containers are functioning properly, you can [Publish application containers to Amazon ECR](#) and then [Submit a simulation job](#).



## Publishing application containers to Amazon ECR

Containers used by AWS RoboMaker in a simulation job must be stored in [Amazon Elastic Container Registry \(ECR\)](#), a fully-managed container registry. Once you have successfully [built your application containers](#), you must push them to Amazon ECR. This section shows you how.

To get started, you can save some repetitive typing by setting a few environment variables that are reused in the commands that follow.

```
export robotapp=robomaker-helloworld-robot-app
export simapp=robomaker-helloworld-sim-app
export account=<YOUR AWS ACCOUNT NUMBER>
export region=<YOUR AWS REGION>
export ecruri=$account.dkr.ecr.$region.amazonaws.com
```

Next, sign in and create two new repositories.

```
aws ecr get-login-password --region $region | docker login --username AWS --password-stdin $ecruri
aws ecr create-repository --repository-name $robotapp
aws ecr create-repository --repository-name $simapp
```

You can tag your Docker images with the Amazon ECR repository's URI.

```
docker tag $robotapp $ecruri/$robotapp:latest
docker tag $simapp $ecruri/$simapp:latest
```

Next, push the Docker images to Amazon ECR.

```
docker push $ecruri/$robotapp
docker push $ecruri/$simapp
```

Finally, you can confirm your images uploaded to Amazon ECR by running the following commands.

```
aws ecr list-images --repository-name $simapp
aws ecr list-images --repository-name $robotapp
```

The following code snippet shows the expected output:

```
Administrator:~/environment/helloworld (ros1) $ aws ecr list-images --repository-name
$simapp
{
  "imageIds": [
    {
      "imageDigest": "sha256:28cad40230402343024kf303f30fk20f2f2fa0a8148",
      "imageTag": "latest"
    }
  ]
}
Administrator:~/environment/helloworld (ros1) $ aws ecr list-images --repository-name
$robotapp
{
  "imageIds": [
    {
      "imageDigest": "sha256:28cad40230402343024kf303f30fk20f2f2fa0a8148",
      "imageTag": "latest"
    }
  ]
}
```

Your robot and simulation Docker images are now being hosted within Amazon ECR. You must associate these images with a [robot application](#) or a [simulation application](#) before proceeding to [submit a simulation job](#).

## Application versioning

AWS RoboMaker supports creating more than one version of your robot applications and simulation applications. This helps you control which code your robots and simulations use. A version is a numbered snapshot of the \$LATEST version of your application. You can create a version to use in different parts of your development workflow. For example, development, beta deployment, or production.

When you version an AWS RoboMaker robot application or simulation application you create a snapshot of the application. Amazon ECR uses image digests to indicate the version of your application. AWS RoboMaker remembers the image digest for each version.

If you have the image uploaded to Amazon ECR and you haven't altered the image digest, you can access and use that version of your application. You can create a maximum of 40 versions per application.

When you create an image, you can also apply tags to it. You can specify the value of the tag field as `latest` for the `$LATEST` version. These values are distinct from each other.

There are two ways that an image gets the `latest` tag:

- You specified a tag with the value of `latest`.
- You push an image that doesn't have tags, in which case Amazon ECR updates the image with the `latest` tag.

When you specify a tag for an image in AWS RoboMaker, the image is always picked as the `$LATEST` version. For example, if you create a robot application with the image name `myImage`, the tag `xyz`, and the image digest `123`, the `$LATEST` version is `myImage:xyz` with the digest `123`.

The following are the scenarios in which you should add a tag:

- You update the `$LATEST` version to use a new tag. For example, if you have the image `myImage`, you can update your image with the tag `abc`. The `$LATEST` version of the image points to `myImage:abc`.
- You update the image and retag it. For example, you can make changes to an image that has the tag `abc`. You can use the tag `xyz` after you update it. The `$LATEST` version points to `myImage:xyz`.

For more information, see [Versioning applications](#).

## Working with robot applications

An AWS RoboMaker robot application is a container image responsible for running your robot's application stack. The robot application image must be hosted in Amazon ECR. Your robot application is often paired with a simulation application to create a simulation job.

### Sections

- [Creating a robot application](#)
- [Creating a robot application version](#)
- [Viewing a robot application](#)
- [Updating a robot application](#)
- [Deleting a robot application](#)

- [Deleting a robot application version](#)

## Creating a robot application

### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left pane, choose **Development**, and then choose **Robot applications**.
3. Select **Create robot application**.
4. In the **Create robot application** page, type a **Name** for the robot application. Choose a name that helps you identify the robot.
5. Provide your Amazon ECR container image. You can use images that you've pushed to Amazon ECR. For more information, see [What is Amazon Elastic Container Registry?](#).
6. For more about tagging, see [Tagging AWS RoboMaker resources](#).
7. Choose **Create**.

### Using the AWS CLI

```
aws robomaker create-robot-application \
--name my-robot-app \
--robot-software-suite name=General \
--environment uri=:<ACCOUNT>.dkr.ecr.<REGION>.amazonaws.com/my-robot-app:latest
```

## Creating a robot application version

### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Development**, and then choose **Robot applications**.

3. Choose the robot application **name**.
4. In the **Robot applications details** page, choose **Create new version**, and then choose **Create**.

Using the AWS CLI

```
aws robomaker create-robot-application-version --name my-robot-app-arn
```

## Viewing a robot application

Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Development**, then choose **Robot applications**.
3. Choose the **Name** of a robot application.

Using the AWS CLI

```
aws robomaker describe-robot-application --application my-robot-application-arn
```

## Updating a robot application

Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Development**, then choose **Robot applications**.
3. Check the box next to the robot application you want to update.
4. Choose **Actions**, then choose **Update**.
5. You can add or remove sources, but you must have at least one source robot application file.

6. Choose **Update** to update the robot application.

### Using the AWS CLI

```
aws robomaker update-robot-application \  
--application my-robot-application-arn \  
--robot-software-suite name=General \  
--environment uri=:<ACCOUNT>.dkr.ecr.<REGION>.amazonaws.com/my-robot-app:latest
```

## Deleting a robot application

### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Development**, then choose **Robot applications**.
3. Choose the **Name** of a robot application to see details including the time it was created and last updated.
4. In the robot application detail page, choose **Delete** and then choose **Delete** to confirm.

### Using the AWS CLI

```
aws robomaker delete-robot-application --application my-robot-application-arn
```

## Deleting a robot application version

### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Development**, then choose **Robot applications**.

3. Choose the **Name** of the robot application to see its versions.
4. In the robot detail page, choose the **Version** to see version details.
5. In the robot application version details page, choose **Delete**, and then choose **Delete** to confirm.

## Using the AWS CLI

```
aws robomaker delete-robot-application-version \  
--application my-robot-application-arn \  
--version 2
```

## Working with simulation applications

An AWS RoboMaker simulation application is a container image responsible for running your robot's simulation stack. The simulation application image must be hosted in Amazon ECR. Your simulation application is often paired with a robot application to create a simulation job.

### Sections

- [Creating a simulation application](#)
- [Creating a simulation application version](#)
- [Viewing a simulation application](#)
- [Updating a simulation application](#)
- [Deleting a simulation application](#)
- [Deleting a simulation application version](#)

## Creating a simulation application

### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Development**, and then choose **Simulation applications**.

3. Select **Create simulation application**.
4. On the **Create simulation application** page, type a **Name** for the simulation application. Choose a name that helps you identify the simulation.
5. Provide your Amazon ECR container image. You can use images that you've pushed to Amazon ECR. For more information, see [What is Amazon ECR?](#).
6. For more about tagging, see [Tagging AWS RoboMaker resources](#).
7. Choose **Create**.

## Using the AWS CLI

```
aws robomaker create-simulation-application \  
--name my-sim-app \  
--simulation-software-suite name=SimulationRuntime \  
--robot-software-suite name=General \  
--environment uri=<ACCOUNT>.dkr.ecr.<REGION>.amazonaws.com/my-sim-app:latest
```

## Creating a simulation application version

### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Development**, and then choose **Simulation applications**.
3. Choose the simulation application **name**.
4. On the **Simulation applications details** page, choose **Create new version**, and then choose **Create**.

### Using the AWS CLI

```
aws robomaker create-simulation-application-version --name my-simulation-  
application-arn
```



## Viewing a simulation application

### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Development**, then choose **Simulation applications**.
3. Select the **Name** of a simulation application to see details including the time it was created and last updated.

### Using the AWS CLI

```
aws robomaker describe-simulation-application --job my-simulation-job-arn
```

## Updating a simulation application

### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Development**, then choose **Simulation applications**.
3. Check the box next to the simulation application you want to update.
4. Select **Actions**, then select **Update**.
5. You can add or remove sources, but you must have at least one source simulation application file.
6. Select **Update** to update the simulation application.

### Using the AWS CLI

```
aws robomaker update-simulation-application \  
--application my-simulation-application-arn \  
--robot-software-suite name=General \  
--simulation-software-suite name=SimulationRuntime \  

```

```
--environment uri=<ACCOUNT>.dkr.ecr.<REGION>.amazonaws.com/my-simulation-app:latest
```

## Deleting a simulation application

### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Development**, then choose **Simulation applications**.
3. Select the **Name** of a simulation application. This shows details such as the time it was created and last updated.
4. In the simulation application detail page, choose **Delete** and then choose **Delete** to delete to confirm.

### Using the AWS CLI

```
aws robomaker delete-simulation-application --application my-simulation-application-arn
```

## Deleting a simulation application version

### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Development**, then choose **Simulation applications**.
3. Select the **Name** of the simulation application to see its versions.
4. On the simulation detail page, choose **Version** to see details.
5. On the details page, choose **Delete**, and then choose **Delete** to confirm.

### Using the AWS CLI

```
aws robomaker delete-simulation-application-version \  
--application my-simulation-application-arn \  
--version 2
```

## Versioning applications

AWS RoboMaker supports creating more than one version of your robot and simulation applications. This helps you control which code your robots and simulations use. A version is a numbered snapshot of the `$LATEST` version of your application. You can create a version to use in different parts of your development workflow, such as development, beta deployment, or production.

When you version an AWS RoboMaker robot application or simulation application, you create a snapshot of the application.

If you're using `colcon` to build your applications, AWS RoboMaker remembers the Amazon S3 path and ETag of the file for each version. You can use the version of the application as it existed when the version was made provided it still exists in the Amazon S3 path and has not been altered (its ETag is unchanged).

If you're using container images for your applications, you upload your images to Amazon ECR. Amazon ECR uses image digests to indicate the version of your application. AWS RoboMaker remembers the image digest for each version.

If you have the image uploaded to Amazon ECR and you haven't altered the image digest, you can access and use that version of your application.

You can create a maximum of 40 versions per application.

### Topics

- [Versioning applications with images](#)
- [The `\$LATEST` version](#)
- [Updating an application version](#)
- [Deleting an application version](#)

## Versioning applications with images

You can update the `$LATEST` version of your container image as you develop your application. When you select the `$LATEST` version, you can get it from the Amazon ECR location that you specify.

When you create an image, you can also apply tags to it. You can specify the value of the tag field as `"latest"` for the `$LATEST` version. These values are distinct from each other.

There are two ways that an image gets the `"latest"` tag:

- You specified a tag with the value of `"latest"`.
- You push an image that doesn't have tags, in which case Amazon ECR updates the image with the `"latest"` tag.

When you specify a tag for an image in AWS RoboMaker, that image is always picked as the `$LATEST` version. For example, if you create a robot application with the image name `"myImage"`, the tag `"xyz"`, and the image digest `"123"`, the `$LATEST` version is `myImage:xyz` with the digest `"123"`.

The following are the scenarios when you want to add a tag:

- You want to update the `$LATEST` version to use a new tag. For example, if you have the image `"myImage"`, you can update your image with the tag `"abc"`. The `$LATEST` version of the image points to `myImage:abc`.
- You want to update the image and retag it. For example, you can make changes to an image that has the tag `"abc"`. You can use the tag `"xyz"` after you update it. The `$LATEST` version points to `myImage:xyz`.

## The `$LATEST` version

When you create a version, AWS RoboMaker takes a snapshot of the `$LATEST` version and increments the version number by 1. AWS RoboMaker remembers the Amazon S3 path and ETag of the file. The path is used to retrieve the file. The ETag is used to confirm it has not changed. Version numbers are never reused. For example, if your latest version is 10 and you remove it and then create a new version, the new version is version 11.

You can update the \$LATEST version as you develop your application. When you select the \$LATEST version, it is retrieved from the Amazon S3 location you specify. For example, when you start a simulation job using the latest version of your robot application and simulation application and then make changes to the robot application at the Amazon S3 path, the updated robot application is used when the simulation job is restarted.

When you deploy a robot application, you must select a specific numbered version to deploy. For more information about how to create a robot application version, see [Creating a robot application version](#).

For more information about how to create a simulation application version, see [Creating a simulation application version](#). For more information about ETags, see [Common Response Headers](#).

## Updating an application version

You can update only the \$LATEST version of an AWS RoboMaker application . When you do this, it is available to use in AWS RoboMaker. For example, if you restart a simulation job, the latest version of the applications is used in the simulation.

For more information, see [Updating a robot application](#) and [Updating a simulation application](#).

## Deleting an application version

When you no longer need an application version, delete it. For more information, see [Deleting a robot application version](#) and [Deleting a simulation application version](#).

## Using images to develop AWS RoboMaker applications

### Important

Starting March 15, 2022, we made changes to AWS RoboMaker simulation which may have affected your pre-existing simulation jobs. To learn more about these changes and migration steps you can follow for your robot applications, simulation applications and simulation jobs, see [Migrating ROS applications to containers](#).

You can use one or more container images to develop and run your simulation and robot applications. For information about images, see [Docker basics for Amazon ECS](#). The images that

you use must meet the requirements listed in [Requirements for AWS RoboMaker compatible containers](#).

You can use your own images with AWS RoboMaker if you use one of the development environments that we support.

There are multiple ways that you can use container images to develop your applications. To see examples of how to develop your applications, see [Creating images to run the hello world sample application](#).

After you've used images to develop your applications, you can test them. To test whether your applications work, you can visualize them on your local Linux machine.

After you've tested that your simulation works, you can push your images to Amazon ECR and run simulation jobs to see how your robot would interact in a virtual environment.

## Topics

- [Migrating ROS applications to containers](#)
- [ROS container FAQs](#)
- [Requirements for AWS RoboMaker compatible containers](#)
- [Creating images to run GPU applications](#)
- [Creating images to run the hello world sample application](#)

## Migrating ROS applications to containers

Starting in October of 2021, AWS RoboMaker expanded support to enable any set of robot and simulation software. Previously, Robot Operating System (ROS) and Gazebo were the only supported robot and simulation software configurations allowed to run in AWS RoboMaker. With this change, you can now configure any robot and simulation software of your choice while running simulations in AWS RoboMaker.

### What does this mean for customers that would like to continue to use ROS and Gazebo?

It means you are required to move to a Docker-based workflow to build your own application containers for use in AWS RoboMaker. Docker is an industry standard tool that allows developers to bundle their application's dependencies and ship their software as a bundled package (container). For more information, see [Docker basics for Amazon ECS](#). The images that you use must meet the requirements listed in [Requirements for AWS RoboMaker compatible containers](#).

## What if I'm already using ROS based containers?

Then you're most of the way there! You must update the software suite of your [Robot](#) and [Simulation](#) application from ROS related software suites to **General** and **Simulation runtime** software suite via the AWS console or CLI. Then, follow the steps for [Running a simulation](#).

## How to migrate to a Docker-based workflow

1. Choose one of the following tutorials depending on your desired version of ROS and follow the steps within it.
  - [Running a sample application with ROS Melodic and Gazebo 9](#)
  - [Running a sample application with ROS 2 Foxy and Gazebo 11](#)
2. After you have created containers, you can proceed to submitting your simulation job.
  - [Running a simulation](#)

## ROS container FAQs

This page lists common questions and answers related to migrating ROS-based robot and simulation applications to Docker containers suitable for running in AWS RoboMaker.

### **Our workflow submits simulation jobs using colcon-bundled robot and simulation applications. Do I need to migrate?**

Yes, you must migrate. The steps to migrate are located at [Migrating ROS applications to containers](#).

### **I am not sure if my robot and simulation applications need to be migrated. How can I tell?**

You can check via the AWS console or the AWS CLI. For instructions, choose the following applicable tab.

Using the console

1. Sign in to the [AWS RoboMaker console](#).
2. In the left navigation pane, choose **Development**, and then choose **Simulation applications**.
3. Select the **Name** of a simulation application to see its details.

If you see **General** and **Simulation runtime**, migration is not necessary. If you see any ROS or Gazebo specific values, you must migrate.

## Using the AWS CLI

### Example

The following is an example AWS CLI command that performs the equivalent of the console-based steps.

```
aws robomaker describe-simulation-application --application YOUR-SIM-APP-ARN
```

This command returns output that indicates the `simulationSoftwareSuite`, the `robotSoftwareSuite` (if applicable), and the environment URI. If you see **Simulation Runtime** as the `simulationSoftwareSuite` and **General** as the `robotSoftwareSuite`, and your environment URI is set, your simulation applications do not require migration.

## How do the robot and simulation application containers communicate with each other?

It is no different from how ROS-based applications typically communicate with each other using ROS middleware. However, you must set some ROS-specific environment variables within the launch configuration objects of your simulation job request.

The following is an example snippet of the settings you must use for the robot application `launchConfig`.

```
"robotApplications": [  
  {  
    "application": "YOUR-ROBOT-APP-ARN",  
    "applicationVersion": "$LATEST",  
    "launchConfig": {  
      "environmentVariables": {  
        "ROS_IP": "ROBOMAKER_ROBOT_APP_IP",  
        "ROS_MASTER_URI": "http://ROBOMAKER_ROBOT_APP_IP:11311",  
        "GAZEBO_MASTER_URI": "http://ROBOMAKER_SIM_APP_IP:11345"  
      },  
      ... # Removed extra data for clarity
```



```
}  
]
```

The following is an example snippet of the settings you must use for the simulation application launchConfig.

```
"simulationApplications": [  
  {  
    "application": "YOUR-SIM-APP-ARN",  
    "applicationVersion": "$LATEST",  
    "launchConfig": {  
      "environmentVariables": {  
        "ROS_IP": "ROBOMAKER_SIM_APP_IP",  
        "ROS_MASTER_URI": "http://ROBOMAKER_ROBOT_APP_IP:11311",  
        "GAZEBO_MASTER_URI": "http://ROBOMAKER_SIM_APP_IP:11345"  
      },  
      ... # Removed extra data for clarity  
    }  
  }  
]
```

The containers communicate with each other as expected if you use the provided ROBOMAKER\_\* strings and port numbers to set ROS\_IP, ROS\_MASTER\_URI, and GAZEBO\_MASTER\_URI.

For more information, see [Running a simulation](#).

### Where did my Real-Time Factor (RTF) metric go? How can I restore it?

AWS RoboMaker no longer publishes this metric automatically. If you want to publish this metric to CloudWatch, you must import the [AWS RoboMaker CloudWatch Publisher](#) package into your simulation application and modify your simulation launch file using the instructions provided in the [README.md](#) file.

### How do I cancel and tag my simulation jobs?

You can use the VPC configuration to self-tag or self-cancel your AWS RoboMaker simulation jobs using the generic AWS APIs. To use the following approach, the container must be running in a VPC with a public route through a [NAT](#) or [IGW](#) to the AWS APIs. The simplest approach is to use a public subnet in your [default VPC](#) to connect to AWS APIs. If you'd like to run simulations in a private subnet, you could alternatively set up a NAT or set up an interface VPC endpoint. For more information, see [AWS RoboMaker and interface VPC endpoints \(AWS PrivateLink\)](#).

**Note**

If you are using the IGW, make sure to set `assignPublicIp=True` as described in the documentation below. If you are using a public IP, ensure that your security groups are sufficiently locked down.

You must add the following block to the request parameters.

```
vpcConfig={
  'subnets': [
    'string',
  ],
  'securityGroups': [
    'string',
  ],
  'assignPublicIp': True|False
},
```

In addition, the AWS RoboMaker simulation job must have an IAM role with permissions to tag and cancel simulation jobs.

In your simulation job, you can either use the AWS CLI or the boto3 Python library to call public AWS RoboMaker APIs. You must have the AWS CLI and boto3 library preinstalled in your container before you can use them in a AWS RoboMaker simulation job. The following Python sample code shows how to cancel a simulation job.

```
class RoboMakerUtils:

    def __init__(self):
        self.job_arn = os.getenv('AWS_ROBOMAKER_SIMULATION_JOB_ARN')
        self.client = boto3.client('robomaker',
region_name=os.getenv('AWS_ROBOMAKER_REGION', 'us-east-1'))

    def tag_robomaker_sim_job(self, key, value):
        self.client.tag_resource(
            resourceArn=self.job_arn,
            tags={
                key: str(value)
            }
        )
```

```
def cancel_robomaker_sim_job(self):
    self.tag_robomaker_sim_job("END_TIME", time.time())
    response = self.client.cancel_simulation_job(
        job=self.job_arn
    )
```

## How do I import Simulation WorldForge worlds into the simulation job?

If you need to import Simulation WorldForge assets into your simulation job, use the [DataSource API](#). This allows you to import world assets from the Amazon S3 output directory of the world export job into a destination of your choosing within your simulation job container.

For more information, see [Using exported worlds in simulation](#).

## My application's log files are not being created. What's going on?

Make sure you have created all the output directories you rely on for debugging related artifacts in your Dockerfile. For example, you can add the following line to your Dockerfile.

```
RUN mkdir -p $YOUR_LOG_DIR
```

For more information, see [Adding a custom upload configuration](#).

## My simulation application failed with 'run\_id on parameter server does not match declared run\_id'. What should I do?

If you are launching your ROS simulation job with both robot application and simulation application, you should add `--wait` to the `roslaunch` commands.

## Requirements for AWS RoboMaker compatible containers

You must meet a set of requirements to run a **AWS RoboMaker Compatible Container (container image)** and to start a simulation successfully. If you've met these requirements, and you're still having trouble running the simulation, see [Simulation jobs](#) and [Simulation WorldForge](#).

### Simulation runtime requirements

Your container image can't use `VOLUME` in the Dockerfile. If `VOLUME` is in the Dockerfile, your simulation **WILL** fail with a 4XX error code.

Your container image can't use EXPOSE in the Dockerfile. If EXPOSE is in the Dockerfile, AWS RoboMaker **WILL** fail your simulation with a 4XX error code.

Your container image **MUST** be less than or equal to 20 GB in size compressed. If your container image is greater than 20 GB compressed, AWS RoboMaker **WILL** fail the simulation with a 4XX error code.

You can't specify CMD in your Dockerfile. If you do, AWS RoboMaker overwrites it with the package name and launch file. Instead, you can use the command parameter in the launchConfig of each simulation application or robot application within your [CreateSimulationJob](#) request to provide a list of launch commands. This is set as CMD in the simulation job. An example command is ["/bin/bash", "-c", "sleep 365d"].

If you want to add tools to your simulation job, you **MUST** install bash to your container image. Your tools are launched with ["/bin/bash", "-c", "<command>"].

If your container is running ROS and you need communication between your robot application and your simulation application, you should set up the following robotics frameworks:

- ROS Master
- Gazebo Master
- ROS IP

You can't customize the /etc/resolv.conf file in your container. AWS RoboMaker overwrites the file with its own file.

If you're running your Dockerfile on AWS, you can't **MOUNT** the image. If you specify Mount in the Dockerfile, AWS RoboMaker **WILL** fail your simulation with a 4XX error code.

Your container image can't use system calls that are blocked by the default Docker seccomp profile. For information about blocked system calls, see [Seccomp security profiles](#).

To specify a user that runs an image, you can specify a USER keyword in the Dockerfile. If you don't specify a user, AWS RoboMaker uses the root user in the container.

In your container image, you can specify either the USER as either a name or a UID:GID. If your container image doesn't have a UID, it has a default value of 1000.

Your container image can't store data in `/opt/amazon/robomaker` or in any of its subfolders. Only AWS RoboMaker can use that directory. Your simulation might not behave properly if you use that directory.

The following runtime configurations are not supported.

	Docker Run Argument	Description
1	<code>--add-host</code>	Add a custom host-to-IP mapping (host:ip)
2	<code>--attach</code> , <code>-a</code>	Attach to STDIN, STDOUT or STDERR
3	<code>--blkio-weight</code>	Block IO (relative weight), between 10 and 1000, or 0 to disable (default 0)
4	<code>--blkio-weight-device</code>	Block IO weight (relative device weight)
5	<code>--cap-add</code>	Add Linux capabilities
6	<code>--cap-drop</code>	Drop Linux capabilities
7	<code>--cgroup-parent</code>	Optional parent cgroup for the container
8	<code>--cgroupns</code>	API 1.41+ < <a href="https://docs.docker.com/engine/api/v1.41/">https://docs.docker.com/engine/api/v1.41/</a> > <code>__Cgroup</code> namespace to use (host private) 'host': Run the container in the Docker host's cgroup namespace 'private': Run the container in its own private cgroup namespace ''': Use the cgroup namespace as configured by the default-c

	Docker Run Argument	Description
		groupns-mode option on the daemon (default)
9	<code>--cidfile</code>	Write the container ID to the file
10	<code>--cpu-count</code>	CPU count (Windows only)
11	<code>--cpu-percent</code>	CPU percent (Windows only)
12	<code>--cpu-period</code>	Limit CPU CFS (Completely Fair Scheduler) period
13	<code>--cpu-quota</code>	Limit CPU CFS (Completely Fair Scheduler) quota
14	<code>--cpu-rt-period</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> >__Limit CPU real-time period in microseconds
15	<code>--cpu-rt-runtime</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> >__Limit CPU real-time runtime in microseconds
16	<code>--cpu-shares</code> , <code>-c</code>	CPU shares (relative weight)
17	<code>--cpus</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> >__Number of CPUs
18	<code>--cpuset-cpus</code>	CPUs in which to allow execution (0-3, 0,1)
19	<code>--cpuset-mems</code>	MEMs in which to allow execution (0-3, 0,1)

	<b>Docker Run Argument</b>	<b>Description</b>
20	<code>--detach , -d</code>	Run container in background and print container ID
21	<code>--detach-keys</code>	Override the key sequence for detaching a container
22	<code>--device</code>	Add a host device to the container
23	<code>--device-cgroup-rule</code>	Add a rule to the cgroup allowed devices list
24	<code>--device-read-bps</code>	Limit read rate (bytes per second) from a device
25	<code>--device-read-iops</code>	Limit read rate (IO per second) from a device
26	<code>--device-write-bps</code>	Limit write rate (bytes per second) to a device
27	<code>--device-write-iops</code>	Limit write rate (IO per second) to a device
28	<code>--disable-content-trust</code>	Skip image verification
29	<code>--dns</code>	Set custom DNS servers
30	<code>--dns-opt</code>	Set DNS options
31	<code>--dns-option</code>	Set DNS options
32	<code>--dns-search</code>	Set custom DNS search domains
33	<code>--domainname</code>	Container NIS domain name

	Docker Run Argument	Description
34	<code>--gpus</code>	API 1.40+ < <a href="https://docs.docker.com/engine/api/v1.40/">https://docs.docker.com/engine/api/v1.40/</a> > __GPU devices to add to the container ('all' to pass all GPUs)
35	<code>--group-add</code>	Add additional groups to join
36	<code>--health-cmd</code>	Command to run to check health
37	<code>--health-interval</code>	Time between running the check (ms m h) (default 0s)
38	<code>--health-retries</code>	Consecutive failures needed to report unhealthy
39	<code>--health-start-period</code>	API 1.29+ < <a href="https://docs.docker.com/engine/api/v1.29/">https://docs.docker.com/engine/api/v1.29/</a> > __Start period for the container to initialize before starting health-retries countdown (ms m h) (default 0s)
40	<code>--health-timeout</code>	Maximum time to allow one check to run (ms m h) (default 0s)
41	<code>--help</code>	Print usage
42	<code>--hostname</code> , <code>-h</code>	Container host name



	Docker Run Argument	Description
43	<code>--init</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> >_Run an init inside the container that forwards signals and reaps processes
44	<code>--interactive , -i</code>	Keep STDIN open even if not attached
45	<code>--io-maxbandwidth</code>	Maximum IO bandwidth limit for the system drive (Windows only)
46	<code>--io-maxiops</code>	Maximum IOps limit for the system drive (Windows only)
47	<code>--ip</code>	IPv4 address (e.g., 172.30.10.0.104)
48	<code>--ip6</code>	IPv6 address (e.g., 2001:db8::33)
49	<code>--ipc</code>	IPC mode to use
50	<code>--isolation</code>	Container isolation technology
51	<code>--kernel-memory</code>	Kernel memory limit
52	<code>--label , -l</code>	Set meta data on a container
53	<code>--label-file</code>	Read in a line delimited file of labels
54	<code>--link</code>	Add link to another container
55	<code>--link-local-ip</code>	Container IPv4/IPv6 link-local addresses

	<b>Docker Run Argument</b>	<b>Description</b>
56	<code>--log-driver</code>	Logging driver for the container
57	<code>--log-opt</code>	Log driver options
58	<code>--mac-address</code>	Container MAC address (e.g., 92:d0:c6:0a:29:33)
59	<code>--memory</code> , <code>-m</code>	Memory limit
60	<code>--memory-reservation</code>	Memory soft limit
61	<code>--memory-swap</code>	Swap limit equal to memory plus swap: '-1' to enable unlimited swap
62	<code>--memory-swappiness</code>	Tune container memory swappiness (0 to 100)
63	<code>--name</code>	Assign a name to the container
64	<code>--net</code>	Connect a container to a network
65	<code>--net-alias</code>	Add network-scoped alias for the container
66	<code>--network</code>	Connect a container to a network
67	<code>--network-alias</code>	Add network-scoped alias for the container
68	<code>--no-healthcheck</code>	Disable any container-specific HEALTHCHECK

	Docker Run Argument	Description
69	<code>--oom-kill-disable</code>	Disable OOM Killer
70	<code>--oom-score-adj</code>	Tune host's OOM preferences (-1000 to 1000)
71	<code>--pid</code>	PID namespace to use
72	<code>--pids-limit</code>	Tune container pids limit (set -1 for unlimited)
73	<code>--platform</code>	API 1.32+ <a href="https://docs.docker.com/engine/api/v1.32/">https://docs.docker.com/engine/api/v1.32/</a> >__Set platform if server is multi-platform capable
74	<code>--privileged</code>	Give extended privileges to this container
75	<code>--publish , -p</code>	Publish a container's port(s) to the host
76	<code>--publish-all , -P</code>	Publish all exposed ports to random ports
77	<code>--pull</code>	Pull image before running ("always" "never")
78	<code>--read-only</code>	Mount the container's root filesystem as read only
79	<code>--restart</code>	Restart policy to apply when a container exits
80	<code>--rm</code>	Automatically remove the container when it exits
81	<code>--runtime</code>	Runtime to use for this container

	Docker Run Argument	Description
82	<code>--security-opt</code>	Security Options
83	<code>--shm-size</code>	Size of /dev/shm
84	<code>--sig-proxy</code>	Proxy received signals to the process
85	<code>--stop-timeout</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> > <code>__Timeout</code> (in seconds) to stop a container
86	<code>--storage-opt</code>	Storage driver options for the container
87	<code>--sysctl</code>	Sysctl options
88	<code>--tmpfs</code>	Mount a tmpfs directory
89	<code>--tty</code> , <code>-t</code>	Allocate a pseudo-TTY
90	<code>--ulimit</code>	Ulimit options
91	<code>--usersns</code>	User namespace to use
92	<code>--uts</code>	UTS namespace to use
93	<code>--volume</code> , <code>-v</code>	Bind mount a volume
94	<code>--volume-driver</code>	Optional volume driver for the container
95	<code>--volumes-from</code>	Mount volumes from the specified container(s)

If you run a simulation job with the preceding runtime configurations, AWS RoboMaker **WILL** fail your simulation with a 4XX error code.

## Metadata requirements

Your container image:

- **MUST** be [Open Container Initiative \(OCI\)](#) complaint.
- **MUST** be built for the X86\_64 architecture. If it's built for a different architecture, AWS RoboMaker **WILL** fail the simulation with a 4XX error code.
- **MUST** be less than or equal to 40 GB in size uncompressed. If your container image is greater than 40 GB uncompressed, AWS RoboMaker **WILL** fail the simulation with a 4XX error code.
- **MUST** have a V2 image manifest, schema version 2 compatible.
- **MUST** use a base image that is based on Linux. If you don't use a base image that is based on Linux, AWS RoboMaker **WILL** fail the simulation with a 4XX error code.
- **MUST** use a development environment and operating system that are compatible with each other. The following are examples of compatible combinations of development environments and operating systems:
  - Robot Operating System (ROS) Melodic – ubuntu:bionic
  - Robot Operating System (ROS) 2 Foxy – ubuntu:focal

If you don't use a compatible combination of robotics framework and operating system, your simulation might show unexpected behavior.

## Binary requirements

The following are the binary requirements for your container image:

To support GUI streaming, we recommend installing and sourcing the following binaries:

- `devilspie`

We recommend that your container image use absolute paths for its executables. We also recommend that the executable inside the container runs correctly. Your simulation **WILL** fail if it can't find the path to your executables.

## GPU requirements

Your container image:

- **MUST** have `glvnd` installed if using OpenGL in your applications.

- **MUST** have NVIDIA CUDA 11.2 or lower if using CUDA in your applications.
- **MUST** have OpenGL version 4.6 or lower if using OpenGL in your applications.
- **MUST** have Vulkan version 1.2 or lower if using Vulkan APIs in your applications.
- **MUST** have OpenCL version 1.2 or lower if using OpenCL in your applications.

### Note

AWS RoboMaker supports Vulkan only for offscreen rendering and is not operational in GUI displays. So, `streamUI` should be set to `false` if using Vulkan.

For detailed instructions on how GPU images can be created, see [Creating images to run GPU applications](#).

## Dockerfile and environment variable requirements

A container image **MUST** provide an entrypoint script for sourcing. The entrypoint script **MUST** have `exec "${@:1}"` as the last line so that AWS RoboMaker can run the entrypoint script. Running the entrypoint script gives you the ability to use the `roslaunch package-name` command. *launch-file* command to run the containers.

Your container image can't use `VOLUME` in the Dockerfile. If `VOLUME` is in the Dockerfile, your simulation **WILL** fail with a 4XX error code.

The `EXPOSE` keyword in your Dockerfile is ignored by AWS RoboMaker. Any ports exposed by the `EXPOSE` keyword are not automatically exposed by the system. If you would like to expose ports on your simulation, you can use AWS RoboMaker [port forwarding configuration](#).

AWS RoboMaker uses the following environment variables. If you run your simulation on AWS, AWS RoboMaker overwrites any value that you specify for these environment variables:

- `ROBOMAKER*`
- `DCV_VIRTUAL_SESSION`
- `XDG_SESSION_ID`
- `DCV_SESSION_ID`
- `XDG_SESSION_TYPE`
- `XDG_RUNTIME_DIR`
- `SHLVL`
- `XAUTHORITY`

You can't specify `CMD` in your Dockerfile. If you do, AWS RoboMaker overwrites with the command in your simulation `launchConfig`.

## Network, mount, security and user requirements

If your container is running ROS and you need communication between your robot application and your simulation application, you should set up the following robotics frameworks:

- ROS Master
- Gazebo Master
- ROS IP

You can't customize the `/etc/resolv.conf` file in your container. AWS RoboMaker overwrites the file with its own file.

If you're running your Dockerfile on AWS, you can't **MOUNT** the image. If you specify `Mount` in the Dockerfile, AWS RoboMaker **WILL** fail your simulation with a 4XX error code.

Your container image can't use system calls that are blocked by the default Docker `seccomp` profile. For information about blocked system calls, see [Seccomp security profiles](#).

To specify a user that runs an image, you can specify a `USER` keyword in the Dockerfile. If you don't specify a user, AWS RoboMaker uses the root user in the container.

In your container image, you can specify the `USER` as either a name or a `UID:GID`. If your container image doesn't have a `UID`, it has a default value of `1000`.

## Other requirements

Your container image can't store data in `/opt/amazon/robomaker` or in any of its subfolders. Only AWS RoboMaker can use that directory. Your simulation might not behave properly if you use that directory.

The following runtime configurations are not supported.

	Docker Run Argument	Description
1	<code>--add-host</code>	Add a custom host-to-IP mapping (host:ip)

	Docker Run Argument	Description
2	<code>--attach , -a</code>	Attach to STDIN, STDOUT or STDERR
3	<code>--blkio-weight</code>	Block IO (relative weight), between 10 and 1000, or 0 to disable (default 0)
4	<code>--blkio-weight-device</code>	Block IO weight (relative device weight)
5	<code>--cap-add</code>	Add Linux capabilities
6	<code>--cap-drop</code>	Drop Linux capabilities
7	<code>--cgroup-parent</code>	Optional parent cgroup for the container
8	<code>--cgroupns</code>	API 1.41+ < <a href="https://docs.docker.com/engine/api/v1.41/">https://docs.docker.com/engine/api/v1.41/</a> > <code>__Cgroup</code> namespace to use (host private) 'host': Run the container in the Docker host's cgroup namespace 'private': Run the container in its own private cgroup namespace ''': Use the cgroup namespace as configured by the default-cgroupns-mode option on the daemon (default)
9	<code>--cidfile</code>	Write the container ID to the file
10	<code>--cpu-count</code>	CPU count (Windows only)
11	<code>--cpu-percent</code>	CPU percent (Windows only)



	Docker Run Argument	Description
12	<code>--cpu-period</code>	Limit CPU CFS (Completely Fair Scheduler) period
13	<code>--cpu-quota</code>	Limit CPU CFS (Completely Fair Scheduler) quota
14	<code>--cpu-rt-period</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> >__Limit CPU real-time period in microseconds
15	<code>--cpu-rt-runtime</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> >__Limit CPU real-time runtime in microseconds
16	<code>--cpu-shares</code> , <code>-c</code>	CPU shares (relative weight)
17	<code>--cpus</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> >__Number of CPUs
18	<code>--cpuset-cpus</code>	CPUs in which to allow execution (0-3, 0,1)
19	<code>--cpuset-mems</code>	MEMs in which to allow execution (0-3, 0,1)
20	<code>--detach</code> , <code>-d</code>	Run container in background and print container ID
21	<code>--detach-keys</code>	Override the key sequence for detaching a container
22	<code>--device</code>	Add a host device to the container

	<b>Docker Run Argument</b>	<b>Description</b>
23	<code>--device-cgroup-rule</code>	Add a rule to the cgroup allowed devices list
24	<code>--device-read-bps</code>	Limit read rate (bytes per second) from a device
25	<code>--device-read-iops</code>	Limit read rate (IO per second) from a device
26	<code>--device-write-bps</code>	Limit write rate (bytes per second) to a device
27	<code>--device-write-iops</code>	Limit write rate (IO per second) to a device
28	<code>--disable-content-trust</code>	Skip image verification
29	<code>--dns</code>	Set custom DNS servers
30	<code>--dns-opt</code>	Set DNS options
31	<code>--dns-option</code>	Set DNS options
32	<code>--dns-search</code>	Set custom DNS search domains
33	<code>--domainname</code>	Container NIS domain name
34	<code>--gpus</code>	API 1.40+ < <a href="https://docs.docker.com/engine/api/v1.40/">https://docs.docker.com/engine/api/v1.40/</a> >__GPU devices to add to the container ('all' to pass all GPUs)
35	<code>--group-add</code>	Add additional groups to join
36	<code>--health-cmd</code>	Run to check health

	Docker Run Argument	Description
37	<code>--health-interval</code>	Time between running the check (msm h) (default 0s)
38	<code>--health-retries</code>	Consecutive failures needed to report unhealthy
39	<code>--health-start-period</code>	API 1.29+ < <a href="https://docs.docker.com/engine/api/v1.29/">https://docs.docker.com/engine/api/v1.29/</a> >__Start period for the container to initialize before starting health-retries countdown (msm h) (default 0s)
40	<code>--health-timeout</code>	Maximum time to allow one check to run (msm h) (default 0s)
41	<code>--help</code>	Print usage
42	<code>--hostname , -h</code>	Container host name
43	<code>--init</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> >__Run an init inside the container that forwards signals and reaps processes
44	<code>--interactive , -i</code>	Keep STDIN open even if not attached
45	<code>--io-maxbandwidth</code>	Maximum IO bandwidth limit for the system drive (Windows only)
46	<code>--io-maxiops</code>	Maximum IOps limit for the system drive (Windows only)

	Docker Run Argument	Description
47	<code>--ip</code>	IPv4 address (e.g., 172.30.100.104)
48	<code>--ip6</code>	IPv6 address (e.g., 2001:db8::33)
49	<code>--ipc</code>	IPC mode to use
50	<code>--isolation</code>	Container isolation technology
51	<code>--kernel-memory</code>	Kernel memory limit
52	<code>--label , -l</code>	Set meta data on a container
53	<code>--label-file</code>	Read in a line delimited file of labels
54	<code>--link</code>	Add link to another container
55	<code>--link-local-ip</code>	Container IPv4/IPv6 link-local addresses
56	<code>--log-driver</code>	Logging driver for the container
57	<code>--log-opt</code>	Log driver options
58	<code>--mac-address</code>	Container MAC address (e.g., 92:d0:c6:0a:29:33)
59	<code>--memory , -m</code>	Memory limit
60	<code>--memory-reservation</code>	Memory soft limit
61	<code>--memory-swap</code>	Swap limit equal to memory plus swap: '-1' to enable unlimited swap

	Docker Run Argument	Description
62	<code>--memory-swappiness</code>	Tune container memory swappiness (0 to 100)
63	<code>--name</code>	Assign a name to the container
64	<code>--net</code>	Connect a container to a network
65	<code>--net-alias</code>	Add network-scoped alias for the container
66	<code>--network</code>	Connect a container to a network
67	<code>--network-alias</code>	Add network-scoped alias for the container
68	<code>--no-healthcheck</code>	Disable any container-specific HEALTHCHECK
69	<code>--oom-kill-disable</code>	Disable OOM Killer
70	<code>--oom-score-adj</code>	Tune host's OOM preferences (-1000 to 1000)
71	<code>--pid</code>	PID namespace to use
72	<code>--pids-limit</code>	Tune container pids limit (set -1 for unlimited)
73	<code>--platform</code>	API 1.32+ <a href="https://docs.docker.com/engine/api/v1.32/">&lt;https://docs.docker.com/engine/api/v1.32/&gt;</a> <code>_Set platform</code> if server is multi-platform capable
74	<code>--privileged</code>	Give extended privileges to this container

	Docker Run Argument	Description
75	<code>--publish , -p</code>	Publish a container's port(s) to the host
76	<code>--publish-all , -P</code>	Publish all exposed ports to random ports
77	<code>--pull</code>	Pull image before running ("always" "never")
78	<code>--read-only</code>	Mount the container's root filesystem as read only
79	<code>--restart</code>	Restart policy to apply when a container exits
80	<code>--rm</code>	Automatically remove the container when it exits
81	<code>--runtime</code>	Runtime to use for this container
82	<code>--security-opt</code>	Security Options
83	<code>--shm-size</code>	Size of /dev/shm
84	<code>--sig-proxy</code>	Proxy received signals to the process
85	<code>--stop-timeout</code>	API 1.25+ < <a href="https://docs.docker.com/engine/api/v1.25/">https://docs.docker.com/engine/api/v1.25/</a> >_Timeout (in seconds) to stop a container
86	<code>--storage-opt</code>	Storage driver options for the container
87	<code>--sysctl</code>	Sysctl options

	Docker Run Argument	Description
88	<code>--tmpfs</code>	Mount a tmpfs directory
89	<code>--tty , -t</code>	Allocate a pseudo-TTY
90	<code>--ulimit</code>	Ulimit options
91	<code>--usersns</code>	User namespace to use
92	<code>--uts</code>	UTS namespace to use
93	<code>--volume , -v</code>	Bind mount a volume
94	<code>--volume-driver</code>	Optional volume driver for the container
95	<code>--volumes-from</code>	Mount volumes from the specified container(s)

If you run a simulation job with the preceding runtime configurations, AWS RoboMaker **WILL** fail your simulation with a 4XX error code.

## Creating images to run GPU applications

AWS RoboMaker GPU simulation jobs support CUDA, OpenGL, OpenCL and Vulkan API access. Therefore, the application using these APIs should have the corresponding drivers installed in their images.

### Note

We recommend using Nvidia base images to get the OpenGL APIs. The example Dockerfile used in the tutorials covers only `nvidia/opengl:1.0-glvnd-runtime-ubuntu20.04` which provides OpenGL support. Refer to the Nvidia documentation to find container images that support CUDA, Vulkan, and OpenCL.

To use DCV display with GPU rendering, you must install `nice-dcv-gl`. Note that X0 is the system's Xorg process that talks to the GPU. X1 and X2 are instead XDCV processes. When you

start an OpenGL application on X1 or X2, `nice-dcv-gl` takes care of redirecting the calls and performing the rendering on X0, where the GPU is available.

To install `nice-dcv-gl`, download the archive, extract it, and install the `nice-dcv-gl` package following the DCV public documentation. See [Install the NICE DCV Server on Linux](#).

The following example demonstrates Dockerfile installing `nice-dcv-gl_2021.2` on a `ubuntu18.04` base image.

```
FROM nvidia/opengl:1.0-glvnd-runtime-ubuntu20.04

ENV DEBIAN_FRONTEND="noninteractive"

RUN apt-get update && apt-get install -y --no-install-recommends \
    ca-certificates \
    gnupg2 \
    wget

RUN wget https://d1uj6qtbmh3dt5.cloudfront.net/NICE-GPG-KEY && gpg --import NICE-GPG-KEY && \
    wget https://d1uj6qtbmh3dt5.cloudfront.net/2021.2/Servers/nice-dcv-2021.2-11048-ubuntu1804-x86_64.tgz && \
    tar xvzf nice-dcv-2021.2-11048-ubuntu1804-x86_64.tgz && \
    cd nice-dcv-2021.2-11048-ubuntu1804-x86_64 && \
    apt install -y ./nice-dcv-gl_2021.2.944-1_amd64.ubuntu1804.deb
```

For detailed instructions on building a GPU application, see [Running a GPU sample application with ROS2 Foxy and Gazebo 11](#).

## Creating images to run the hello world sample application

You can use the Hello World sample application that we provide to help you understand how to create and run your simulation and robot applications. In the following sections, we show you how to create and run images for the following development environments:

- ROS Melodic and Gazebo 9
- ROS 2 Foxy and Gazebo 11

ROS is the robot operating system used for your robot applications. Gazebo is the operating system for your simulation applications. AWS RoboMaker uses both software suites to use container images and provide validation checks.



The tutorials guide you through using AWS RoboMaker container images to set up the Hello World robot and simulation applications. The Hello World applications are example applications that help you understand how to work with AWS RoboMaker.

For each tutorial, you create images for both your robot and simulation applications. You can run the images locally to test how they work. If your simulations work properly, you can push them to Amazon ECR and run simulation jobs in the cloud. For more information about simulation jobs, see [Simulation with AWS RoboMaker](#).

## Running a sample application with ROS 2 Foxy and Gazebo 11

The following tutorial shows you how to use container images to develop with ROS 2 Foxy and Gazebo 11, by creating and running the Hello World robot application and simulation application. You can get the sample application to work by running the commands described in this document.

For this tutorial, we create and use three container images. The following shows the directory structure that we use for this example application.

```
### HelloWorldSampleAppROS2FoxyGazebo11 // Base Image
#   ### Dockerfile
### HelloWorldSampleAppROS2FoxyGazebo11RobotApp // Image for Robot App
#   ### Dockerfile
#   ### robot-entrypoint.sh
### HelloWorldSampleAppROS2FoxyGazebo11SimApp // Image for Simulation App
#   ### Dockerfile
#   ### simulation-entrypoint.sh
```

Each Dockerfile has the instructions needed to build each image;

- The Dockerfile for the base image has the commands to set up ROS and Gazebo.
- The Dockerfile for the robot application has the commands to set up the Hello World robot application.
- The Dockerfile for the simulation application has the commands to set up the Hello World simulation application.

Both the robot application and the simulation application have an entrypoint script. These scripts source the environments for their respective applications. They set up the path for you to run commands to start your robot and simulation applications.

## Creating a base image

To create a base image, you save the commands to create your environment in a Dockerfile. You then build the Dockerfile.

- Save the following commands in a Dockerfile.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
FROM ros:foxy

ENV DEBIAN_FRONTEND noninteractive

RUN apt-get clean
RUN apt-get update && apt-get install -y \
    lsb \
    unzip \
    wget \
    curl \
    sudo \
    python3-vcstool \
    python3-rosinstall \
    python3-colcon-common-extensions \
    ros-foxy-rviz2 \
    ros-foxy-rqt \
    ros-foxy-rqt-common-plugins \
    devilspie \
    xfce4-terminal

RUN wget https://packages.osrfoundation.org/gazebo.key -O - | sudo apt-key add -; \
    sh -c 'echo "deb http://packages.osrfoundation.org/gazebo/ubuntu-stable
`lsb_release -cs` main" > /etc/apt/sources.list.d/gazebo-stable.list'
RUN apt-get update && apt-get install -y gazebo11

ENV QT_X11_NO_MITSHM=1

ARG USERNAME=robomaker
RUN groupadd $USERNAME
RUN useradd -ms /bin/bash -g $USERNAME $USERNAME
RUN sh -c 'echo "$USERNAME ALL=(root) NOPASSWD:ALL" >> /etc/sudoers'
USER $USERNAME

RUN sh -c 'cd /home/$USERNAME'
```

```
# Download and build our Robot and Simulation application
RUN sh -c 'mkdir -p /home/robomaker/workspace'
RUN sh -c 'cd /home/robomaker/workspace && wget https://github.com/aws-robotics/aws-robomaker-sample-application-helloworld/archive/3527834.zip && unzip 3527834.zip && mv aws-robomaker-sample-application-helloworld-3527834771373beff0ed3630c13479567db4149e aws-robomaker-sample-application-helloworld-ros2'
RUN sh -c 'cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros2'

RUN sudo rosdep fix-permissions
RUN rosdep update
```

After you've created the Dockerfile, build it using the following commands on your terminal.

```
cd ../HelloWorldSampleAppROS2FoxyGazebo11
docker build -t helloworldsampleappros2foxygazebo11:latest .
```

Building the base image installs ROS 2 Foxy and Gazebo 11. You need both libraries installed to successfully run your applications.

### Creating an Image for the Robot Application

After you've created the base image, you can create the image for your robot application. You save the following script in a Dockerfile and build it. This script downloads the Hello World robot application and sets it up.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
FROM helloworldsampleappros2foxygazebo11:latest

# Build the Robot application
RUN cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros2/robot_ws && \
/bin/bash -c "source /opt/ros/foxy/setup.bash && vcs import < .rosinstall && rosdep install --rosdistro foxy --from-paths src --ignore-src -r -y && colcon build"
```

```

COPY robot-entrypoint.sh /home/robomaker/robot-entrypoint.sh
RUN sh -c 'sudo chmod +x /home/robomaker/robot-entrypoint.sh'
RUN sh -c 'sudo chown robomaker:robomaker /home/robomaker/robot-entrypoint.sh'

CMD ros2 launch hello_world_robot rotate.launch.py
ENTRYPOINT [ "/home/robomaker/robot-entrypoint.sh" ]

```

The following command creates the image for the robot application from the Dockerfile.

```

cd HelloWorldSampleAppROS2FoxyGazebo11RobotApp/
HelloWorldSampleAppROS2FoxyGazebo11RobotApp
docker build -t helloworldsampleappros2foxygazebo11robotapp:latest .

```

The following are the contents of the script that you can save as `robot-entrypoint.sh`. This script sources the environment for the robot application.

```

#!/bin/bash

if [ ! -z $GAZEBO_MASTER_URI ]; then
    tmp_GAZEBO_MASTER_URI=$GAZEBO_MASTER_URI
fi

cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros2/robot_ws
source /opt/ros/foxy/setup.bash
source /usr/share/gazebo-11/setup.sh
source ./install/setup.sh

if [ ! -z $tmp_GAZEBO_MASTER_URI ]; then
    export GAZEBO_MASTER_URI=$tmp_GAZEBO_MASTER_URI
    unset tmp_GAZEBO_MASTER_URI
fi

printenv

exec "${@:1}"

```

## Creating an image for the simulation application

After you've created the base image and the image for the robot application, you can create the image for your simulation application. You save the following script in a Dockerfile and build it. This script downloads the Hello World robot application and sets it up.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
FROM helloworldsampleappros2foxygazebo11:latest

# Build the Simulation application
RUN cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros2/
simulation_ws && \
/bin/bash -c "source /opt/ros/foxy/setup.bash && vcs import < .rosinstall && rosdep
install --rosdistro foxy --from-paths src --ignore-src -r -y && colcon build"

COPY simulation-entrypoint.sh /home/robomaker/simulation-entrypoint.sh

RUN sh -c 'sudo chmod +x /home/robomaker/simulation-entrypoint.sh'
RUN sh -c 'sudo chown robomaker:robomaker /home/robomaker/simulation-entrypoint.sh'

CMD ros2 launch hello_world_simulation empty_world.launch.py
ENTRYPOINT [ "/home/robomaker/simulation-entrypoint.sh" ]

```

The following command creates the image.

```

cd HelloWorldSampleAppROS2FoxyGazebo11SimApp/HelloWorldSampleAppROS2FoxyGazebo11SimApp
docker build -t helloworldsampleappros2foxygazebo11simapp:latest .

```

The following are the contents of the script that you can save as `simulation-entrypoint.sh`. This script sources the environment for the simulation application.

```

#!/bin/bash

if [ ! -z $GAZEBO_MASTER_URI ]; then
    tmp_GAZEBO_MASTER_URI=$GAZEBO_MASTER_URI
fi

cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros2/
simulation_ws
source /opt/ros/foxy/setup.bash
source /usr/share/gazebo-11/setup.sh
source ./install/setup.sh

if [ ! -z $tmp_GAZEBO_MASTER_URI ]; then
    export GAZEBO_MASTER_URI=$tmp_GAZEBO_MASTER_URI

```

```
unset tmp_GAZEBO_MASTER_URI
fi

printenv

exec "${@:1}"
```

## Running the application and pushing it to Amazon ECR

After you've created your images, make sure that they run properly in your local Linux environment. After you've checked that your image runs, you can push your Docker image to Amazon ECR and create a simulation job.

The following commands give you the ability to run the Hello World application in your local Linux environment.

```
docker run -it -e DISPLAY -v /tmp/.X11-unix:/tmp/.X11-unix/ --name robot_app \
-u robomaker -e ROBOMAKER_GAZEBO_MASTER_URI=http://localhost:5555 \
-e ROBOMAKER_ROS_MASTER_URI=http://localhost:11311 \
helloworldsampleappros2foxygazebo11robotapp:latest
```

```
docker run -it -e DISPLAY -v /tmp/.X11-unix:/tmp/.X11-unix/ --name sim_app \
-u robomaker -e ROBOMAKER_GAZEBO_MASTER_URI=http://localhost:5555 \
-e ROBOMAKER_ROS_MASTER_URI=http://localhost:11311 \
helloworldsampleappros2foxygazebo11simapp:latest
```

When you run the robot application and simulation application containers, you can visualize the simulation using the Gazebo GUI tool. Use the following commands to:

1. Connect to your container running the simulation application.
2. Visualize your application by running the Gazebo Graphical User Interface (GUI).

```
# Enable access to X server to launch Gazebo from docker container
$ xhost +

# Check that the robot_app and sim_app containers are running. The command should list
both containers
$ docker container ls
```

```
# Connect to the sim app container
$ docker exec -it sim_app bash

# Launch Gazebo from within the container
$ /home/robomaker/simulation-entrypoint.sh ros2 launch gazebo_ros gzclient.launch.py
```

You can add tags to your images. The following commands give you the ability to tag your images.

```
docker tag helloworldsampleappros2foxygazebo11robotapp:latest accountID.dkr.ecr.us-west-2.amazonaws.com/helloworldsampleappros2foxygazebo11robotapp:latest
```

```
docker tag helloworldsampleappros2foxygazebo11simapp:latest accountID.dkr.ecr.us-west-2.amazonaws.com/helloworldsampleappros2foxygazebo11simapp:latest
```

After you've verified that the application is working properly, you can push to Amazon ECR using the following commands.

```
aws ecr get-login-password --region us-west-2 | docker login --username AWS --password-stdin accountID.dkr.ecr.us-west-2.amazonaws.com
docker push accountID.dkr.ecr.us-west-2.amazonaws.com/helloworldsampleappros2foxygazebo11robotapp:latest
docker push accountID.dkr.ecr.us-west-2.amazonaws.com/helloworldsampleappros2foxygazebo11simapp:latest
```

You can then run a simulation job on the image. For more information about simulation jobs, see [Simulation with AWS RoboMaker](#).

## Running a sample application with ROS Melodic and Gazebo 9

The following tutorial shows you how to use container images to develop with ROS and Gazebo 9 by creating and running the Hello World robot application and simulation application. You can get the sample application to work by running the commands described in this document.

For this tutorial, we create and use three container images. The following shows the directory structure that we use for this example application.

```
### HelloWorldSampleAppROSMelodicGazebo9 // Base Image
#   ### Dockerfile
### HelloWorldSampleAppROSMelodicGazebo9RobotApp // Image for Robot App
#   ### Dockerfile
#   ### robot-entrypoint.sh
### HelloWorldSampleAppROSMelodicGazebo9SimApp // Image for Simulation App
#   ### Dockerfile
#   ### simulation-entrypoint.sh
```

Each Dockerfile has the instructions needed to build each image.

- The Dockerfile for the base image has the commands to set up ROS and Gazebo.
- The Dockerfile for the robot application has the commands to set up the Hello World robot application.
- The Dockerfile for the simulation application has the commands to set up the Hello World simulation application.

Both the robot application and the simulation application have entrypoint scripts. These scripts source the environments for their respective applications. They set up the path for you to run commands that give you the ability to run your robot and simulation applications.

## Creating a base image

To create a base image, save the commands in the example to create your environment in a Dockerfile. Then, build the Dockerfile.

1. Save the following commands in a Dockerfile.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
FROM ros:melodic

ENV DEBIAN_FRONTEND noninteractive

RUN apt-get clean
RUN apt-get update && apt-get install -y \
    lsb \
    unzip \
    wget \
    curl \
```



```
sudo \  
python-vcstool \  
python-rosinstall \  
python3-colcon-common-extensions \  
ros-melodic-rviz \  
ros-melodic-rqt \  
ros-melodic-rqt-common-plugins \  
devilspie \  
xfce4-terminal \  
ros-melodic-gazebo-ros-pkgs \  
ros-melodic-gazebo-ros-control \  
ros-melodic-turtlebot3  
  
ENV QT_X11_NO_MITSHM=1  
  
ARG USERNAME=robomaker  
RUN groupadd $USERNAME  
RUN useradd -ms /bin/bash -g $USERNAME $USERNAME  
RUN sh -c 'echo "$USERNAME ALL=(root) NOPASSWD:ALL" >> /etc/sudoers'  
USER $USERNAME  
  
RUN sh -c 'cd /home/$USERNAME'  
  
# Download and build our Robot and Simulation application  
RUN sh -c 'mkdir -p /home/robomaker/workspace'  
RUN sh -c 'cd /home/robomaker/workspace && wget https://github.com/aws-robotics/  
aws-robomaker-sample-application-helloworld/archive/ros1.zip && unzip ros1.zip'  
RUN sh -c 'cd /home/robomaker/workspace/aws-robomaker-sample-application-  
helloworld-ros1'  
  
RUN sudo rosdep fix-permissions  
RUN rosdep update
```

2. After you've created the Dockerfile, build it using the following commands on your terminal.

```
cd ../HelloWorldSampleAppROSMelodicGazebo9  
docker build -t helloworldsampleapprosmelodicgazebo9:latest .
```

Building the base image installs ROS Melodic and Gazebo 9. You need both libraries installed to successfully run your applications.

## Creating an image for the robot application

After you've created the base image, create the image for your robot application.

1. Save the following script in a Dockerfile and build it. This script downloads the Hello World robot application and sets it up.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
FROM helloworldsampleapprosmelodicgazebo9:latest

# Build the Robot application
RUN cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros1/robot_ws && \
  /bin/bash -c "source /opt/ros/melodic/setup.bash && vcs import < .rosinstall && rosdep install --rosdistro melodic --from-paths src --ignore-src -r -y && colcon build"

COPY robot-entrypoint.sh /home/robomaker/robot-entrypoint.sh

RUN sh -c 'sudo chmod +x /home/robomaker/robot-entrypoint.sh'
RUN sh -c 'sudo chown robomaker:robomaker /home/robomaker/robot-entrypoint.sh'

CMD roslaunch hello_world_robot rotate.launch
ENTRYPOINT [ "/home/robomaker/robot-entrypoint.sh" ]
```

2. Use the following command to create the image for the robot application from the Dockerfile.

```
cd HelloWorldSampleAppROSMelodicGazebo9RobotApp/
HelloWorldSampleAppROSMelodicGazebo9RobotApp
docker build -t helloworldsampleapprosmelodicgazebo9robotapp:latest image/.
```

3. The following are the contents of the script that you can save as `robot-entrypoint.sh`. This script sources the environment for the robot application.

```
#!/bin/bash

if [ ! -z $GAZEBO_MASTER_URI ]; then
    tmp_GAZEBO_MASTER_URI=$GAZEBO_MASTER_URI
fi

cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros1/robot_ws
```

```

source /opt/ros/melodic/setup.bash
source /usr/share/gazebo-9/setup.sh
source ./install/setup.sh

if [ ! -z $tmp_GAZEBO_MASTER_URI ]; then
    export GAZEBO_MASTER_URI=$tmp_GAZEBO_MASTER_URI
    unset tmp_GAZEBO_MASTER_URI
fi

printenv

exec "${@:1}"

```

## Creating an image for the simulation application

After you've created the base image and the image for the robot application, you can create the image for your simulation application.

1. Save the following script in a Dockerfile and build it. This script downloads the Hello World robot application and sets it up.

```

# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
FROM helloworldsampleapprosmelodicgazebo9:latest

# Build the Simulation application
RUN cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros1/
simulation_ws && \
    /bin/bash -c "source /opt/ros/melodic/setup.bash && vcs import < .rosinstall &&
rosdep install --rosdistro melodic --from-paths src --ignore-src -r -y && colcon
build"

COPY simulation-entrypoint.sh /home/robomaker/simulation-entrypoint.sh

RUN sh -c 'sudo chmod +x /home/robomaker/simulation-entrypoint.sh'
RUN sh -c 'sudo chown robomaker:robomaker /home/robomaker/simulation-entrypoint.sh'

CMD roslaunch hello_world_simulation empty_world.launch
ENTRYPOINT [ "/home/robomaker/simulation-entrypoint.sh" ]

```

2. Save the following `simulation-entrypoint.sh` script. This script sources the environment for the simulation application.

```
#!/bin/bash

if [ ! -z $GAZEBO_MASTER_URI ]; then
    tmp_GAZEBO_MASTER_URI=$GAZEBO_MASTER_URI
fi

cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros1/
simulation_ws
source /opt/ros/melodic/setup.bash
source /usr/share/gazebo-9/setup.sh
source ./install/setup.sh

if [ ! -z $tmp_GAZEBO_MASTER_URI ]; then
    export GAZEBO_MASTER_URI=$tmp_GAZEBO_MASTER_URI
    unset tmp_GAZEBO_MASTER_URI
fi

printenv

exec "${@:1}"
```

## Running the application and pushing it to ECR

After you've created your images, make sure that they run properly in your local Linux environment. After you've checked that Docker image runs, you can push it to Amazon ECR and create a simulation job.

1. Use the following commands to run the Hello World application in your local Linux environment.

```
docker run -it -e DISPLAY -v /tmp/.X11-unix/:/tmp/.X11-unix/ \
-u robomaker -e ROBOMAKER_GAZEBO_MASTER_URI=http://localhost:5555 \
-e ROBOMAKER_ROS_MASTER_URI=http://localhost:11311 \
helloworldsampleapprosmelodicgazebo9robotapp:latest
```

```
docker run -it -e DISPLAY -v /tmp/.X11-unix/:/tmp/.X11-unix/ \
-u robomaker -e ROBOMAKER_GAZEBO_MASTER_URI=http://localhost:5555 \
-e ROBOMAKER_ROS_MASTER_URI=http://localhost:11311 \
helloworldsampleapprosmelodicgazebo9simapp:latest
```

2. Run the robot application and simulation application containers to visualize the simulation using the Gazebo GUI tool. Use the following commands to:
  1. Connect to your container running the simulation application.
  2. Visualize your application by running the Gazebo Graphical User Interface (GUI).

```
# Enable access to X server to launch Gazebo from docker container
$ xhost +

# Check that the robot_app and sim_app containers are running. The command should
  list both containers
$ docker container ls

# Connect to the sim app container
$ docker exec -it sim_app bash

# Launch Gazebo from within the container
$ rosrun gazebo_ros gzclient
```

3. Add tags to your images to keep them organized. Use the following commands to tag your images.

```
docker tag
  helloworldsampleapprosmelodicgazebo9robotapp:latest accountID.dkr.ecr.us-
west-2.amazonaws.com/helloworldsampleapprosmelodicgazebo9robotapp:latest
```

```
docker tag helloworldsampleapprosmelodicgazebo9simapp:latest accountID.dkr.ecr.us-
west-2.amazonaws.com/helloworldsampleapprosmelodicgazebo9simapp:latest
```

4. After you've verified that the application is working properly, you can push to Amazon ECR using the following commands.

```
aws ecr get-login-password --region us-west-2 | docker login --username AWS --
password-stdin accountID.dkr.ecr.us-west-2.amazonaws.com
docker push accountID.dkr.ecr.us-west-2.amazonaws.com/
helloworldsampleapprosmelodicgazebo9robotapp:latest
docker push accountID.dkr.ecr.us-west-2.amazonaws.com/
helloworldsampleapprosmelodicgazebo9simapp:latest
```

You can then run a simulation job on the image. For more information about simulation jobs, see [Simulation with AWS RoboMaker](#).

## Running a GPU sample application with ROS2 Foxy and Gazebo 11

This tutorial explains how to use GPU drivers within container images to develop with ROS 2 Foxy and Gazebo 11 by creating and running the Hello World robot application and simulation application using three container images outlined in the following example.

```
### SampleGPUBaseApp // Base Image
#   ### Dockerfile
### SampleGPURobotApp // Image for Robot App
#   ### Dockerfile
#   ### robot-entrypoint.sh
### SampleGPUSimulationApp // Image for Simulation App
#   ### Dockerfile
#   ### simulation-entrypoint.sh
```

Each Dockerfile contains the instructions needed to build each image.

- The Dockerfile for the base image includes commands to set up ROS, Gazebo and GPU drivers.
- The Dockerfile for the robot application includes the commands to set up the Hello World robot application.
- The Dockerfile for the simulation application includes the commands to set up the Hello World simulation application.

Both the robot application and the simulation application have an entrypoint script. These scripts source the environments for their respective applications and set up the path for you to run commands to start your robot and simulation applications.

### Creating a base GPU image

The following Dockerfile contains the commands to create a base image from NVIDIA OpenGL and install DCV.

- Save the following commands in the Dockerfile in the `SampleGPUBaseApp` directory.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
FROM nvidia/opengl:1.0-glvnd-runtime-ubuntu20.04

ENV DEBIAN_FRONTEND="noninteractive"
ENV QT_X11_NO_MITSHM=1

RUN apt-get clean
RUN apt-get update && apt-get install -y --no-install-recommends \
    ca-certificates \
    devilspie \
    gnupg2 \
    mesa-utils \
    sudo \
    unzip \
    wget \
    xfce4-terminal

RUN wget https://d1uj6qtbmh3dt5.cloudfront.net/NICE-GPG-KEY && gpg --import NICE-GPG-KEY && \
    wget https://d1uj6qtbmh3dt5.cloudfront.net/2021.2/Servers/nice-dcv-2021.2-11048-ubuntu1804-x86_64.tgz && \
    tar xvzf nice-dcv-2021.2-11048-ubuntu1804-x86_64.tgz && \
    cd nice-dcv-2021.2-11048-ubuntu1804-x86_64 && \
    apt install -y ./nice-dcv-gl_2021.2.944-1_amd64.ubuntu1804.deb

RUN apt update && apt -y install locales && \
    locale-gen en_US en_US.UTF-8 && \
    update-locale LC_ALL=en_US.UTF-8 LANG=en_US.UTF-8

ENV LANG=en_US.UTF-8

RUN apt-get update && apt-get install -y --no-install-recommends curl lsb-release

RUN curl -sSL https://raw.githubusercontent.com/ros/rosdistro/master/ros.key -o /usr/share/keyrings/ros-archive-keyring.gpg && \
    curl -s https://raw.githubusercontent.com/ros/rosdistro/master/ros.asc | apt-key add - && \
    echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/ros-archive-keyring.gpg] http://packages.ros.org/ros2/ubuntu $(lsb_release -cs) main" | tee /etc/apt/sources.list.d/ros2.list > /dev/null && \
    apt update && \
    apt install -y ros-foxy-desktop && \
```

```
    /bin/bash -c "source /opt/ros/foxy/setup.bash"

RUN apt -y install ros-foxy-gazebo-ros-pkgs

RUN apt-key adv --fetch-keys 'http://packages.osrfoundation.org/gazebo.key' && \
    apt update && \
    apt install -y python3-rosdep git

RUN if [ ! -f "/etc/ros/rosdep/sources.list.d/20-default.list" ]; then \
    rosdep init; \
fi

RUN rosdep update

RUN apt-get install -y python3-apt python3-pip python3-vcstool python3-testresources

RUN pip3 install -U pytest setuptools colcon-ros-bundle

RUN useradd --create-home robomaker && \
    sh -c 'echo "robomaker ALL=(root) NOPASSWD:ALL" >> /etc/sudoers'

RUN sh -c 'mkdir -p /home/robomaker/workspace' && \
    sh -c 'cd /home/robomaker/workspace && wget https://github.com/aws-robotics/
aws-robomaker-sample-application-helloworld/archive/ros2.zip && unzip ros2.zip'
```

After you've created the Dockerfile, build it using the following commands on your terminal.

```
cd SampleGPUBaseApp
docker build -t samplegpubaseapp:latest .
```

Building the base image installs ROS 2 Foxy, Gazebo 11, NVIDIA OpenGL, and NICE-DCV.

## Creating an Image for the Robot Application

After you've created the base image, you can create the image for your robot application. Save the following script in the Dockerfile in the SampleGPURobotApp directory and build it. This script downloads the Hello World robot application and sets it up.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
FROM samplegpubaseapp:latest
```



```
# Build the Robot application
RUN cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros2/
robot_ws && \
/bin/bash -c "source /opt/ros/foxy/setup.bash && vcs import < .rosinstall && rosdep
install --rosdistro foxy --from-paths src --ignore-src -r -y && colcon build"

COPY robot-entrypoint.sh /home/robomaker/robot-entrypoint.sh
RUN sh -c 'sudo chmod +x /home/robomaker/robot-entrypoint.sh'
RUN sh -c 'sudo chown robomaker:robomaker /home/robomaker/robot-entrypoint.sh'

CMD ros2 launch hello_world_robot rotate.launch.py
ENTRYPOINT [ "/home/robomaker/robot-entrypoint.sh" ]
```

The following are the contents of the script that you save as `robot-entrypoint.sh`. This script sources the environment for the robot application.

```
#!/bin/bash
cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros2/robot_ws
source /opt/ros/foxy/setup.bash
source /usr/share/gazebo-11/setup.sh
source ./install/setup.sh
printenv

exec "${@:1}"
```

The following command creates the image for the robot application from the Dockerfile.

```
cd SampleGPURobotApp
docker build -t samplegpurobotapp:latest .
```

## Creating an Image for the Simulation Application

### Creating an Image for the Simulation Application

After you've created the base image and the image for the robot application, you can create the image for your simulation application. You save the following script in a Dockerfile in the `SampleGPUSimulationApp` directory and then build it. This script downloads the Hello World simulation application and sets it up.

```
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: MIT-0
```

```
FROM samplegpubaseapp:latest

# Build the Simulation application
RUN cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros2/
simulation_ws && \
/bin/bash -c "source /opt/ros/foxy/setup.bash && vcs import < .rosinstall && rosdep
install --rosdistro foxy --from-paths src --ignore-src -r -y && colcon build"

COPY simulation-entrypoint.sh /home/robomaker/simulation-entrypoint.sh

RUN sh -c 'sudo chmod +x /home/robomaker/simulation-entrypoint.sh'
RUN sh -c 'sudo chown robomaker:robomaker /home/robomaker/simulation-entrypoint.sh'

CMD ros2 launch hello_world_simulation empty_world.launch.py
ENTRYPOINT [ "/home/robomaker/simulation-entrypoint.sh" ]
```

The following are the contents of the script that you save as `simulation-entrypoint.sh`. This script sources the environment for the simulation application.

```
#!/bin/bash
if [ ! -z $GAZEBO_MASTER_URI ]; then
    tmp_GAZEBO_MASTER_URI=$GAZEBO_MASTER_URI
fi

cd /home/robomaker/workspace/aws-robomaker-sample-application-helloworld-ros2/
simulation_ws
source /opt/ros/foxy/setup.bash
source /usr/share/gazebo-11/setup.sh

if [ ! -z $tmp_GAZEBO_MASTER_URI ]; then
    export GAZEBO_MASTER_URI=$tmp_GAZEBO_MASTER_URI
    unset tmp_GAZEBO_MASTER_URI
fi

source ./install/setup.sh
printenv

exec "${@:1}"
```

The following command creates the image.

```
cd SampleGPUSimulationApp
```

```
docker build -t samplegpusimulationapp:latest .
```

## Running the application and pushing it to Amazon ECR

After you've created your images, make sure they run properly in your local Linux environment. After you've checked that your image runs, you can push your Docker image to Amazon ECR and create a simulation job.

The following commands give you the ability to run the Hello World application in your local Linux environment.

```
docker run -it -e DISPLAY -v /tmp/.X11-unix:/tmp/.X11-unix/ --name gpu_robot_app \
-u robomaker -e ROBOMAKER_GAZEBO_MASTER_URI=http://localhost:5555 \
-e ROBOMAKER_ROS_MASTER_URI=http://localhost:11311 \
samplegपुरobotapp:latest

docker run -it -e DISPLAY -v /tmp/.X11-unix:/tmp/.X11-unix/ --name gpu_sim_app \
-u robomaker -e ROBOMAKER_GAZEBO_MASTER_URI=http://localhost:5555 \
-e ROBOMAKER_ROS_MASTER_URI=http://localhost:11311 \
samplegpusimulationapp:latest
```

When you run the robot application and simulation application containers, you can visualize the simulation using the Gazebo GUI tool. Use the following commands to:

- Connect to your container running the simulation application.
- Visualize your application by running the Gazebo Graphical User Interface (GUI).

```
# Enable access to X server to launch Gazebo from docker container
$ xhost +

# Check that the robot_app and sim_app containers are running. The command should list
both containers
$ docker container ls

# Connect to the sim app container
$ docker exec -it gpu_sim_app bash

# Launch Gazebo from within the container
$ /home/robomaker/simulation-entrypoint.sh ros2 launch gazebo_ros gzclient.launch.py
```

You can add tags to your images. The following commands give you the ability to tag your images.

```
docker tag samplegpurobotapp:latest accountID.dkr.ecr.us-west-2.amazonaws.com/  
samplegpurobotapp:latest
```

```
docker tag samplegpusimulationapp:latest accountID.dkr.ecr.us-west-2.amazonaws.com/  
samplegpusimulationapp:latest
```

After you've verified that the application is working properly, you can push it to Amazon ECR using the following commands.

```
aws ecr get-login-password --region us-west-2 | docker login --username AWS --password-  
stdin accountID.dkr.ecr.us-west-2.amazonaws.com  
docker push accountID.dkr.ecr.us-west-2.amazonaws.com/samplegpurobotapp:latest  
docker push accountID.dkr.ecr.us-west-2.amazonaws.com/samplegpusimulationapp:latest
```

You can now run a simulation job with GPU Compute using these images. For more information about simulation jobs, see [Simulation with AWS RoboMaker](#).

# Simulation with AWS RoboMaker

An AWS RoboMaker simulation job is a pairing of a robot application and a simulation application running in the cloud. While a simulation job is running, you can interact with it using graphical tools and a terminal to visualize sensor data and control components of the robot. The following topics describe how to run, configure, manage, log, and batch AWS RoboMaker simulation jobs.

## Topics

- [Running a simulation](#)
- [Configuring a simulation](#)
- [Managing a simulation](#)
- [Logging a simulation](#)
- [Batching simulations](#)

## Running a simulation

To get started running a simulation, use the following AWS CLI commands to describe your applications. These commands provide output that you can examine to confirm you're ready to proceed to creating a simulation job.

The following command retrieves data associated with your robot application.

```
aws robomaker describe-robot-application --application YOUR-ROBOT-APP-ARN
```

The output for `describe-robot-application` contains the following data.

```
{
  "arn": "YOUR-ROBOT-APP-ARN",
  "name": "YOUR-ROBOT-APP-NAME",
  ... # Removed extra data for clarity
  "robotSoftwareSuite": {
    "name": "General"
  },
}
```

```
... # Removed extra data for clarity

"environment": {
  "uri": "YOUR-ROBOT-APP-ECR-URI"
}
}
```

The next command retrieves the data associated with your simulation application.

```
aws robomaker describe-simulation-application --application YOUR-SIM-APP-ARN
```

The output for the `describe-simulation-application` contains the following data.

```
{
  "arn": "YOUR-SIM-APP-ARN",
  "name": "YOUR-SIM-APP-NAME",
  ... # Removed extra data for clarity

  "simulationSoftwareSuite": {
    "name": "SimulationRuntime"
  },
  "robotSoftwareSuite": {
    "name": "General"
  },
  ... # Removed extra data for clarity

  "environment": {
    "uri": "YOUR-SIM-APP-ECR-URI"
  }
}
```

Save the returned values for `YOUR-ROBOT-APP-ARN` and `YOUR-SIM-APP-ARN`. You need them both to submit your simulation job. If you need to import WorldForge assets into your simulation job, use the [DataSource](#) API. This allows you to import world assets from the Amazon S3 output directory of the world export job into a destination of your choosing within your simulation job container. For more information, see [Using exported worlds in simulation](#).

To submit your simulation job, create a JSON file in your working directory named `create_simulation_job.json`. Copy, paste, and edit the strings identified by red italic

text to contain YOUR-IAM-ROLE-ARN, YOUR-ROBOT-APP-ARN, and YOUR-SIM-APP-ARN. The roslaunch command, TurtleBot environment variable, and tool settings below are specific to the [hello world sample application](#). You must update these settings to your own custom values depending on the needs of your simulation job. For more information, see the [CreateSimulationJob](#) API.

```
{
  "maxJobDurationInSeconds": 3600,
  "iamRole": "IAM-ROLE-ARN",
  "robotApplications": [
    {
      "application": "YOUR-ROBOT-APP-ARN",
      "applicationVersion": "$LATEST",
      "launchConfig": {
        "environmentVariables": {
          "ROS_IP": "ROBOMAKER_ROBOT_APP_IP",
          "ROS_MASTER_URI": "http://ROBOMAKER_ROBOT_APP_IP:11311",
          "GAZEBO_MASTER_URI": "http://ROBOMAKER_SIM_APP_IP:11345"
        },
        "streamUI": false,
        "command": [
          "/bin/bash", "-c", "roslaunch hello_world_robot rotate.launch"
        ]
      },
      "tools": [
        {
          "streamUI": true,
          "name": "robot-terminal",
          "command": "/entrypoint.sh && xfce4-terminal",
          "streamOutputToCloudWatch": true,
          "exitBehavior": "RESTART"
        }
      ]
    }
  ],
  "simulationApplications": [
    {
      "application": "YOUR-SIM-APP-ARN",
      "launchConfig": {
        "environmentVariables": {
          "ROS_IP": "ROBOMAKER_SIM_APP_IP",
          "ROS_MASTER_URI": "http://ROBOMAKER_ROBOT_APP_IP:11311",
          "GAZEBO_MASTER_URI": "http://ROBOMAKER_SIM_APP_IP:11345",

```

```
        "TURTLEBOT3_MODEL": "waffle_pi"
    },
    "streamUI": true,
    "command": [
        "/bin/bash", "-c", "roslaunch hello_world_simulation
empty_world.launch --wait"
    ]
},
"tools": [
    {
        "streamUI": true,
        "name": "gzclient",
        "command": "/entrypoint.sh && gzclient",
        "streamOutputToCloudWatch": true,
        "exitBehavior": "RESTART"
    }
]
}
]
```

### Note

Take special note of the ROS\_ and GAZEBO\_ specific environment variable settings in the launchConfig object in the robot and simulation application. The ROBOMAKER\_\* string values and port numbers are required to enable your robot application container to communicate with your simulation application container.

Once you have confirmed your job settings, you can submit the job using the following command.

```
aws robomaker create-simulation-job --cli-input-json file://create_simulation_job.json
```

To confirm that your simulation is running in AWS RoboMaker, visit the **Simulation Jobs** page within the [AWS RoboMaker console](#). Look for your running job and choose it to view the details and to launch any associated tools. Congratulations, your simulation job is now running in AWS RoboMaker!



# Configuring a simulation

The following sections describe how to configure simulation jobs. For more information, see the concept that describes [Application configuration](#).

## Sections

- [Configuring a simulation job for Amazon VPC access](#)
- [Internet access for simulation jobs](#)
- [Configuring SimulationJob compute](#)
- [Configuring custom simulation tools](#)
- [Root access and system capabilities](#)

## Configuring a simulation job for Amazon VPC access

When you create resources in the Amazon Virtual Private Cloud (Amazon VPC), they cannot be read through the public internet. Example resources could be Amazon Redshift data warehouses or Amazon ElastiCache clusters. They could also be your services on an Amazon Elastic Compute Cloud instance. By default, resources in an Amazon VPC are not accessible to an AWS RoboMaker simulation job.

### Note

AWS RoboMaker runs your simulation job on an isolated network with no external connectivity. To allow your job to access resources in your Amazon VPC, you must provide VPC-specific data that includes Amazon VPC subnet IDs and security group IDs. AWS RoboMaker uses this data to set up elastic network interfaces ([ENIs](#)). ENIs help your job to connect securely to other resources in your private Amazon VPC.

AWS RoboMaker does not connect to resources within dedicated tenancy VPCs. For more information, see [Dedicated VPCs](#).

You can add Amazon VPC data to your AWS RoboMaker simulation job by using the `VpcConfig` parameter at the time you create a job (see [CreateSimulationJob](#)). The following is an AWS CLI example where a public IP is assigned.

```
aws robomaker create-simulation-job \
```

```
--output-location s3Bucket=my-bucket,s3Prefix=my-output-folder \  
--max-job-duration-in-seconds 3600 \  
--iam-role my-role-arn \  
--failure-behavior Continue \  
--robot-applications application='my-robot-application-  
arn,launchConfig={command=["roslaunch", "hello_world_robot", "rotate.launch"]}\' \  
--simulation-applications application='my-simulation-application-  
arn,launchConfig={command=["roslaunch", "hello_world_simulation",  
"empty_world.launch"]}\' \  
--vpc-config assignPublicIp=true,subnets=comma-separated-vpc-subnet-  
ids,securityGroups=comma-separated-security-group-ids
```

### Note

When a simulation job is configured to run in a VPC, it incurs an ENI penalty. Address resolution may be delayed when you try to connect to network resources.

## Internet access for simulation jobs

AWS RoboMaker uses the VPC data you provide to set up ENIs. ENIs allow your job to access VPC resources. Each ENI is assigned a private IP address from the range in the subnets you specify. The ENI is not assigned any public IP addresses by default.

If your job requires internet access (perhaps to find AWS services that do not have VPC endpoints), and you are using *private* Subnets, you can set up a NAT inside your VPC. You can use the Amazon VPC NAT gateway and request AWS RoboMaker to assign a public IP. For more information, see [NAT Gateways](#) in the *Amazon VPC User Guide*.

### Note

You cannot use an internet gateway that is directly attached to your VPC, as an internet connection requires the ENI to have public IP addresses. By default, your ENI has a private IP address.

To configure internet access when using *public* Subnets, set `assignPublicIp=true` to assign a public IP to your ENI.

If your simulation job only requires access to public AWS APIs and you want more privacy, see [AWS RoboMaker and interface VPC endpoints \(AWS PrivateLink\)](#). With this information, you can create your interface VPC endpoint and add the VPC using the [CreateSimulationJob](#) API.

## Configuring SimulationJob compute

To use GPU in your SimulationJobs, you can configure the ComputeType of the SimulationJob to use GPU Compute. You realize the following benefits when using Graphics Processing Unit (GPU)-based simulation jobs in AWS RoboMaker.

- GPU-based simulation jobs allow execution of applications that require GPU-enabled sensor plugins and high fidelity rendering and performance using OpenGL, CUDA, OpenCL, and Vulkan.
- GPU-based simulation jobs ensure that the AWS RoboMaker GUI tools have high-quality HD resolution so you can see objects in greater detail. The GUI tools experience is ideal because the GPU ensures a higher rate of frames per second.
- GPU-based simulation accelerates the simulation job completion time. With GPU, you can run complex simulation scenes without taking a performance hit on the real-time factor and the frames per second.
- GPU-based simulation jobs improve the training of reinforcement learning models.

### Compute

The Compute parameter in the CreateSimulationJob request can be used to configure which kind of Compute is needed for the SimulationJob.

### ComputeType

ComputeType specifies the type of Compute required for the job. Valid values are CPU and GPU\_AND\_CPU. The default value is CPU. If GPU\_AND\_CPU is specified, the job created can use GPU along with CPU.

### GpuUnitLimit

Using the GpuUnitLimit parameter, you can specify the number of GPU units that need to be allocated to your job. For GPU\_AND\_CPU ComputeType, it must be 1. For CPU ComputeType, it must be 0.

For information about building containers to utilize GPU, see [Creating images to run GPU applications](#).

## Configuring custom simulation tools

With AWS RoboMaker, you can configure custom tools for the applications in a simulation job. Use custom tools to interact with the simulation, as diagnostic utilities, or for other purposes. You can also configure default tools like `rqt` or `rviz` provided by AWS RoboMaker. If your simulation job is part of an automated pipeline, you can disable default tools and use fewer resources.

You can configure up to 10 custom tools. Custom tools are started after the main ROS process is started.

A custom tool configuration includes the following elements:

- **Tool name** — The name of the tool.
- **Command** — The command to invoke the tool in a bash shell. You must include the tool executable name. You can use environment variables including custom variables in your arguments. For example, to use the current simulation job ID, you can reference `AWS_ROBOMAKER_SIMULATION_JOB_ID`.
- **Exit behavior** — Determines what action is taken if the custom tool exits. If you specify `fail`, the simulation job fails. If you specify `restart`, the tool is restarted. The default is `restart`.
- **UI streaming** — Specifies whether a streaming session is configured for the tool. If `True`, AWS RoboMaker configures a connection so you can interact with the tool as it is running in the simulation. It must have a graphical user interface. The default is `false`.
- **Log behavior** — Specifies whether tool `stdout` and `stderr` is streamed to CloudWatch Logs. The default is `false`.

## Root access and system capabilities

AWS RoboMaker provides limited root (`sudo`) access to applications running in a simulation job. The following list contains significant (but not all) syscalls that are blocked.

- `acct`
- `add_key`
- `bpf`
- `clock_adjtime`
- `clock_settime`
- `clone`
- `create_module`

- delete\_module
- finit\_module
- get\_kernel\_syms
- get\_mempolicy
- init\_module
- ioperm
- iopl
- kcmp
- kexec\_file\_load
- kexec\_load
- keyctl
- lookup\_dcookie
- mbind
- mount
- move\_pages
- name\_to\_handle\_at
- nfsservctl
- open\_by\_handle\_at
- perf\_event\_open
- personality
- pivot\_root
- process\_vm\_readv
- process\_vm\_writev
- ptrace
- query\_module
- quotactl
- reboot
- request\_key
- set\_mempolicy
- setns
- settimeofday
- stime
- swapon
- swapoff
- sysfs

- `_sysctl`
- `umount`
- `umount2`
- `unshare`
- `uselib`
- `userfaultfd`
- `ustat`
- `vm86`
- `vm86old`

## Managing a simulation

The following sections describe how to create, view, cancel, clone, and restart simulation jobs.

### Sections

- [Creating a simulation job](#)
- [Viewing a simulation job](#)
- [Cancelling a simulation job](#)
- [Cloning a simulation job](#)
- [Restarting a simulation job](#)

## Creating a simulation job

Create a simulation job when you want to run your robot application in a virtual world using any simulation platform. You select the software suite name when you specify the simulation application. We currently support **General** and **SimulationRuntime** software suites.

### To create a simulation job

Follow the steps under one of the following tabs:

Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Simulation run**, and then choose **Simulation jobs**.

3. Choose **Create simulation job**.
4. On the **Simulation configuration** page, select a **simulation job duration**. Select any value between 5 minutes and 14 days.

**⚠ Important**

To learn more about how you are charged for AWS RoboMaker see [AWS RoboMaker Pricing](#).

5. Select a **Failure behavior**. Choose **Fail** to terminate the host instance if the simulation job fails. Choose **Continue** to keep the host instance so you can connect and investigate.

If you specify an optional S3 folder in a following step, it contains simulation data. It is available independent of the selected failure behavior.

6. For **IAM Role**, select a role or select **Create new role** to create one. AWS RoboMaker uses this role to access resources on your behalf. It is also used by your application to access AWS resources like Amazon Rekognition or Amazon Lex.
7. *Optional:* In **Compute**, select a simulation unit limit. Your simulation is allocated CPU and memory proportional to the supplied simulation unit limit. A simulation unit is 1 vcpu and 2GB of memory. The default is 15.
8. *Optional:* In **Output destination**, type in an Amazon S3 folder name where simulation job output is stored. Optionally, select **Create new S3 folder** to create a new Amazon S3 folder.
9. *Optional:* In **Networking**, if your robot application or simulation application accesses resources on an Amazon VPC, select the **VPC**, subnets and security groups. Select all available subnets to ensure all of your resource limits are available. For more information, see [VPCs and Subnets](#).


If you want to access the simulation job from outside of the VPC, select **Assign public IP**.

10. Optionally, under **Tags**, specify one or more tags for the simulation job. Tags are words or phrases that act as metadata for identifying and organizing your AWS resources. Each tag consists of a key and a value. You can manage tags for your simulation job on the **Simulation Job details** page.

For more about tagging, see [Using Cost Allocation Tags](#) in the *AWS Billing and Cost Management User Guide*.

11. Choose **Next**.

12. On the **Specify robot application** page, under **Robot application**, select **Create new application**. Optionally, you can select **Choose existing application** to use a robot application that you have already created.
13. Type a **name** for the robot application.
14. Under **Container image**, specify the Amazon ECR repository location for the robot application container. For more information, see [Requirements for AWS RoboMaker compatible containers](#).

 **Note**

Using \$LATEST doesn't protect you from changes in Amazon ECR. When AWS RoboMaker accesses the repository, it sets it to read-only.

For more information about versioning, see [Versioning applications](#).

15. In **Robot application configuration**, provide the **Launch Command** for your robot application.
16. *Optional:* To configure robot application tools, expand **Robot application tools**. Select **Use default tools** to use preconfigured tools. Select **Customize tools** to add, remove, or edit custom tools to use with the application.

To add a new custom tool:

- a. Select **Add tool**.
  - b. On the **Add application tool**, specify a **Tool name**.
  - c. Specify the command-line arguments for the tool. You must include the tool executable name.
  - d. Choose an **Exit behavior**. If you select **Fail**, the simulation job fails if the tool exits. Select **Restart** to restart the tool. The default is **Restart**.
  - e. Choose to enable or disable UI streaming. UI streaming is disabled by default.
  - f. Select **Send output to CloudWatch** to record logs for the tool. The logs are available in CloudWatch. Output is not sent to CloudWatch by default. Custom tools start only after the main ROS launch process has started.
17. *Optional:* If your application includes a graphical user interface, select **Run with streaming session**. AWS RoboMaker configures a connection so you can interact with your application



as it is running in the simulation. You can connect by selecting **Robot Application** under **Simulation tools** on the simulation job detail page.

18. *Optional:* If your robot application uses environment variables, specify the **Name** and **Value** pairs. Environment variable names must start with A-Z or underscore and consist of A-Z, 0-9 and underscore. Names beginning with AWS are reserved.

Select **Add environment variable** to add additional variables.

You can read environment variables in a launch file using roslaunch [substitution args](#).

19. *Optional:* Configure traffic forwarding from the simulation job port to the application port. Simulation job networking must be configured in order to specify port mapping for your robot and simulation applications.
20. *Optional:* Specify one or more **Robot application upload configurations**. A simulation job output destination must be configured in order to specify upload configurations. Each configuration specifies an upload behavior, a Unix glob file matching rule, and a location to place matching files. For more information about custom uploads, see [Adding a custom upload configuration](#).
21. Choose **Next**.
22. On the **Specify simulation application** page, select **Create new application**. Optionally, you can select **Choose existing application** to use a simulation application that you have already created.
23. Type a **name** for the simulation application.
24. Under **Container image**, specify the Amazon ECR repository location for the robot application container. For more information, see [???](#). Using \$LATEST doesn't protect you from changes in Amazon ECR. When AWS RoboMaker accesses the repository, it sets the repository to read-only.

For more information about versioning, see [Versioning applications](#).

25. In **Simulation application configuration**, provide the **Launch Command** for your robot application.
26. *Optional:* To configure robot application tools, expand **Simulation application tools**. Select **Use default tools** to use preconfigured tools. Select **Customize tools** to add, remove, or edit custom tools to use with application.

To add a new custom tool:

- a. Select **Add tool**.
- b. On the **Add application tool**, specify a **Tool name**.
- c. Specify the command-line arguments for the tool. You must include the tool executable name.
- d. Choose an **Exit behavior**. If you select **Fail**, the simulation job fails if the tool exits. Select **Restart** to restart the tool. The default is **Restart**.
- e. Choose to enable or disable UI streaming. UI streaming is disabled by default.
- f. Select **Send output to CloudWatch** to record logs for the tool. The logs are available in CloudWatch. Output is not sent to CloudWatch by default.

Custom tools start only after the main launch process has started.

27. *Optional:* If your application includes a graphical user interface, select **Run with streaming session**. AWS RoboMaker configures a connection so you can interact with your application as it is running in the simulation. You can connect by selecting **Simulation Application** under **Simulation tools** on the simulation job detail page.
28. *Optional:* If your simulation application uses environment variables, specify the **Name** and **Value** pairs. Select **Add environment variable** to add additional variables.
29. *Optional:* Configure traffic forwarding from the simulation job port to the application port. Simulation job networking must be configured in order to specify port mapping for your robot and simulation applications.
30. *Optional:* Specify one or more **Simulation application upload configurations**. A simulation job output destination must be configured in order to specify upload configurations. Each configuration specifies an upload behavior, a Unix glob file matching rule, and a location to place matching files.

Default upload configurations maintain backwards compatibility with past simulation job output configurations. The default configuration is added to additional upload configurations you create. For more information about custom uploads, see [Adding a custom upload configuration](#).

31. Choose **Next**.
32. Select **Create** to create the simulation job.

## Using the AWS CLI

### Example

The following is an example AWS CLI command that performs the equivalent of the console-based simulation job creation on the other tab.

```
aws robomaker create-simulation-job --max-job-duration-in-seconds 3600
--iam-role arn:aws:iam::111111111111:role/MyRole --robot-applications
application=arn:aws:robomaker:us-west-2:111111111111:robot-application/
MyRobotApplication/1551203485821,launchConfig="{command=["roslaunch",
"hello_world_robot", "rotate.launch"]}" --simulation-applications
application=arn:aws:robomaker:us-west-2:111111111111:simulation-application/
MySimulationApplication/1551203427605,launchConfig="{command=["roslaunch",
"hello_world_simulation", "empty_world.launch"]}" --tags Region=North
```

## Viewing a simulation job

You can view information about a simulation job and, if the job is running, launch graphical tools and a terminal to interact with the simulation. You can also view details about the simulation job and manage tags.

### To view a simulation job

Follow the steps under one of the following tabs:

#### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Simulations**, then choose **Simulation jobs**.
3. Select the **Id** of a simulation job to view its details including the time it was created and launch commands for the robot application and simulation application.

#### Using the AWS CLI

## Example

Following is an example AWS CLI command that performs the equivalent of the console-based view simulation job on the other tab.

```
aws robomaker list-simulation-jobs
aws robomaker describe-simulation-job --job my-simulation-job-arn
```

## Canceling a simulation job

A simulation job can be cancelled if it is running and no longer needed.

### To cancel a simulation job

Follow the steps under one of the following tabs:

#### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Simulations**, then choose **Simulation jobs**.
3. Select the **Id** of the simulation job you want to cancel.
4. In the **Simulation job detail** page, under **Actions**, choose **Cancel**.
5. In the **Cancel simulation job** page, select **Yes, cancel**.

#### Using the AWS CLI

### Example

Here's an example AWS CLI command that performs the equivalent of the console-based cancel simulation job on the other tab.

```
aws robomaker list-simulation-jobs
aws robomaker cancel-simulation-job --job my-simulation-job-arn
```

For information about cancelling from within a simulation job, see the [ROS container FAQs](#).

## Cloning a simulation job

You can create a new simulation job from an existing simulation job by *cloning* it from the **Simulation job detail** page on the AWS Management Console.

### Note

Simulation jobs with ROS and Gazebo software suite are not available for cloning. For more information, see [Support policy](#).

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Simulations**, then choose **Simulation jobs**.
3. Select the **Id** of a running simulation job you would like to restart.
4. In the **Simulation job detail** page, under **Actions**, choose **Clone**.
5. In the **Review and create simulation job**, select **Edit** to make changes.
6. Select **Create** to create the simulation job.

## Restarting a simulation job

Running simulation jobs can be restarted. When restarted, the simulation job uses the robot application and simulation application source files in the Amazon S3 location and all other configuration settings specified when the simulation job was created.

### To restart a simulation job

Follow the steps under one of the following tabs:

#### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Simulations**, then choose **Simulation jobs**.
3. Select the **Id** of a running simulation job you would like to restart.
4. In the **Simulation job detail** page, under **Actions**, choose **Restart**.
5. In the **Restart simulation job** page, select **Yes, restart**.

## Using the AWS CLI

### Example

The following is an example AWS CLI command that performs the equivalent of the console-based restart simulation job on the other tab. The simulation job must be running.

```
aws robomaker restart-simulation-job --job my-simulation-job-arn
```

## Logging a simulation

To capture output files and other artifacts from your simulation job, you can configure custom uploads. You can configure custom uploads for your robot application and your simulation application. When you configure a custom upload, files you specify are uploaded from the simulation job to the Amazon S3 simulation output location you provide. This can be useful when you want to review or analyze application output generated during a simulation run or reuse artifacts.

Before you can configure custom uploads, you must provide an Amazon S3 output destination for your simulation job. AWS RoboMaker uploads matching files to a folder using a name you specify. Matching files can be uploaded when all of the simulation job tools shut down or uploaded as they are produced and then removed.

Default upload configurations are automatically added to your custom upload configurations unless you turn them off. The default upload configuration uploads ROS and Gazebo default logging output. This maintains compatibility with past simulation job output configurations, which uploaded ROS and Gazebo default logging output. You can turn off the default upload configuration when you configure a simulation job in the console. You can also turn it off by setting `useDefaultUploadConfigurations` to `false` in the [CreateSimulationJob](#) API.

Your simulation applications are extraded onto a single 128gb partition and you have write access to the partition.

### Sections

- [Adding a custom upload configuration](#)
- [Environment variables created by AWS RoboMaker](#)

## Adding a custom upload configuration

To create a custom upload configuration, you need to specify a *name* prefix that specifies where the files are uploaded in Amazon S3, a Unix glob *path* specifying the files to upload, and an *upload behavior* specifying when the files are uploaded.

### Name

A name is a prefix that specifies how files are uploaded in Amazon S3. It is appended to the simulation output location to determine the final path.

For example, if your simulation output location is `s3://my-bucket` and your upload configuration name is `robot-test`, your files are uploaded to `s3://my-bucket/<simid>/<runid>/robot-test`.

### Path

The path specifies which files are uploaded. Standard Unix glob matching rules are accepted subject to the following:

- The path must begin with `/home/robomaker/` or `/var/log`.
- The path must not contain a reverse path expression (`/.`).
- Symbolic links are not followed.
- You can use `**` as a *super asterisk* in your path. For example, specifying `/var/log/**/*.log` causes all `.log` files in the `/var/log` directory tree to be collected.

You can also use the standard asterisk as a standard wildcard. For example, `/var/log/system.log*` matches files such as `system.log_1111`, `system.log_2222`, and so on in `/var/log`.

### Upload behavior

You can select one of the following upload behaviors:

- **Upload on terminate** (`UPLOAD_ON_TERMINATE`) uploads all files matching the path once the simulation job enters the terminating state. AWS RoboMaker attempts to upload logs for a maximum of 60 minutes.

AWS RoboMaker does not begin uploading files until all of your tools running in the simulation have stopped.

- **Upload rolling with auto remove** (UPLOAD\_ROLLING\_AUTO\_REMOVE) uploads all files matching the path as they are generated. Paths are checked every 5 seconds. When the files are uploaded, the source files are deleted. Once a file is deleted, if a new file is generated with the same name, it replaces the previously uploaded file. AWS RoboMaker performs a final check for files once all of your applications running in the simulation have stopped.

Upload rolling with auto remove is useful for uploading rolling logs. Write or stream output to an "active" file which is not covered by the path glob. Once you're done writing to the active file, roll the file into a location covered by the path glob to be uploaded and removed.

This setting can help you conserve space in your simulation job. It can also help you access files before your simulation job terminates.

The simulation job partition size is 128gb. If your simulation job ends for any reason, AWS RoboMaker tries to upload all files specified in your custom upload configuration.

## Environment variables created by AWS RoboMaker

AWS RoboMaker defines the following simulation job environment variables.

- AWS\_ROBOMAKER\_SIMULATION\_JOB\_ID
- AWS\_ROBOMAKER\_SIMULATION\_JOB\_ARN
- AWS\_ROBOMAKER\_SIMULATION\_RUN\_ID

You can get these variables from your application or from the command line. For example, to get the current simulation job Amazon Resource Name (ARN) in Python, use `os.environ.get("AWS_ROBOMAKER_SIMULATION_JOB_ARN")`.

If you specified an Amazon Simple Storage Service output bucket for the simulation job, you can use the environment variables to find the output path. AWS RoboMaker writes output to `s3://bucket-name/AWS_ROBOMAKER_SIMULATION_JOB_ID/AWS_ROBOMAKER_SIMULATION_RUN_ID`. Use this to manage objects in Amazon S3 from code or the command line.

AWS RoboMaker also handles specific environment variables set up in `CreateSimulationJobRequest` to allow robot and simulation application containers to communicate with each other. For more information, see [ROS container FAQs](#).



# Batching simulations

This section provides information about how you can start and manage simulation job batches. Using a simulation job batch, you can launch and run many simulations using a single API call to perform regression testing, parameter optimization, machine learning model training, and synthetic data generation.

## Note

Simulation job batches can only be started using the AWS RoboMaker SDK or AWS CLI. You can view, clone, and cancel simulation batches using the AWS RoboMaker console.

## Sections

- [Starting a simulation job batch](#)
- [View a simulation job batch](#)
- [Cancelling a simulation job batch](#)
- [Cloning a simulation job batch](#)

## Starting a simulation job batch

Simulation job batches are started from the AWS SDK or AWS CLI. A simulation job batch includes one or more simulation job requests. Each simulation job request identifies which applications to use in each simulation, the maximum duration of the job, and other information. You can apply tags to the simulation job batch and each simulation job request.

**To start a simulation job batch, you must do the following:**

1. Install the AWS Command Line Interface. For more information about installing the AWS CLI, see [Installing the AWS CLI](#).
2. Copy the following JSON into a file named `startsimjobbatch.json`. Modify the file to match your desired configuration, and then save it.

```
{
  "batchPolicy": {
    "timeoutInSeconds": 400,
    "maxConcurrency": 2
  }
}
```

```

    },
    "createSimulationJobRequests": [
      {
        "maxJobDurationInSeconds": 300,
        "iamRole": "arn:aws:iam::111111111111:role/MyRole",
        "failureBehavior": "Fail",
        "robotApplications": [
          {
            "application": "arn:aws:robomaker:us-east-1:111111111111:robot-
application/MyRobotApplicationArn",
            "launchConfig": {
              "packageName": "hello_world_robot",
              "launchFile": "rotate.launch"
            }
          }
        ],
        "simulationApplications": [
          {
            "application": "arn:aws:robomaker:us-
east-1:111111111111:simulation-applicationMySimulationApplicationArn",
            "launchConfig": {
              "command": [
                "roslaunch", "hello_world_robot", "rotate.launch"
              ]
            }
          }
        ],
        "tags": {
          "myRequestTagKey" : "myRequestTagValue"
        }
      },
      {
        "maxJobDurationInSeconds": 300,
        "iamRole": "arn:aws:iam::111111111111:role/MyRole",
        "failureBehavior": "Fail",
        "simulationApplications": [
          {
            "application": "arn:aws:robomaker:us-
east-1:111111111111:simulation-applicationMySimulationApplicationArn",
            "launchConfig": {
              "command": [
                "roslaunch", "hello_world_simulation",
                "empty_world.launch"
              ]
            }
          }
        ]
      }
    ]
  }
}

```

```
    }
  }
],
"tags": {
  "myBatchTagKey" : "myBatchTagValue"
}
}
```

3. Open a command prompt, then run the following AWS CLI command:

```
$ aws robomaker start-simulation-job-batch --cli-input-json
file://startsimjobbatch.json
```

To view the simulation job batch, see [View a simulation job batch](#).

## View a simulation job batch

You can view information about a simulation job batch including details about simulation job requests in the batch.

### To see the details of a simulation job batch

Follow the steps under one of the following tabs:

#### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Simulations**, then choose **Simulation job batches**.
3. Select the **Id** of a simulation job batch to view its details.

#### Using the AWS CLI

### Example

The following is an example AWS CLI command that performs the equivalent of the console-based view simulation job on the other tab.

```
aws robomaker list-simulation-job-batches
aws robomaker describe-simulation-job-batch --job my-simulation-job-batch-arn
```

## Canceling a simulation job batch

A simulation job can be cancelled if it is running and no longer needed.

### To cancel a simulation job

Follow the steps under one of the following tabs:

#### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Simulations**, then choose **Simulation job batches**.
3. Select the **Id** of the simulation job batch you want to cancel.
4. In the **Simulation job batch detail** page, under **Batch actions**, choose **Cancel batch**.
5. In the **Cancel simulation job batch** page, select **Cancel**.

#### Using the AWS CLI

### Example

The following is an example AWS CLI command that performs the equivalent of the console-based cancel simulation job batch on the other tab.

```
$ aws robomaker list-simulation-job-batches
$ aws robomaker cancel-simulation-job-batch --job my-simulation-job-batch-arn
```

## Cloning a simulation job batch

You can start a new simulation job batch by cloning an existing batch. When you clone, you can include all of the simulation job requests or select a subset of requests.

**Note**

Simulation job batches with ROS and Gazebo software suite is not available for cloning. For more information, see [Support policy](#).

**To clone a simulation job batch:**

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Simulations**, then choose **Simulation job batches**.
3. Select the **Id** of the simulation job batch you want to clone.
4. To clone the entire batch, in the **Simulation job batch detail** page, under **Batch actions**, choose **Clone batch**.

To clone specific simulation job requests from the batch, under **Simulation job requests**, check the simulation job requests you want to clone, then select **Request actions** and choose **Clone request**.

5. In the **Clone simulation job batch** page, select **Submit**.

# Creating worlds with Simulation WorldForge

With Simulation WorldForge, worlds are generated from the simulation world templates you define. The simulation world template specifies the world layout, room dimensions, furnishings, how rooms are connected, and other details. Walls and floors and other room features can have material properties. Rooms can be furnished by room type automatically or you can select potential furnishings. Generated worlds can be used in your simulation jobs and exported to use on your developer machine.

Simulation WorldForge can help you manage simulation workloads that require a large number of simulation worlds with domain randomization. Common Simulation WorldForge scenarios include the following:

- **Regression testing** — Test your robotics applications in hundreds of worlds to verify correct behavior.
- **Synthetic imagery data generation** — You can capture images from the generated worlds to use in other robotic applications. For example, you can capture images of rooms with different furniture layout and material composition.
- **Reinforcement learning** — Create hundreds of unique worlds with an interior structure for your robotic application to explore. You control the composition of the world.
- **Developing algorithms** — A robotics navigation engineer can verify a navigation algorithm succeeds in a known layout with different furniture placement. A robotics localization engineer can ensure a layout algorithm detects different structural elements in different floor plans.

You do not need to know world generation algorithms or how to create and manage infrastructure. Simulation WorldForge and AWS RoboMaker are fully managed services.

## Simulation WorldForge concepts

Simulation WorldForge uses a collection of parameters (a *simulation world template*) to determine how to generate new worlds. One simulation world template can be used to generate hundreds of worlds. Each *world* contains a building. The building has a single floor. The floor has a *floorplan template* that describes the rooms' sizes and shapes. It also suggests how the rooms might be connected. Floors also have an *interior template* that specifies how structural elements like walls and floors in the floor plan are finished. The interior template also has parameters describing

how to populate each room with furnishing like tables and sofas and accessories like clothes and kitchenware.

You can create a simulation world template from a sample template, clone an existing template, or start from scratch using the console at <https://console.aws.amazon.com/robomaker/>. For example, if you want to generate worlds containing one bedroom, you can start with the one bedroom apartment sample template. It is a one-bedroom, one-bathroom open floor plan with a kitchen and a living room. It uses typical materials and furnishings and accessories appropriate for each room type. After you save it, you can start a *world generator job* to generate worlds. You can generate up to 50 worlds in a world generation job.

You can also create a simulation world template using the SDK or the AWS Command Line Interface. For example, to create a template from the AWS CLI, first create a *world template JSON document* with the template body. It specifies the parameters for the building, floor plans, interiors, and other details. After you save it, you can create a simulation world template by calling `create-world-template` and specifying the JSON file:

```
aws robomaker create-world-template --name "my-template" --templateBody file://  
my_template_body.json
```

After you have configured and saved a simulation world template, you can create a world generation job and generate worlds. Hundreds of worlds can be generated from a single simulation world template. You can generate up to 100 worlds in a single world generation job. Worlds can be used with a simulation in AWS RoboMaker. You can also export worlds to modify and use in your own ROS environment.

## Understanding simulation world templates

This section describes the components of a simulation world template. Components include the floor plan and preferences for interior materials and furniture. Simulation WorldForge provides defaults for many of the components including materials, furniture selection, and room connectivity. You can override the defaults with your own preferences. Simulation WorldForge makes a best effort to follow your preferences when generating worlds.

### Floor Plan

The floor plan specifies an indoor floor plan for a single-story, residential building. It includes the world dimensions, the number and types of rooms, and parameters that influence how rooms are connected.

Every world is guaranteed to have a clear 1-meter cylinder centered at coordinate  $(0, 0, 0)$ , the default robot starting position. Simulation WorldForge determines the room.

## World dimensions

You can configure an aspect ratio and a ceiling height for the building. Valid aspect ratios are from 1:4 to 4:1. Valid ceiling height is 2.4 to 4.0 meters. All measurements are in meters and square meters. The console supports conversion between the US and metric system.

## Rooms

You can specify the number of rooms, room type, room name, desired area, desired aspect ratio, and interior features. The following room types are supported:

- Bedroom
- Bathroom
- Living
- Dining
- Kitchen
- Hallway
- Closet

Furniture, wall material, and floor material are selected from types appropriate for the room type. For example, a bathroom might be assigned a tile wall, a linoleum floor, and have a toilet and a shower.

## Connections

Simulation WorldForge automatically connects all rooms by default. You can connect rooms by an opening or a doorway. When rooms are connected by an opening, the rooms are in an open floor plan. There is no wall. Rooms connected by a doorway have a narrow, doorless opening. Doorway openings are randomly placed along the adjacent wall.

You can override default connections with desired connections. For example, if you have a kitchen, dining room, and a bedroom, you can request a door connection between the kitchen and bedroom. Simulation WorldForge makes a best effort to make the connection, but it is not guaranteed.



## Interiors

You can select from a number of different interior material and furniture types. Simulation WorldForge randomly assigns flooring, walls, and furniture to rooms by room type. For example, a kitchen might be assigned an oven and dining room table and chairs.

You can select material types for flooring and walls as a custom set. When you create a custom set, you can apply the custom assignment by **room type** or **room name**. You can have multiple custom sets. If there is a conflict, a custom assignment for a room always has precedence over one for room type.

For example, assume you have a "Modern Flooring" custom set assigned to all bedrooms and a "Chic Flooring" custom set assigned to the "Master Bedroom" room. When Simulation WorldForge assigns flooring materials, "Master Bedroom" is assigned flooring material from the "Chic Flooring" set. Other bedrooms have flooring material selected from the "Modern Flooring" set.

This rule also applies to custom furniture sets.

### Flooring Material Types

Supported flooring types include the following:

- Carpet
- Concrete
- Floorboards
- Linoleum
- Parquetry
- Tiles

Flooring material is randomly chosen from all of the flooring material types selected. For example, if you specify Carpet, Concrete, linoleum and parquetry, the floor of your room might be concrete.

### Wall material types

Supported wall material types include the following:

- Brick
- Concrete

- Stone
- Tiles
- Wood panels
- Wall paint
- Wallpaper

Wall material is randomly chosen from all of the wall material types selected. For example, if you specify `Brick`, `Tiles`, and `Wallpaper`, your room might have walls that use tile and wallpaper. Simulation WorldForge might not assign wall material from all chosen wall material types.

## Furniture types

Simulation WorldForge supports the following furniture types:

- Baths
- Bar cabinets
- Beds
- Bookcases
- Coffee tables
- Console tables
- Corner cabinets
- Desk chairs
- Desks
- Dining chairs
- Dining tables
- Dish washers
- Dressers
- End and side tables
- Floor lamps
- Fridges
- Living room chairs
- Kitchen islands and carts
- Media storage

- Nightstands
- Ottomans
- Ovens
- Serving carts
- Showers
- Sideboards and buffets
- Sofas
- Storage
- Storage benches
- Toilets
- Vanity counters
- Washing machines and dryers

Furniture is randomly chosen from all of the furniture types selected. For example, if you specify Sideboards and buffets, Sofas and Console tables, your room might have a sofa and two console tables, but no sideboard or buffet. Simulation WorldForge might not assign material types from all chosen furniture types.

## Common tasks

This section contains common tasks for creating simulation world templates. Many of the tasks specify desired connections or desired shapes. Simulation WorldForge makes a best effort to generate worlds according to simulation world template parameters. Generated worlds might not always include all desired properties.

### Topics

- [Specifying a list of rooms for a floor](#)
- [Requesting a long hallway](#)
- [Requesting a doorway between rooms](#)
- [Applying a configuration to all rooms](#)
- [Requesting doors in doorways](#)
- [Requesting no doors in doorways](#)
- [Requesting a wide floor plan footprint](#)

- [Requesting a custom ceiling height](#)
- [Specifying the same material types for floors in different rooms](#)
- [Specifying different material types for floors between rooms of the same type](#)
- [Specifying more and less furniture in rooms](#)
- [Adding specific furniture types to all bedrooms and a single shared living/dining room](#)
- [Specifying a room without furniture](#)

## Specifying a list of rooms for a floor

The room type influences the floor plan by contributing to which rooms are adjacent. The room type is also used to determine the types of material for its flooring and walls and the types of furniture to randomly place by default. You may override the default flooring and walls material types and furniture types by room type or room name.

You can select from the following room types: Bedroom, Bathroom, Living Room, Dining Room, Kitchen, Hallway, Closet.

The following examples specifies a three-room house. The sizes and shapes of the rooms are determined by default.

Using the console

1. In the **Simulation world template edit** screen, under **Floor plan**, choose **Rooms**.
2. In the **Rooms** pane, choose **Add room**.
3. Add details for the room. You can specify a room **Name**, **Room type**, **Desired area** and **Desired aspect ratio**.
4. Choose **Save** to save the new room. Repeat until you have the rooms you desire. If you add too many, you can delete them from the **Rooms** pane.

Using the AWS CLI

### Example

You can use the following JSON in the `templateBody` as part of a call to `create-world-template`.

```
"Rooms": [
```

```
{
  "Type": "Bedroom",
  "Name": "My Master Bedroom",
},
{
  "Type": "Bathroom",
  "Name": "My Ensuite",
},
{
  "Type": "Kitchen",
  "Name": "My Kitchen",
}
]
```

## Requesting a long hallway

You can use the `DesiredShape` property to request the preferred shape of a room. `Type` has no effect on the shape. In the following example, the `Hallway` aspect ratio is low. When it is combined with a large enough `Area`, it indicates a desire for a long, narrow hallway. Simulation WorldForge attempts to generate rooms similar to the desired shape.

### Using the console

1. On the **Simulation world template edit** screen, under **Floor plan**, choose **Rooms**.
2. In the **Rooms** pane, choose **Add room**.
3. Specify a room **Name**, then choose **Hallway** for **Room type**.
4. Specify a **Desired area** of 20 and a **Desired aspect ratio** of 4:1.
5. Choose **Save** to save the hallway.

### Using the AWS CLI

#### Example

You can use the following JSON in the `templateBody` as part of a call to `create-world-template`.

```
"Rooms": [
  {
    "Type": "Hallway",
```

```
    "Name": "My Hallway",
    "DesiredShape": {
      "Area": 20.0,
      "AspectRatio": {
        "x": 4, "y": 1
      }
    }
  }
]
```

The valid room area range is 10 meters to 300 meters. The valid room aspect ratio range is 1:4 to 4:1.

## Requesting a doorway between rooms

If you have two rooms, and the rooms share at least one wall, you can request a `DesiredConnections` between the two rooms. Simulation WorldForge tries to place the rooms adjacent and, depending on the `ConnectionType`, either place a `Doorway` in a random location along an adjacent wall or create an `Opening` by removing an adjacent wall entirely.

The following example requests an open connection for the living room and kitchen. It also requests a separate doorway connection for the bedroom and bathroom:

Using the console

1. On the **Simulation world template edit** screen, under **Floor plan**, choose **Connections**.
2. In the **Connections** pane, choose **Add connection**.
3. In the **Desired connections** pane, select **Opening** for **Connection type** and then select a room for **Location 1** and **Location 2**, such as "My Living Room" and "My Kitchen".
4. Choose **Save** to save desired connections.
5. Repeat to add a **Door** as a desired connection between two other locations, such as "My Bedroom" and "My Bathroom".

Using the AWS CLI

### Example

You can use the following JSON in the `templateBody` as part of a call to `create-world-template`.

```
"DesiredConnections": [
  {
    "Location": [ "My Living Room", "My Kitchen" ],
    "ConnectionType": "Opening"
  },
  {
    "Location": [ "My Bedroom", "My Bathroom" ],
    "ConnectionType": "Doorway"
  }
]
```

The valid number of connections per room is 4 and a maximum of one opening connection for each pair of rooms.

## Applying a configuration to all rooms

### Note

You can only apply a configuration to all rooms by using templates that are Version 2 and above. For more information, see [Applying a configuration to all rooms](#).

You can use the `Target.All` keyword to apply a configuration to all rooms.

The following example changes the door state for all doors.

Using the console

The following procedure gives you the ability to apply a configuration for all the doors in your world. You can also apply a single configuration to all floors, material sets, walls, and furniture.

1. On the **Simulation world template edit** screen, under **Interiors**, choose **Doors**.
2. In the **Doors** pane, choose **Add custom doors**.
3. For **Set name**, specify a name for your set of custom doors.
4. For **Rooms affected**, specify **All rooms**.
5. For **Door state**, choose the open state of the door.
6. Choose **Save** to save the doors configuration.

## Using the AWS CLI

### Example

You can use the following JSON in the `templateBody` as part of a call to `create-world-template`. The following example targets all doors within a doorway set.

```
"Interior": {
  "Doorways": {
    "DoorwaySets": [
      {
        "Name": "your-doorway-set",
        "TargetSet": "Target.All",
        "Door": {
          "InitialState": {
            "OpenPosition": {
              "Percent": "percentage-that-you-specify"
            }
          }
        }
      }
    ]
  }
}
```

## Requesting doors in doorways

### Note

You can only configure doorways to have doors using world templates that are Version 2 and above.

You can use a template to specify doors in doorways in your AWS RoboMaker Simulation WorldForge world.

You can specify the following types of doors:

- Hinged door



You can configure the percentage in which these doors are open. For example, these are some open states that you can specify:

- 0% open – closed
- 50% open – halfway open
- 70% open – mostly open
- 100% open – entirely open

You can also choose to have AWS RoboMaker assign a randomized open percentage to each door.

You can use the following procedure to add doors to your doorways.

Using the console

1. On the **Simulation world template edit** screen, under **Interiors**, choose **Doors**.
2. In the **Doors** pane, choose **Add custom doors**.
3. For **Set name**, name your custom door set.
4. For **Rooms affected** under **Location**, choose the rooms that you want to have doors.
5. For **Door type** under **Customizations**, choose the type of door that you're adding.
6. Under **Door state**, choose whether the door is open, closed, partially open, or in a randomized state.
7. Choose **Save** to save the configuration.

Using the AWS CLI

### Example

You can use the following JSON in the `templateBody` as part of a call to `create-world-template`.

```
"Interior": {
  "Doorways": {
    "DoorwaySets": [
      {
        "Name": "your-doorway-set",
        "TargetSet": "the-doorways-that-you-want-to-target",
        "Door": {
```

```
        "InitialState": {
          "OpenPosition": {
            "Percent": "the-open-percentage-that-you-specify-for-the-doors-that-
you're-targeting"
          }
        }
      ]
    }
  }
}
```

## Requesting no doors in doorways

### Note

You can only explicitly specify no doors in doorways using world templates that are Version 2 and above.

You can use a template to explicitly specify that the doorways in your AWS RoboMaker Simulation WorldForge world have no doors in the doorways.

The following example requests that there are no doors in the doorways between rooms.

Using the console

1. On the **Simulation world template edit** screen, under **Interiors**, choose **Doors**.
2. In the **Doors** pane, choose **Add custom doors**.
3. For the **Rooms affected** pane under **Location**, choose **All rooms**.
4. For **Door type** under **Customizations**, choose **No door in doorway**.
5. Choose **Save**.

Using the AWS CLI

### Example

You can use the following JSON in the `templateBody` as part of a call to `create-world-template`.

```
"Interior": {
  "Doorways": {
    "DoorwaySets": [
      {
        "Name": "doorway-set-name",
        "TargetSet": "Target.All",
        "Door": null
      }
    ]
  }
}
```

## Requesting a wide floor plan footprint

If you want a longer or wider floor plan layout that affects all of the rooms, you can request a `DesiredAspectRatio` for the `Footprint`. Simulation WorldForge uses this preference to influence the overall shape and positions of the rooms so the floor plan better fits the requested footprint aspect ratio. The desired aspect ratio is optional and defaults to a square.

The following example overrides the default square ratio (1:1) to a prefer a wider layout where all of the rooms are more likely to be stretched and placed to create a non-square footprint:

Using the console

1. On the **Simulation world template edit** screen, under **Floor plan**, choose **World dimensions**.
2. In the **World dimensions** pane, under **Desired aspect ratio**, specify a **Width** of 1 and a **Length** of 4.
3. Choose **Save** to save the new room.

Using the AWS CLI

### Example

You can use the following JSON in the `templateBody` as part of a call to `create-world-template`.

```
"Footprint": {
  "DesiredAspectRatio": {
```

```
    "x": 1, "y": 4
  }
}
```

The valid range for `DesiredAspectRatio` is a range from 1:4 to 4:1.

## Requesting a custom ceiling height

The floor plan ceiling height determines the height of the walls for all the rooms. The default ceiling height is 2.4 meters. In this example, we override the default to 3.2 meters:

Using the console

1. On the **Simulation world template edit** screen, under **Floor plan**, choose **World dimensions**.
2. In the **World dimensions** pane, specify a **Ceiling height** of 3.2.
3. Choose **Save** to save the new room.

Using the AWS CLI

### Example

You can use the following JSON in the `templateBody` as part of a call to `create-world-template`.

```
"Ceiling": {
  "Height": 3.2
}
```

## Specifying the same material types for floors in different rooms

Use either room types or room names and list multiple rooms for the interior flooring section. In the following example, all of the bedrooms, living rooms and dining rooms have a random floorboard material assigned.

Using the console

1. On the **Simulation world template edit** screen, under **Interiors**, choose **Flooring**.

2. In the **Flooring** pane, choose **Add flooring**.
3. In the **Custom flooring** pane, specify a flooring **Set name**, such asor example, "Flooring Material Set 1".
4. Under **Filter type**, choose **By room type**.
5. Under **Room types**, select **Bedrooms**, **Living rooms**, and **Dining rooms**.
6. Under **Custom flooring**, choose **Add material** and then choose **Floorboard**.
7. Choose **Save** to save the flooring set.

## Using the AWS CLI

### Example

You can use the following JSON in the `templateBody` as part of a call to `create-world-template`.

```
"Flooring": {
  "MaterialSets": [
    {
      "Name": "Flooring Material Set 1",
      "TargetSet": {
        "RoomTypes": [ "Bedroom", "Living", "Dining" ]
      },
      "SampleSet": {
        "MaterialTypes": [ "Floorboards" ]
      }
    }
  ]
}
```

## Specifying different material types for floors between rooms of the same type

In the following example, all of the bedrooms, living rooms and dining rooms have a random floorboard material assigned except for Bedroom 3. It is assigned a random carpet material.

### Using the console

1. On the **Simulation world template edit** screen, under **Interiors**, choose **Flooring**.

2. In the **Flooring** pane, choose **Add flooring**.
3. In the **Custom flooring** pane, specify a flooring **Set name**, such as **Flooring Material Set 1**.
4. Under **Filter type**, choose **By room type**.
5. Under **Room types**, select **Bedrooms**, **Living rooms**, and **Dining rooms**.
6. Under **Custom flooring**, choose **Add material** and then choose **Floorboard**.
7. Choose **Save** to save the flooring set.
8. In the **Flooring** pane, choose **Add flooring**.
9. In the **Custom flooring** pane, specify a flooring **Set name**, such as **Flooring Material Set for Bedroom 3**.
10. Under **Filter type**, choose **By room name**.
11. Under **Room name**, select a room name, such as **Bedroom 3**.
12. Under **Custom flooring**, choose **Add material** and then choose **Carpet**.
13. Choose **Save** to save the flooring set.

## Using the AWS CLI

### Example

You can use the following JSON in the `templateBody` as part of a call to `create-world-template`.

```
"Flooring": {
  "MaterialSets": [
    {
      "Name": "Flooring Material Set 1",
      "TargetSet": {
        "RoomTypes": [ "Bedroom", "Living", "Dining" ]
      },
      "SampleSet": {
        "MaterialTypes": [ "Floorboards" ]
      }
    },
    {
      "Name": "Flooring Material Set for Bedroom 3",
      "TargetSet": {
```

```
    "RoomNames": [ "Bedroom 3" ]
  },
  "SampleSet": {
    "MaterialTypes": [ "Carpet" ]
  }
}
]
```

## Specifying more and less furniture in rooms

You can specify how densely furniture is spaced by room names or room types. By default, rooms are randomly furnished with moderate spacing. In the following example, all bedrooms are randomly furnished with dense spacings. The living room and dining room are furnished sparsely. All other rooms are furnished by default.

Using the console

1. On the **Simulation world template edit** screen, under **Interiors**, choose **Furniture**.
2. In the **Furniture** pane, choose **Add custom furniture**.
3. In the **Custom furniture** pane, specify a custom furniture **Set name**, such as Dense Furniture Arrangement.
4. Under **Filter type**, choose **By room type**.
5. Under **Room types**, select **Bedrooms**.
6. Toggle **Override furniture** to use default furniture.
7. Under **Furniture density**, choose **Dense**.
8. Choose **Save** to save the furniture set.
9. In the **Furniture** pane, choose **Add custom furniture**.
10. In the **Custom furniture** pane, specify a custom furniture **Set name**, such as Sparse Furniture Arrangement.
11. Under **Filter type**, choose **By room name**.
12. Under **Room names**, select the rooms you want to have sparse furniture density, such as My Living Room and My Dining Room.
13. Toggle **Override furniture** to use default furniture.

14. Under **Furniture density**, choose **Sparse**.
15. Choose **Save** to save the furniture set.

## Using the AWS CLI

### Example

You can use the following JSON in the `templateBody` as part of a call to `create-world-template`.

```
"Furniture": {
  "FurnitureArrangements": [
    {
      "Name": "Dense Furniture Arrangement",
      "TargetSet": {
        "RoomTypes": [ "Bedroom" ]
      },
      "DesiredSpatialDensity": "Dense"
    },
    {
      "Name": "Sparse Furniture Arrangement",
      "TargetSet": {
        "RoomNames": [ "My Living Room", "My Dining Room" ]
      },
      "DesiredSpatialDensity": "Sparse"
    }
  ]
}
```

## Adding specific furniture types to all bedrooms and a single shared living/dining room

You can specify the types of furniture for a room by room names or room types. In the following example, all bedrooms are moderately furnished with random beds, desks, dressers, and floor lamps. The room "My living/dining room" is densely furnished with random dining tables, dining chairs, floor lamps, sofas, and coffee tables. All other rooms are furnished by default.

### Using the console

1. On the **Simulation world template edit** screen, under **Interiors**, choose **Furniture**.



2. In the **Furniture** pane, choose **Add custom furniture**.
3. In the **Custom furniture** pane, specify a custom furniture **Set name**, such as Bedroom Furniture.
4. Under **Filter type**, choose **By room type**.
5. Under **Room types**, select **Bedrooms**.
6. Ensure **Override furniture** is selected. If it is not selected, Simulation WorldForge uses default furniture.
7. Under **Furniture types**, choose **Add furniture** and then select **Beds, Desks, Dressers, and Floorlamps**.
8. Choose **Save** to save the furniture set.
9. In the **Furniture** pane, choose **Add custom furniture**.
10. In the **Custom furniture** pane, specify a custom furniture **Set name**, such as Living and Dining Furniture.
11. Under **Filter type**, choose **By room name**.
12. Under **Room names**, select a room, such as My living and dining room.
13. Ensure **Override furniture** is selected. If it is not selected, Simulation WorldForge uses default furniture.
14. Under **Furniture types**, choose **DiningTables, DiningChairs, FloorLamps, Sofas, and CoffeeTables**.
15. Under **Furniture density**, choose **Dense**.
16. Choose **Save** to save the furniture set.

## Using the AWS CLI

### Example

You can use the following JSON in the `templateBody` as part of a call to `create-world-template`.

```
"Furniture": {
  "FurnitureArrangements": [
    {
      "Name": "Bedroom Furniture",
      "TargetSet": {
```

```
    "RoomTypes": [ "Bedroom" ]
  },
  "SampleSet": {
    "ModelTypes": [
      "Beds",
      "Desks",
      "Dressers",
      "FloorLamps"
    ]
  }
}
{
  "Name": "Living and Dining Furniture",
  "TargetSet": {
    "RoomNames": [ "My living and dining room" ]
  },
  "SampleSet": {
    "ModelTypes": [
      "DiningTables",
      "DiningChairs",
      "FloorLamps",
      "Sofas",
      "CoffeeTables"
    ],
    "DesiredSpatialDensity": "Dense"
  }
}
]
```

## Specifying a room without furniture

Specify an empty list for the model set for the furnishing arrangement. All other rooms are furnished by default:

Using the console

1. On the **Simulation world template edit** screen, under **Interiors**, choose **Furniture**.
2. In the **Furniture** pane, choose **Add custom furniture**.
3. In the **Custom furniture** pane, specify a custom furniture **Set name**, such as No furniture.

4. Under **Filter type**, choose **By room name**.
5. Under **Room names**, select the rooms that you want to have no furniture, such as My Spare Room.
6. Ensure **Override furniture** is selected. If it is not selected, Simulation WorldForge uses default furniture.
7. Under **Furniture types**, make sure no types are chosen.
8. Choose **Save** to save the furniture set.

## Using the AWS CLI

### Example

You can use the following JSON in the `templateBody` as part of a call to `create-world-template`.

```
"Furniture": {
  "FurnitureArrangements": [
    {
      "Name": "No Furniture",
      "TargetSet": {
        "RoomNames": [ "My Spare Room" ]
      },
      "SampleSet": {
        "ModelTypes": []
      }
    }
  ]
}
```

## JSON schema for simulation world template body

The `templateBody` (simulation world template body) is an input parameter of the [CreateWorldTemplate](#) operation. This parameter is a JSON-formatted string. The JSON specifies a simulation world template and contains the parameters Simulation WorldForge uses to generate worlds.

The following shows the schema for the different versions of the world template.

## Version 2

The following is the template for the Version 2 schema

```
{
  "title": "WorldTemplate",
  "description": "The top-level template for parameterizing a randomly generated world.
  By default, a single\nresidential building with one floor and one room is generated.",
  "type": "object",
  "properties": {
    "Version": {
      "title": "Version",
      "type": "string"
    },
    "Buildings": {
      "title": "Buildings",
      "default": [
        {
          "Floors": [
            {
              "Floorplan": {
                "Footprint": {
                  "DesiredAspectRatio": {
                    "x": 1.0,
                    "y": 1.0
                  }
                },
                "Ceiling": {
                  "Height": 3.0
                },
                "Rooms": [
                  {
                    "Type": "Living",
                    "Name": "My_Living_Room",
                    "OriginalName": "My Living Room",
                    "DesiredShape": {
                      "Area": 20.0,
                      "AspectRatio": {
                        "x": 1.0,
                        "y": 1.0
                      }
                    }
                  }
                ]
              }
            }
          ]
        }
      ]
    }
  }
}
```

```

        "DesiredConnections": []
    },
    "Interior": {
        "Doorways": {
            "DoorwaySets": []
        },
        "Flooring": {
            "MaterialSets": []
        },
        "Walls": {
            "MaterialSets": []
        },
        "Furniture": {
            "FurnitureArrangements": []
        }
    }
}
]
}
],
"type": "array",
"items": {
    "$ref": "#/definitions/BuildingTemplate"
},
"minItems": 1,
"maxItems": 1
}
},
"required": [
    "Version"
],
"additionalProperties": false,
"definitions": {
    "AspectRatio": {
        "title": "AspectRatio",
        "type": "object",
        "properties": {
            "x": {
                "title": "X",
                "default": 1,
                "minimum": 1,
                "maximum": 4,
                "type": "number"
            }
        }
    },
}

```

```
    "y": {
      "title": "Y",
      "default": 1,
      "minimum": 1,
      "maximum": 4,
      "type": "number"
    }
  },
  "additionalProperties": false
},
"FloorplanFootprint": {
  "title": "FloorplanFootprint",
  "description": "The desired footprint of this floorplan.",
  "type": "object",
  "properties": {
    "DesiredAspectRatio": {
      "title": "Desiredaspectratio",
      "default": {
        "x": 1.0,
        "y": 1.0
      },
      "allOf": [
        {
          "$ref": "#/definitions/AspectRatio"
        }
      ]
    }
  },
  "additionalProperties": false
},
"FloorplanCeiling": {
  "title": "FloorplanCeiling",
  "description": "The height of the ceiling for this floorplan in metres.",
  "type": "object",
  "properties": {
    "Height": {
      "title": "Height",
      "default": 3.0,
      "type": "number",
      "minimum": 2.4,
      "maximum": 4.0
    }
  },
  "additionalProperties": false
}
```

```
},
"Rectangle": {
  "title": "Rectangle",
  "description": "A rectangle defined by area in square metres and aspect ratio.",
  "type": "object",
  "properties": {
    "Area": {
      "title": "Area",
      "type": "number"
    },
    "AspectRatio": {
      "$ref": "#/definitions/AspectRatio"
    }
  },
  "required": [
    "Area",
    "AspectRatio"
  ],
  "additionalProperties": false
},
"FloorplanRoom": {
  "title": "FloorplanRoom",
  "description": "A description for single room for this floorplan.",
  "type": "object",
  "properties": {
    "Type": {
      "title": "Type",
      "enum": [
        "Bedroom",
        "Bathroom",
        "Living",
        "Dining",
        "Kitchen",
        "Hallway",
        "Closet"
      ],
      "type": "string"
    },
    "Name": {
      "title": "Name",
      "maxLength": 255,
      "minLength": 1,
      "pattern": "^[a-zA-Z0-9_\\- ]*$",
      "type": "string"
    }
  }
}
```

```

    },
    "OriginalName": {
      "title": "Originalname",
      "type": "string"
    },
    },
    "DesiredShape": {
      "title": "Desiredshape",
      "default": {
        "Area": 20.0,
        "AspectRatio": {
          "x": 1.0,
          "y": 1.0
        }
      }
    },
    },
    "allOf": [
      {
        "$ref": "#/definitions/Rectangle"
      }
    ]
  }
},
"required": [
  "Type",
  "Name"
],
"additionalProperties": false
},
"FloorplanConnection": {
  "title": "FloorplanConnection",
  "description": "Describes the desired layout of the rooms and their adjacent rooms. A connection can be either a doorway or\nan open space without any walls. Two rooms cannot both share an interior doorway and an opening.\n\nThe same two rooms can have multiple doorways, up to a limit.",
  "type": "object",
  "properties": {
    "Location": {
      "title": "Location",
      "type": "array",
      "items": {
        "type": "string"
      }
    },
    "minItems": 2,
    "maxItems": 2
  }
},

```



```

    "ConnectionType": {
      "title": "Connectiontype",
      "enum": [
        "Doorway",
        "Opening"
      ],
      "type": "string"
    }
  },
  "required": [
    "Location",
    "ConnectionType"
  ],
  "additionalProperties": false
},
"FloorplanTemplate": {
  "title": "FloorplanTemplate",
  "description": "The top-level floorplan template that parameterizes the randomly generated architectural layout. By default, a residential floorplan with bedroom and living room are generated with a random doorway or opening connection. The footprint contributes to the overall shape of the floor layout along with rooms. The footprint shape is desired as it is a preference and not guaranteed. The ceiling determines the height of the walls. There are minimum and maximum ceiling heights. The ceiling height is guaranteed. Rooms are required. Each room has a desired shape. Together, the room shapes and footprint determine floor layout. The room types contribute to the layout and are used when randomly selecting furniture and materials for the walls and floors. DesiredConnections are optional. Two rooms are connected if they share a wall and doorway or adjacent without any wall aka \"opening\". All rooms are guaranteed to be connected randomly if they are not specified in the connections list. Connections that are specified are _not_ guaranteed but will be attempted as best-effort.",
  "type": "object",
  "properties": {
    "Footprint": {
      "title": "Footprint",
      "default": {
        "DesiredAspectRatio": {
          "x": 1.0,
          "y": 1.0
        }
      }
    },
    "allOf": [
      {
        "$ref": "#/definitions/FloorplanFootprint"
      }
    ]
  }
}

```

```
    }
  ]
},
"Ceiling": {
  "title": "Ceiling",
  "default": {
    "Height": 3.0
  },
  "allOf": [
    {
      "$ref": "#/definitions/FloorplanCeiling"
    }
  ]
},
"Rooms": {
  "title": "Rooms",
  "default": [
    {
      "Type": "Living",
      "Name": "My_Living_Room",
      "OriginalName": "My Living Room",
      "DesiredShape": {
        "Area": 20.0,
        "AspectRatio": {
          "x": 1.0,
          "y": 1.0
        }
      }
    }
  ]
},
"type": "array",
"items": {
  "$ref": "#/definitions/FloorplanRoom"
},
"minItems": 1,
"maxItems": 6
},
"DesiredConnections": {
  "title": "Desiredconnections",
  "default": [],
  "type": "array",
  "items": {
    "$ref": "#/definitions/FloorplanConnection"
  }
},
```

```
        "minItems": 0,
        "maxItems": 12
    }
},
"additionalProperties": false
},
"RoomNameList": {
    "title": "RoomNameList",
    "description": "The set of all rooms matching any of the listed room names.",
    "type": "object",
    "properties": {
        "RoomNames": {
            "title": "Roomnames",
            "type": "array",
            "items": {
                "type": "string"
            },
            "minItems": 1,
            "maxItems": 6
        }
    },
    "required": [
        "RoomNames"
    ],
    "additionalProperties": false
},
"RoomTypeList": {
    "title": "RoomTypeList",
    "description": "The set of all rooms matching any of the listed room types.",
    "type": "object",
    "properties": {
        "RoomTypes": {
            "title": "Roomtypes",
            "type": "array",
            "items": {
                "enum": [
                    "Bedroom",
                    "Bathroom",
                    "Living",
                    "Dining",
                    "Kitchen",
                    "Hallway",
                    "Closet"
                ]
            }
        }
    },
    "required": [
        "RoomTypes"
    ],
    "additionalProperties": false
},
"additionalProperties": false
}
```

```

        "type": "string"
      },
      "minItems": 1,
      "maxItems": 7
    }
  },
  "required": [
    "RoomTypes"
  ],
  "additionalProperties": false
},
"RoomPairTargetFilter": {
  "title": "RoomPairTargetFilter",
  "description": "Defines a target set as a pair of rooms. The pairs are defined as
the cross product of two lists\nFrom and To.",
  "type": "object",
  "properties": {
    "From": {
      "title": "From",
      "anyOf": [
        {
          "$ref": "#/definitions/RoomNameList"
        },
        {
          "$ref": "#/definitions/RoomTypeList"
        }
      ]
    },
    "To": {
      "title": "To",
      "anyOf": [
        {
          "$ref": "#/definitions/RoomNameList"
        },
        {
          "$ref": "#/definitions/RoomTypeList"
        }
      ]
    }
  }
},
"required": [
  "From",
  "To"
],

```

```

    "additionalProperties": false
  },
  "DoorOpenPosition": {
    "title": "DoorOpenPosition",
    "description": "Defines the amount of openness of an InteriorDoor.\n\nThe range
for Percent is [0., 100.]",
    "type": "object",
    "properties": {
      "Percent": {
        "title": "Percent",
        "default": 100.0,
        "anyOf": [
          {
            "type": "number",
            "minimum": 0.0,
            "maximum": 100.0
          },
          {
            "const": "Random",
            "type": "string"
          }
        ]
      }
    }
  },
  "additionalProperties": false
},
  "DoorInitialState": {
    "title": "DoorInitialState",
    "description": "Defines the initial state for an InteriorDoor object\n
\nOpenPosition specifies how much the door should be open.",
    "type": "object",
    "properties": {
      "OpenPosition": {
        "title": "Openposition",
        "default": {
          "Percent": 100.0
        },
        "allOf": [
          {
            "$ref": "#/definitions/DoorOpenPosition"
          }
        ]
      }
    }
  },
},

```

```

    "additionalProperties": false
  },
  "InteriorDoor": {
    "title": "InteriorDoor",
    "description": "Custom configuration for each Doorway Set.\n\nInitial State of doors includes the ability to configure how much the door should be open in\npercent [0., 100.]",
    "type": "object",
    "properties": {
      "InitialState": {
        "title": "Initialstate",
        "default": {
          "OpenPosition": {
            "Percent": 100.0
          }
        },
      },
      "allOf": [
        {
          "$ref": "#/definitions/DoorInitialState"
        }
      ]
    }
  },
  "additionalProperties": false
},
"InteriorDoorwaySet": {
  "title": "InteriorDoorwaySet",
  "description": "A set of doors to randomly assign to a set of interior target elements.\n\nThe target set determines *what room pairs* are receive the doors as specified in `Door`.\nRooms may be targeted by room type or room name.\n\nThe Door customizes the configuration for doors added in the specified target set.",
  "type": "object",
  "properties": {
    "Name": {
      "title": "Name",
      "maxLength": 255,
      "minLength": 1,
      "pattern": "^[a-zA-Z0-9_\\- ]*$",
      "type": "string"
    },
    "TargetSet": {
      "title": "Targetset",
      "anyOf": [
        {

```

```

        "const": "Target.All",
        "type": "string"
    },
    {
        "$ref": "#/definitions/RoomPairTargetFilter"
    }
]
},
"Door": {
    "title": "Door",
    "anyOf": [
        {
            "$ref": "#/definitions/InteriorDoor"
        },
        {
            "const": null
        }
    ]
}
},
"required": [
    "Name",
    "TargetSet"
],
"additionalProperties": false
},
"InteriorDoorways": {
    "title": "InteriorDoorways",
    "description": "Describes the interior template parameters for all doorways for
this floorplan.\nAll doorways not explicitly targeted will have a random door assigned
fully opened.",
    "type": "object",
    "properties": {
        "DoorwaySets": {
            "title": "Doorwaysets",
            "default": [],
            "type": "array",
            "items": {
                "$ref": "#/definitions/InteriorDoorwaySet"
            },
            "minItems": 0,
            "maxItems": 13
        }
    }
},
},

```

```

    "additionalProperties": false
  },
  "MaterialSetByMaterialType": {
    "title": "MaterialSetByMaterialType",
    "description": "The set of materials that match any of the material types listed.
An empty\nset is invalid since all targets require materials.",
    "type": "object",
    "properties": {
      "MaterialTypes": {
        "title": "Materialtypes",
        "type": "array",
        "items": {
          "type": "string"
        },
        "minItems": 1
      }
    },
    "required": [
      "MaterialTypes"
    ],
    "additionalProperties": false
  },
  "InteriorMaterialSet": {
    "title": "InteriorMaterialSet",
    "description": "A set of sample materials to randomly assign to a set of interior
target elements.\n\nThe target set determines *what rooms* receive the materials
in the sample\nset. The targets in a room are the walls and flooring. Rooms may be
targeted\nby room type or room name.\n\nThe sample set determines *what materials* to
randomly select for the\ntarget rooms' walls and floors.\n\nThe sample set is optional
and when not specified (null) materials are\nrandomly selected according to the room
type for each room in the target\nset.\n\nA sample set with an empty material set is
invalid since all wall\nand flooring targets require materials.",
    "type": "object",
    "properties": {
      "Name": {
        "title": "Name",
        "maxLength": 255,
        "minLength": 1,
        "pattern": "^[a-zA-Z0-9_\\- ]*$",
        "type": "string"
      },
      "TargetSet": {
        "title": "Targetset",
        "anyOf": [

```



```

    {
      "const": "Target.All",
      "type": "string"
    },
    {
      "anyOf": [
        {
          "$ref": "#/definitions/RoomNameList"
        },
        {
          "$ref": "#/definitions/RoomTypeList"
        }
      ]
    }
  ],
  "SampleSet": {
    "$ref": "#/definitions/MaterialSetByMaterialType"
  }
},
"required": [
  "Name",
  "TargetSet"
],
"additionalProperties": false
},
"InteriorFlooring": {
  "title": "InteriorFlooring",
  "description": "Describes the interior template parameters for all floors for
this floorplan.\nAll floors not explicitly targeted will have a random floor material
assigned by room type.",
  "type": "object",
  "properties": {
    "MaterialSets": {
      "title": "Materialsets",
      "default": [],
      "type": "array",
      "items": {
        "$ref": "#/definitions/InteriorMaterialSet"
      }
    },
    "minItems": 0,
    "maxItems": 6
  }
},
},

```

```
    "additionalProperties": false
  },
  "InteriorWalls": {
    "title": "InteriorWalls",
    "description": "Describes the interior template parameters for all walls for
this floorplan.\nAll walls not explicitly targeted will have a random wall material
assigned by room type.",
    "type": "object",
    "properties": {
      "MaterialSets": {
        "title": "Materialsets",
        "default": [],
        "type": "array",
        "items": {
          "$ref": "#/definitions/InteriorMaterialSet"
        },
        "minItems": 0,
        "maxItems": 6
      }
    },
    "additionalProperties": false
  },
  "ModelTypeList": {
    "title": "ModelTypeList",
    "description": "The set of all models matching any of the listed model types.\nAn
empty set means zero models to sample/select.",
    "type": "object",
    "properties": {
      "ModelTypes": {
        "title": "Modeltypes",
        "type": "array",
        "items": {
          "enum": [
            "Baths",
            "BarCabinets",
            "Beds",
            "Bookcases",
            "CoffeeTables",
            "ConsoleTables",
            "CornerCabinets",
            "DeskChairs",
            "Desks",
            "DiningChairs",
            "DiningTables",
```

```

        "DishWashers",
        "Dressers",
        "EndAndSideTables",
        "FloorLamps",
        "Fridges",
        "LivingRoomChairs",
        "KitchenIslandsAndCarts",
        "MediaStorage",
        "Nightstands",
        "Ottomans",
        "Ovens",
        "ServingCarts",
        "Showers",
        "SideboardsAndBuffets",
        "Sofas",
        "Storage",
        "StorageBenches",
        "Toilets",
        "VanityCounters",
        "WashingMachinesAndDryers"
    ],
    "type": "string"
},
"minItems": 0
}
},
"required": [
    "ModelTypes"
],
"additionalProperties": false
},
"FurnitureArrangementSet": {
    "title": "FurnitureArrangementSet",
    "description": "Describes the interior template for placing furniture in one
or more rooms.\n\n- TargetSet is the set of rooms to furnish, filter by room name or
room\n type.\n- SampleSet is a set of all furnishing models to randomly choose and\n
place.\n- DesiredSpatialDensity is the desired level of free space after placing\n
furniture.",
    "type": "object",
    "properties": {
        "Name": {
            "title": "Name",
            "maxLength": 255,
            "minLength": 1,

```

```
    "pattern": "^[a-zA-Z0-9_\\- ]*$",
    "type": "string"
  },
  "TargetSet": {
    "title": "Targetset",
    "anyOf": [
      {
        "const": "Target.All",
        "type": "string"
      },
      {
        "anyOf": [
          {
            "$ref": "#/definitions/RoomNameList"
          },
          {
            "$ref": "#/definitions/RoomTypeList"
          }
        ]
      }
    ]
  },
  "SampleSet": {
    "$ref": "#/definitions/ModelTypeList"
  },
  "DesiredSpatialDensity": {
    "title": "Desiredspatialdensity",
    "default": "Moderate",
    "enum": [
      "Sparse",
      "Moderate",
      "Dense"
    ],
    "type": "string"
  }
},
"required": [
  "Name",
  "TargetSet"
],
"additionalProperties": false
},
"InteriorFurnishings": {
  "title": "InteriorFurnishings",
```

```

    "description": "Describes the types of furniture models for randomly placing into
each room\nin the world. Rooms are targeted by room type or room name. Rooms that
are\nnot targeted are furnished at random by their room type with moderate density.
\ndensity. For an empty room, specify an empty sample set.",
    "type": "object",
    "properties": {
      "FurnitureArrangements": {
        "title": "Furniturearrangements",
        "default": [],
        "type": "array",
        "items": {
          "$ref": "#/definitions/FurnitureArrangementSet"
        },
        "minItems": 0,
        "maxItems": 6
      }
    },
    "additionalProperties": false
  },
  "InteriorTemplate": {
    "title": "InteriorTemplate",
    "description": "Top-level template for parameterizing the interior finishes and
furnishings for\nthis floorplan.",
    "type": "object",
    "properties": {
      "Doorways": {
        "title": "Doorways",
        "default": {
          "DoorwaySets": []
        },
        "allOf": [
          {
            "$ref": "#/definitions/InteriorDoorways"
          }
        ]
      },
      "Flooring": {
        "title": "Flooring",
        "default": {
          "MaterialSets": []
        },
        "allOf": [
          {
            "$ref": "#/definitions/InteriorFlooring"
          }
        ]
      }
    }
  }
}

```

```

    }
  ]
},
"Walls": {
  "title": "Walls",
  "default": {
    "MaterialSets": []
  },
  "allOf": [
    {
      "$ref": "#/definitions/InteriorWalls"
    }
  ]
},
"Furniture": {
  "title": "Furniture",
  "default": {
    "FurnitureArrangements": []
  },
  "allOf": [
    {
      "$ref": "#/definitions/InteriorFurnishings"
    }
  ]
},
"additionalProperties": false
},
"FloorTemplate": {
  "title": "FloorTemplate",
  "description": "Describes a single floor within a building. Defaults to a
single residential room\nof a random type and size, and the interior is randomly
furnished.",
  "type": "object",
  "properties": {
    "Floorplan": {
      "title": "Floorplan",
      "default": {
        "Footprint": {
          "DesiredAspectRatio": {
            "x": 1.0,
            "y": 1.0
          }
        }
      }
    }
  }
},

```

```
    "Ceiling": {
      "Height": 3.0
    },
    "Rooms": [
      {
        "Type": "Living",
        "Name": "My_Living_Room",
        "OriginalName": "My Living Room",
        "DesiredShape": {
          "Area": 20.0,
          "AspectRatio": {
            "x": 1.0,
            "y": 1.0
          }
        }
      }
    ],
    "DesiredConnections": []
  },
  "allOf": [
    {
      "$ref": "#/definitions/FloorplanTemplate"
    }
  ]
},
"Interior": {
  "title": "Interior",
  "default": {
    "Doorways": {
      "DoorwaySets": []
    },
    "Flooring": {
      "MaterialSets": []
    },
    "Walls": {
      "MaterialSets": []
    },
    "Furniture": {
      "FurnitureArrangements": []
    }
  }
},
"allOf": [
  {
    "$ref": "#/definitions/InteriorTemplate"
```

```
    }
  ]
}
},
"additionalProperties": false
},
"BuildingTemplate": {
  "title": "BuildingTemplate",
  "description": "Describes a building to be randomly generated. Defaults to one residential floor.",
  "type": "object",
  "properties": {
    "Floors": {
      "title": "Floors",
      "default": [
        {
          "Floorplan": {
            "Footprint": {
              "DesiredAspectRatio": {
                "x": 1.0,
                "y": 1.0
              }
            },
            "Ceiling": {
              "Height": 3.0
            },
            "Rooms": [
              {
                "Type": "Living",
                "Name": "My_Living_Room",
                "OriginalName": "My Living Room",
                "DesiredShape": {
                  "Area": 20.0,
                  "AspectRatio": {
                    "x": 1.0,
                    "y": 1.0
                  }
                }
              }
            ],
            "DesiredConnections": []
          },
          "Interior": {
            "Doorways": {
```



```

        "DoorwaySets": []
    },
    "Flooring": {
        "MaterialSets": []
    },
    "Walls": {
        "MaterialSets": []
    },
    "Furniture": {
        "FurnitureArrangements": []
    }
}
],
"type": "array",
"items": {
    "$ref": "#/definitions/FloorTemplate"
},
"minItems": 1,
"maxItems": 1
}
},
"additionalProperties": false
}
}
}

```

## Version 1

The following is the template for the Version 1 schema.

```

{
  "title": "WorldTemplate",
  "description": "The top-level template for parameterizing a randomly generated world. By default, a single\nresidential building with one floor and one room is generated.",
  "type": "object",
  "properties": {
    "Version": {
      "title": "Version",
      "default": "1",
      "type": "string"
    },
    "Buildings": {

```

```
"title": "Buildings",
"default": [
  {
    "Floors": [
      {
        "Floorplan": {
          "Footprint": {
            "DesiredAspectRatio": {
              "x": 1.0,
              "y": 1.0
            }
          },
          "Ceiling": {
            "Height": 3.0
          },
          "Rooms": [
            {
              "Type": "Living",
              "Name": "My Living Room",
              "DesiredShape": {
                "Area": 20.0,
                "AspectRatio": {
                  "x": 1.0,
                  "y": 1.0
                }
              }
            },
            {
              "Type": "Bedroom",
              "Name": "My Bedroom",
              "DesiredShape": {
                "Area": 20.0,
                "AspectRatio": {
                  "x": 1.0,
                  "y": 1.0
                }
              }
            }
          ],
          "DesiredConnections": []
        },
        "Interior": {
          "Flooring": {
            "MaterialSets": []
          }
        }
      }
    ]
  }
]
```

```

        },
        "Walls": {
            "MaterialSets": []
        },
        "Furniture": {
            "FurnitureArrangements": []
        }
    }
}
]
}
],
"type": "array",
"items": {
    "$ref": "#/definitions/BuildingTemplate"
},
"minItems": 1,
"maxItems": 1
}
},
"additionalProperties": false,
"definitions": {
    "AspectRatio": {
        "title": "AspectRatio",
        "type": "object",
        "properties": {
            "x": {
                "title": "X",
                "default": 1,
                "minimum": 1,
                "maximum": 4,
                "type": "number"
            },
            "y": {
                "title": "Y",
                "default": 1,
                "minimum": 1,
                "maximum": 4,
                "type": "number"
            }
        },
    },
    "additionalProperties": false
},
"FloorplanFootprint": {

```

```
"title": "FloorplanFootprint",
"description": "The desired footprint of this floorplan.",
"type": "object",
"properties": {
  "DesiredAspectRatio": {
    "title": "Desiredaspectratio",
    "default": {
      "x": 1.0,
      "y": 1.0
    },
    "allOf": [
      {
        "$ref": "#/definitions/AspectRatio"
      }
    ]
  }
},
"additionalProperties": false
},
"FloorplanCeiling": {
  "title": "FloorplanCeiling",
  "description": "The height of the ceiling for this floorplan in metres.",
  "type": "object",
  "properties": {
    "Height": {
      "title": "Height",
      "default": 3.0,
      "type": "number",
      "minimum": 2.4,
      "maximum": 4.0
    }
  },
  "additionalProperties": false
},
"Rectangle": {
  "title": "Rectangle",
  "description": "A rectangle defined by area in square metres and aspect ratio.",
  "type": "object",
  "properties": {
    "Area": {
      "title": "Area",
      "type": "number"
    },
    "AspectRatio": {
```

```
        "$ref": "#/definitions/AspectRatio"
    }
},
"required": [
    "Area",
    "AspectRatio"
],
"additionalProperties": false
},
"FloorplanRoom": {
    "title": "FloorplanRoom",
    "description": "A description for single room for this floorplan.",
    "type": "object",
    "properties": {
        "Type": {
            "title": "Type",
            "enum": [
                "Bedroom",
                "Bathroom",
                "Living",
                "Dining",
                "Kitchen",
                "Hallway",
                "Closet"
            ],
            "type": "string"
        },
        "Name": {
            "title": "Name",
            "type": "string"
        },
        "DesiredShape": {
            "title": "Desiredshape",
            "default": {
                "Area": 20.0,
                "AspectRatio": {
                    "x": 1.0,
                    "y": 1.0
                }
            },
            "type": "object",
            "required": [
                "Area",
                "AspectRatio"
            ],
            "additionalProperties": false
        },
        "allOf": [
            {
                "$ref": "#/definitions/Rectangle"
            }
        ]
    }
}
```

```

    ]
  }
},
"required": [
  "Type",
  "Name"
],
"additionalProperties": false
},
"FloorplanConnection": {
  "title": "FloorplanConnection",
  "description": "Describes the desired layout of the rooms and their adjacent
rooms. A connection can be either a doorway or \nan open space without any walls. Two
rooms cannot both share an interior doorway and an opening. \nThe same two rooms can
have multiple doorways, up to a limit.",
  "type": "object",
  "properties": {
    "Location": {
      "title": "Location",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 2,
      "maxItems": 2
    },
    "ConnectionType": {
      "title": "Connectiontype",
      "enum": [
        "Doorway",
        "Opening"
      ],
      "type": "string"
    }
  },
  "required": [
    "Location",
    "ConnectionType"
  ],
  "additionalProperties": false
},
"FloorplanTemplate": {
  "title": "FloorplanTemplate",

```

```

    "description": "The top-level floorplan template that parameterizes the randomly
generated \narchitectural layout. By default, a residential floorplan with bedroom
and \nliving room are generated with a random doorway or opening connection. \n\nThe
footprint contributes to the overall shape of the floor layout along\nwith rooms. The
footprint shape is desired as it is a preference and not\nnguaranteed.\n\nThe ceiling
determines the height of the walls. There are minimum and\nmaximum ceiling heights.
The ceiling height is guaranteed.\n\nRooms are required. Each room has a desired
shape. Together, the room\nshapes and footprint determine floor layout. The room
types contribute to\nthe layout and are used when randomly selecting furniture and
materials for\nthe walls and floors.\n\nDesiredConnections are optional. Two rooms are
connected if they share a\nwall and doorway or adjacent without any wall aka \n"opening
\n". All rooms are\nnguaranteed to be connected randomly if they are not specified in the
\nconnections list. Connections that are specified are _not_ guaranteed but\nwill be
attempted as best-effort.",
    "type": "object",
    "properties": {
      "Footprint": {
        "title": "Footprint",
        "default": {
          "DesiredAspectRatio": {
            "x": 1.0,
            "y": 1.0
          }
        },
        "allOf": [
          {
            "$ref": "#/definitions/FloorplanFootprint"
          }
        ]
      },
      "Ceiling": {
        "title": "Ceiling",
        "default": {
          "Height": 3.0
        },
        "allOf": [
          {
            "$ref": "#/definitions/FloorplanCeiling"
          }
        ]
      },
      "Rooms": {
        "title": "Rooms",
        "default": [

```

```
{
  "Type": "Living",
  "Name": "My Living Room",
  "DesiredShape": {
    "Area": 20.0,
    "AspectRatio": {
      "x": 1.0,
      "y": 1.0
    }
  }
},
{
  "Type": "Bedroom",
  "Name": "My Bedroom",
  "DesiredShape": {
    "Area": 20.0,
    "AspectRatio": {
      "x": 1.0,
      "y": 1.0
    }
  }
}
],
"type": "array",
"items": {
  "$ref": "#/definitions/FloorplanRoom"
},
"minItems": 1,
"maxItems": 6
},
"DesiredConnections": {
  "title": "Desiredconnections",
  "default": [],
  "type": "array",
  "items": {
    "$ref": "#/definitions/FloorplanConnection"
  },
  "minItems": 0,
  "maxItems": 12
}
},
"additionalProperties": false
},
"RoomNameList": {
```



```
"title": "RoomNameList",
"description": "The set of all rooms matching any of the listed room names.",
"type": "object",
"properties": {
  "RoomNames": {
    "title": "Roomnames",
    "type": "array",
    "items": {
      "type": "string"
    }
  }
},
"required": [
  "RoomNames"
],
"additionalProperties": false
},
"RoomTypeList": {
  "title": "RoomTypeList",
  "description": "The set of all rooms matching any of the listed room types.",
  "type": "object",
  "properties": {
    "RoomTypes": {
      "title": "Roomtypes",
      "type": "array",
      "items": {
        "enum": [
          "Bedroom",
          "Bathroom",
          "Living",
          "Dining",
          "Kitchen",
          "Hallway",
          "Closet"
        ],
        "type": "string"
      }
    }
  },
  "required": [
    "RoomTypes"
  ],
  "additionalProperties": false
},
```

```

"MaterialSetByMaterialType": {
  "title": "MaterialSetByMaterialType",
  "description": "The set of materials that match any of the material types listed.
An empty\nset is invalid since all targets require materials.",
  "type": "object",
  "properties": {
    "MaterialTypes": {
      "title": "Materialtypes",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 1
    }
  },
  "required": [
    "MaterialTypes"
  ],
  "additionalProperties": false
},
"InteriorMaterialSet": {
  "title": "InteriorMaterialSet",
  "description": "A set of sample materials to randomly assign to a set of interior
target elements.\n\nThe target set determines *what rooms* receive the materials
in the sample\nset. The targets in a room are the walls and flooring. Rooms may be
targeted \nby room type or room name. \n\nThe sample set determines *what materials*
to randomly select for the\ntarget rooms' walls and floors. \n\nThe sample set is
optional and when not specified (null) materials are\nrandomly selected according to
the room type for each room in the target\nset.\n\nA sample set with an empty material
set is invalid since all wall \nand flooring targets require materials.",
  "type": "object",
  "properties": {
    "Name": {
      "title": "Name",
      "type": "string"
    },
    "TargetSet": {
      "title": "Targetset",
      "anyOf": [
        {
          "$ref": "#/definitions/RoomNameList"
        },
        {
          "$ref": "#/definitions/RoomTypeList"
        }
      ]
    }
  }
}

```

```

    }
  ]
},
"SampleSet": {
  "$ref": "#/definitions/MaterialSetByMaterialType"
}
},
"required": [
  "Name",
  "TargetSet"
],
"additionalProperties": false
},
"InteriorFlooring": {
  "title": "InteriorFlooring",
  "description": "Describes the interior template parameters for all floors for
this floorplan.\nAll floors not explicitly targeted will have a random floor material
assigned by room type.",
  "type": "object",
  "properties": {
    "MaterialSets": {
      "title": "Materialsets",
      "default": [],
      "type": "array",
      "items": {
        "$ref": "#/definitions/InteriorMaterialSet"
      },
      "minItems": 0,
      "maxItems": 6
    }
  },
  "additionalProperties": false
},
"InteriorWalls": {
  "title": "InteriorWalls",
  "description": "Describes the interior template parameters for all walls for
this floorplan.\nAll walls not explicitly targeted will have a random wall material
assigned by room type.",
  "type": "object",
  "properties": {
    "MaterialSets": {
      "title": "Materialsets",
      "default": [],
      "type": "array",

```

```

        "items": {
            "$ref": "#/definitions/InteriorMaterialSet"
        },
        "minItems": 0,
        "maxItems": 6
    }
},
"additionalProperties": false
},
"ModelTypeList": {
    "title": "ModelTypeList",
    "description": "The set of all models matching any of the listed model types.\nAn empty set means zero models to sample/select.",
    "type": "object",
    "properties": {
        "ModelTypes": {
            "title": "Modeltypes",
            "type": "array",
            "items": {
                "type": "string"
            },
            "minItems": 0
        }
    },
    "required": [
        "ModelTypes"
    ],
    "additionalProperties": false
},
"FurnitureArrangementSet": {
    "title": "FurnitureArrangementSet",
    "description": "Describes the interior template for placing furniture in one or more rooms.\n\n- TargetSet is the set of rooms to furnish, filter by room name or room type.\n- SampleSet is a set of all furnishing models to randomly choose and place. \n- DesiredSpatialDensity is the desired level of free space after placing furniture.",
    "type": "object",
    "properties": {
        "Name": {
            "title": "Name",
            "type": "string"
        },
        "TargetSet": {
            "title": "Targetset",

```

```

    "anyOf": [
      {
        "$ref": "#/definitions/RoomNameList"
      },
      {
        "$ref": "#/definitions/RoomTypeList"
      }
    ],
    "SampleSet": {
      "$ref": "#/definitions/ModelTypeList"
    },
    "DesiredSpatialDensity": {
      "title": "Desiredspatialdensity",
      "default": "Moderate",
      "enum": [
        "Sparse",
        "Moderate",
        "Dense"
      ],
      "type": "string"
    }
  },
  "required": [
    "Name",
    "TargetSet"
  ],
  "additionalProperties": false
},
"InteriorFurnishings": {
  "title": "InteriorFurnishings",
  "description": "Describes the types of furniture models for randomly placing into each room\nin the world. Rooms are targeted by room type or room name. Rooms that are\nnot targeted are furnished at random by their room type with moderate density.\ndensity. For an empty room, specify an empty sample set.",
  "type": "object",
  "properties": {
    "FurnitureArrangements": {
      "title": "Furniturearrangements",
      "default": [],
      "type": "array",
      "items": {
        "$ref": "#/definitions/FurnitureArrangementSet"
      }
    },

```

```
        "minItems": 0,
        "maxItems": 6
    }
},
"additionalProperties": false
},
"InteriorTemplate": {
    "title": "InteriorTemplate",
    "description": "Top-level template for parameterizing the interior finishes and
furnishings for\nthis floorplan.",
    "type": "object",
    "properties": {
        "Flooring": {
            "title": "Flooring",
            "default": {
                "MaterialSets": []
            },
            "allOf": [
                {
                    "$ref": "#/definitions/InteriorFlooring"
                }
            ]
        },
        "Walls": {
            "title": "Walls",
            "default": {
                "MaterialSets": []
            },
            "allOf": [
                {
                    "$ref": "#/definitions/InteriorWalls"
                }
            ]
        },
        "Furniture": {
            "title": "Furniture",
            "default": {
                "FurnitureArrangements": []
            },
            "allOf": [
                {
                    "$ref": "#/definitions/InteriorFurnishings"
                }
            ]
        }
    }
}
```

```
    }
  },
  "additionalProperties": false
},
"FloorTemplate": {
  "title": "FloorTemplate",
  "description": "Describes a single floor within a building. Defaults to a
single residential room\nof a random type and size, and the interior is randomly
furnished.",
  "type": "object",
  "properties": {
    "Floorplan": {
      "title": "Floorplan",
      "default": {
        "Footprint": {
          "DesiredAspectRatio": {
            "x": 1.0,
            "y": 1.0
          }
        },
        "Ceiling": {
          "Height": 3.0
        }
      },
      "Rooms": [
        {
          "Type": "Living",
          "Name": "My Living Room",
          "DesiredShape": {
            "Area": 20.0,
            "AspectRatio": {
              "x": 1.0,
              "y": 1.0
            }
          }
        },
        {
          "Type": "Bedroom",
          "Name": "My Bedroom",
          "DesiredShape": {
            "Area": 20.0,
            "AspectRatio": {
              "x": 1.0,
              "y": 1.0
            }
          }
        }
      ]
    }
  }
}
```

```

        }
    }
    ],
    "DesiredConnections": []
},
"allOf": [
    {
        "$ref": "#/definitions/FloorplanTemplate"
    }
]
},
"Interior": {
    "title": "Interior",
    "default": {
        "Flooring": {
            "MaterialSets": []
        },
        "Walls": {
            "MaterialSets": []
        },
        "Furniture": {
            "FurnitureArrangements": []
        }
    },
    "allOf": [
        {
            "$ref": "#/definitions/InteriorTemplate"
        }
    ]
}
},
"additionalProperties": false
},
"BuildingTemplate": {
    "title": "BuildingTemplate",
    "description": "Describes a building to be randomly generated. Defaults to one residential floor.",
    "type": "object",
    "properties": {
        "Floors": {
            "title": "Floors",
            "default": [
                {
                    "Floorplan": {

```



```
"Footprint": {
  "DesiredAspectRatio": {
    "x": 1.0,
    "y": 1.0
  }
},
"Ceiling": {
  "Height": 3.0
},
"Rooms": [
  {
    "Type": "Living",
    "Name": "My Living Room",
    "DesiredShape": {
      "Area": 20.0,
      "AspectRatio": {
        "x": 1.0,
        "y": 1.0
      }
    }
  },
  {
    "Type": "Bedroom",
    "Name": "My Bedroom",
    "DesiredShape": {
      "Area": 20.0,
      "AspectRatio": {
        "x": 1.0,
        "y": 1.0
      }
    }
  }
],
"DesiredConnections": [],
"Interior": {
  "Flooring": {
    "MaterialSets": []
  },
  "Walls": {
    "MaterialSets": []
  },
  "Furniture": {
    "FurnitureArrangements": []
  }
}
```

```
        }
      }
    }
  ],
  "type": "array",
  "items": {
    "$ref": "#/definitions/FloorTemplate"
  },
  "minItems": 1,
  "maxItems": 1
}
},
"additionalProperties": false
}
}
```

## Sample world templates in JSON

The `templateBody` (simulation world template body) is an input parameter of the [CreateWorldTemplate](#) API. This parameter is a JSON-formatted string. The JSON specifies a simulation world template and contains the parameters Simulation WorldForge uses to generate worlds.

This section contains sample simulation world template bodies.

### Topics

- [One-bedroom house](#)
- [One room only](#)
- [Two rooms](#)

## One-bedroom house

The following example specifies a one-bedroom house. It specifies interior materials and furniture.

```
{
  "name": "OneBedroomHouse",
  "templateBody": {
    "Version": "2",
    "Buildings": [
```

```
{
  "Floors": [
    {
      "Floorplan": {
        "Footprint": {
          "DesiredAspectRatio": {
            "x": 1,
            "y": 1
          }
        },
        "Ceiling": {
          "Height": 3
        },
        "Rooms": [
          {
            "Type": "Bedroom",
            "Name": "Bedroom",
            "DesiredShape": {
              "Area": 25,
              "AspectRatio": {
                "x": 1,
                "y": 1.2
              }
            }
          },
          {
            "Type": "Living",
            "Name": "Living room",
            "DesiredShape": {
              "Area": 30,
              "AspectRatio": {
                "x": 1,
                "y": 1.5
              }
            }
          },
          {
            "Type": "Bathroom",
            "Name": "Bathroom",
            "DesiredShape": {
              "Area": 10,
              "AspectRatio": {
                "x": 1,
                "y": 1.5
              }
            }
          }
        ]
      }
    }
  ]
}
```

```
    }
  }
},
{
  "Type": "Kitchen",
  "Name": "Kitchen",
  "DesiredShape": {
    "Area": 15,
    "AspectRatio": {
      "x": 1.5,
      "y": 1
    }
  }
}
],
"DesiredConnections": [
  {
    "Location": [
      "Bathroom",
      "Living room"
    ],
    "ConnectionType": "Doorway"
  },
  {
    "Location": [
      "Living room",
      "Kitchen"
    ],
    "ConnectionType": "Opening"
  },
  {
    "Location": [
      "Bedroom",
      "Living room"
    ],
    "ConnectionType": "Doorway"
  }
]
},
"Interior": {
  "Flooring": {
    "MaterialSets": [
      {
        "Name": "Floorboard room types",
```

```

    "TargetSet": {
      "RoomTypes": [
        "Kitchen"
      ]
    },
    "SampleSet": {
      "MaterialTypes": [
        "Floorboards"
      ]
    }
  },
  {
    "Name": "Carpet room types",
    "TargetSet": {
      "RoomTypes": [
        "Living",
        "Bedroom"
      ]
    },
    "SampleSet": {
      "MaterialTypes": [
        "Carpet"
      ]
    }
  },
  {
    "Name": "Bathroom",
    "TargetSet": {
      "RoomNames": [
        "Bathroom"
      ]
    },
    "SampleSet": {
      "MaterialTypes": [
        "Parquetry"
      ]
    }
  }
],
"Walls": {
  "MaterialSets": [
    {
      "Name": "Brick room types",

```

```
        "TargetSet": {
          "RoomTypes": [
            "Living"
          ]
        },
        "SampleSet": {
          "MaterialTypes": [
            "Brick"
          ]
        }
      },
      {
        "Name": "Tiles room types",
        "TargetSet": {
          "RoomTypes": [
            "Bathroom"
          ]
        },
        "SampleSet": {
          "MaterialTypes": [
            "Tiles"
          ]
        }
      }
    ]
  },
  "Furniture": {
    "FurnitureArrangements": [
      {
        "Name": "Dense furniture room types",
        "TargetSet": {
          "RoomTypes": [
            "Living",
            "Bedroom",
            "Kitchen",
            "Bathroom"
          ]
        },
        "DesiredSpatialDensity": "Dense"
      }
    ]
  }
}
```

```
    ]
  }
]
}
```

## One room only

The following example specifies a one-bedroom house. It specifies interior furniture.

```
{
  "Version": "2",
  "Buildings": [
    {
      "Floors": [
        {
          "Floorplan": {
            "Footprint": {
              "DesiredAspectRatio": {
                "x": 1,
                "y": 1
              }
            },
            "Ceiling": {
              "Height": 3
            },
            "Rooms": [
              {
                "Type": "Bedroom",
                "Name": "Bedroom",
                "DesiredShape": {
                  "Area": 40,
                  "AspectRatio": {
                    "x": 1,
                    "y": 1.61
                  }
                }
              }
            ],
            "DesiredConnections": []
          },
          "Interior": {
            "Furniture": {
```

```

        "FurnitureArrangements": [
            {
                "Name": "Bedroom furniture",
                "TargetSet": {
                    "RoomNames": [
                        "Bedroom"
                    ]
                },
                "DesiredSpatialDensity": "Dense"
            }
        ]
    }
}

```

## Two rooms

The following example specifies a one-bedroom house. Simulation WorldForge determines details including floor material, wall material, furniture placement, and connectivity.

```

{
  "name": "TwoRooms",
  "templateBody": {
    "Version": "2",
    "Buildings": [
      {
        "Floors": [
          {
            "Floorplan": {
              "Footprint": {
                "DesiredAspectRatio": {
                  "x": 1,
                  "y": 1
                }
              }
            },
            "Ceiling": {
              "Height": 3
            },
            "Rooms": [

```



```
    {
      "Type": "Living",
      "Name": "Living room",
      "DesiredShape": {
        "Area": 30,
        "AspectRatio": {
          "x": 1,
          "y": 1.5
        }
      }
    },
    {
      "Type": "Dining",
      "Name": "Dining room",
      "DesiredShape": {
        "Area": 30,
        "AspectRatio": {
          "x": 1,
          "y": 1.5
        }
      }
    }
  ],
  "DesiredConnections": [],
  "Interior": {}
}
]
}
}
```

## Managing simulation world templates

This section provides information about how you can create and manage simulation world templates. You use a simulation world template to specify how Simulation WorldForge generates worlds. You can specify the number of rooms, how they are connected, furniture, and the material types used for interior elements.

To learn more about simulation world templates, begin with [Understanding simulation world templates](#). You can also review the JSON templateBody that describes a simulation world template. For more information, see [JSON schema for simulation world template body](#).

## Topics

- [Creating a simulation world template](#)
- [Viewing a simulation world template](#)
- [Modifying a simulation world template](#)
- [Deleting a simulation world template](#)
- [Simulation world template versions, features, and changes](#)

## Creating a simulation world template

Create a simulation world template to specify how Simulation WorldForge generates worlds. When your simulation world template is complete, create a world generation job to generate worlds with different room and interior configurations.

You can create a simulation world template from a sample template, a saved template, or from scratch. After the template is created, you can modify the floor plan, the interiors, and other details. For more information about modifying the simulation world template, see [Modifying a simulation world template](#).

### To create a simulation world template

Follow the steps on one of the following tabs:

Using the console

#### To create a simulation world template

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. On the AWS RoboMaker console, expand **Simulation WorldForge** on the left and then choose **World templates**.
3. On the **World templates** page, choose **Create template**.

4. On the **Create a world template** page, choose one of the template options. You can choose one of the preconfigured **Sample templates**, clone and modify a **Saved template**, or **Start from scratch** with a default world.
5. On the **Template detail** page, in the upper left, choose **Rename** and then specify a name for the template.
6. (Optional) Customize the floor plan and interior details. For more information, see [Understanding simulation world templates](#).
7. On the **Template detail** page, choose **Save and exit**.

## Using the AWS CLI

### Example

You can update the simulation world template using the AWS CLI. First, create a JSON document that specifies the worlds Simulation WorldForge generates. Next, use `create-world-template` to create the simulation world template.

For example, the following JSON document specifies a one-bedroom house.

```
{
  "title": "WorldTemplate",
  "description": "The top-level template for parameterizing a randomly generated world. By default, a single\nresidential building with one floor and one room is generated.",
  "type": "object",
  "properties": {
    "Version": {
      "title": "Version",
      "default": "1",
      "type": "string"
    },
    "Buildings": {
      "title": "Buildings",
      "default": [
        {
          "Floors": [
            {
              "Floorplan": {
                "Footprint": {
                  "DesiredAspectRatio": {
```

```
        "x": 1.0,
        "y": 1.0
      }
    },
    "Ceiling": {
      "Height": 3.0
    },
    "Rooms": [
      {
        "Type": "Living",
        "Name": "My Living Room",
        "DesiredShape": {
          "Area": 20.0,
          "AspectRatio": {
            "x": 1.0,
            "y": 1.0
          }
        }
      }
    ],
    "DesiredConnections": []
  },
  "Interior": {
    "Flooring": {
      "MaterialSets": []
    },
    "Walls": {
      "MaterialSets": []
    },
    "Furniture": {
      "FurnitureArrangements": []
    }
  }
]
}
],
"type": "array",
"items": {
  "$ref": "#/definitions/BuildingTemplate"
},
"minItems": 1,
"maxItems": 1
}
```

```
},
"additionalProperties": false,
"definitions": {
  "AspectRatio": {
    "title": "AspectRatio",
    "type": "object",
    "properties": {
      "x": {
        "title": "X",
        "default": 1,
        "minimum": 1,
        "maximum": 4,
        "type": "number"
      },
      "y": {
        "title": "Y",
        "default": 1,
        "minimum": 1,
        "maximum": 4,
        "type": "number"
      }
    }
  },
  "additionalProperties": false
},
"FloorplanFootprint": {
  "title": "FloorplanFootprint",
  "description": "The desired footprint of this floorplan.",
  "type": "object",
  "properties": {
    "DesiredAspectRatio": {
      "title": "Desiredaspectratio",
      "default": {
        "x": 1.0,
        "y": 1.0
      },
      "allOf": [
        {
          "$ref": "#/definitions/AspectRatio"
        }
      ]
    },
    "additionalProperties": false
  },
  "additionalProperties": false
},
```

```
"FloorplanCeiling": {
  "title": "FloorplanCeiling",
  "description": "The height of the ceiling for this floorplan in metres.",
  "type": "object",
  "properties": {
    "Height": {
      "title": "Height",
      "default": 3.0,
      "type": "number",
      "minimum": 2.4,
      "maximum": 4.0
    }
  },
  "additionalProperties": false
},
"Rectangle": {
  "title": "Rectangle",
  "description": "A rectangle defined by area in square metres and aspect
ratio.",
  "type": "object",
  "properties": {
    "Area": {
      "title": "Area",
      "type": "number"
    },
    "AspectRatio": {
      "$ref": "#/definitions/AspectRatio"
    }
  },
  "required": [
    "Area",
    "AspectRatio"
  ],
  "additionalProperties": false
},
"FloorplanRoom": {
  "title": "FloorplanRoom",
  "description": "A description for single room for this floorplan.",
  "type": "object",
  "properties": {
    "Type": {
      "title": "Type",
      "enum": [
        "Bedroom",
```

```

        "Bathroom",
        "Living",
        "Dining",
        "Kitchen",
        "Hallway",
        "Closet"
    ],
    "type": "string"
},
"Name": {
    "title": "Name",
    "maxLength": 255,
    "minLength": 1,
    "pattern": "^[a-zA-Z0-9_\\- ]*$",
    "type": "string"
},
"DesiredShape": {
    "title": "Desiredshape",
    "default": {
        "Area": 20.0,
        "AspectRatio": {
            "x": 1.0,
            "y": 1.0
        }
    }
},
"allOf": [
    {
        "$ref": "#/definitions/Rectangle"
    }
]
}
},
"required": [
    "Type",
    "Name"
],
"additionalProperties": false
},
"FloorplanConnection": {
    "title": "FloorplanConnection",
    "description": "Describes the desired layout of the rooms and their adjacent
rooms. A connection can be either a doorway or \nan open space without any walls.
Two rooms cannot both share an interior doorway and an opening. \nThe same two
rooms can have multiple doorways, up to a limit.",

```

```

    "type": "object",
    "properties": {
      "Location": {
        "title": "Location",
        "type": "array",
        "items": {
          "type": "string"
        },
        "minItems": 2,
        "maxItems": 2
      },
      "ConnectionType": {
        "title": "Connectiontype",
        "enum": [
          "Doorway",
          "Opening"
        ],
        "type": "string"
      }
    },
    "required": [
      "Location",
      "ConnectionType"
    ],
    "additionalProperties": false
  },
  "FloorplanTemplate": {
    "title": "FloorplanTemplate",
    "description": "The top-level floorplan template that parameterizes the randomly generated \narchitectural layout. By default, a residential floorplan with bedroom and \nliving room are generated with a random doorway or opening connection. \n\nThe footprint contributes to the overall shape of the floor layout along\nwith rooms. The footprint shape is desired as it is a preference and not \nguaranteed.\n\nThe ceiling determines the height of the walls. There are minimum and\nmaximum ceiling heights. The ceiling height is guaranteed.\n\nRooms are required. Each room has a desired shape. Together, the room\nshapes and footprint determine floor layout. The room types contribute to\nthe layout and are used when randomly selecting furniture and materials for\nthe walls and floors.\n\nDesiredConnections are optional. Two rooms are connected if they share a\nwall and doorway or adjacent without any wall aka \"opening\". All rooms are\nguaranteed to be connected randomly if they are not specified in the\nconnections list. Connections that are specified are _not_ guaranteed but\nwill be attempted as best-effort.",
    "type": "object",

```



```
"properties": {
  "Footprint": {
    "title": "Footprint",
    "default": {
      "DesiredAspectRatio": {
        "x": 1.0,
        "y": 1.0
      }
    },
    "allOf": [
      {
        "$ref": "#/definitions/FloorplanFootprint"
      }
    ]
  },
  "Ceiling": {
    "title": "Ceiling",
    "default": {
      "Height": 3.0
    },
    "allOf": [
      {
        "$ref": "#/definitions/FloorplanCeiling"
      }
    ]
  },
  "Rooms": {
    "title": "Rooms",
    "default": [
      {
        "Type": "Living",
        "Name": "My Living Room",
        "DesiredShape": {
          "Area": 20.0,
          "AspectRatio": {
            "x": 1.0,
            "y": 1.0
          }
        }
      }
    ],
    "type": "array",
    "items": {
      "$ref": "#/definitions/FloorplanRoom"
    }
  }
}
```

```
    },
    "minItems": 1,
    "maxItems": 6
  },
  "DesiredConnections": {
    "title": "Desiredconnections",
    "default": [],
    "type": "array",
    "items": {
      "$ref": "#/definitions/FloorplanConnection"
    },
    "minItems": 0,
    "maxItems": 12
  }
},
"additionalProperties": false
},
"RoomNameList": {
  "title": "RoomNameList",
  "description": "The set of all rooms matching any of the listed room names.",
  "type": "object",
  "properties": {
    "RoomNames": {
      "title": "Roomnames",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 1,
      "maxItems": 6
    }
  },
  "required": [
    "RoomNames"
  ],
  "additionalProperties": false
},
"RoomTypeList": {
  "title": "RoomTypeList",
  "description": "The set of all rooms matching any of the listed room types.",
  "type": "object",
  "properties": {
    "RoomTypes": {
      "title": "Roomtypes",
```

```
    "type": "array",
    "items": {
      "enum": [
        "Bedroom",
        "Bathroom",
        "Living",
        "Dining",
        "Kitchen",
        "Hallway",
        "Closet"
      ],
      "type": "string"
    },
    "minItems": 1,
    "maxItems": 7
  }
},
"required": [
  "RoomTypes"
],
"additionalProperties": false
},
"MaterialSetByMaterialType": {
  "title": "MaterialSetByMaterialType",
  "description": "The set of materials that match any of the material types listed. An empty\nset is invalid since all targets require materials.",
  "type": "object",
  "properties": {
    "MaterialTypes": {
      "title": "Materialtypes",
      "type": "array",
      "items": {
        "type": "string"
      },
      "minItems": 1
    }
  },
  "required": [
    "MaterialTypes"
  ],
  "additionalProperties": false
},
"InteriorMaterialSet": {
  "title": "InteriorMaterialSet",
```

```

    "description": "A set of sample materials to randomly assign to a set of
interior target elements.\n\nThe target set determines *what rooms* receive the
materials in the sample\nset. The targets in a room are the walls and flooring.
Rooms may be targeted\nby room type or room name.\n\nThe sample set determines
*what materials* to randomly select for the\ntarget rooms' walls and floors.\n
\nThe sample set is optional and when not specified (null) materials are\nrandomly
selected according to the room type for each room in the target\nset.\n\nA sample
set with an empty material set is invalid since all wall\nand flooring targets
require materials.",
    "type": "object",
    "properties": {
        "Name": {
            "title": "Name",
            "maxLength": 255,
            "minLength": 1,
            "pattern": "^[a-zA-Z0-9_\\- ]*$",
            "type": "string"
        },
        "TargetSet": {
            "title": "Targetset",
            "anyOf": [
                {
                    "$ref": "#/definitions/RoomNameList"
                },
                {
                    "$ref": "#/definitions/RoomTypeList"
                }
            ]
        },
        "SampleSet": {
            "$ref": "#/definitions/MaterialSetByMaterialType"
        }
    },
    "required": [
        "Name",
        "TargetSet"
    ],
    "additionalProperties": false
},
"InteriorFlooring": {
    "title": "InteriorFlooring",
    "description": "Describes the interior template parameters for all floors
for this floorplan.\nAll floors not explicitly targeted will have a random floor
material assigned by room type.",

```

```
"type": "object",
"properties": {
  "MaterialSets": {
    "title": "Materialsets",
    "default": [],
    "type": "array",
    "items": {
      "$ref": "#/definitions/InteriorMaterialSet"
    },
    "minItems": 0,
    "maxItems": 6
  }
},
"additionalProperties": false
},
"InteriorWalls": {
  "title": "InteriorWalls",
  "description": "Describes the interior template parameters for all walls for
this floorplan.\nAll walls not explicitly targeted will have a random wall material
assigned by room type.",
  "type": "object",
  "properties": {
    "MaterialSets": {
      "title": "Materialsets",
      "default": [],
      "type": "array",
      "items": {
        "$ref": "#/definitions/InteriorMaterialSet"
      },
      "minItems": 0,
      "maxItems": 6
    }
  },
  "additionalProperties": false
},
"ModelTypeList": {
  "title": "ModelTypeList",
  "description": "The set of all models matching any of the listed model types.
\nAn empty set means zero models to sample/select.",
  "type": "object",
  "properties": {
    "ModelTypes": {
      "title": "Modeltypes",
      "type": "array",
```

```
    "items": {
      "enum": [
        "Baths",
        "BarCabinets",
        "Beds",
        "Bookcases",
        "CoffeeTables",
        "ConsoleTables",
        "CornerCabinets",
        "DeskChairs",
        "Desks",
        "DiningChairs",
        "DiningTables",
        "DishWashers",
        "Dressers",
        "EndAndSideTables",
        "FloorLamps",
        "Fridges",
        "LivingRoomChairs",
        "KitchenIslandsAndCarts",
        "MediaStorage",
        "Nightstands",
        "Ottomans",
        "Ovens",
        "ServingCarts",
        "Showers",
        "SideboardsAndBuffets",
        "Sofas",
        "Storage",
        "StorageBenches",
        "Toilets",
        "VanityCounters",
        "WashingMachinesAndDryers"
      ],
      "type": "string"
    },
    "minItems": 0
  }
},
"required": [
  "ModelTypes"
],
"additionalProperties": false
},
```

```

"FurnitureArrangementSet": {
  "title": "FurnitureArrangementSet",
  "description": "Describes the interior template for placing furniture in one
or more rooms.\n\n- TargetSet is the set of rooms to furnish, filter by room name
or room\n type.\n- SampleSet is a set of all furnishing models to randomly choose
and\n place.\n- DesiredSpatialDensity is the desired level of free space after
placing\n furniture.",
  "type": "object",
  "properties": {
    "Name": {
      "title": "Name",
      "maxLength": 255,
      "minLength": 1,
      "pattern": "^[a-zA-Z0-9_\\- ]*$",
      "type": "string"
    },
    "TargetSet": {
      "title": "Targetset",
      "anyOf": [
        {
          "$ref": "#/definitions/RoomNameList"
        },
        {
          "$ref": "#/definitions/RoomTypeList"
        }
      ]
    },
    "SampleSet": {
      "$ref": "#/definitions/ModelTypeList"
    },
    "DesiredSpatialDensity": {
      "title": "Desiredspatialdensity",
      "default": "Moderate",
      "enum": [
        "Sparse",
        "Moderate",
        "Dense"
      ],
      "type": "string"
    }
  },
  "required": [
    "Name",
    "TargetSet"
  ]
}

```

```

    ],
    "additionalProperties": false
  },
  "InteriorFurnishings": {
    "title": "InteriorFurnishings",
    "description": "Describes the types of furniture models for randomly placing
into each room\n\nin the world. Rooms are targeted by room type or room name. Rooms
that are\n\nnot targeted are furnished at random by their room type with moderate
density.\n\ndensity. For an empty room, specify an empty sample set.",
    "type": "object",
    "properties": {
      "FurnitureArrangements": {
        "title": "Furniturearrangements",
        "default": [],
        "type": "array",
        "items": {
          "$ref": "#/definitions/FurnitureArrangementSet"
        },
        "minItems": 0,
        "maxItems": 6
      }
    },
    "additionalProperties": false
  },
  "InteriorTemplate": {
    "title": "InteriorTemplate",
    "description": "Top-level template for parameterizing the interior finishes
and furnishings for\n\nthis floorplan.",
    "type": "object",
    "properties": {
      "Flooring": {
        "title": "Flooring",
        "default": {
          "MaterialSets": []
        },
        "allOf": [
          {
            "$ref": "#/definitions/InteriorFlooring"
          }
        ]
      },
      "Walls": {
        "title": "Walls",
        "default": {

```



```

        "MaterialSets": []
    },
    "allOf": [
        {
            "$ref": "#/definitions/InteriorWalls"
        }
    ]
},
"Furniture": {
    "title": "Furniture",
    "default": {
        "FurnitureArrangements": []
    },
    "allOf": [
        {
            "$ref": "#/definitions/InteriorFurnishings"
        }
    ]
}
},
"additionalProperties": false
},
"FloorTemplate": {
    "title": "FloorTemplate",
    "description": "Describes a single floor within a building. Defaults to a
single residential room\nof a randomy type and size, and the interior is randomly
furnished.",
    "type": "object",
    "properties": {
        "Floorplan": {
            "title": "Floorplan",
            "default": {
                "Footprint": {
                    "DesiredAspectRatio": {
                        "x": 1.0,
                        "y": 1.0
                    }
                }
            },
            "Ceiling": {
                "Height": 3.0
            },
            "Rooms": [
                {
                    "Type": "Living",

```

```

        "Name": "My Living Room",
        "DesiredShape": {
            "Area": 20.0,
            "AspectRatio": {
                "x": 1.0,
                "y": 1.0
            }
        }
    ],
    "DesiredConnections": [],
},
"allOf": [
    {
        "$ref": "#/definitions/FloorplanTemplate"
    }
]
},
"Interior": {
    "title": "Interior",
    "default": {
        "Flooring": {
            "MaterialSets": []
        },
        "Walls": {
            "MaterialSets": []
        },
        "Furniture": {
            "FurnitureArrangements": []
        }
    },
    "allOf": [
        {
            "$ref": "#/definitions/InteriorTemplate"
        }
    ]
}
},
"additionalProperties": false
},
"BuildingTemplate": {
    "title": "BuildingTemplate",
    "description": "Describes a building to be randomly generated. Defaults to one residential floor.",

```

```
"type": "object",
"properties": {
  "Floors": {
    "title": "Floors",
    "default": [
      {
        "Floorplan": {
          "Footprint": {
            "DesiredAspectRatio": {
              "x": 1.0,
              "y": 1.0
            }
          },
          "Ceiling": {
            "Height": 3.0
          },
          "Rooms": [
            {
              "Type": "Living",
              "Name": "My Living Room",
              "DesiredShape": {
                "Area": 20.0,
                "AspectRatio": {
                  "x": 1.0,
                  "y": 1.0
                }
              }
            }
          ],
          "DesiredConnections": []
        },
        "Interior": {
          "Flooring": {
            "MaterialSets": []
          },
          "Walls": {
            "MaterialSets": []
          },
          "Furniture": {
            "FurnitureArrangements": []
          }
        }
      }
    ]
  },

```

```
        "type": "array",
        "items": {
            "$ref": "#/definitions/FloorTemplate"
        },
        "minItems": 1,
        "maxItems": 1
    }
},
"additionalProperties": false
}
}
```

If you save the JSON into a file named `one-bedroom-house.json`, you can use it with the AWS CLI to create a simulation world template:

```
$ aws robomaker create-world-template --template my-simulation-world-template-arn --
template-body file://one-bedroom-house.json
```

## Viewing a simulation world template

View details about a simulation world template.

### To see the details of a simulation world template

Follow the steps on one of the following tabs:

#### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Simulation WorldForge**, and then choose **World templates**.
3. Choose the **Id** of a simulation world template to view its details, including its floor plan and interiors. You can also generate worlds from the detail view.

#### Using the AWS CLI

## Example

The following AWS CLI example uses `list-world-templates` to list existing templates, and then it uses `describe-world-template` and `get-world-template-body` to view the details of a simulation world template.

```
$ aws robomaker list-world-templates
$ aws robomaker describe-world-template --template my-simulation-world-template-arn
$ aws robomaker get-world-template-body --template my-simulation-world-template-arn
```

## Modifying a simulation world template

Select the floor plan to customize the number and types of rooms and the connections between rooms in the floor plan. Choose interiors to customize flooring, walls, and furniture.

### To modify a simulation world template

Follow the steps on one of the following tabs:

Using the console

#### To modify the simulation world template

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. On the AWS RoboMaker console, expand **Simulation WorldForge** in the left navigation pane and then select **World templates**.
3. On the **World templates** page, choose the simulation world template you want to modify.
4. Choose **Edit** or **Override** next to each element you want to modify. For more information about simulation world template components, see [Understanding simulation world templates](#).

Using the AWS CLI

## Example

The following AWS CLI example uses `list-world-templates` to list existing templates, and then it uses `describe-world-template` to view the details of a simulation world template and `get-world-template-body` to retrieve the template body JSON and write it to a file.

```
$ aws robomaker list-world-templates
$ aws robomaker describe-world-template --template my-simulation-world-template-arn
$ aws robomaker get-world-template-body --template my-simulation-world-template-arn
  --output json > myTemplateBody.json
$ aws robomaker update-world-template-body --template my-simulation-world-template-arn
  --template-body file://myTemplateBody.json
```

## Deleting a simulation world template

When you no longer need a simulation world template, you can delete it.

Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Simulation WorldForge**, and then choose **World templates**.
3. Choose the **Id** of a simulation world template, choose **Template actions**, choose **Delete**, and then confirm the deletion by selecting **Delete** in the dialog box.

Using the AWS CLI

## Example

The following AWS CLI example uses `list-world-templates` to list existing templates, and then it uses `delete-world-template` to delete a simulation world template.

```
$ aws robomaker list-world-templates
$ aws robomaker delete-world-template --template my-simulation-world-template-arn
```

## Simulation world template versions, features, and changes

AWS RoboMaker Simulation WorldForge releases new versions of the world templates. You can use the new features and improvements in these templates to create worlds that are better suited for your use case.

To use all of the features of a world template, upgrade your world template to the latest version. The latest version of a world template has all of the features that are present in previous versions.

You can update your world templates using either the AWS RoboMaker console or the AWS CLI. If you're using the AWS RoboMaker console, you see a prompt that you can use to upgrade your template.

To upgrade your world template to the latest version using the API, set the `Version` field of the JSON that defines the world template to the numeric value for the latest version. For example, if Version 2 is the latest version, you would specify `"Version": "2"` in the body of the world template. To view the latest schema, see [JSON schema for simulation world template body](#).

The following descriptions provide information about features and updates for the world templates. The updates for the latest version are shown first.

### Simulation world template version 2 release

The updates for Version 2 include:

- The ability to add hinged doors to your worlds.
- The ability to apply a configuration to all rooms.
- A new field that describes your world.
- Changes to the floor friction values.
- Version agnostic updates.

#### Doors

You can use Version 2 of the AWS RoboMaker Simulation WorldForge template to create a world that has hinged doors.

You can configure the percentage that these doors are open. For example, these are some open states that you can specify:

- 0% open – closed

- 50% open – halfway open
- 70% open – mostly open
- 100% open – entirely open

You can also specify that Simulation WorldForge randomize the openness of the doors by setting the open percentage to a random state.

You can configure the doors that you want to see in your world under the `Interior` section of your world template. To learn how to use a world template to create a room with doors, see [Requesting doors in doorways](#).

### Applying a configuration to all rooms

You can use the `Target.All` keyword of the world template to apply a configuration change to all rooms. These are some of the things that you can change in all of the rooms:

- Flooring material
- Wall material
- Doorways
- Furniture arrangements

For example, if you want to specify that every door is closed in your world template, you can specify that the doors are zero percent open and use the `Target.All` keyword to apply that condition to all doors. For more information, see [Applying a configuration to all rooms](#).

### A new field that describes your worlds

The worlds that are created with a Version 2 template have a `world_description.json` file. This file appears in the same directory as the Gazebo WorldForge `.world` file.

The `world_description.json` file lists all the doors in your Simulation WorldForge world. You can use the [DescribeWorld](#) operation to see a description of your world. The description is the value of the `worldDescriptionBody` field. If your world was created with a Version 1 template, the value of the field is empty.

### Version 2 changes to the floor friction values

In Version 2, the floors have the same floor friction values as the Gazebo ground plane. The floor friction values in Version 1 are unchanged.



## Version agnostic updates

For all world templates, the spaces in the room names are replaced with underscores in your Gazebo model names. This change gives you the ability to use ROS topics for all your Simulation WorldForge Gazebo models. You can use ROS topics to get information about your model, or make changes to your model.

## Managing world generation jobs

Use a world generation job to generate worlds from a simulation world template. When you create a world generation job, you specify the number of different floor plans and interior configurations. You can generate up to 50 worlds per world generation job.

### Topics

- [Creating a world generation job](#)
- [Viewing a world generation job](#)
- [Canceling a world generation job](#)

## Creating a world generation job

Create a world generation job to generate worlds with different room and interior configurations. Each world generation job can generate up to 50 worlds.

### To create a world generation job

Follow the steps on one of the following tabs:

Using the console

#### To create a simulation world template

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. On the AWS RoboMaker console, expand **Simulation WorldForge** on the left and then choose **World templates**.
3. On the **World templates** page, choose the simulation world template you want to use to generate worlds, and then choose **Generate worlds**.

4. On the **Generate worlds** page, specify the **Number of floor plans**. The number of floor plans multiplied by the number of interior variations per floor plan must not exceed 50.
5. Specify the number of **Interior variations per floor plan**. The number of floor plans multiplied by the number of interior variations per floor plan must not exceed 50.
6. *Optional:* Add **World tags** that are assigned to all of the worlds you generate.
7. *Optional:* Add **Generation job tags** that are assigned to the generation job. These tags do not apply to worlds you generate.
8. Choose **Generate**.

You can track the progress of your world generation job in the **World generation detail** page. The time required to generate your worlds depends on the complexity of the simulation world template and the number of worlds you are generating.

## Using the AWS CLI

### Example

You can generate worlds from a simulation world template using the AWS CLI. Use `create-world-generation-job` to create the world generation job.

The following AWS CLI example shows how to generate 4 worlds with 2 floor plans with 2 different interior floor plans.

```
$ aws robomaker list-world-templates
$ aws robomaker create-world-generation-job --template my-simulation-world-template-arn --worldCount floorplanCount=2,interiorCountPerFloorplan=2
$ aws robomaker list-world-generation-jobs
$ aws robomaker describe-world-generation-job --job my-world-generation-job-arn
```

## Viewing a world generation job

You can view world generation progress, summary information, and other details about a world generation job.

### To see the details of a world generation job

Follow the steps on one of the following tabs:

### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Simulation WorldForge**, then choose **World templates**.
3. Choose the **Id** of a world generation job to view its details. You can find generation jobs using the search bar.

### Using the AWS CLI

#### Example

The following AWS CLI example uses the `list-world-generation-jobs` to list existing world generation jobs, and then it uses `describe-world-generation-job` to view the details of a specific world generation job.

```
$ aws robomaker list-world-generation-jobs
$ aws robomaker describe-world-generation-job --job my-world-generation-job-arn
```

## Canceling a world generation job

You can cancel a world generation job that is in progress.

### To cancel a world generation job

Follow the steps under one of the following tabs:

#### Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Simulation WorldForge**, and then choose **Generation jobs**.
3. On the **Generation jobs** page, choose the world generation job you want to cancel.

4. Choose **Cancel**. On the **Cancel generation job** page, choose **Cancel job** to cancel the job.

## Using the AWS CLI

### Example

The following AWS CLI example uses the `list-world-generation-jobs` to list existing world generation jobs, and then it uses `cancel-world-generation-job` to cancel a specific world generation job.

```
$ aws robomaker list-world-generation-jobs
$ aws robomaker cancel-world-generation-job --job my-world-generation-job-arn
```

## Managing world export jobs

You can export worlds generated by Simulation WorldForge to use in your own environment. Worlds are exported to your Amazon S3 bucket in a .zip file. The .zip file includes Gazebo assets and an ROS workspace for the worlds.

### Topics

- [Creating a world export job](#)
- [Viewing a world export job](#)

## Creating a world export job

You can select worlds to export to your Amazon S3 bucket. All worlds selected for the export are in a single .zip file.

### To create a world export job

Follow the steps on one of the following tabs:

#### Using the console

You can export one world per export job.

## To create a simulation world template

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. On the AWS RoboMaker console, expand **Simulation WorldForge** in the left navigation pane, and then choose **Worlds**.
3. On the **Worlds** page, choose **Create export job**.
4. On the **Create export job** page, choose a **World** to export.
5. Choose an **IAM role** with PutObject, GetObject and AbortMultipartUpload permissions to your Amazon S3 bucket. Choose **Create** to have a role with appropriate permissions created for you.
6. Choose an **S3 destination for worlds output**. You can also create a new Amazon S3 bucket by choosing **Create new S3 bucket** near the bottom of the page.
7. *Optional:* On the **Create export job** page, add tags that are assigned to the exported world.
8. Choose **Create** to create the world export job.

You can track its progress of the export job in the world export job details page. You are taken there automatically after you create the job.

## Using the AWS CLI

### Example

You can export worlds using the AWS CLI. Use `create-world-export-job` to create the world export job. You can export one world per export job.

The following AWS CLI example shows how to export a world. First, you can list worlds using `list-worlds`, and then call `create-world-export-job` specifying a world Amazon Resource Name (ARN). You can check status by calling `list-world-export-jobs` and `describe-world-export-job`.

```
aws robomaker list-worlds
aws robomaker create-world-export-job --worlds my-simulation-world-arn --iam-role
  my-iam-role-arn --outputLocation s3Bucket=my-bucket,s3prefix=prefix
aws robomaker list-world-export-jobs
aws robomaker describe-world-export-job --job my-world-export-job-arn
```

## Viewing a world export job

View the status and other details of a world export job.

### To see the details of a world export job

Follow the steps on one of the following tabs:

Using the console

1. Sign in to the AWS RoboMaker console at <https://console.aws.amazon.com/robomaker/>.
2. In the left navigation pane, choose **Simulation WorldForge**, and then choose **Export jobs**.
3. Choose the **ID** of a world export job to view its details. You can also search and cancel world export jobs.

Using the AWS CLI

### Example

The following AWS CLI example uses the `list-world-export-jobs` to list existing world export jobs, and then it uses `describe-world-export-job` to view the details of a specific world export job.

```
aws robomaker list-world-export-jobs
aws robomaker describe-world-export-job --job my-world-export-job-arn
```

## Using exported worlds in simulation

Simulation WorldForge can be used to create worlds for use with AWS RoboMaker. Once worlds have been created, they must be exported to use in simulations. You may also upload worlds to use in your simulation.

Exporting worlds offers the opportunity to use:

- Physics that are different from the [default SDF physics](#)
- Specialized lighting

- Custom models

The following sections provide more information about using generated worlds in your simulation.

### Important

To learn more about how you are charged for AWS RoboMaker, see [AWS RoboMaker pricing](#).

## Sections

- [Using an exported world as a data source](#)
- [Using an exported world in ROS and Gazebo](#)
- [Using an exported world with custom physics, lights, and models](#)

## Using an exported world as a data source

With Simulation WorldForge, you can export worlds you can use in your ROS environment. The world you choose to export is copied to a single .zip file in an Amazon S3 bucket. This section describes how to use an exported world in an Amazon S3 bucket in a simulation job by giving instructions on how to adjust the launch file, then creating a simulation job either through the AWS Management Console or using the command line interface.

You must first update your simulation application launch file before using the AWS Management Console or AWS CLI to add a data source.

### To update your simulation launch file:

1. Run the following launch commands:

```
<launch>
  <!-- Always set GUI to false for AWS RoboMaker Simulation
  Use gui:=true on roslaunch command-line to run with gzclient.
  -->
  <arg name="gui" default="false"/>

  <include file="$(find aws_robomaker_worldforge_worlds)/launch/
  launch_world.launch">
```

```
<arg name="gui" value="$(arg gui)"/>
</include>

<!-- Your other launch commands go here. -->
</launch>
```

You can spawn your robot at (0, 0, 0). The worlds Simulation WorldForge generates are guaranteed to have a 1 meter cylinder clear at (0, 0, 0).

2. Rebuild your image and push as normal. For more information, see [Using images to develop AWS RoboMaker applications](#).

## To add a data source

Follow the steps under one of the following tabs:

### Using the console

1. Follow the procedure in [Creating a world export job](#) to export a world.
2. While creating a simulation job, add a new data source. Give the data source a meaningful name, such as `WorldForge`, used in the example.
3. Optionally, identify a destination directory into which to place the world.

#### Note

Do not use your workspace directory as the destination, as AWS RoboMaker overwrites the destination with `dataSource` files. Instead, you can specify a separate directory under the workspace such as `your_workspace/src/aws_exported_world` as the destination.

4. Choose **Archive** as the type. AWS RoboMaker unzips the world into the destination directory.
5. Choose **Browse S3** and find the correct world-generated export.
6. Continue with creating the simulation job as normal.

### Using the AWS CLI



## Example

Follow the procedure in [Creating a world export job](#) to export a world.

Assuming the workspace is located in the container at `/home/simulation_ws`, the following command extracts the world into the default destination directory:

```
aws robomaker create-simulation-job \
  --max-job-duration-in-seconds <time> \
  --iam-role <IAM role ARN> \
  --data-sources '[{
    "name": "WorldForge",
    "type": "Archive",
    "destination": "/home/simulation_ws/src/aws_exported_world",
    "s3Bucket": "worldforge-test",
    "s3Keys": ["aws-robomaker-worldforge-export-q376mqk4z7gm.zip"]
  }]' \
  --robot-applications <config> \
  --simulation-applications <config>
```

## Using an exported world in ROS and Gazebo

As stated in the previous section, Simulation WorldForge exports your chosen world to a single `.zip` file. The `.zip` file includes all of the assets needed to modify and visualize the worlds using ROS and Gazebo. It includes the following important folders:

- The root folder, `workspace_src`, is the ROS workspace. It contains shared models, world data, and other information for the worlds. It is compatible with ROS 1 and ROS 2.
- **Shared models** is copied to `workspace_src/src/aws_robomaker_worldforge_shared_models/models`. For example, if the same chair is used in more than one world, it is placed in the shared model folder.
- **World data** is copied to `workspace_src/src/aws_robomaker_worldforge_worlds/worlds/`.

### To update your simulation launch file:

1. Follow the procedure in [Creating a world export job](#) to export a world.
2. Unzip the world into an ROS workspace.

```
cd MyApplication/simulation_ws
unzip MyExportedWorld.zip
```

### 3. Build the world.

```
rosdep install --from-paths src --ignore-src -r -y$ colcon build
```

### 4. Launch the world.

```
source install/setup.sh
roslaunch aws_robomaker_worldforge_worlds launch_world.launch gui:=true
```

## To build and launch the world in a simulation job:

1. Follow the procedure in [Creating a world export job](#) to export a world.
2. Import the exported world into the workspace source package directory `/home/simulation_ws/src/aws_exported_world` using `dataSource`
3. Modify the LaunchConfig of simulation application.

```
"launchConfig": {
  "environmentVariables": {
    "ROS_IP": "ROBOMAKER_SIM_APP_IP",
    "ROS_MASTER_URI": "http://ROBOMAKER_ROBOT_APP_IP:11311",
    "GAZEBO_MASTER_URI": "http://ROBOMAKER_SIM_APP_IP:11345",
    "GAZEBO_MODEL_PATH": "@GAZEBO_MODEL_PATH:/home/
simulation_ws/src/aws_exported_world/aws_robomaker_worldforge_pkgs/
aws_robomaker_worldforge_shared_models/models"
  },
  "streamUI": true,
  "command": [
    "/bin/bash", "-c", "cd /home/simulation_ws && colcon build && source
install/setup.sh && roslaunch hello_world_simulation worldforge_world.launch"
  ]
},
```

## Using an exported world with custom physics, lights, and models

If your simulation scenario requires customization, you can export and modify the world. For example, you can apply custom physics, different lighting effects, add custom models, or make other modifications.

After the world is exported, you need to modify the `.world` file to include the exported world model. The `.world` file uses SDF. For more information about SDF, see [SDFFormat](#).

### To modify your `.world` file to include the exported world model:

1. Follow the procedure in [Creating a world export job](#) to export a world.
2. Copy the following code to your `.world` file. Make sure the world name matches the exported model name.

```
<sdf version="1.6">
  <world name="generation_82856b0yq33y_world_16">
    <model name="WorldForge World">
      <include>
        <uri>model://generation_82856b0yq33y_world_16</uri>
      </include>
    </model>
    <!-- Your other <world> elements go here -->
  </world>
</sdf>
```

3. Verify that your launch file includes the modified `.world` file. Use the updated launch file to launch your simulation.

# Security

This section provides guidelines for securing different aspects of AWS RoboMaker.

## Topics

- [Data protection in AWS RoboMaker](#)
- [Authentication and access control for AWS RoboMaker](#)
- [Logging and monitoring in AWS RoboMaker](#)
- [Tagging AWS RoboMaker resources](#)
- [Security compliance](#)
- [Resilience in AWS RoboMaker](#)
- [Infrastructure security in AWS RoboMaker](#)
- [AWS RoboMaker and interface VPC endpoints \(AWS PrivateLink\)](#)

## Data protection in AWS RoboMaker

The AWS [shared responsibility model](#) applies to data protection in AWS RoboMaker. As described in this model, AWS is responsible for protecting the global infrastructure that runs all of the AWS Cloud. You are responsible for maintaining control over your content that is hosted on this infrastructure. You are also responsible for the security configuration and management tasks for the AWS services that you use. For more information about data privacy, see the [Data Privacy FAQ](#). For information about data protection in Europe, see the [AWS Shared Responsibility Model and GDPR](#) blog post on the *AWS Security Blog*.

For data protection purposes, we recommend that you protect AWS account credentials and set up individual users with AWS IAM Identity Center or AWS Identity and Access Management (IAM). That way, each user is given only the permissions necessary to fulfill their job duties. We also recommend that you secure your data in the following ways:

- Use multi-factor authentication (MFA) with each account.
- Use SSL/TLS to communicate with AWS resources. We require TLS 1.2 and recommend TLS 1.3.
- Set up API and user activity logging with AWS CloudTrail.
- Use AWS encryption solutions, along with all default security controls within AWS services.

- Use advanced managed security services such as Amazon Macie, which assists in discovering and securing sensitive data that is stored in Amazon S3.
- If you require FIPS 140-2 validated cryptographic modules when accessing AWS through a command line interface or an API, use a FIPS endpoint. For more information about the available FIPS endpoints, see [Federal Information Processing Standard \(FIPS\) 140-2](#).

We strongly recommend that you never put confidential or sensitive information, such as your customers' email addresses, into tags or free-form text fields such as a **Name** field. This includes when you work with AWS RoboMaker or other AWS services using the console, API, AWS CLI, or AWS SDKs. Any data that you enter into tags or free-form text fields used for names may be used for billing or diagnostic logs. If you provide a URL to an external server, we strongly recommend that you do not include credentials information in the URL to validate your request to that server.

## Authentication and access control for AWS RoboMaker

AWS Identity and Access Management (IAM) is an AWS service that helps an administrator securely control access to AWS RoboMaker resources. Administrators use IAM to control who is *authenticated* (signed in) and *authorized* (has permissions) to use AWS RoboMaker resources. IAM is a feature of your AWS account offered at no additional charge.

### Important

To get started quickly, review the introductory information on this page, and then see [Getting started with IAM](#), and [What are policies?](#)

### Topics

- [Introduction to authorization and access control](#)
- [Permissions required](#)
- [Understanding how AWS RoboMaker works with IAM](#)
- [Troubleshooting authentication and access control](#)

## Introduction to authorization and access control

AWS RoboMaker is integrated with AWS Identity and Access Management (IAM), which offers a wide range of features:

- Create users and groups in your AWS account.
- Easily share your AWS resources between the users in your AWS account.
- Assign unique security credentials to each user.
- Control each user's access to services and resources.
- Get a single bill for all users in your AWS account.

For more information about IAM, see the following:

- [AWS Identity and Access Management \(IAM\)](#)
- [Getting started](#)
- [IAM User Guide](#)

## Permissions required

To use AWS RoboMaker or to manage authorization and access control for yourself or others, you must have the correct permissions.

### Permissions Required to Use the AWS RoboMaker Console

To access the AWS RoboMaker console, you must have a minimum set of permissions that allows you to list and view details about the AWS RoboMaker resources in your AWS account. If you create an identity-based permissions policy that is more restrictive than the minimum required permissions, the console won't function as intended for entities with that policy.

For read-only access to the AWS RoboMaker console, use the **AWSRoboMakerReadOnlyAccess** policy.

If an IAM user wants to create a simulation job, you need to grant `iam:PassRole` permission to that user. For more information about passing a role, see [Granting a User Permissions to Pass a Role to an AWS Service](#).

For example, you can attach the following policy to a user. It provides permission to create a simulation job:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "iam:PassRole",
      "Resource": "arn:aws:iam::123456789012:role/S3AndCloudWatchAccess"
    }
  ]
}
```

You don't need to allow minimum console permissions for users that are making calls only to the AWS CLI or the AWS API. Instead, you need only the permissions that match the API operation you're trying to perform.

## Permissions Required to View Worlds in the AWS RoboMaker in the Console

You can grant permissions required to view AWS RoboMaker worlds in the AWS RoboMaker console by attaching the following policy to a user:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "robomaker: DescribeWorld"
      ],
      "Resource": "*",
      "Effect": "Allow"
    }
  ]
}
```

## Permissions required to use the AWS RoboMaker simulation tools

The IAM user or role used to create simulation will automatically have permission to access the simulation tools. If it is a different user or role, it should have the `robomaker:CreateSimulationJob` privilege.

## Permissions Required for Authentication Management

To manage your own credentials, such as your password, access keys, and multi-factor authentication (MFA) devices, your administrator must grant you the required permissions. To view the policy that includes these permissions, see [Allow users to self-manage their credentials](#).

As an AWS administrator, you need full access to IAM so that you can create and manage users, groups, roles, and policies in IAM. You should use the [AdministratorAccess](#) AWS managed policy that includes full access to all of AWS. This policy does not provide access to the AWS Billing and Cost Management console or allow tasks that require root user credentials. For more information, see [AWS Tasks That Require AWS account root user Credentials](#) in the *AWS General Reference*.

### Warning

Only an administrator user should have full access to AWS. Anyone with this policy has permission to fully manage authentication and access control, in addition to modifying every resource in AWS. To learn how to create this user, see [Create your IAM Admin user](#).

## Permissions required for access control

If your administrator provided you with IAM user credentials, they attached policies to your IAM user to control what resources you can access. To view the policies attached to your user in the AWS Management Console, you must have the following permissions:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
        "iam:GetUserPolicy",
        "iam:ListGroupsForUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
      ],
      "Resource": [
        "arn:aws:iam::*:user/${aws:username}"
      ]
    }
  ]
}
```



```

    ]
  },
  {
    "Sid": "ListUsersViewGroupsAndPolicies",
    "Effect": "Allow",
    "Action": [
      "iam:GetGroupPolicy",
      "iam:GetPolicyVersion",
      "iam:GetPolicy",
      "iam:ListAttachedGroupPolicies",
      "iam:ListGroupPolicies",
      "iam:ListPolicyVersions",
      "iam:ListPolicies",
      "iam:ListUsers"
    ],
    "Resource": "*"
  }
]
}

```

If you need additional permissions, ask your administrator to update your policies to allow you to access the actions that you require.

## Permissions required for a simulation job

When you create a simulation job, it must have an IAM role with the permissions below.

- Replace `my-input-bucket` with the name of the bucket containing the robot and simulation application bundles.
- Replace `my-output-bucket` to point to the bucket where AWS RoboMaker will write output files.
- Replace `account#` with your account number.

Public ECR jobs require separate permissions, such as `ecr-public:GetAuthorizationToken`, `sts:GetServiceBearerToken`, and any other permissions required for your final implementation. For more information, see [Public repository policies](#) in the *Amazon ECR User Guide*.

### Jobs with Private ECR images

```
{
```

```

"Version": "2012-10-17",
"Statement": [
  {
    "Action": "s3:ListBucket",
    "Resource": [
      "arn:aws:s3:::my-input-bucket"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "s3:Get*",
      "s3:List*"
    ],
    "Resource": [
      "arn:aws:s3:::my-input-bucket/*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": "s3:Put*",
    "Resource": [
      "arn:aws:s3:::my-output-bucket/*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "logs:CreateLogGroup",
      "logs:CreateLogStream",
      "logs:PutLogEvents",
      "logs:DescribeLogStreams"
    ],
    "Resource": [
      "arn:aws:logs:*:account#:log-group:/aws/robomaker/SimulationJobs*"
    ],
    "Effect": "Allow"
  },
  {
    "Action": [
      "ecr:BatchGetImage",
      "ecr:GetAuthorizationToken",
      "ecr:BatchCheckLayerAvailability",
      "ecr:GetDownloadUrlForLayer"
    ]
  }
]

```

```

    ],
    "Resource":
"arn:partition:ecr:region:account#:repository/repository_name",
    "Effect": "Allow"
  }
]
}

```

## Jobs with Public ECR images

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": "s3:ListBucket",
      "Resource": [
        "arn:aws:s3::my-input-bucket"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [
        "s3:Get*",
        "s3:List*"
      ],
      "Resource": [
        "arn:aws:s3::my-input-bucket/*"
      ],
      "Effect": "Allow"
    },
    {
      "Action": "s3:Put*",
      "Resource": [
        "arn:aws:s3::my-output-bucket/*"
      ],
      "Effect": "Allow"
    },
    {
      "Action": [
        "logs:CreateLogGroup",
        "logs:CreateLogStream",
        "logs:PutLogEvents",

```

```

        "logs:DescribeLogStreams"
    ],
    "Resource": [
        "arn:aws:logs:*:account:log-group:/aws/robomaker/SimulationJobs*"
    ],
    "Effect": "Allow"
},
{
    "Action": [
        "ecr-public:GetAuthorizationToken",
        "sts:GetServiceBearerToken"
    ],
    "Resource": "*",
    "Effect": "Allow"
}
]
}

```

The policy must be attached to a role with the following trust policy.

```

{
    "Version": "2012-10-17",
    "Statement": {
        "Effect": "Allow",
        "Principal": { "Service": "robomaker.amazonaws.com" },
        "Action": "sts:AssumeRole",
        "Condition": {
            "StringEquals": {
                "aws:SourceAccount": "account" // Account where the simulation job
resource is created
            },
            "StringEquals": {
                "aws:SourceArn": "arn:aws:robomaker:region:account:simulation-job/*"
            }
        }
    }
}

```

Condition keys prevent an AWS service from being used as a [confused deputy](#) during transactions between services. See [SourceAccount](#) and [SourceArn](#) for additional information about condition keys.

## Permissions Required to use Tags from a ROS Application or ROS Command Line

You can tag, untag, and list tags in your simulation job from the ROS command-line or in your ROS application while it is running. You must have an IAM role with the permissions below. Replace `account#` with your account number.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "robomaker:TagResource",
        "robomaker:UntagResource",
        "robomaker:ListTagsForResource",
      ],
      "Resource": [
        "arn:aws:robomaker:*:account#:simulation-job*"
      ],
      "Effect": "Allow"
    }
  ]
}
```

The policy must be attached to a role with the following trust policy:

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Principal": { "Service": "robomaker.amazonaws.com" },
    "Action": "sts:AssumeRole",
    "Condition": {
      "StringEquals": {
        "aws:SourceAccount": "account#" // Account where the simulation job
resource is created
      },
      "StringEquals": {
        "aws:SourceArn": "arn:aws:robomaker:region:account#:simulation-job/*"
      }
    }
  }
}
```

```
}
```

Condition keys prevent an AWS service from being used as a [confused deputy](#) during transactions between services. See [SourceAccount](#) and [SourceArn](#) for additional information about condition keys.

## Understanding how AWS RoboMaker works with IAM

Services can work with IAM in several ways:

- **Actions** – AWS RoboMaker supports using actions in a policy. This allows an administrator to control whether an entity can complete an operation in AWS RoboMaker. For example, to allow an entity to view a policy by performing the `GetPolicy` AWS API operation, an administrator must attach a policy that allows the `iam:GetPolicy` action.
- **Resource-level permissions** – AWS RoboMaker does not support resource-level permissions. Resource-level permissions allow you to use [ARNs](#) to specify individual resources in the policy. Because AWS RoboMaker does not support this feature, then you must choose **All resources** in the [policy visual editor](#). In a JSON policy document, you must use `*` in the `Resource` element.
- **Authorization based on tags** – AWS RoboMaker does support authorization based tags. This feature allows you to use [resource tags](#) in the condition of a policy.
- **Temporary credentials** – AWS RoboMaker supports temporary credentials. This feature allows you to sign in with federation, assume an IAM role, or to assume a cross-account role. You obtain temporary security credentials by calling AWS STS API operations such as [AssumeRole](#) or [GetFederationToken](#).
- **Service-linked roles** – AWS RoboMaker supports service roles. This feature allows a service to assume a [service-linked role](#) on your behalf. This role allows the service to access resources in other services to complete an action on your behalf. Service-linked roles appear in your IAM account, and are owned by the service. An IAM administrator can view, but not edit the permissions for service-linked roles.
- **Service roles** – AWS RoboMaker supports service roles. This feature allows a service to assume a [service role](#) on your behalf. This role allows the service to access resources in other services to complete an action on your behalf. Service roles appear in your IAM account, and are owned by the account. This means that an IAM administrator can change the permissions for this role. However, this might break the functionality of the service.

## Troubleshooting authentication and access control

Use the following information to help you diagnose and fix common issues that you might encounter when working with IAM.

### Topics

- [I am not authorized to perform an action in AWS RoboMaker](#)
- [I'm an administrator and want to allow others to access AWS RoboMaker](#)

### I am not authorized to perform an action in AWS RoboMaker

If you receive an error in the AWS Management Console that tells you that you're not authorized to perform an action, then you must contact the administrator that provided you with your user name and password.

The following example error occurs when an IAM user named `my-user-name` tries to use the console to perform the `CreateRobotApplication` action, but does not have permissions.

```
User: arn:aws:iam::123456789012:user/my-user-name is not authorized to perform: aws-robomaker:CreateRobotApplication on resource: my-example-robot-application
```

For this example, ask your administrator to update your policies to allow you to access the `my-example-robot-application` resource using the `aws-robomaker:CreateRobotApplication` action.

### I'm an administrator and want to allow others to access AWS RoboMaker

To allow others to access AWS RoboMaker you must create an IAM entity (user or role) for the person or application that needs access. They will use the credentials for that entity to access AWS. You must then attach a policy to the entity that grants them the correct permissions in AWS RoboMaker.

To get started right away, see [Getting started with IAM](#).

## What are policies?

You control access in AWS by creating policies and attaching them to IAM identities or AWS resources.

**Note**

To get started quickly, review the introductory information on [Authentication and access control for AWS RoboMaker](#) and then see [Getting started with IAM](#).

A policy is an object in AWS that, when associated with an entity or resource, defines their permissions. AWS evaluates these policies when a principal, such as a user, makes a request. Permissions in the policies determine whether the request is allowed or denied. Most policies are stored in AWS as JSON documents.

IAM policies define permissions for an action regardless of the method that you use to perform the operation. For example, if a policy allows the [GetUser](#) action, then a user with that policy can get user information from the AWS Management Console, the AWS CLI, or the AWS API. When you create an IAM user, you can set up the user to allow console or programmatic access. The IAM user can sign in to the console using a user name and password. Or they can use access keys to work with the CLI or API.

To provide access, add permissions to your users, groups, or roles:

- Users and groups in AWS IAM Identity Center:

Create a permission set. Follow the instructions in [Create a permission set](#) in the *AWS IAM Identity Center User Guide*.

- Users managed in IAM through an identity provider:

Create a role for identity federation. Follow the instructions in [Creating a role for a third-party identity provider \(federation\)](#) in the *IAM User Guide*.

- IAM users:

- Create a role that your user can assume. Follow the instructions in [Creating a role for an IAM user](#) in the *IAM User Guide*.
- (Not recommended) Attach a policy directly to a user or add a user to a user group. Follow the instructions in [Adding permissions to a user \(console\)](#) in the *IAM User Guide*.



### **Unsupported policies with AWS RoboMaker**

Resource-based policies and Access control lists (ACLs) are not supported by AWS RoboMaker. For more information, see [Policy types](#) in the *IAM User Guide*.

## Topics

- [Identity-based policies](#)
- [Policy access level classifications](#)

## Identity-based policies

You can attach policies to IAM identities. For example, you can do the following:

- **Attach a permissions policy to a user or a group in your account** – To grant a user permissions to create an AWS RoboMaker resource, such as a robot applications, you can attach a permissions policy to a user or a group to which the user belongs.
- **Attach a permissions policy to a role (grant cross-account permissions)** – You can attach an identity-based permissions policy to an IAM role to grant cross-account permissions. For example, the administrator in account A can create a role to grant cross-account permissions to another AWS account (for example, account B) or an AWS service as follows:
  1. Account A administrator creates an IAM role and attaches a permissions policy to the role that grants permissions on resources in account A.
  2. Account A administrator attaches a trust policy to the role identifying account B as the principal who can assume the role.
  3. Account B administrator can then delegate permissions to assume the role to any users in account B. Doing this allows users in account B to create or access resources in account A. The principal in the trust policy can also be an AWS service principal if you want to grant an AWS service permissions to assume the role.

For more information about using IAM to delegate permissions, see [Access Management](#) in the *IAM User Guide*.

For more information about users, groups, roles, and permissions, see [Identities \(Users, Groups, and Roles\)](#) in the *IAM User Guide*.

## Policy access level classifications

In the IAM console, actions are grouped using the following access level classifications:

- **List** – Provide permission to list resources within the service to determine whether an object exists. Actions with this level of access can list objects but cannot see the contents of a resource. Most actions with the **List** access level cannot be performed on a specific resource. When you create a policy statement with these actions, you must specify **All resources** ("\*").
- **Read** – Provide permission to read but not edit the contents and attributes of resources in the service. For example, the Amazon S3 actions `GetObject` and `GetBucketLocation` have the **Read** access level.
- **Write** – Provide permission to create, delete, or modify resources in the service. For example, the Amazon S3 actions `CreateBucket`, `DeleteBucket` and `PutObject` have the **Write** access level.
- **Permissions management** – Provide permission to grant or modify resource permissions in the service. For example, most IAM and AWS Organizations policy actions have the **Permissions management** access level.

### Tip

To improve the security of your AWS account, restrict or regularly monitor policies that include the **Permissions management** access level classification.

- **Tagging** – Provide permission to create, delete, or modify tags that are attached to a resource in the service. For example, the Amazon EC2 `CreateTags` and `DeleteTags` actions have the **Tagging** access level.

## AWS managed policies for AWS RoboMaker

To add permissions to users, groups, and roles, it is easier to use AWS managed policies than to write policies yourself. It takes time and expertise to [create IAM customer managed policies](#) that provide your team with only the permissions they need. To get started quickly, you can use our AWS managed policies. These policies cover common use cases and are available in your AWS account. For more information about AWS managed policies, see [AWS managed policies](#) in the *IAM User Guide*.

AWS services maintain and update AWS managed policies. You can't change the permissions in AWS managed policies. Services occasionally add additional permissions to an AWS managed policy to support new features. This type of update affects all identities (users, groups, and roles) where the policy is attached. Services are most likely to update an AWS managed policy when a new feature is launched or when new operations become available. Services do not remove permissions from an AWS managed policy, so policy updates won't break your existing permissions.

Additionally, AWS supports managed policies for job functions that span multiple services. For example, the **ReadOnlyAccess** AWS managed policy provides read-only access to all AWS services and resources. When a service launches a new feature, AWS adds read-only permissions for new operations and resources. For a list and descriptions of job function policies, see [AWS managed policies for job functions](#) in the *IAM User Guide*.

## AWS managed policy: AWSRoboMaker\_FullAccess

This policy grants contributor permissions that allows AWS RoboMaker to read images or bundles that you can use to create applications. Additionally, this policy gives you access to all AWS RoboMaker resources and operations. It also creates an IAM role in your account that manages Amazon EC2 resources in your account.

### Permissions details

This policy includes the following permissions.

- `s3:GetObject` — If you're using a bundle for either your robot or simulation application, it allows AWS RoboMaker to get the zip files from your Amazon S3 bucket.
- `ecr:BatchGetImage` — If you're using an image for either your robot or simulation application, it allows AWS RoboMaker to get the image from your Amazon ECR repository.
- `ecr-public:DescribeImages` — If you're using a publicly available image for either your robot or simulation application, it allows AWS RoboMaker to get information about that image from the Amazon ECR repository.
- `iam:CreateServiceLinkedRole` — Provides AWS RoboMaker with access with the Amazon EC2 resources it needs to operate successfully. For more information, see [Using service-linked roles for AWS RoboMaker](#)

```
{
```

```
"Version": "2012-10-17",
"Statement": [
  {
    "Effect": "Allow",
    "Action": "robomaker:*",
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": "s3:GetObject",
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "aws:CalledViaFirst": "robomaker.amazonaws.com"
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": "ecr:BatchGetImage",
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "aws:CalledViaFirst": "robomaker.amazonaws.com"
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": "ecr-public:DescribeImages",
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "aws:CalledViaFirst": "robomaker.amazonaws.com"
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": "iam:CreateServiceLinkedRole",
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "iam:AWSServiceName": "robomaker.amazonaws.com"
      }
    }
  }
]
```

```
    }
  }
]
}
```

## AWS managed policy: AWSRoboMakerReadOnlyAccess

This managed policy example provides read only access to AWS RoboMaker via the AWS Management Console and SDK.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "robomaker:List*",
        "robomaker:BatchDescribe*",
        "robomaker:Describe*",
        "robomaker:Get*"
      ],
      "Resource": "*"
    }
  ]
}
```

## AWS RoboMaker updates to AWS managed policies

View details about updates to AWS managed policies for AWS RoboMaker since this service began tracking these changes. For automatic alerts about changes to this page, subscribe to the RSS feed on the [AWS RoboMaker Document history page](#).

Change	Description	Date
<a href="#">AWSRoboMaker_FullAccess</a> – New policy	<p>AWS RoboMaker added a new policy to allow access to resources it needs to successfully run.</p> <p>This policy gives AWS RoboMaker access to the Amazon ECR images or zip files that you've stored on Amazon S3 to create your robot and simulation applications. It also gives AWS RoboMaker the ability to access the Amazon EC2 it needs to run successfully.</p>	July 27, 2021
<a href="#">AWSRoboMakerReadOnlyAccess</a> – New policy	AWS RoboMaker added a new policy to allow read only access to AWS RoboMaker resources.	January 11, 2022
AWS RoboMaker started tracking changes	AWS RoboMaker started tracking changes for its AWS managed policies.	July 27, 2021

## Using service-linked roles for AWS RoboMaker

AWS RoboMaker uses AWS Identity and Access Management (IAM) [service-linked roles](#). A service-linked role is a unique type of IAM role that is linked directly to AWS RoboMaker. Service-linked roles are predefined by AWS RoboMaker and include all the permissions that the service requires to call other AWS services on your behalf.

A service-linked role makes setting up AWS RoboMaker easier because you don't have to manually add the necessary permissions. AWS RoboMaker defines the permissions of its service-linked roles, and unless defined otherwise, only AWS RoboMaker can assume its roles. The defined permissions

include the trust policy and the permissions policy, and that permissions policy cannot be attached to any other IAM entity.

You can delete a service-linked role only after first deleting their related resources. This protects your AWS RoboMaker resources because you can't inadvertently remove permission to access the resources.

For information about other services that support service-linked roles, see [AWS Services That Work with IAM](#) and look for the services that have **Yes** in the **Service-Linked Role** column. Choose a **Yes** with a link to view the service-linked role documentation for that service.

## Service-linked role permissions for AWS RoboMaker

AWS RoboMaker uses the service-linked role named **AWSServiceRoleForRoboMaker** – Allows RoboMaker to access EC2 and Lambda resources on your behalf.

The AWSServiceRoleForRoboMaker service-linked role trusts the following services to assume the role:

- `robomaker.amazonaws.com`

The role permissions policy allows AWS RoboMaker to complete the following actions on the specified resources:

- Create and cancel a simulation job created as part of a simulation job batch
- Manage Amazon EC2 networking resources
- Create and get AWS Lambda functions

You must configure permissions to allow an IAM entity (such as a user, group, or role) to create, edit, or delete a service-linked role. For more information, see [Service-Linked Role Permissions](#) in the *IAM User Guide*.

## Creating a service-linked role

You don't need to manually create a service-linked role. When you `SimulationJob` or `DeploymentJob` in the AWS Management Console, the AWS CLI, or the AWS API, AWS RoboMaker creates the service-linked role for you.

If you delete this service-linked role, and then need to create it again, you can use the same process to recreate the role in your account. When you create a `SimulationJob`, `SimulationJobBatch`, or `DeploymentJob`, AWS RoboMaker creates the service-linked role for you again.

You can also use the IAM console to create a service-linked role with the **RoboMaker** use case. In the AWS CLI or the AWS API, create a service-linked role with the `robomaker.amazonaws.com` service name. For more information, see [Creating a Service-Linked Role](#) in the *IAM User Guide*. If you delete this service-linked role, you can use this same process to create the role again.

## Editing a service-linked role

AWS RoboMaker does not allow you to edit the `AWSServiceRoleForRoboMaker` service-linked role. After you create a service-linked role, you cannot change the name of the role because various entities might reference the role. However, you can edit the description of the role using IAM. For more information, see [Editing a Service-Linked Role](#) in the *IAM User Guide*.

## Deleting a service-linked role

If you no longer need to use a feature or service that requires a service-linked role, we recommend that you delete that role. That way you don't have an unused entity that is not actively monitored or maintained. However, you must clean up the resources for your service-linked role before you can manually delete it.

### Note

If the AWS RoboMaker service is using the role when you try to delete the resources, then the deletion might fail. If that happens, wait for a few minutes and try the operation again.

## To manually delete the service-linked role using IAM

Use the IAM console, the AWS CLI, or the AWS API to delete the `AWSServiceRoleForRoboMaker` service-linked role. For more information, see [Deleting a Service-Linked Role](#) in the *IAM User Guide*.

## Supported regions for AWS RoboMaker service-linked roles

AWS RoboMaker supports using service-linked roles in all of the regions where the service is available. For more information, see [AWS Regions and Endpoints](#).

AWS RoboMaker does not support using service-linked roles in every region where the service is available. You can use the `AWSServiceRoleForRoboMaker` role in the following regions.



Region name	Region identity	Support in AWS RoboMaker
US East (N. Virginia)	us-east-1	Yes
US East (Ohio)	us-east-2	Yes
US West (N. California)	us-west-1	Yes
US West (Oregon)	us-west-2	Yes
Asia Pacific (Mumbai)	ap-south-1	Yes
Asia Pacific (Osaka)	ap-northeast-3	Yes
Asia Pacific (Seoul)	ap-northeast-2	Yes
Asia Pacific (Singapore)	ap-southeast-1	Yes
Asia Pacific (Sydney)	ap-southeast-2	Yes
Asia Pacific (Tokyo)	ap-northeast-1	Yes
Canada (Central)	ca-central-1	Yes
Europe (Frankfurt)	eu-central-1	Yes
Europe (Ireland)	eu-west-1	Yes
Europe (London)	eu-west-2	Yes
Europe (Paris)	eu-west-3	Yes
South America (São Paulo)	sa-east-1	Yes
AWS GovCloud (US)	us-gov-west-1	No

## Getting started with IAM

AWS Identity and Access Management (IAM) is an AWS service that allows you manage access to services and resources securely. IAM is a feature of your AWS account offered at no additional charge.

### Note

Before you start with IAM, review the introductory information on [Authentication and access control for AWS RoboMaker](#).

When you create an AWS account, you begin with one sign-in identity that has complete access to all AWS services and resources in the account. This identity is called the AWS account *root user* and is accessed by signing in with the email address and password that you used to create the account. We strongly recommend that you don't use the root user for your everyday tasks. Safeguard your root user credentials and use them to perform the tasks that only the root user can perform. For the complete list of tasks that require you to sign in as the root user, see [Tasks that require root user credentials](#) in the *IAM User Guide*.

## Create your IAM Admin user

To create an administrator user, choose one of the following options.

Choose one way to manage your administrator	To	By	You can also
In IAM Identity Center  (Recommended)	Use short-term credentials to access AWS.  This aligns with the security best practices	Following the instructions in <a href="#">Getting started</a> in the <i>AWS IAM Identity Center User Guide</i> .	Configure programmatic access by <a href="#">Configuring the AWS CLI to use AWS IAM Identity Center</a> in the AWS

Choose one way to manage your administrator	To	By	You can also
	. For information about best practices, see <a href="#">Security best practices in IAM</a> in the <i>IAM User Guide</i> .		<i>Command Line Interface User Guide</i> .
In IAM (Not recommended)	Use long-term credentials to access AWS.	Following the instructions in <a href="#">Creating your first IAM admin user and user group</a> in the <i>IAM User Guide</i> .	Configure programmatic access by <a href="#">Managing access keys for IAM users</a> in the <i>IAM User Guide</i> .

## Create delegated users for AWS RoboMaker

To support multiple users in your AWS account, you must delegate permission to allow other people to perform only the actions you want to allow. To do this, create an IAM group with the permissions those people need and then add IAM users to the necessary groups as you create them. You can use this process to set up the groups, users, and permissions for your entire AWS account. This solution is best used by small and medium organizations where an AWS administrator can manually manage the users and groups. For large organizations, you can use [custom IAM roles](#), [federation](#), or [single sign-on](#).

See [Creating a role to delegate permissions to an IAM user](#) in the *IAM User Guide* for examples and more information on delegated users.

## Allow users to self-manage their credentials

You must have physical access to the hardware that will host the user's virtual MFA device in order to configure MFA. For example, you might configure MFA for a user who will use a virtual MFA device running on a smartphone. In that case, you must have the smartphone available in order

to finish the wizard. Because of this, you might want to let users configure and manage their own virtual MFA devices. In that case, you must grant users the permissions to perform the necessary IAM actions.

See [IAM: Allows IAM users to self-manage an MFA device](#) in the *IAM User Guide* for an example policy to grant necessary permissions.

## Enable MFA for your IAM user

For increased security, we recommend that all IAM users configure multi-factor authentication (MFA) to help protect your AWS RoboMaker resources. MFA adds extra security because it requires users to provide unique authentication from an AWS-supported MFA device in addition to their regular sign-in credentials. See [Enabling MFA devices for users in AWS](#) in the *IAM User Guide* for set up instructions and more information on MFA options.

### Note

You must have physical access to the mobile device that will host the user's virtual MFA device in order to configure MFA for an IAM user.

## Logging and monitoring in AWS RoboMaker

Monitoring is an important part of maintaining the reliability, availability, and performance of AWS RoboMaker and your AWS solutions. You should collect monitoring data from all of the parts of your AWS solution so that you can more easily debug a multi-point failure if one occurs.

### Topics

- [Monitoring AWS RoboMaker with Amazon CloudWatch](#)
- [Logging calls with AWS CloudTrail](#)

## Monitoring AWS RoboMaker with Amazon CloudWatch

AWS RoboMaker sends metrics to Amazon CloudWatch. You can use the AWS Management Console, the AWS CLI, or an API to list the metrics that AWS RoboMaker sends to CloudWatch.

Metrics exist only in the region in which they are created. Metrics cannot be deleted, but they automatically expire after 15 months if no new data is published to them.

For more information about Amazon CloudWatch, see the [Amazon CloudWatch User Guide](#).

## Topics

- [AWS RoboMaker simulation metrics](#)
- [AWS RoboMaker usage metrics](#)

## AWS RoboMaker simulation metrics

You can monitor AWS RoboMaker simulation jobs using Amazon CloudWatch, which collects information from your simulation job and creates readable, near real-time metrics. Information is provided at 1-minute frequency.

The following metrics are available in the `SimulationJobId` dimension.

Metric	Description
<code>RealTimeFactor</code>	<p>The ratio of the amount of time that was simulated versus wall clock time. For example, if it takes an hour to simulate 30 minutes, the factor is .5.</p> <p>More complex simulations have a lower real time factor.</p>
<code>vCPU*</code>	<p>Number of virtual CPU cores used by the simulation job</p> <p>Unit: Count</p>
<code>Memory*</code>	<p>Amount of memory, in GB, used by the simulation job</p> <p>Unit: GB</p>
<code>SimulationUnit*</code>	<p><code>SimulationUnit</code> is calculated based on vCPU and memory consumption of the simulation job</p> <p>Unit: Count</p>

**⚠ Important**

Metrics marked with \* are for estimation purposes. AWS RoboMaker emits metrics while preparing to run a simulation job. Charges do not accrue until the simulation job is in the Running state.

**AWS RoboMaker usage metrics**

You can use CloudWatch usage metrics to provide visibility into your account's usage of resources. Use these metrics to visualize your current service usage on CloudWatch graphs and dashboards.

AWS RoboMaker usage metrics correspond to AWS service quotas. You can configure alarms that alert you when your usage approaches a service quota. For more information about CloudWatch integration with service quotas, see [Service Quotas Integration and Usage Metrics](#).

The following metrics are available in the AWS/Usage dimension.

Metric	Description
ResourceCount	<p>The number of the specified resources running in your account. The resources are defined by the dimensions associated with the metric.</p> <p>The most useful statistic for this metric is MAXIMUM, which represents the maximum number of resources used during the 1-minute period.</p>

The following dimensions are used to refine the usage metrics that are published by AWS RoboMaker.

Dimension	Description
Service	The name of the AWS service containing the resource. For AWS RoboMaker usage metrics, the value for this dimension is RoboMaker .

Dimension	Description
Type	The type of entity that is being reported. Currently, the only valid value for AWS RoboMaker usage metrics is Resource.
Resource	The type of resource that is running. Currently, the valid values for AWS RoboMaker usage metrics are RobotApplication, SimulationApplication, ActiveSimulationJob, and ActiveSimulationJobBatch.
Class	The class of resource being tracked. For AWS RoboMaker usage metrics with ActiveSimulationJob as the value of the Resource dimension, the valid values are CPU   GPU_AND_CPU. The value for this dimension defines the kind of compute resources used by the simulation jobs reported by that metric. For others, the class value is None.

These metrics are emitted every minute. Use these metrics to monitor usage and then request a corresponding limit increase if needed. For more information about monitoring your usage, see [Visualizing your service quotas and setting alarms](#).

## Logging calls with AWS CloudTrail

AWS RoboMaker is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in AWS RoboMaker. CloudTrail captures all API calls for AWS RoboMaker as events. The calls captured include calls from the AWS RoboMaker console and code calls to the AWS RoboMaker API operations. If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for AWS RoboMaker. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in **Event history**. Using the information collected by CloudTrail, you can determine the request that was made to AWS RoboMaker, the IP address from which the request was made, who made the request, when it was made, and additional details.

To learn more about CloudTrail, see the [AWS CloudTrail User Guide](#).

## AWS RoboMaker information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in AWS RoboMaker, that activity is recorded in a CloudTrail event along with other AWS service events in **Event history**. You can view, search, and download recent events in your AWS account. For more information, see [Viewing Events with CloudTrail Event History](#).

For an ongoing record of events in your AWS account, including events for AWS RoboMaker, create a trail. A *trail* enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all AWS Regions. The trail logs events from all Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see the following:

- [Overview for Creating a Trail](#)
- [CloudTrail Supported Services and Integrations](#)
- [Configuring Amazon SNS Notifications for CloudTrail](#)
- [Receiving CloudTrail Log Files from Multiple Regions](#) and [Receiving CloudTrail Log Files from Multiple Accounts](#)

All AWS RoboMaker actions are logged by CloudTrail and are documented in the [AWS RoboMaker API Reference](#). For example, calls to the `CreateSimulationJob`, `RegisterRobot` and `UpdateRobotApplication` actions generate entries in the CloudTrail log files.

Every event or log entry contains information about who generated the request. The identity information helps you determine the following:

- Whether the request was made with root or AWS Identity and Access Management (IAM) user credentials.
- Whether the request was made with temporary security credentials for a role or federated user.
- Whether the request was made by another AWS service.

For more information, see the [CloudTrail userIdentity Element](#).



## Understanding AWS RoboMaker log file entries

A trail is a configuration that enables delivery of events as log files to an Amazon S3 bucket that you specify. CloudTrail log files contain one or more log entries. An event represents a single request from any source and includes information about the requested action, the date and time of the action, request parameters, and so on. CloudTrail log files aren't an ordered stack trace of the public API calls, so they don't appear in any specific order.

The following example shows a CloudTrail log entry that demonstrates the DescribeRobot action.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "my-principal-id",
    "arn": "my-arn",
    "accountId": "my-account-id",
    "accessKeyId": "my-access-key",
    "userName": "my-user-name"
  },
  "eventTime": "2018-12-07T00:28:03Z",
  "eventSource": "robomaker.amazonaws.com",
  "eventName": "DescribeRobot",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "my-ip-address",
  "userAgent": "aws-internal/3 aws-sdk-java/1.11.455
Linux/4.4.83-0.1.fm.327.54.326.metal1.x86_64 OpenJDK_64-Bit_Server_VM/25.192-b12
java/1.8.0_192,",
  "requestParameters": {
    "robot": "my-robot-arn"
  },
  "responseElements": null,
  "requestID": "f54cdf8b-f9b6-11e8-8883-c3f04579eca3",
  "eventID": "affb0303-ff48-4f65-af8e-d7d19710bac3",
  "readOnly": true,
  "eventType": "AwsApiCall",
  "recipientAccountId": "my-recipient-account-id"
}
```

# Tagging AWS RoboMaker resources

To help manage and organize your fleets, robots, robot applications, simulation applications, and simulation jobs you can optionally assign your own metadata to specific resources in the form of tags. This section describes tags and shows you how to create them.

## Tag basics

Tags enable you to categorize your AWS RoboMaker resources in different ways, for example, by purpose, owner, or environment. This is useful when you have many resources of the same type—you can quickly identify a specific resource based on the tags you've assigned to it. Each tag consists of a key and optional value, both of which you define. For example, you could define a set of tags for your robots that helps you track devices by function. We recommend that you create a set of tag keys that meets your needs for each kind of resource. Using a consistent set of tag keys makes it easier for you to manage your resources.

You can search for and filter resources based on the tags you add or apply. You can also use tags to control access to your resources as described in [Using tags with IAM policies](#).

For ease of use, the Tag Editor in the AWS Management Console provides a central, unified way to create and manage your tags. For more information, see [Working with Tag Editor](#) in [Working with the AWS Management Console](#).

You can also work with tags using the AWS CLI and the AWS RoboMaker API. You can associate tags with thing groups, thing types, topic rules, jobs, security profiles, and billing groups when you create them by using the **Tags** field in the following commands:

- [CreateRobotApplication](#)
- [CreateSimulationApplication](#)
- [CreateSimulationJob](#)
- [CreateWorldExportJob](#)
- [CreateWorldGenerationJob](#)
- [CreateWorldTemplate](#)
- [StartSimulationJobBatch](#)

You can add, modify, or delete tags for existing resources that support tagging by using the following commands:

- [TagResource](#)
- [ListTagsForResource](#)
- [UntagResource](#)

You can edit tag keys and values, and you can remove tags from a resource at any time. You can set the value of a tag to an empty string, but you can't set the value of a tag to null. If you add a tag that has the same key as an existing tag on that resource, the new value overwrites the old value. If you delete a resource, any tags associated with the resource are also deleted.

## Tag restrictions and limitations

The following basic restrictions apply to tags:

- Maximum number of tags per resource: 50
- Maximum key length: 127 Unicode characters in UTF-8
- Maximum value length: 255 Unicode characters in UTF-8
- Tag keys and values are case-sensitive.
- Do not use the `aws:` prefix in your tag names or values because it's reserved for AWS use. You can't edit or delete tag names or values with this prefix. Tags with this prefix don't count against your tags per resource limit.
- If your tagging schema is used across multiple services and resources, remember that other services may have restrictions on allowed characters. Generally, allowed characters are: letters, spaces, and numbers representable in UTF-8, and the following special characters: `+ - = . _ : / @`.

## Using tags with IAM policies

You can apply tag-based resource-level permissions in the IAM policies you use for AWS RoboMaker API actions. This gives you better control over what resources a user can create, modify, or use. You use the `Condition` element (also called the `Condition` block) with the following condition context keys and values in an IAM policy to control user access (permissions) based on a resource's tags:

- Use `aws:ResourceTag/tag-key: tag-value` to allow or deny user actions on resources with specific tags.
- Use `aws:RequestTag/tag-key: tag-value` to require that a specific tag be used (or not used) when making an API request to create or modify a resource that allows tags.

- Use `aws:TagKeys`: [*tag-key*, ...] to require that a specific set of tag keys be used (or not used) when making an API request to create or modify a resource that allows tags.

### Note

The condition context keys and values in an IAM policy apply only to those AWS RoboMaker actions where an identifier for a resource capable of being tagged is a required parameter. For example, the use of [ListFleets](#) will not be allowed or denied on the basis of condition context keys and values because no taggable resource (fleet, robot, robot application, simulation application, simulation job, deployment job) is referenced in this request.

For more information, see [Controlling access to AWS resources using tags](#) in the *AWS Identity and Access Management User Guide*. The [IAM JSON policy reference](#) section of that guide has detailed syntax, descriptions, and examples of the elements, variables, and evaluation logic of JSON policies in IAM.

The following example policy applies two tag-based restrictions. An IAM user restricted by this policy:

- Cannot create a robot with tag "env=prod" (in the example, see the line `"aws:RequestTag/env" : "prod"`).
- Cannot delete a robot that has an existing tag "env=prod" (in the example, see the line `"aws:ResourceTag/env" : "prod"`).

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Effect" : "Deny",
      "Action" : "robomaker:CreateRobot",
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
          "aws:RequestTag/env" : "prod"
        }
      }
    }
  ],
}
```

```
{
  "Effect" : "Deny",
  "Action" : "robomaker:DeleteRobot",
  "Resource" : "*",
  "Condition" : {
    "StringEquals" : {
      "aws:ResourceTag/env" : "prod"
    }
  }
},
{
  "Effect": "Allow",
  "Action": "robomaker:*",
  "Resource": "*"
}
]
```

You can also specify multiple tag values for a given tag key by enclosing them in a list, like this:

```
  "StringEquals" : {
    "aws:ResourceTag/env" : ["dev", "test"]
  }
```

### Note

If you allow or deny users access to resources based on tags, you must consider explicitly denying users the ability to add those tags to or remove them from the same resources. Otherwise, it's possible for a user to circumvent your restrictions and gain access to a resource by modifying its tags.

## Security compliance

The AWS HIPAA Compliance Program includes AWS RoboMaker as a HIPAA eligible service. The AWS PCI DSS Compliance Program includes AWS RoboMaker as a PCI-compliant service.

For general information about AWS Cloud and HIPAA compliance, see the following:

- [HIPAA compliance](#)

- [Architecting for HIPAA Security and Compliance on Amazon Web Services](#)

## Resilience in AWS RoboMaker

The AWS global infrastructure is built around AWS regions and availability zones. AWS Regions provide multiple physically separated and isolated Availability Zones, which are connected with low-latency, high-throughput, and highly redundant networking. With Availability Zones, you can design and operate applications and databases that automatically fail over between zones without interruption. Availability Zones are more highly available, fault tolerant, and scalable than traditional single or multiple data center infrastructures.

For more information about AWS Regions and Availability Zones, see [AWS global infrastructure](#).

In addition to the AWS global infrastructure, AWS RoboMaker offers several features to help support your data resiliency and backup needs.

## Infrastructure security in AWS RoboMaker

As a managed service, AWS RoboMaker is protected by AWS global network security. For information about AWS security services and how AWS protects infrastructure, see [AWS Cloud Security](#). To design your AWS environment using the best practices for infrastructure security, see [Infrastructure Protection](#) in *Security Pillar AWS Well-Architected Framework*.

You use AWS published API calls to access AWS RoboMaker through the network. Clients must support the following:

- Transport Layer Security (TLS). We require TLS 1.2 and recommend TLS 1.3.
- Cipher suites with perfect forward secrecy (PFS) such as DHE (Ephemeral Diffie-Hellman) or ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Most modern systems such as Java 7 and later support these modes.

Additionally, requests must be signed by using an access key ID and a secret access key that is associated with an IAM principal. Or you can use the [AWS Security Token Service](#) (AWS STS) to generate temporary security credentials to sign requests.

# AWS RoboMaker and interface VPC endpoints (AWS PrivateLink)

You can establish a private connection between your VPC and AWS RoboMaker by creating an *interface VPC endpoint*. Interface endpoints are powered by [AWS PrivateLink](#), a technology that enables you to privately access AWS RoboMaker APIs without an internet gateway, NAT device, VPN connection, or AWS Direct Connect connection. Instances in your VPC don't need public IP addresses to communicate with AWS RoboMaker APIs. Traffic between your VPC and AWS RoboMaker does not leave the Amazon network.

Each interface endpoint is represented by one or more [Elastic Network Interfaces](#) in your subnets.

For more information, see [Interface VPC endpoints \(AWS PrivateLink\)](#) in the *AWS PrivateLink Guide*.

## Considerations for AWS RoboMaker VPC endpoints

Before you set up an interface VPC endpoint for AWS RoboMaker, ensure that you review [Interface endpoint properties and limitations](#) in the *AWS PrivateLink Guide*.

AWS RoboMaker supports making calls to all of its API actions from your VPC.

## Creating an interface VPC endpoint for AWS RoboMaker

You can create a VPC endpoint for the AWS RoboMaker service using either the Amazon VPC console or the AWS Command Line Interface (AWS CLI). For more information, see [Creating an interface endpoint](#) in the *AWS PrivateLink Guide*.

Create a VPC endpoint for AWS RoboMaker using the following service name:

- `com.amazonaws.region.robomaker`

If you enable private DNS for the endpoint, you can make API requests to AWS RoboMaker using its default DNS name for the Region, for example, `robomaker.us-east-1.amazonaws.com`.

For more information, see [Accessing a service through an interface endpoint](#) in the *AWS PrivateLink Guide*.

## Creating a VPC endpoint policy for AWS RoboMaker

You can attach an endpoint policy to your VPC endpoint that controls access to AWS RoboMaker. The policy specifies the following information:

- The principal that can perform actions.
- The actions that can be performed.
- The resources on which actions can be performed.

For more information, see [Controlling access to services with VPC endpoints](#) in the *AWS PrivateLink Guide*.

### Example: VPC endpoint policy for AWS RoboMaker actions

The following is an example of an endpoint policy for AWS RoboMaker. When attached to an endpoint, this policy grants access to the listed AWS RoboMaker actions for all principals on all resources.

```
{
  "Statement": [
    {
      "Principal": "*",
      "Effect": "Allow",
      "Action": [
        "robomaker:ListSimulationJobs",
        "robomaker:ListSimulationJobBatches"
      ],
      "Resource": "*"
    }
  ]
}
```



# API reference

This chapter contains the API reference documentation for AWS RoboMaker. It is comprised of the following sections.

## Sections

- [Actions](#)
- [Data Types](#)
- [Common Errors](#)
- [Common Parameters](#)

## Actions

The following actions are supported:

- [BatchDeleteWorlds](#)
- [BatchDescribeSimulationJob](#)
- [CancelDeploymentJob](#)
- [CancelSimulationJob](#)
- [CancelSimulationJobBatch](#)
- [CancelWorldExportJob](#)
- [CancelWorldGenerationJob](#)
- [CreateDeploymentJob](#)
- [CreateFleet](#)
- [CreateRobot](#)
- [CreateRobotApplication](#)
- [CreateRobotApplicationVersion](#)
- [CreateSimulationApplication](#)
- [CreateSimulationApplicationVersion](#)
- [CreateSimulationJob](#)
- [CreateWorldExportJob](#)
- [CreateWorldGenerationJob](#)

- [CreateWorldTemplate](#)
- [DeleteFleet](#)
- [DeleteRobot](#)
- [DeleteRobotApplication](#)
- [DeleteSimulationApplication](#)
- [DeleteWorldTemplate](#)
- [DeregisterRobot](#)
- [DescribeDeploymentJob](#)
- [DescribeFleet](#)
- [DescribeRobot](#)
- [DescribeRobotApplication](#)
- [DescribeSimulationApplication](#)
- [DescribeSimulationJob](#)
- [DescribeSimulationJobBatch](#)
- [DescribeWorld](#)
- [DescribeWorldExportJob](#)
- [DescribeWorldGenerationJob](#)
- [DescribeWorldTemplate](#)
- [GetWorldTemplateBody](#)
- [ListDeploymentJobs](#)
- [ListFleets](#)
- [ListRobotApplications](#)
- [ListRobots](#)
- [ListSimulationApplications](#)
- [ListSimulationJobBatches](#)
- [ListSimulationJobs](#)
- [ListTagsForResource](#)
- [ListWorldExportJobs](#)
- [ListWorldGenerationJobs](#)
- [ListWorlds](#)

- [ListWorldTemplates](#)
- [RegisterRobot](#)
- [RestartSimulationJob](#)
- [StartSimulationJobBatch](#)
- [SyncDeploymentJob](#)
- [TagResource](#)
- [UntagResource](#)
- [UpdateRobotApplication](#)
- [UpdateSimulationApplication](#)
- [UpdateWorldTemplate](#)

## BatchDeleteWorlds

Deletes one or more worlds in a batch operation.

### Request Syntax

```
POST /batchDeleteWorlds HTTP/1.1
Content-type: application/json

{
  "worlds": [ "string" ]
}
```

### URI Request Parameters

The request does not use any URI parameters.

### Request Body

The request accepts the following data in JSON format.

#### worlds

A list of Amazon Resource Names (arns) that correspond to worlds to delete.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

### Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
```

```
"unprocessedWorlds": [ "string" ]  
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### unprocessedWorlds

A list of unprocessed worlds associated with the call. These worlds were not deleted.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **ThrottlingException**

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

## BatchDescribeSimulationJob

Describes one or more simulation jobs.

### Request Syntax

```
POST /batchDescribeSimulationJob HTTP/1.1
Content-type: application/json

{
  "jobs": [ "string" ]
}
```

### URI Request Parameters

The request does not use any URI parameters.

### Request Body

The request accepts the following data in JSON format.

#### jobs

A list of Amazon Resource Names (ARNs) of simulation jobs to describe.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

### Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "jobs": [
    {
```

```

"arn": "string",
"clientRequestToken": "string",
"compute": {
  "computeType": "string",
  "gpuUnitLimit": number,
  "simulationUnitLimit": number
},
"dataSources": [
  {
    "destination": "string",
    "name": "string",
    "s3Bucket": "string",
    "s3Keys": [
      {
        "etag": "string",
        "s3Key": "string"
      }
    ],
    "type": "string"
  }
],
"failureBehavior": "string",
"failureCode": "string",
"failureReason": "string",
"iamRole": "string",
"lastStartedAt": number,
"lastUpdatedAt": number,
"loggingConfig": {
  "recordAllRosTopics": boolean
},
"maxJobDurationInSeconds": number,
"name": "string",
"networkInterface": {
  "networkInterfaceId": "string",
  "privateIpAddress": "string",
  "publicIpAddress": "string"
},
"outputLocation": {
  "s3Bucket": "string",
  "s3Prefix": "string"
},
"robotApplications": [
  {
    "application": "string",

```



```

    "applicationVersion": "string",
    "launchConfig": {
      "command": [ "string" ],
      "environmentVariables": {
        "string" : "string"
      },
      "launchFile": "string",
      "packageName": "string",
      "portForwardingConfig": {
        "portMappings": [
          {
            "applicationPort": number,
            "enableOnPublicIp": boolean,
            "jobPort": number
          }
        ]
      },
      "streamUI": boolean
    },
    "tools": [
      {
        "command": "string",
        "exitBehavior": "string",
        "name": "string",
        "streamOutputToCloudWatch": boolean,
        "streamUI": boolean
      }
    ],
    "uploadConfigurations": [
      {
        "name": "string",
        "path": "string",
        "uploadBehavior": "string"
      }
    ],
    "useDefaultTools": boolean,
    "useDefaultUploadConfigurations": boolean
  }
],
"simulationApplications": [
  {
    "application": "string",
    "applicationVersion": "string",
    "launchConfig": {

```

```

    "command": [ "string" ],
    "environmentVariables": {
      "string": "string"
    },
    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean,
  "worldConfigs": [
    {
      "world": "string"
    }
  ]
}
],
"simulationTimeMillis": number,
"status": "string",

```

```
    "tags": {
      "string" : "string"
    },
    "vpcConfig": {
      "assignPublicIp": boolean,
      "securityGroups": [ "string" ],
      "subnets": [ "string" ],
      "vpcId": "string"
    }
  }
],
"unprocessedJobs": [ "string" ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### jobs

A list of simulation jobs.

Type: Array of [SimulationJob](#) objects

### unprocessedJobs

A list of unprocessed simulation job Amazon Resource Names (ARNs).

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **ResourceNotFoundException**

The specified resource does not exist.

HTTP Status Code: 400

### **ThrottlingException**

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# CancelDeploymentJob

*This action has been deprecated.*

## Important

This API is no longer supported. For more information, see the May 2, 2022 update in the [Support policy](#) page.

Cancels the specified deployment job.

## Request Syntax

```
POST /cancelDeploymentJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### job

The deployment job ARN to cancel.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: Yes

## Response Syntax

```
HTTP/1.1 200
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)

- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

## CancelSimulationJob

Cancels the specified simulation job.

### Request Syntax

```
POST /cancelSimulationJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

### URI Request Parameters

The request does not use any URI parameters.

### Request Body

The request accepts the following data in JSON format.

#### job

The simulation job ARN to cancel.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

### Response Syntax

```
HTTP/1.1 200
```

### Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.



## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

## CancelSimulationJobBatch

Cancels a simulation job batch. When you cancel a simulation job batch, you are also cancelling all of the active simulation jobs created as part of the batch.

### Request Syntax

```
POST /cancelSimulationJobBatch HTTP/1.1
Content-type: application/json
```

```
{
  "batch": "string"
}
```

### URI Request Parameters

The request does not use any URI parameters.

### Request Body

The request accepts the following data in JSON format.

#### batch

The id of the batch to cancel.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

### Response Syntax

```
HTTP/1.1 200
```

### Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

## CancelWorldExportJob

Cancels the specified export job.

### Request Syntax

```
POST /cancelWorldExportJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

### URI Request Parameters

The request does not use any URI parameters.

### Request Body

The request accepts the following data in JSON format.

#### job

The Amazon Resource Name (arn) of the world export job to cancel.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

### Response Syntax

```
HTTP/1.1 200
```

### Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



## CancelWorldGenerationJob

Cancels the specified world generator job.

### Request Syntax

```
POST /cancelWorldGenerationJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

### URI Request Parameters

The request does not use any URI parameters.

### Request Body

The request accepts the following data in JSON format.

#### job

The Amazon Resource Name (arn) of the world generator job to cancel.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

### Response Syntax

```
HTTP/1.1 200
```

### Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# CreateDeploymentJob

*This action has been deprecated.*

## Important

This API is no longer supported and will throw an error if used. For more information, see the January 31, 2022 update in the [Support policy](#) page.

Deploys a specific version of a robot application to robots in a fleet.

The robot application must have a numbered `applicationVersion` for consistency reasons. To create a new version, use `CreateRobotApplicationVersion` or see [Creating a Robot Application Version](#).

## Note

After 90 days, deployment jobs expire and will be deleted. They will no longer be accessible.

## Request Syntax

```
POST /createDeploymentJob HTTP/1.1
Content-type: application/json

{
  "clientRequestToken": "string",
  "deploymentApplicationConfigs": [
    {
      "application": "string",
      "applicationVersion": "string",
      "launchConfig": {
        "environmentVariables": {
          "string" : "string"
        },
        "launchFile": "string",
        "packageName": "string",
        "postLaunchFile": "string",
        "preLaunchFile": "string"
      }
    }
  ]
}
```

```
    }
  }
],
"deploymentConfig": {
  "concurrentDeploymentPercentage": number,
  "downloadConditionFile": {
    "bucket": "string",
    "etag": "string",
    "key": "string"
  },
  "failureThresholdPercentage": number,
  "robotDeploymentTimeoutInSeconds": number
},
"fleet": "string",
"tags": {
  "string" : "string"
}
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### clientRequestToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: [a-zA-Z0-9\_\-=]\*

Required: Yes

### deploymentApplicationConfigs

The deployment application configuration.

Type: Array of [DeploymentApplicationConfig](#) objects

Array Members: Fixed number of 1 item.

Required: Yes

### deploymentConfig

The requested deployment configuration.

Type: [DeploymentConfig](#) object

Required: No

### fleet

The Amazon Resource Name (ARN) of the fleet to deploy.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

### tags

A map that contains tag keys and tag values that are attached to the deployment job.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Required: No

## Response Syntax

```
HTTP/1.1 200
```

Content-type: application/json

```
{
  "arn": "string",
  "createdAt": number,
  "deploymentApplicationConfigs": [
    {
      "application": "string",
      "applicationVersion": "string",
      "launchConfig": {
        "environmentVariables": {
          "string" : "string"
        },
        "launchFile": "string",
        "packageName": "string",
        "postLaunchFile": "string",
        "preLaunchFile": "string"
      }
    }
  ],
  "deploymentConfig": {
    "concurrentDeploymentPercentage": number,
    "downloadConditionFile": {
      "bucket": "string",
      "etag": "string",
      "key": "string"
    },
    "failureThresholdPercentage": number,
    "robotDeploymentTimeoutInSeconds": number
  },
  "failureCode": "string",
  "failureReason": "string",
  "fleet": "string",
  "status": "string",
  "tags": {
    "string" : "string"
  }
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

## arn

The Amazon Resource Name (ARN) of the deployment job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

## createdAt

The time, in milliseconds since the epoch, when the fleet was created.

Type: Timestamp

## deploymentApplicationConfigs

The deployment application configuration.

Type: Array of [DeploymentApplicationConfig](#) objects

Array Members: Fixed number of 1 item.

## deploymentConfig

The deployment configuration.

Type: [DeploymentConfig](#) object

## failureCode

The failure code of the simulation job if it failed:

`BadPermissionError`

AWS Greengrass requires a service-level role permission to access other services. The role must include the [AWSGreengrassResourceAccessRolePolicy managed policy](#).

`ExtractingBundleFailure`

The robot application could not be extracted from the bundle.

`FailureThresholdBreached`

The percentage of robots that could not be updated exceeded the percentage set for the deployment.



### GreengrassDeploymentFailed

The robot application could not be deployed to the robot.

### GreengrassGroupVersionDoesNotExist

The AWS Greengrass group or version associated with a robot is missing.

### InternalServerError

An internal error has occurred. Retry your request, but if the problem persists, contact us with details.

### MissingRobotApplicationArchitecture

The robot application does not have a source that matches the architecture of the robot.

### MissingRobotDeploymentResource

One or more of the resources specified for the robot application are missing. For example, does the robot application have the correct launch package and launch file?

### PostLaunchFileFailure

The post-launch script failed.

### PreLaunchFileFailure

The pre-launch script failed.

### ResourceNotFound

One or more deployment resources are missing. For example, do robot application source bundles still exist?

### RobotDeploymentNoResponse

There is no response from the robot. It might not be powered on or connected to the internet.

### Type: String

Valid Values: ResourceNotFound | EnvironmentSetupError | EtagMismatch  
| FailureThresholdBreach | RobotDeploymentAborted |  
RobotDeploymentNoResponse | RobotAgentConnectionTimeout  
| GreengrassDeploymentFailed | InvalidGreengrassGroup |  
MissingRobotArchitecture | MissingRobotApplicationArchitecture |

MissingRobotDeploymentResource | GreengrassGroupVersionDoesNotExist  
| LambdaDeleted | ExtractingBundleFailure | PreLaunchFileFailure |  
PostLaunchFileFailure | BadPermissionError | DownloadConditionFailed |  
BadLambdaAssociated | InternalServerError | RobotApplicationDoesNotExist  
| DeploymentFleetDoesNotExist | FleetDeploymentTimeout

### failureReason

The failure reason of the deployment job if it failed.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

### fleet

The target fleet for the deployment job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### status

The status of the deployment job.

Type: String

Valid Values: Pending | Preparing | InProgress | Failed | Succeeded |  
Canceled

### tags

The list of all tags added to the deployment job.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **ConcurrentDeploymentException**

The failure percentage threshold percentage was met.

HTTP Status Code: 400

### **IdempotentParameterMismatchException**

The request uses the same client token as a previous, but non-identical request. Do not reuse a client token with different requests, unless the requests are identical.

HTTP Status Code: 400

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **LimitExceededException**

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

HTTP Status Code: 400

### **ResourceNotFoundException**

The specified resource does not exist.

HTTP Status Code: 400

## ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# CreateFleet

*This action has been deprecated.*

## Important

This API is no longer supported and will throw an error if used. For more information, see the January 31, 2022 update in the [Support policy](#) page.

Creates a fleet, a logical group of robots running the same robot application.

## Request Syntax

```
POST /createFleet HTTP/1.1
Content-type: application/json

{
  "name": "string",
  "tags": {
    "string" : "string"
  }
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### name

The name of the fleet.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

Required: Yes

### tags

A map that contains tag keys and tag values that are attached to the fleet.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "createdAt": number,
  "name": "string",
  "tags": {
    "string" : "string"
  }
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the fleet.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `a1n:.*`

### createdAt

The time, in milliseconds since the epoch, when the fleet was created.

Type: Timestamp

### name

The name of the fleet.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[a-zA-Z0-9_\-\ ]*`

### tags

The list of all tags added to the fleet.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\|/+=: ]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\|/+=: ]*`

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerError

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **LimitExceededException**

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

HTTP Status Code: 400

### **ThrottlingException**

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



# CreateRobot

*This action has been deprecated.*

## Important

This API is no longer supported and will throw an error if used. For more information, see the January 31, 2022 update in the [Support policy](#) page.

Creates a robot.

## Request Syntax

```
POST /createRobot HTTP/1.1
Content-type: application/json

{
  "architecture": "string",
  "greengrassGroupId": "string",
  "name": "string",
  "tags": {
    "string" : "string"
  }
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### architecture

The target architecture of the robot.

Type: String

Valid Values: X86\_64 | ARM64 | ARMHF

Required: Yes

### greengrassGroupId

The Greengrass group id.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: .\*

Required: Yes

### name

The name for the robot.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-\ ]\*

Required: Yes

### tags

A map that contains tag keys and tag values that are attached to the robot.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\|/+=: ]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: [a-zA-Z0-9 \_.\-\|/+=: ]\*

Required: No

## Response Syntax

```
HTTP/1.1 200
```

```
Content-type: application/json

{
  "architecture": "string",
  "arn": "string",
  "createdAt": number,
  "greengrassGroupId": "string",
  "name": "string",
  "tags": {
    "string" : "string"
  }
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### architecture

The target architecture of the robot.

Type: String

Valid Values: X86\_64 | ARM64 | ARMHF

### arn

The Amazon Resource Name (ARN) of the robot.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### createdAt

The time, in milliseconds since the epoch, when the robot was created.

Type: Timestamp

### greengrassGroupId

The Amazon Resource Name (ARN) of the Greengrass group associated with the robot.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: .\*

### name

The name of the robot.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

### tags

The list of all tags added to the robot.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **LimitExceededException**

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

HTTP Status Code: 400

### **ResourceAlreadyExistsException**

The specified resource already exists.

HTTP Status Code: 400

### **ThrottlingException**

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# CreateRobotApplication

Creates a robot application.

## Request Syntax

```
POST /createRobotApplication HTTP/1.1
Content-type: application/json
```

```
{
  "environment": {
    "uri": "string"
  },
  "name": "string",
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
      "architecture": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ],
  "tags": {
    "string" : "string"
  }
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### environment

The object that contains that URI of the Docker image that you use for your robot application.

Type: [Environment](#) object

Required: No

### name

The name of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

Required: Yes

### robotSoftwareSuite

The robot software suite used by the robot application.

Type: [RobotSoftwareSuite](#) object

Required: Yes

### sources

The sources of the robot application.

Type: Array of [SourceConfig](#) objects

Required: No

### tags

A map that contains tag keys and tag values that are attached to the robot application.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "environment": {
    "uri": "string"
  },
  "lastUpdatedAt": number,
  "name": "string",
  "revisionId": "string",
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
      "architecture": "string",
      "etag": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ],
  "tags": {
    "string" : "string"
  },
  "version": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the robot application.



Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

### environment

An object that contains the Docker image URI used to create your robot application.

Type: [Environment](#) object

### lastUpdatedAt

The time, in milliseconds since the epoch, when the robot application was last updated.

Type: Timestamp

### name

The name of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[a-zA-Z0-9_\-]*`

### revisionId

The revision id of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 40.

Pattern: `[a-zA-Z0-9_\. \-]*`

### robotSoftwareSuite

The robot software suite used by the robot application.

Type: [RobotSoftwareSuite](#) object

### sources

The sources of the robot application.

Type: Array of [Source](#) objects

### tags

The list of all tags added to the robot application.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

### version

The version of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `(\$\{LATEST})|[0-9]*`

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **IdempotentParameterMismatchException**

The request uses the same client token as a previous, but non-identical request. Do not reuse a client token with different requests, unless the requests are identical.

HTTP Status Code: 400

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

## **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

## **LimitExceededException**

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

HTTP Status Code: 400

## **ResourceAlreadyExistsException**

The specified resource already exists.

HTTP Status Code: 400

## **ThrottlingException**

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



# CreateRobotApplicationVersion

Creates a version of a robot application.

## Request Syntax

```
POST /createRobotApplicationVersion HTTP/1.1
Content-type: application/json
```

```
{
  "application": "string",
  "currentRevisionId": "string",
  "imageDigest": "string",
  "s3Etags": [ "string" ]
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### application

The application information for the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: Yes

### currentRevisionId

The current revision id for the robot application. If you provide a value and it matches the latest revision ID, a new version will be created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 40.

Pattern: `[a-zA-Z0-9_.\-]*`

Required: No

### imageDigest

A SHA256 identifier for the Docker image that you use for your robot application.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 72.

Pattern: `[Ss][Hh][Aa]256:[0-9a-fA-F]{64}`

Required: No

### s3Etags

The Amazon S3 identifier for the zip file bundle that you use for your robot application.

Type: Array of strings

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "environment": {
    "uri": "string"
  },
  "lastUpdatedAt": number,
  "name": "string",
  "revisionId": "string",
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
```

```
{
  "architecture": "string",
  "etag": "string",
  "s3Bucket": "string",
  "s3Key": "string"
},
"version": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### environment

The object that contains the Docker image URI used to create your robot application.

Type: [Environment](#) object

### lastUpdatedAt

The time, in milliseconds since the epoch, when the robot application was last updated.

Type: Timestamp

### name

The name of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

### revisionId

The revision id of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 40.

Pattern: [a-zA-Z0-9\_.\-]\*

### robotSoftwareSuite

The robot software suite used by the robot application.

Type: [RobotSoftwareSuite](#) object

### sources

The sources of the robot application.

Type: Array of [Source](#) objects

### version

The version of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: (\\$LATEST) | [0-9]\*

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **IdempotentParameterMismatchException**

The request uses the same client token as a previous, but non-identical request. Do not reuse a client token with different requests, unless the requests are identical.

HTTP Status Code: 400



## InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

## InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

## LimitExceededException

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

HTTP Status Code: 400

## ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



# CreateSimulationApplication

Creates a simulation application.

## Request Syntax

```
POST /createSimulationApplication HTTP/1.1
Content-type: application/json
```

```
{
  "environment": {
    "uri": "string"
  },
  "name": "string",
  "renderingEngine": {
    "name": "string",
    "version": "string"
  },
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "simulationSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
      "architecture": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ],
  "tags": {
    "string" : "string"
  }
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### environment

The object that contains the Docker image URI used to create your simulation application.

Type: [Environment](#) object

Required: No

### name

The name of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

Required: Yes

### renderingEngine

The rendering engine for the simulation application.

Type: [RenderingEngine](#) object

Required: No

### robotSoftwareSuite

The robot software suite used by the simulation application.

Type: [RobotSoftwareSuite](#) object

Required: Yes

### simulationSoftwareSuite

The simulation software suite used by the simulation application.

Type: [SimulationSoftwareSuite](#) object

Required: Yes

## sources

The sources of the simulation application.

Type: Array of [SourceConfig](#) objects

Required: No

## tags

A map that contains tag keys and tag values that are attached to the simulation application.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "environment": {
    "uri": "string"
  },
  "lastUpdatedAt": number,
  "name": "string",
  "renderingEngine": {
    "name": "string",
    "version": "string"
  },
  "revisionId": "string",
```

```

"robotSoftwareSuite": {
  "name": "string",
  "version": "string"
},
"simulationSoftwareSuite": {
  "name": "string",
  "version": "string"
},
"sources": [
  {
    "architecture": "string",
    "etag": "string",
    "s3Bucket": "string",
    "s3Key": "string"
  }
],
"tags": {
  "string" : "string"
},
"version": "string"
}

```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### environment

The object that contains the Docker image URI that you used to create your simulation application.

Type: [Environment](#) object

## lastUpdatedAt

The time, in milliseconds since the epoch, when the simulation application was last updated.

Type: Timestamp

## name

The name of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

## renderingEngine

The rendering engine for the simulation application.

Type: [RenderingEngine](#) object

## revisionId

The revision id of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 40.

Pattern: [a-zA-Z0-9\_.\-]\*

## robotSoftwareSuite

Information about the robot software suite.

Type: [RobotSoftwareSuite](#) object

## simulationSoftwareSuite

The simulation software suite used by the simulation application.

Type: [SimulationSoftwareSuite](#) object

## sources

The sources of the simulation application.

Type: Array of [Source](#) objects

### tags

The list of all tags added to the simulation application.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

### version

The version of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `(\$\{LATEST})|[0-9]*`

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **IdempotentParameterMismatchException**

The request uses the same client token as a previous, but non-identical request. Do not reuse a client token with different requests, unless the requests are identical.

HTTP Status Code: 400

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500



## **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

## **LimitExceededException**

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

HTTP Status Code: 400

## **ResourceAlreadyExistsException**

The specified resource already exists.

HTTP Status Code: 400

## **ThrottlingException**

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



# CreateSimulationApplicationVersion

Creates a simulation application with a specific revision id.

## Request Syntax

```
POST /createSimulationApplicationVersion HTTP/1.1
Content-type: application/json
```

```
{
  "application": "string",
  "currentRevisionId": "string",
  "imageDigest": "string",
  "s3Etags": [ "string" ]
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### application

The application information for the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

### currentRevisionId

The current revision id for the simulation application. If you provide a value and it matches the latest revision ID, a new version will be created.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 40.

Pattern: `[a-zA-Z0-9_.\-]*`

Required: No

### imageDigest

The SHA256 digest used to identify the Docker image URI used to create the simulation application.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 72.

Pattern: `[Ss][Hh][Aa]256:[0-9a-fA-F]{64}`

Required: No

### s3Etags

The Amazon S3 eTag identifier for the zip file bundle that you use to create the simulation application.

Type: Array of strings

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json
```

```
{
  "arn": "string",
  "environment": {
    "uri": "string"
  },
  "lastUpdatedAt": number,
  "name": "string",
  "renderingEngine": {
    "name": "string",
    "version": "string"
  },
}
```

```
"revisionId": "string",
"robotSoftwareSuite": {
  "name": "string",
  "version": "string"
},
"simulationSoftwareSuite": {
  "name": "string",
  "version": "string"
},
"sources": [
  {
    "architecture": "string",
    "etag": "string",
    "s3Bucket": "string",
    "s3Key": "string"
  }
],
"version": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### environment

The object that contains the Docker image URI used to create the simulation application.

Type: [Environment](#) object

### lastUpdatedAt

The time, in milliseconds since the epoch, when the simulation application was last updated.

Type: Timestamp

### name

The name of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

### renderingEngine

The rendering engine for the simulation application.

Type: [RenderingEngine](#) object

### revisionId

The revision ID of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 40.

Pattern: [a-zA-Z0-9\_.\-]\*

### robotSoftwareSuite

Information about the robot software suite.

Type: [RobotSoftwareSuite](#) object

### simulationSoftwareSuite

The simulation software suite used by the simulation application.

Type: [SimulationSoftwareSuite](#) object

### sources

The sources of the simulation application.

Type: Array of [Source](#) objects

## version

The version of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: (\\$LATEST) | [0-9]\*

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **IdempotentParameterMismatchException**

The request uses the same client token as a previous, but non-identical request. Do not reuse a client token with different requests, unless the requests are identical.

HTTP Status Code: 400

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **LimitExceededException**

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

HTTP Status Code: 400

### **ThrottlingException**

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



# CreateSimulationJob

Creates a simulation job.

## Note

After 90 days, simulation jobs expire and will be deleted. They will no longer be accessible.

## Request Syntax

```
POST /createSimulationJob HTTP/1.1
Content-type: application/json

{
  "clientRequestToken": "string",
  "compute": {
    "computeType": "string",
    "gpuUnitLimit": number,
    "simulationUnitLimit": number
  },
  "dataSources": [
    {
      "destination": "string",
      "name": "string",
      "s3Bucket": "string",
      "s3Keys": [ "string" ],
      "type": "string"
    }
  ],
  "failureBehavior": "string",
  "iamRole": "string",
  "loggingConfig": {
    "recordAllRosTopics": boolean
  },
  "maxJobDurationInSeconds": number,
  "outputLocation": {
    "s3Bucket": "string",
    "s3Prefix": "string"
  },
  "robotApplications": [
    {
```

```
"application": "string",
"applicationVersion": "string",
"launchConfig": {
  "command": [ "string" ],
  "environmentVariables": {
    "string": "string"
  },
  "launchFile": "string",
  "packageName": "string",
  "portForwardingConfig": {
    "portMappings": [
      {
        "applicationPort": number,
        "enableOnPublicIp": boolean,
        "jobPort": number
      }
    ]
  },
  "streamUI": boolean
},
"tools": [
  {
    "command": "string",
    "exitBehavior": "string",
    "name": "string",
    "streamOutputToCloudWatch": boolean,
    "streamUI": boolean
  }
],
"uploadConfigurations": [
  {
    "name": "string",
    "path": "string",
    "uploadBehavior": "string"
  }
],
"useDefaultTools": boolean,
"useDefaultUploadConfigurations": boolean
}
],
"simulationApplications": [
  {
    "application": "string",
    "applicationVersion": "string",
```

```

"launchConfig": {
  "command": [ "string" ],
  "environmentVariables": {
    "string" : "string"
  },
  "launchFile": "string",
  "packageName": "string",
  "portForwardingConfig": {
    "portMappings": [
      {
        "applicationPort": number,
        "enableOnPublicIp": boolean,
        "jobPort": number
      }
    ]
  },
  "streamUI": boolean
},
"tools": [
  {
    "command": "string",
    "exitBehavior": "string",
    "name": "string",
    "streamOutputToCloudWatch": boolean,
    "streamUI": boolean
  }
],
"uploadConfigurations": [
  {
    "name": "string",
    "path": "string",
    "uploadBehavior": "string"
  }
],
"useDefaultTools": boolean,
"useDefaultUploadConfigurations": boolean,
"worldConfigs": [
  {
    "world": "string"
  }
]
}
],
"tags": {

```

```
    "string" : "string"
  },
  "vpcConfig": {
    "assignPublicIp": boolean,
    "securityGroups": [ "string" ],
    "subnets": [ "string" ]
  }
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### clientRequestToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: [a-zA-Z0-9\_\-=]\*

Required: No

### compute

Compute information for the simulation job.

Type: [Compute](#) object

Required: No

### dataSources

Specify data sources to mount read-only files from S3 into your simulation. These files are available under `/opt/robomaker/datasources/data_source_name`.

**Note**

There is a limit of 100 files and a combined size of 25GB for all DataSourceConfig objects.

Type: Array of [DataSourceConfig](#) objects

Array Members: Minimum number of 1 item. Maximum number of 6 items.

Required: No

**[failureBehavior](#)**

The failure behavior the simulation job.

Continue

Leaves the instance running for its maximum timeout duration after a 4XX error code.

Fail

Stop the simulation job and terminate the instance.

Type: String

Valid Values: Fail | Continue

Required: No

**[iamRole](#)**

The IAM role name that allows the simulation instance to call the AWS APIs that are specified in its associated policies on your behalf. This is how credentials are passed in to your simulation job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `arn:aws:iam::\w+:role/.*`

Required: Yes

**[loggingConfig](#)**

The logging configuration.

Type: [LoggingConfig](#) object

Required: No

### [maxJobDurationInSeconds](#)

The maximum simulation job duration in seconds (up to 14 days or 1,209,600 seconds. When maxJobDurationInSeconds is reached, the simulation job will status will transition to Completed.

Type: Long

Required: Yes

### [outputLocation](#)

Location for output files generated by the simulation job.

Type: [OutputLocation](#) object

Required: No

### [robotApplications](#)

The robot application to use in the simulation job.

Type: Array of [RobotApplicationConfig](#) objects

Array Members: Fixed number of 1 item.

Required: No

### [simulationApplications](#)

The simulation application to use in the simulation job.

Type: Array of [SimulationApplicationConfig](#) objects

Array Members: Fixed number of 1 item.

Required: No

### [tags](#)

A map that contains tag keys and tag values that are attached to the simulation job.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Required: No

### [vpcConfig](#)

If your simulation job accesses resources in a VPC, you provide this parameter identifying the list of security group IDs and subnet IDs. These must belong to the same VPC. You must provide at least one security group and one subnet ID.

Type: [VPCConfig](#) object

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "clientRequestToken": "string",
  "compute": {
    "computeType": "string",
    "gpuUnitLimit": number,
    "simulationUnitLimit": number
  },
  "dataSources": [
    {
      "destination": "string",
      "name": "string",
      "s3Bucket": "string",
      "s3Keys": [
        {
          "etag": "string",
          "s3Key": "string"
        }
      ]
    }
  ],
}
```

```

    "type": "string"
  }
],
"failureBehavior": "string",
"failureCode": "string",
"iamRole": "string",
"lastStartedAt": number,
"lastUpdatedAt": number,
"loggingConfig": {
  "recordAllRosTopics": boolean
},
"maxJobDurationInSeconds": number,
"outputLocation": {
  "s3Bucket": "string",
  "s3Prefix": "string"
},
"robotApplications": [
  {
    "application": "string",
    "applicationVersion": "string",
    "launchConfig": {
      "command": [ "string" ],
      "environmentVariables": {
        "string" : "string"
      },
      "launchFile": "string",
      "packageName": "string",
      "portForwardingConfig": {
        "portMappings": [
          {
            "applicationPort": number,
            "enableOnPublicIp": boolean,
            "jobPort": number
          }
        ]
      },
      "streamUI": boolean
    },
    "tools": [
      {
        "command": "string",
        "exitBehavior": "string",
        "name": "string",
        "streamOutputToCloudWatch": boolean,

```



```

        "streamUI": boolean
    }
],
"uploadConfigurations": [
    {
        "name": "string",
        "path": "string",
        "uploadBehavior": "string"
    }
],
"useDefaultTools": boolean,
"useDefaultUploadConfigurations": boolean
}
],
"simulationApplications": [
    {
        "application": "string",
        "applicationVersion": "string",
        "launchConfig": {
            "command": [ "string " ],
            "environmentVariables": {
                "string": "string"
            },
            "launchFile": "string",
            "packageName": "string",
            "portForwardingConfig": {
                "portMappings": [
                    {
                        "applicationPort": number,
                        "enableOnPublicIp": boolean,
                        "jobPort": number
                    }
                ]
            },
            "streamUI": boolean
        },
        "tools": [
            {
                "command": "string",
                "exitBehavior": "string",
                "name": "string",
                "streamOutputToCloudWatch": boolean,
                "streamUI": boolean
            }
        ]
    }
]

```

```

    ],
    "uploadConfigurations": [
      {
        "name": "string",
        "path": "string",
        "uploadBehavior": "string"
      }
    ],
    "useDefaultTools": boolean,
    "useDefaultUploadConfigurations": boolean,
    "worldConfigs": [
      {
        "world": "string"
      }
    ]
  }
],
"simulationTimeMillis": number,
"status": "string",
"tags": {
  "string" : "string"
},
"vpcConfig": {
  "assignPublicIp": boolean,
  "securityGroups": [ "string" ],
  "subnets": [ "string" ],
  "vpcId": "string"
}
}

```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the simulation job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

### clientRequestToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `[a-zA-Z0-9_\-=]*`

### compute

Compute information for the simulation job.

Type: [ComputeResponse](#) object

### dataSources

The data sources for the simulation job.

Type: Array of [DataSource](#) objects

### failureBehavior

the failure behavior for the simulation job.

Type: String

Valid Values: `Fail` | `Continue`

### failureCode

The failure code of the simulation job if it failed:

`InternalServerError`

Internal service error.

`RobotApplicationCrash`

Robot application exited abnormally.

`SimulationApplicationCrash`

Simulation application exited abnormally.

### BadPermissionsRobotApplication

Robot application bundle could not be downloaded.

### BadPermissionsSimulationApplication

Simulation application bundle could not be downloaded.

### BadPermissionsS3Output

Unable to publish outputs to customer-provided S3 bucket.

### BadPermissionsCloudwatchLogs

Unable to publish logs to customer-provided CloudWatch Logs resource.

### SubnetIpLimitExceeded

Subnet IP limit exceeded.

### ENILimitExceeded

ENI limit exceeded.

### BadPermissionsUserCredentials

Unable to use the Role provided.

### InvalidBundleRobotApplication

Robot bundle cannot be extracted (invalid format, bundling error, or other issue).

### InvalidBundleSimulationApplication

Simulation bundle cannot be extracted (invalid format, bundling error, or other issue).

### RobotApplicationVersionMismatchedEtag

Etag for RobotApplication does not match value during version creation.

### SimulationApplicationVersionMismatchedEtag

Etag for SimulationApplication does not match value during version creation.

Type: String

Valid Values: `InternalServerError` | `RobotApplicationCrash` | `SimulationApplicationCrash` | `RobotApplicationHealthCheckFailure` | `SimulationApplicationHealthCheckFailure` | `BadPermissionsRobotApplication`

| BadPermissionsSimulationApplication | BadPermissionsS3Object  
| BadPermissionsS3Output | BadPermissionsCloudwatchLogs |  
SubnetIpLimitExceeded | ENILimitExceeded | BadPermissionsUserCredentials  
| InvalidBundleRobotApplication | InvalidBundleSimulationApplication  
| InvalidS3Resource | ThrottlingError | LimitExceeded |  
MismatchedEtag | RobotApplicationVersionMismatchedEtag |  
SimulationApplicationVersionMismatchedEtag | ResourceNotFound |  
RequestThrottled | BatchTimedOut | BatchCanceled | InvalidInput |  
WrongRegionS3Bucket | WrongRegionS3Output | WrongRegionRobotApplication  
| WrongRegionSimulationApplication | UploadContentMismatchError

### iamRole

The IAM role that allows the simulation job to call the AWS APIs that are specified in its associated policies on your behalf.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `arn:aws:iam::\w+:role/.*`

### lastStartedAt

The time, in milliseconds since the epoch, when the simulation job was last started.

Type: Timestamp

### lastUpdatedAt

The time, in milliseconds since the epoch, when the simulation job was last updated.

Type: Timestamp

### loggingConfig

The logging configuration.

Type: [LoggingConfig](#) object

### maxJobDurationInSeconds

The maximum simulation job duration in seconds.

Type: Long

## outputLocation

Simulation job output files location.

Type: [OutputLocation](#) object

## robotApplications

The robot application used by the simulation job.

Type: Array of [RobotApplicationConfig](#) objects

Array Members: Fixed number of 1 item.

## simulationApplications

The simulation application used by the simulation job.

Type: Array of [SimulationApplicationConfig](#) objects

Array Members: Fixed number of 1 item.

## simulationTimeMillis

The simulation job execution duration in milliseconds.

Type: Long

## status

The status of the simulation job.

Type: String

Valid Values: Pending | Preparing | Running | Restarting | Completed | Failed | RunningFailed | Terminating | Terminated | Canceled

## tags

The list of all tags added to the simulation job.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

## [vpcConfig](#)

Information about the vpc configuration.

Type: [VPCConfigResponse](#) object

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **IdempotentParameterMismatchException**

The request uses the same client token as a previous, but non-identical request. Do not reuse a client token with different requests, unless the requests are identical.

HTTP Status Code: 400

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **LimitExceededException**

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

HTTP Status Code: 400

### **ResourceNotFoundException**

The specified resource does not exist.

HTTP Status Code: 400

### **ServiceUnavailableException**

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

### **ThrottlingException**

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



# CreateWorldExportJob

Creates a world export job.

## Request Syntax

```
POST /createWorldExportJob HTTP/1.1
Content-type: application/json
```

```
{
  "clientRequestToken": "string",
  "iamRole": "string",
  "outputLocation": {
    "s3Bucket": "string",
    "s3Prefix": "string"
  },
  "tags": {
    "string" : "string"
  },
  "worlds": [ "string" ]
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### clientRequestToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: [a-zA-Z0-9\_\-=]\*

Required: No

## iamRole

The IAM role that the world export process uses to access the Amazon S3 bucket and put the export.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `arn:aws:iam::\w+:role/.*`

Required: Yes

## outputLocation

The output location.

Type: [OutputLocation](#) object

Required: Yes

## tags

A map that contains tag keys and tag values that are attached to the world export job.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Required: No

## worlds

A list of Amazon Resource Names (arns) that correspond to worlds to export.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "clientRequestToken": "string",
  "createdAt": number,
  "failureCode": "string",
  "iamRole": "string",
  "outputLocation": {
    "s3Bucket": "string",
    "s3Prefix": "string"
  },
  "status": "string",
  "tags": {
    "string" : "string"
  }
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the world export job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

## clientRequestToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `[a-zA-Z0-9_\-=]*`

## createdAt

The time, in milliseconds since the epoch, when the world export job was created.

Type: Timestamp

## failureCode

The failure code of the world export job if it failed:

**InternalServerError**

Internal service error.

**LimitExceeded**

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

**ResourceNotFound**

The specified resource could not be found.

**RequestThrottled**

The request was throttled.

**InvalidInput**

An input parameter in the request is not valid.

**AllWorldGenerationFailed**

All of the worlds in the world generation job failed. This can happen if your `worldCount` is greater than 50 or less than 1.

For more information about troubleshooting WorldForge, see [Troubleshooting Simulation WorldForge](#).

Type: String

Valid Values: `InternalServerError` | `LimitExceeded` | `ResourceNotFound` | `RequestThrottled` | `InvalidInput` | `AccessDenied`

### iamRole

The IAM role that the world export process uses to access the Amazon S3 bucket and put the export.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `arn:aws:iam::\w+:role/.*`

### outputLocation

The output location.

Type: [OutputLocation](#) object

### status

The status of the world export job.

Pending

The world export job request is pending.

Running

The world export job is running.

Completed

The world export job completed.

Failed

The world export job failed. See `failureCode` for more information.

Canceled

The world export job was cancelled.

Canceling

The world export job is being cancelled.

Type: String

Valid Values: Pending | Running | Completed | Failed | Canceling | Canceled

### tags

A map that contains tag keys and tag values that are attached to the world export job.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **IdempotentParameterMismatchException**

The request uses the same client token as a previous, but non-identical request. Do not reuse a client token with different requests, unless the requests are identical.

HTTP Status Code: 400

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

## ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

## ServiceUnavailableException

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

## ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# CreateWorldGenerationJob

Creates worlds using the specified template.

## Request Syntax

```
POST /createWorldGenerationJob HTTP/1.1
Content-type: application/json

{
  "clientRequestToken": "string",
  "tags": {
    "string" : "string"
  },
  "template": "string",
  "worldCount": {
    "floorplanCount": number,
    "interiorCountPerFloorplan": number
  },
  "worldTags": {
    "string" : "string"
  }
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### clientRequestToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: [a-zA-Z0-9\_\-=]\*

Required: No



## tags

A map that contains tag keys and tag values that are attached to the world generator job.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Required: No

## template

The Amazon Resource Name (arn) of the world template describing the worlds you want to create.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: Yes

## worldCount

Information about the world count.

Type: [WorldCount](#) object

Required: Yes

## worldTags

A map that contains tag keys and tag values that are attached to the generated worlds.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "clientRequestToken": "string",
  "createdAt": number,
  "failureCode": "string",
  "status": "string",
  "tags": {
    "string" : "string"
  },
  "template": "string",
  "worldCount": {
    "floorplanCount": number,
    "interiorCountPerFloorplan": number
  },
  "worldTags": {
    "string" : "string"
  }
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the world generator job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

### clientRequestToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `[a-zA-Z0-9_\-=]*`

### createdAt

The time, in milliseconds since the epoch, when the world generator job was created.

Type: Timestamp

### failureCode

The failure code of the world generator job if it failed:

**InternalServerError**

Internal service error.

**LimitExceeded**

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

**ResourceNotFound**

The specified resource could not be found.

**RequestThrottled**

The request was throttled.

**InvalidInput**

An input parameter in the request is not valid.

Type: String

Valid Values: `InternalServerError` | `LimitExceeded` | `ResourceNotFound` | `RequestThrottled` | `InvalidInput` | `AllWorldGenerationFailed`

### status

The status of the world generator job.

Pending

The world generator job request is pending.

Running

The world generator job is running.

Completed

The world generator job completed.

Failed

The world generator job failed. See `failureCode` for more information.

PartialFailed

Some worlds did not generate.

Canceled

The world generator job was cancelled.

Canceling

The world generator job is being cancelled.

Type: String

Valid Values: `Pending` | `Running` | `Completed` | `Failed` | `PartialFailed` | `Canceling` | `Canceled`

### tags

A map that contains tag keys and tag values that are attached to the world generator job.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

### template

The Amazon Resource Name (arn) of the world template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

### worldCount

Information about the world count.

Type: [WorldCount](#) object

### worldTags

A map that contains tag keys and tag values that are attached to the generated worlds.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **IdempotentParameterMismatchException**

The request uses the same client token as a previous, but non-identical request. Do not reuse a client token with different requests, unless the requests are identical.

HTTP Status Code: 400

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **LimitExceededException**

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

HTTP Status Code: 400

### **ResourceNotFoundException**

The specified resource does not exist.

HTTP Status Code: 400

### **ServiceUnavailableException**

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

### **ThrottlingException**

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)

- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# CreateWorldTemplate

Creates a world template.

## Request Syntax

```
POST /createWorldTemplate HTTP/1.1
Content-type: application/json
```

```
{
  "clientRequestToken": "string",
  "name": "string",
  "tags": {
    "string" : "string"
  },
  "templateBody": "string",
  "templateLocation": {
    "s3Bucket": "string",
    "s3Key": "string"
  }
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### clientRequestToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: [a-zA-Z0-9\_\-=]\*

Required: No



## name

The name of the world template.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 255.

Pattern: .\*

Required: No

## tags

A map that contains tag keys and tag values that are attached to the world template.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Required: No

## templateBody

The world template body.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 262144.

Pattern: [\\S\\s]+

Required: No

## templateLocation

The location of the world template.

Type: [TemplateLocation](#) object

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "clientRequestToken": "string",
  "createdAt": number,
  "name": "string",
  "tags": {
    "string" : "string"
  }
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the world template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### clientRequestToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: [a-zA-Z0-9\_\-=]\*

### createdAt

The time, in milliseconds since the epoch, when the world template was created.

Type: Timestamp

### name

The name of the world template.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 255.

Pattern: .\*

### tags

A map that contains tag keys and tag values that are attached to the world template.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerError

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **LimitExceededException**

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

HTTP Status Code: 400

### **ResourceAlreadyExistsException**

The specified resource already exists.

HTTP Status Code: 400

### **ResourceNotFoundException**

The specified resource does not exist.

HTTP Status Code: 400

### **ThrottlingException**

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



# DeleteFleet

*This action has been deprecated.*

## Important

This API is no longer supported. For more information, see the May 2, 2022 update in the [Support policy](#) page.

Deletes a fleet.

## Request Syntax

```
POST /deleteFleet HTTP/1.1
Content-type: application/json

{
  "fleet": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### fleet

The Amazon Resource Name (ARN) of the fleet.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: Yes

## Response Syntax

```
HTTP/1.1 200
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)

- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



# DeleteRobot

*This action has been deprecated.*

## Important

This API is no longer supported. For more information, see the May 2, 2022 update in the [Support policy](#) page.

Deletes a robot.

## Request Syntax

```
POST /deleteRobot HTTP/1.1
Content-type: application/json

{
  "robot": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### robot

The Amazon Resource Name (ARN) of the robot.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

## Response Syntax

```
HTTP/1.1 200
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)

- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DeleteRobotApplication

Deletes a robot application.

## Request Syntax

```
POST /deleteRobotApplication HTTP/1.1
Content-type: application/json

{
  "application": "string",
  "applicationVersion": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### application

The Amazon Resource Name (ARN) of the the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

### applicationVersion

The version of the robot application to delete.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: (\\$LATEST) | [0-9]\*

Required: No

## Response Syntax

```
HTTP/1.1 200
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)

- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DeleteSimulationApplication

Deletes a simulation application.

## Request Syntax

```
POST /deleteSimulationApplication HTTP/1.1
Content-type: application/json

{
  "application": "string",
  "applicationVersion": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### application

The application information for the simulation application to delete.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

### applicationVersion

The version of the simulation application to delete.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: (\\$LATEST) | [0-9]\*

Required: No

## Response Syntax

```
HTTP/1.1 200
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)



- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DeleteWorldTemplate

Deletes a world template.

## Request Syntax

```
POST /deleteWorldTemplate HTTP/1.1
Content-type: application/json

{
  "template": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### template

The Amazon Resource Name (arn) of the world template you want to delete.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

## Response Syntax

```
HTTP/1.1 200
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DeregisterRobot

*This action has been deprecated.*

## Important

This API is no longer supported. For more information, see the May 2, 2022 update in the [Support policy](#) page.

Deregisters a robot.

## Request Syntax

```
POST /deregisterRobot HTTP/1.1
Content-type: application/json

{
  "fleet": "string",
  "robot": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### fleet

The Amazon Resource Name (ARN) of the fleet.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

## robot

The Amazon Resource Name (ARN) of the robot.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "fleet": "string",
  "robot": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### fleet

The Amazon Resource Name (ARN) of the fleet.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

### robot

The Amazon Resource Name (ARN) of the robot.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



# DescribeDeploymentJob

*This action has been deprecated.*

## Important

This API is no longer supported. For more information, see the May 2, 2022 update in the [Support policy](#) page.

Describes a deployment job.

## Request Syntax

```
POST /describeDeploymentJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### job

The Amazon Resource Name (ARN) of the deployment job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "createdAt": number,
  "deploymentApplicationConfigs": [
    {
      "application": "string",
      "applicationVersion": "string",
      "launchConfig": {
        "environmentVariables": {
          "string": "string"
        },
        "launchFile": "string",
        "packageName": "string",
        "postLaunchFile": "string",
        "preLaunchFile": "string"
      }
    }
  ],
  "deploymentConfig": {
    "concurrentDeploymentPercentage": number,
    "downloadConditionFile": {
      "bucket": "string",
      "etag": "string",
      "key": "string"
    },
    "failureThresholdPercentage": number,
    "robotDeploymentTimeoutInSeconds": number
  },
  "failureCode": "string",
  "failureReason": "string",
  "fleet": "string",
  "robotDeploymentSummary": [
    {
      "arn": "string",
      "deploymentFinishTime": number,
      "deploymentStartTime": number,
      "failureCode": "string",
      "failureReason": "string",
```

```
    "progressDetail": {
      "currentProgress": "string",
      "estimatedTimeRemainingSeconds": number,
      "percentDone": number,
      "targetResource": "string"
    },
    "status": "string"
  }
],
"status": "string",
"tags": {
  "string" : "string"
}
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the deployment job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### createdAt

The time, in milliseconds since the epoch, when the deployment job was created.

Type: Timestamp

### deploymentApplicationConfigs

The deployment application configuration.

Type: Array of [DeploymentApplicationConfig](#) objects

Array Members: Fixed number of 1 item.

## deploymentConfig

The deployment configuration.

Type: [DeploymentConfig](#) object

## failureCode

The deployment job failure code.

Type: String

Valid Values: ResourceNotFound | EnvironmentSetupError | EtagMismatch | FailureThresholdBreached | RobotDeploymentAborted | RobotDeploymentNoResponse | RobotAgentConnectionTimeout | GreengrassDeploymentFailed | InvalidGreengrassGroup | MissingRobotArchitecture | MissingRobotApplicationArchitecture | MissingRobotDeploymentResource | GreengrassGroupVersionDoesNotExist | LambdaDeleted | ExtractingBundleFailure | PreLaunchFileFailure | PostLaunchFileFailure | BadPermissionError | DownloadConditionFailed | BadLambdaAssociated | InternalServerError | RobotApplicationDoesNotExist | DeploymentFleetDoesNotExist | FleetDeploymentTimeout

## failureReason

A short description of the reason why the deployment job failed.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

## fleet

The Amazon Resource Name (ARN) of the fleet.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

## robotDeploymentSummary

A list of robot deployment summaries.

Type: Array of [RobotDeployment](#) objects

### status

The status of the deployment job.

Type: String

Valid Values: Pending | Preparing | InProgress | Failed | Succeeded | Canceled

### tags

The list of all tags added to the specified deployment job.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

## ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

## ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DescribeFleet

*This action has been deprecated.*

## Important

This API is no longer supported. For more information, see the May 2, 2022 update in the [Support policy](#) page.

Describes a fleet.

## Request Syntax

```
POST /describeFleet HTTP/1.1
Content-type: application/json

{
  "fleet": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### fleet

The Amazon Resource Name (ARN) of the fleet.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "createdAt": number,
  "lastDeploymentJob": "string",
  "lastDeploymentStatus": "string",
  "lastDeploymentTime": number,
  "name": "string",
  "robots": [
    {
      "architecture": "string",
      "arn": "string",
      "createdAt": number,
      "fleetArn": "string",
      "greenGrassGroupId": "string",
      "lastDeploymentJob": "string",
      "lastDeploymentTime": number,
      "name": "string",
      "status": "string"
    }
  ],
  "tags": {
    "string" : "string"
  }
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the fleet.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.



Pattern: `arn:.*`

### createdAt

The time, in milliseconds since the epoch, when the fleet was created.

Type: Timestamp

### lastDeploymentJob

The Amazon Resource Name (ARN) of the last deployment job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

### lastDeploymentStatus

The status of the last deployment.

Type: String

Valid Values: Pending | Preparing | InProgress | Failed | Succeeded | Canceled

### lastDeploymentTime

The time of the last deployment.

Type: Timestamp

### name

The name of the fleet.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[a-zA-Z0-9_\-\ ]*`

### robots

A list of robots.

Type: Array of [Robot](#) objects

Array Members: Minimum number of 0 items. Maximum number of 1000 items.

## tags

The list of all tags added to the specified fleet.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DescribeRobot

*This action has been deprecated.*

## Important

This API is no longer supported. For more information, see the May 2, 2022 update in the [Support policy](#) page.

Describes a robot.

## Request Syntax

```
POST /describeRobot HTTP/1.1
Content-type: application/json

{
  "robot": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### robot

The Amazon Resource Name (ARN) of the robot to be described.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "architecture": "string",
  "arn": "string",
  "createdAt": number,
  "fleetArn": "string",
  "greengrassGroupId": "string",
  "lastDeploymentJob": "string",
  "lastDeploymentTime": number,
  "name": "string",
  "status": "string",
  "tags": {
    "string" : "string"
  }
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### architecture

The target architecture of the robot application.

Type: String

Valid Values: X86\_64 | ARM64 | ARMHF

### arn

The Amazon Resource Name (ARN) of the robot.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### createdAt

The time, in milliseconds since the epoch, when the robot was created.

Type: Timestamp

### fleetArn

The Amazon Resource Name (ARN) of the fleet.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### greengrassGroupId

The Greengrass group id.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: .\*

### lastDeploymentJob

The Amazon Resource Name (ARN) of the last deployment job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### lastDeploymentTime

The time of the last deployment job.

Type: Timestamp

### name

The name of the robot.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[a-zA-Z0-9_\-\ ]*`

### status

The status of the fleet.

Type: String

Valid Values: Available | Registered | PendingNewDeployment | Deploying | Failed | InSync | NoResponse

### tags

The list of all tags added to the specified robot.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\|/+=: ]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\|/+=: ]*`

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

## ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

## ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



# DescribeRobotApplication

Describes a robot application.

## Request Syntax

```
POST /describeRobotApplication HTTP/1.1
Content-type: application/json

{
  "application": "string",
  "applicationVersion": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### application

The Amazon Resource Name (ARN) of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: Yes

### applicationVersion

The version of the robot application to describe.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `(\\$LATEST) | [0-9]*`

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "environment": {
    "uri": "string"
  },
  "imageDigest": "string",
  "lastUpdatedAt": number,
  "name": "string",
  "revisionId": "string",
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
      "architecture": "string",
      "etag": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ],
  "tags": {
    "string" : "string"
  },
  "version": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

### environment

The object that contains the Docker image URI used to create the robot application.

Type: [Environment](#) object

### imageDigest

A SHA256 identifier for the Docker image that you use for your robot application.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 72.

Pattern: `[Ss][Hh][Aa]256:[0-9a-fA-F]{64}`

### lastUpdatedAt

The time, in milliseconds since the epoch, when the robot application was last updated.

Type: Timestamp

### name

The name of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[a-zA-Z0-9_\-]*`

### revisionId

The revision id of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 40.

Pattern: `[a-zA-Z0-9_\. \-]*`

## [robotSoftwareSuite](#)

The robot software suite used by the robot application.

Type: [RobotSoftwareSuite](#) object

## [sources](#)

The sources of the robot application.

Type: Array of [Source](#) objects

## [tags](#)

The list of all tags added to the specified robot application.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

## [version](#)

The version of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `(\$\{LATEST})|[0-9]*`

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

## InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **ResourceNotFoundException**

The specified resource does not exist.

HTTP Status Code: 400

### **ThrottlingException**

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DescribeSimulationApplication

Describes a simulation application.

## Request Syntax

```
POST /describeSimulationApplication HTTP/1.1
Content-type: application/json

{
  "application": "string",
  "applicationVersion": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### application

The application information for the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: Yes

### applicationVersion

The version of the simulation application to describe.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `(\\$LATEST) | [0-9]*`

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "environment": {
    "uri": "string"
  },
  "imageDigest": "string",
  "lastUpdatedAt": number,
  "name": "string",
  "renderingEngine": {
    "name": "string",
    "version": "string"
  },
  "revisionId": "string",
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "simulationSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
      "architecture": "string",
      "etag": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ],
  "tags": {
    "string" : "string"
  },
  "version": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the robot simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### environment

The object that contains the Docker image URI used to create the simulation application.

Type: [Environment](#) object

### imageDigest

A SHA256 identifier for the Docker image that you use for your simulation application.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 72.

Pattern: [Ss][Hh][Aa]256:[0-9a-fA-F]{64}

### lastUpdatedAt

The time, in milliseconds since the epoch, when the simulation application was last updated.

Type: Timestamp

### name

The name of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

### renderingEngine

The rendering engine for the simulation application.



Type: [RenderingEngine](#) object

### [revisionId](#)

The revision id of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 40.

Pattern: `[a-zA-Z0-9_.\-]*`

### [robotSoftwareSuite](#)

Information about the robot software suite.

Type: [RobotSoftwareSuite](#) object

### [simulationSoftwareSuite](#)

The simulation software suite used by the simulation application.

Type: [SimulationSoftwareSuite](#) object

### [sources](#)

The sources of the simulation application.

Type: Array of [Source](#) objects

### [tags](#)

The list of all tags added to the specified simulation application.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

### [version](#)

The version of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: (\\$LATEST) | [0-9]\*

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)

- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DescribeSimulationJob

Describes a simulation job.

## Request Syntax

```
POST /describeSimulationJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### job

The Amazon Resource Name (ARN) of the simulation job to be described.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "clientRequestToken": "string",
  "compute": {
    "computeType": "string",
```

```

    "gpuUnitLimit": number,
    "simulationUnitLimit": number
  },
  "dataSources": [
    {
      "destination": "string",
      "name": "string",
      "s3Bucket": "string",
      "s3Keys": [
        {
          "etag": "string",
          "s3Key": "string"
        }
      ],
      "type": "string"
    }
  ],
  "failureBehavior": "string",
  "failureCode": "string",
  "failureReason": "string",
  "iamRole": "string",
  "lastStartedAt": number,
  "lastUpdatedAt": number,
  "loggingConfig": {
    "recordAllRosTopics": boolean
  },
  "maxJobDurationInSeconds": number,
  "name": "string",
  "networkInterface": {
    "networkInterfaceId": "string",
    "privateIpAddress": "string",
    "publicIpAddress": "string"
  },
  "outputLocation": {
    "s3Bucket": "string",
    "s3Prefix": "string"
  },
  "robotApplications": [
    {
      "application": "string",
      "applicationVersion": "string",
      "launchConfig": {
        "command": [ "string" ],
        "environmentVariables": {

```

```

        "string" : "string"
    },
    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
        "portMappings": [
            {
                "applicationPort": number,
                "enableOnPublicIp": boolean,
                "jobPort": number
            }
        ]
    },
    "streamUI": boolean
},
"tools": [
    {
        "command": "string",
        "exitBehavior": "string",
        "name": "string",
        "streamOutputToCloudWatch": boolean,
        "streamUI": boolean
    }
],
"uploadConfigurations": [
    {
        "name": "string",
        "path": "string",
        "uploadBehavior": "string"
    }
],
"useDefaultTools": boolean,
"useDefaultUploadConfigurations": boolean
}
],
"simulationApplications": [
    {
        "application": "string",
        "applicationVersion": "string",
        "launchConfig": {
            "command": [ "string" ],
            "environmentVariables": {
                "string" : "string"
            }
        }
    },

```

```

    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean,
  "worldConfigs": [
    {
      "world": "string"
    }
  ]
}
],
"simulationTimeMillis": number,
"status": "string",
"tags": {
  "string" : "string"
},
"vpcConfig": {

```

```
    "assignPublicIp": boolean,
    "securityGroups": [ "string" ],
    "subnets": [ "string" ],
    "vpcId": "string"
  }
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the simulation job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

### clientRequestToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `[a-zA-Z0-9_\-=]*`

### compute

Compute information for the simulation job.

Type: [ComputeResponse](#) object

### dataSources

The data sources for the simulation job.

Type: Array of [DataSource](#) objects



## failureBehavior

The failure behavior for the simulation job.

Type: String

Valid Values: Fail | Continue

## failureCode

The failure code of the simulation job if it failed:

InternalServerError

Internal service error.

RobotApplicationCrash

Robot application exited abnormally.

SimulationApplicationCrash

Simulation application exited abnormally.

BadPermissionsRobotApplication

Robot application bundle could not be downloaded.

BadPermissionsSimulationApplication

Simulation application bundle could not be downloaded.

BadPermissionsS3Output

Unable to publish outputs to customer-provided S3 bucket.

BadPermissionsCloudwatchLogs

Unable to publish logs to customer-provided CloudWatch Logs resource.

SubnetIpLimitExceeded

Subnet IP limit exceeded.

ENILimitExceeded

ENI limit exceeded.

BadPermissionsUserCredentials

Unable to use the Role provided.

### InvalidBundleRobotApplication

Robot bundle cannot be extracted (invalid format, bundling error, or other issue).

### InvalidBundleSimulationApplication

Simulation bundle cannot be extracted (invalid format, bundling error, or other issue).

### RobotApplicationVersionMismatchedEtag

Etag for RobotApplication does not match value during version creation.

### SimulationApplicationVersionMismatchedEtag

Etag for SimulationApplication does not match value during version creation.

Type: String

Valid Values: InternalServiceError | RobotApplicationCrash | SimulationApplicationCrash | RobotApplicationHealthCheckFailure | SimulationApplicationHealthCheckFailure | BadPermissionsRobotApplication | BadPermissionsSimulationApplication | BadPermissionsS3Object | BadPermissionsS3Output | BadPermissionsCloudwatchLogs | SubnetIpLimitExceeded | ENILimitExceeded | BadPermissionsUserCredentials | InvalidBundleRobotApplication | InvalidBundleSimulationApplication | InvalidS3Resource | ThrottlingError | LimitExceeded | MismatchedEtag | RobotApplicationVersionMismatchedEtag | SimulationApplicationVersionMismatchedEtag | ResourceNotFound | RequestThrottled | BatchTimedOut | BatchCanceled | InvalidInput | WrongRegionS3Bucket | WrongRegionS3Output | WrongRegionRobotApplication | WrongRegionSimulationApplication | UploadContentMismatchError

### failureReason

Details about why the simulation job failed. For more information about troubleshooting, see [Troubleshooting](#).

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

## iamRole

The IAM role that allows the simulation instance to call the AWS APIs that are specified in its associated policies on your behalf.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `arn:aws:iam::\w+:role/.*`

## lastStartedAt

The time, in milliseconds since the epoch, when the simulation job was last started.

Type: Timestamp

## lastUpdatedAt

The time, in milliseconds since the epoch, when the simulation job was last updated.

Type: Timestamp

## loggingConfig

The logging configuration.

Type: [LoggingConfig](#) object

## maxJobDurationInSeconds

The maximum job duration in seconds. The value must be 8 days (691,200 seconds) or less.

Type: Long

## name

The name of the simulation job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[a-zA-Z0-9_\-\ ]*`

## networkInterface

The network interface information for the simulation job.

Type: [NetworkInterface](#) object

### [outputLocation](#)

Location for output files generated by the simulation job.

Type: [OutputLocation](#) object

### [robotApplications](#)

A list of robot applications.

Type: Array of [RobotApplicationConfig](#) objects

Array Members: Fixed number of 1 item.

### [simulationApplications](#)

A list of simulation applications.

Type: Array of [SimulationApplicationConfig](#) objects

Array Members: Fixed number of 1 item.

### [simulationTimeMillis](#)

The simulation job execution duration in milliseconds.

Type: Long

### [status](#)

The status of the simulation job.

Type: String

Valid Values: Pending | Preparing | Running | Restarting | Completed | Failed | RunningFailed | Terminating | Terminated | Canceled

### [tags](#)

The list of all tags added to the specified simulation job.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

## **vpcConfig**

The VPC configuration.

Type: [VPCConfigResponse](#) object

## **Errors**

For information about the errors that are common to all actions, see [Common Errors](#).

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **ResourceNotFoundException**

The specified resource does not exist.

HTTP Status Code: 400

### **ThrottlingException**

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DescribeSimulationJobBatch

Describes a simulation job batch.

## Request Syntax

```
POST /describeSimulationJobBatch HTTP/1.1
Content-type: application/json

{
  "batch": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### batch

The id of the batch to describe.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "batchPolicy": {
    "maxConcurrency": number,
    "timeoutInSeconds": number
  }
}
```

```

},
"clientRequestToken": "string",
"createdAt": number,
"createdRequests": [
  {
    "arn": "string",
    "computeType": "string",
    "dataSourceNames": [ "string" ],
    "lastUpdatedAt": number,
    "name": "string",
    "robotApplicationNames": [ "string" ],
    "simulationApplicationNames": [ "string" ],
    "status": "string"
  }
],
"failedRequests": [
  {
    "failedAt": number,
    "failureCode": "string",
    "failureReason": "string",
    "request": {
      "compute": {
        "computeType": "string",
        "gpuUnitLimit": number,
        "simulationUnitLimit": number
      },
      "dataSources": [
        {
          "destination": "string",
          "name": "string",
          "s3Bucket": "string",
          "s3Keys": [ "string" ],
          "type": "string"
        }
      ],
      "failureBehavior": "string",
      "iamRole": "string",
      "loggingConfig": {
        "recordAllRosTopics": boolean
      },
      "maxJobDurationInSeconds": number,
      "outputLocation": {
        "s3Bucket": "string",
        "s3Prefix": "string"
      }
    }
  }
]

```



```

    },
    "robotApplications": [
      {
        "application": "string",
        "applicationVersion": "string",
        "launchConfig": {
          "command": [ "string " ],
          "environmentVariables": {
            "string": "string"
          },
          "launchFile": "string",
          "packageName": "string",
          "portForwardingConfig": {
            "portMappings": [
              {
                "applicationPort": number,
                "enableOnPublicIp": boolean,
                "jobPort": number
              }
            ]
          },
          "streamUI": boolean
        },
        "tools": [
          {
            "command": "string",
            "exitBehavior": "string",
            "name": "string",
            "streamOutputToCloudWatch": boolean,
            "streamUI": boolean
          }
        ],
        "uploadConfigurations": [
          {
            "name": "string",
            "path": "string",
            "uploadBehavior": "string"
          }
        ],
        "useDefaultTools": boolean,
        "useDefaultUploadConfigurations": boolean
      }
    ],
    "simulationApplications": [

```

```

{
  "application": "string",
  "applicationVersion": "string",
  "launchConfig": {
    "command": [ "string" ],
    "environmentVariables": {
      "string" : "string"
    },
    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean,
  "worldConfigs": [
    {
      "world": "string"
    }
  ]
}

```

```

    }
  ],
  "tags": {
    "string" : "string"
  },
  "useDefaultApplications": boolean,
  "vpcConfig": {
    "assignPublicIp": boolean,
    "securityGroups": [ "string" ],
    "subnets": [ "string" ]
  }
}
],
"failureCode": "string",
"failureReason": "string",
"lastUpdatedAt": number,
"pendingRequests": [
  {
    "compute": {
      "computeType": "string",
      "gpuUnitLimit": number,
      "simulationUnitLimit": number
    },
    "dataSources": [
      {
        "destination": "string",
        "name": "string",
        "s3Bucket": "string",
        "s3Keys": [ "string" ],
        "type": "string"
      }
    ],
    "failureBehavior": "string",
    "iamRole": "string",
    "loggingConfig": {
      "recordAllRosTopics": boolean
    },
    "maxJobDurationInSeconds": number,
    "outputLocation": {
      "s3Bucket": "string",
      "s3Prefix": "string"
    },
    "robotApplications": [

```

```

{
  "application": "string",
  "applicationVersion": "string",
  "launchConfig": {
    "command": [ "string" ],
    "environmentVariables": {
      "string" : "string"
    },
    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean
}
],
"simulationApplications": [
  {
    "application": "string",

```

```

    "applicationVersion": "string",
    "launchConfig": {
      "command": [ "string" ],
      "environmentVariables": {
        "string" : "string"
      },
      "launchFile": "string",
      "packageName": "string",
      "portForwardingConfig": {
        "portMappings": [
          {
            "applicationPort": number,
            "enableOnPublicIp": boolean,
            "jobPort": number
          }
        ]
      },
      "streamUI": boolean
    },
    "tools": [
      {
        "command": "string",
        "exitBehavior": "string",
        "name": "string",
        "streamOutputToCloudWatch": boolean,
        "streamUI": boolean
      }
    ],
    "uploadConfigurations": [
      {
        "name": "string",
        "path": "string",
        "uploadBehavior": "string"
      }
    ],
    "useDefaultTools": boolean,
    "useDefaultUploadConfigurations": boolean,
    "worldConfigs": [
      {
        "world": "string"
      }
    ]
  }
],

```

```
    "tags": {
      "string" : "string"
    },
    "useDefaultApplications": boolean,
    "vpcConfig": {
      "assignPublicIp": boolean,
      "securityGroups": [ "string" ],
      "subnets": [ "string" ]
    }
  }
],
"status": "string",
"tags": {
  "string" : "string"
}
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the batch.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

### batchPolicy

The batch policy.

Type: [BatchPolicy](#) object

### clientRequestToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `[a-zA-Z0-9_\-=]*`

### **createdAt**

The time, in milliseconds since the epoch, when the simulation job batch was created.

Type: Timestamp

### **createdRequests**

A list of created simulation job summaries.

Type: Array of [SimulationJobSummary](#) objects

Array Members: Minimum number of 0 items. Maximum number of 100 items.

### **failedRequests**

A list of failed create simulation job requests. The request failed to be created into a simulation job. Failed requests do not have a simulation job ID.

Type: Array of [FailedCreateSimulationJobRequest](#) objects

### **failureCode**

The failure code of the simulation job batch.

Type: String

Valid Values: `InternalServerError`

### **failureReason**

The reason the simulation job batch failed.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: `.*`

### **lastUpdatedAt**

The time, in milliseconds since the epoch, when the simulation job batch was last updated.

Type: Timestamp

## pendingRequests

A list of pending simulation job requests. These requests have not yet been created into simulation jobs.

Type: Array of [SimulationJobRequest](#) objects

Array Members: Minimum number of 1 item. Maximum number of 1000 items.

## status

The status of the batch.

### Pending

The simulation job batch request is pending.

### InProgress

The simulation job batch is in progress.

### Failed

The simulation job batch failed. One or more simulation job requests could not be completed due to an internal failure (like `InternalServerError`). See `failureCode` and `failureReason` for more information.

### Completed

The simulation batch job completed. A batch is complete when (1) there are no pending simulation job requests in the batch and none of the failed simulation job requests are due to `InternalServerError` and (2) when all created simulation jobs have reached a terminal state (for example, `Completed` or `Failed`).

### Canceled

The simulation batch job was cancelled.

### Canceling

The simulation batch job is being cancelled.

### Completing

The simulation batch job is completing.

### TimingOut

The simulation job batch is timing out.



If a batch timing out, and there are pending requests that were failing due to an internal failure (like `InternalServerError`), the batch status will be `Failed`. If there are no such failing request, the batch status will be `TimedOut`.

### TimedOut

The simulation batch job timed out.

Type: String

Valid Values: `Pending` | `InProgress` | `Failed` | `Completed` | `Canceled` | `Canceling` | `Completing` | `TimingOut` | `TimedOut`

### tags

A map that contains tag keys and tag values that are attached to the simulation job batch.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9_.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9_.\-\/+=:]*`

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **InternalServerError**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

## **ResourceNotFoundException**

The specified resource does not exist.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DescribeWorld

Describes a world.

## Request Syntax

```
POST /describeWorld HTTP/1.1
Content-type: application/json

{
  "world": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### world

The Amazon Resource Name (arn) of the world you want to describe.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "createdAt": number,
}
```

```
"generationJob": "string",
"tags": {
  "string" : "string"
},
"template": "string",
"worldDescriptionBody": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### [arn](#)

The Amazon Resource Name (arn) of the world.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### [createdAt](#)

The time, in milliseconds since the epoch, when the world was created.

Type: Timestamp

### [generationJob](#)

The Amazon Resource Name (arn) of the world generation job that generated the world.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### [tags](#)

A map that contains tag keys and tag values that are attached to the world.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

## template

The world template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

## worldDescriptionBody

Returns the JSON formatted string that describes the contents of your world.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 262144.

Pattern: `[\S\s]+`

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **ResourceNotFoundException**

The specified resource does not exist.

HTTP Status Code: 400

### **ThrottlingException**

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

## DescribeWorldExportJob

Describes a world export job.

### Request Syntax

```
POST /describeWorldExportJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

### URI Request Parameters

The request does not use any URI parameters.

### Request Body

The request accepts the following data in JSON format.

#### job

The Amazon Resource Name (arn) of the world export job to describe.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

### Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "clientRequestToken": "string",
}
```

```
"createdAt": number,
"failureCode": "string",
"failureReason": "string",
"iamRole": "string",
"outputLocation": {
  "s3Bucket": "string",
  "s3Prefix": "string"
},
"status": "string",
"tags": {
  "string" : "string"
},
"worlds": [ "string" ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the world export job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### clientRequestToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: [a-zA-Z0-9\_\-=]\*

### createdAt

The time, in milliseconds since the epoch, when the world export job was created.



Type: Timestamp

### failureCode

The failure code of the world export job if it failed:

InternalServerError

Internal service error.

LimitExceeded

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

ResourceNotFound

The specified resource could not be found.

RequestThrottled

The request was throttled.

InvalidInput

An input parameter in the request is not valid.

Type: String

Valid Values: InternalServerError | LimitExceeded | ResourceNotFound | RequestThrottled | InvalidInput | AccessDenied

### failureReason

The reason why the world export job failed.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

### iamRole

The IAM role that the world export process uses to access the Amazon S3 bucket and put the export.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `arn:aws:iam::\w+:role/.*`

### outputLocation

The output location.

Type: [OutputLocation](#) object

### status

The status of the world export job.

Pending

The world export job request is pending.

Running

The world export job is running.

Completed

The world export job completed.

Failed

The world export job failed. See `failureCode` and `failureReason` for more information.

Canceled

The world export job was cancelled.

Canceling

The world export job is being cancelled.

Type: String

Valid Values: Pending | Running | Completed | Failed | Canceling | Canceled

### tags

A map that contains tag keys and tag values that are attached to the world export job.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

## worlds

A list of Amazon Resource Names (arns) that correspond to worlds to be exported.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DescribeWorldGenerationJob

Describes a world generation job.

## Request Syntax

```
POST /describeWorldGenerationJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### job

The Amazon Resource Name (arn) of the world generation job to describe.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "clientRequestToken": "string",
}
```

```

"createdAt": number,
"failureCode": "string",
"failureReason": "string",
"finishedWorldsSummary": {
  "failureSummary": {
    "failures": [
      {
        "failureCode": "string",
        "failureCount": number,
        "sampleFailureReason": "string"
      }
    ],
    "totalFailureCount": number
  },
  "finishedCount": number,
  "succeededWorlds": [ "string" ]
},
"status": "string",
"tags": {
  "string" : "string"
},
"template": "string",
"worldCount": {
  "floorplanCount": number,
  "interiorCountPerFloorplan": number
},
"worldTags": {
  "string" : "string"
}
}

```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the world generation job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

### clientRequestToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `[a-zA-Z0-9_\-=]*`

### createdAt

The time, in milliseconds since the epoch, when the world generation job was created.

Type: Timestamp

### failureCode

The failure code of the world generation job if it failed:

`InternalServerError`

Internal service error.

`LimitExceeded`

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

`ResourceNotFound`

The specified resource could not be found.

`RequestThrottled`

The request was throttled.

`InvalidInput`

An input parameter in the request is not valid.

Type: String

Valid Values: `InternalServerError` | `LimitExceeded` | `ResourceNotFound` | `RequestThrottled` | `InvalidInput` | `AllWorldGenerationFailed`

## failureReason

The reason why the world generation job failed.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

## finishedWorldsSummary

Summary information about finished worlds.

Type: [FinishedWorldsSummary](#) object

## status

The status of the world generation job:

Pending

The world generation job request is pending.

Running

The world generation job is running.

Completed

The world generation job completed.

Failed

The world generation job failed. See `failureCode` for more information.

PartialFailed

Some worlds did not generate.

Canceled

The world generation job was cancelled.

Canceling

The world generation job is being cancelled.

Type: String



Valid Values: Pending | Running | Completed | Failed | PartialFailed | Canceling | Canceled

### tags

A map that contains tag keys and tag values that are attached to the world generation job.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

### template

The Amazon Resource Name (arn) of the world template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

### worldCount

Information about the world count.

Type: [WorldCount](#) object

### worldTags

A map that contains tag keys and tag values that are attached to the generated worlds.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# DescribeWorldTemplate

Describes a world template.

## Request Syntax

```
POST /describeWorldTemplate HTTP/1.1
Content-type: application/json

{
  "template": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### template

The Amazon Resource Name (arn) of the world template you want to describe.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "clientRequestToken": "string",
}
```

```
"createdAt": number,  
"lastUpdatedAt": number,  
"name": "string",  
"tags": {  
  "string" : "string"  
},  
"version": "string"  
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the world template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### clientRequestToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: [a-zA-Z0-9\_\-=]\*

### createdAt

The time, in milliseconds since the epoch, when the world template was created.

Type: Timestamp

### lastUpdatedAt

The time, in milliseconds since the epoch, when the world template was last updated.

Type: Timestamp

## name

The name of the world template.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 255.

Pattern: .\*

## tags

A map that contains tag keys and tag values that are attached to the world template.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

## version

The version of the world template that you're using.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

## InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

## ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

## ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

## GetWorldTemplateBody

Gets the world template body.

### Request Syntax

```
POST /getWorldTemplateBody HTTP/1.1
Content-type: application/json

{
  "generationJob": "string",
  "template": "string"
}
```

### URI Request Parameters

The request does not use any URI parameters.

### Request Body

The request accepts the following data in JSON format.

#### generationJob

The Amazon Resource Name (arn) of the world generator job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: No

#### template

The Amazon Resource Name (arn) of the world template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*



Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "templateBody": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### templateBody

The world template body.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 262144.

Pattern: [S\s]+

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **ResourceNotFoundException**

The specified resource does not exist.

HTTP Status Code: 400

### **ThrottlingException**

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# ListDeploymentJobs

*This action has been deprecated.*

## Important

This API is no longer supported. For more information, see the May 2, 2022 update in the [Support policy](#) page.

Returns a list of deployment jobs for a fleet. You can optionally provide filters to retrieve specific deployment jobs.

## Request Syntax

```
POST /listDeploymentJobs HTTP/1.1
Content-type: application/json

{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### filters

Optional filters to limit results.

The filter names `status` and `fleetName` are supported. When filtering, you must use the complete value of the filtered item. You can use up to three filters, but they must be for the same named item. For example, if you are looking for items with the status `InProgress` or the status `Pending`.

Type: Array of [Filter](#) objects

Array Members: Fixed number of 1 item.

Required: No

### [maxResults](#)

When this parameter is used, `ListDeploymentJobs` only returns `maxResults` results in a single page along with a `nextToken` response element. The remaining results of the initial request can be seen by sending another `ListDeploymentJobs` request with the returned `nextToken` value. This value can be between 1 and 200. If this parameter is not used, then `ListDeploymentJobs` returns up to 200 results and a `nextToken` value if applicable.

Type: Integer

Required: No

### [nextToken](#)

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListDeploymentJobs` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json
```

```

{
  "deploymentJobs": [
    {
      "arn": "string",
      "createdAt": number,
      "deploymentApplicationConfigs": [
        {
          "application": "string",
          "applicationVersion": "string",
          "launchConfig": {
            "environmentVariables": {
              "string" : "string"
            },
            "launchFile": "string",
            "packageName": "string",
            "postLaunchFile": "string",
            "preLaunchFile": "string"
          }
        }
      ],
      "deploymentConfig": {
        "concurrentDeploymentPercentage": number,
        "downloadConditionFile": {
          "bucket": "string",
          "etag": "string",
          "key": "string"
        },
        "failureThresholdPercentage": number,
        "robotDeploymentTimeoutInSeconds": number
      },
      "failureCode": "string",
      "failureReason": "string",
      "fleet": "string",
      "status": "string"
    }
  ],
  "nextToken": "string"
}

```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### deploymentJobs

A list of deployment jobs that meet the criteria of the request.

Type: Array of [DeploymentJob](#) objects

Array Members: Minimum number of 0 items. Maximum number of 200 items.

### nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListDeploymentJobs` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **ResourceNotFoundException**

The specified resource does not exist.

HTTP Status Code: 400

## ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# ListFleets

*This action has been deprecated.*

## Important

This API is no longer supported. For more information, see the May 2, 2022 update in the [Support policy](#) page.

Returns a list of fleets. You can optionally provide filters to retrieve specific fleets.

## Request Syntax

```
POST /listFleets HTTP/1.1
Content-type: application/json

{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### filters

Optional filters to limit results.

The filter name `name` is supported. When filtering, you must use the complete value of the filtered item. You can use up to three filters.



Type: Array of [Filter](#) objects

Array Members: Fixed number of 1 item.

Required: No

### [maxResults](#)

When this parameter is used, `ListFleets` only returns `maxResults` results in a single page along with a `nextToken` response element. The remaining results of the initial request can be seen by sending another `ListFleets` request with the returned `nextToken` value. This value can be between 1 and 200. If this parameter is not used, then `ListFleets` returns up to 200 results and a `nextToken` value if applicable.

Type: Integer

Required: No

### [nextToken](#)

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListFleets` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

#### Note

This token should be treated as an opaque identifier that is only used to retrieve the next items in a list and not for other programmatic purposes.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

Required: No

## Response Syntax

```
HTTP/1.1 200
```

```
Content-type: application/json

{
  "fleetDetails": [
    {
      "arn": "string",
      "createdAt": number,
      "lastDeploymentJob": "string",
      "lastDeploymentStatus": "string",
      "lastDeploymentTime": number,
      "name": "string"
    }
  ],
  "nextToken": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### fleetDetails

A list of fleet details meeting the request criteria.

Type: Array of [Fleet](#) objects

Array Members: Minimum number of 0 items. Maximum number of 200 items.

### nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListFleets` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

## ListRobotApplications

Returns a list of robot application. You can optionally provide filters to retrieve specific robot applications.

### Request Syntax

```
POST /listRobotApplications HTTP/1.1
Content-type: application/json
```

```
{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string",
  "versionQualifier": "string"
}
```

### URI Request Parameters

The request does not use any URI parameters.

### Request Body

The request accepts the following data in JSON format.

#### filters

Optional filters to limit results.

The filter name `name` is supported. When filtering, you must use the complete value of the filtered item. You can use up to three filters.

Type: Array of [Filter](#) objects

Array Members: Fixed number of 1 item.

Required: No

## maxResults

When this parameter is used, `ListRobotApplications` only returns `maxResults` results in a single page along with a `nextToken` response element. The remaining results of the initial request can be seen by sending another `ListRobotApplications` request with the returned `nextToken` value. This value can be between 1 and 100. If this parameter is not used, then `ListRobotApplications` returns up to 100 results and a `nextToken` value if applicable.

Type: Integer

Required: No

## nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListRobotApplications` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

Required: No

## versionQualifier

The version qualifier of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ALL

Required: No

## Response Syntax

```
HTTP/1.1 200
```

```
Content-type: application/json

{
  "nextToken": "string",
  "robotApplicationSummaries": [
    {
      "arn": "string",
      "lastUpdatedAt": number,
      "name": "string",
      "robotSoftwareSuite": {
        "name": "string",
        "version": "string"
      },
      "version": "string"
    }
  ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListRobotApplications` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

### robotApplicationSummaries

A list of robot application summaries that meet the criteria of the request.

Type: Array of [RobotApplicationSummary](#) objects

Array Members: Minimum number of 0 items. Maximum number of 100 items.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)





# ListRobots

*This action has been deprecated.*

## Important

This API is no longer supported. For more information, see the May 2, 2022 update in the [Support policy](#) page.

Returns a list of robots. You can optionally provide filters to retrieve specific robots.

## Request Syntax

```
POST /listRobots HTTP/1.1
Content-type: application/json

{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### filters

Optional filters to limit results.

The filter names `status` and `fleetName` are supported. When filtering, you must use the complete value of the filtered item. You can use up to three filters, but they must be for the

same named item. For example, if you are looking for items with the status `Registered` or the status `Available`.

Type: Array of [Filter](#) objects

Array Members: Fixed number of 1 item.

Required: No

### [maxResults](#)

When this parameter is used, `ListRobots` only returns `maxResults` results in a single page along with a `nextToken` response element. The remaining results of the initial request can be seen by sending another `ListRobots` request with the returned `nextToken` value. This value can be between 1 and 200. If this parameter is not used, then `ListRobots` returns up to 200 results and a `nextToken` value if applicable.

Type: Integer

Required: No

### [nextToken](#)

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListRobots` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json
```

```
{
  "nextToken": "string",
  "robots": [
    {
      "architecture": "string",
      "arn": "string",
      "createdAt": number,
      "fleetArn": "string",
      "greenGrassGroupId": "string",
      "lastDeploymentJob": "string",
      "lastDeploymentTime": number,
      "name": "string",
      "status": "string"
    }
  ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListRobots` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

### robots

A list of robots that meet the criteria of the request.

Type: Array of [Robot](#) objects

Array Members: Minimum number of 0 items. Maximum number of 1000 items.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)

- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

## ListSimulationApplications

Returns a list of simulation applications. You can optionally provide filters to retrieve specific simulation applications.

### Request Syntax

```
POST /listSimulationApplications HTTP/1.1
Content-type: application/json
```

```
{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string",
  "versionQualifier": "string"
}
```

### URI Request Parameters

The request does not use any URI parameters.

### Request Body

The request accepts the following data in JSON format.

#### filters

Optional list of filters to limit results.

The filter name `name` is supported. When filtering, you must use the complete value of the filtered item. You can use up to three filters.

Type: Array of [Filter](#) objects

Array Members: Fixed number of 1 item.

Required: No

## maxResults

When this parameter is used, `ListSimulationApplications` only returns `maxResults` results in a single page along with a `nextToken` response element. The remaining results of the initial request can be seen by sending another `ListSimulationApplications` request with the returned `nextToken` value. This value can be between 1 and 100. If this parameter is not used, then `ListSimulationApplications` returns up to 100 results and a `nextToken` value if applicable.

Type: Integer

Required: No

## nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListSimulationApplications` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

Required: No

## versionQualifier

The version qualifier of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: ALL

Required: No

## Response Syntax

```
HTTP/1.1 200
```



```
Content-type: application/json

{
  "nextToken": "string",
  "simulationApplicationSummaries": [
    {
      "arn": "string",
      "lastUpdatedAt": number,
      "name": "string",
      "robotSoftwareSuite": {
        "name": "string",
        "version": "string"
      },
      "simulationSoftwareSuite": {
        "name": "string",
        "version": "string"
      },
      "version": "string"
    }
  ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListSimulationApplications` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

## [simulationApplicationSummaries](#)

A list of simulation application summaries that meet the criteria of the request.

Type: Array of [SimulationApplicationSummary](#) objects

Array Members: Minimum number of 0 items. Maximum number of 100 items.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)

- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

## ListSimulationJobBatches

Returns a list simulation job batches. You can optionally provide filters to retrieve specific simulation batch jobs.

### Request Syntax

```
POST /listSimulationJobBatches HTTP/1.1
Content-type: application/json
```

```
{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

### URI Request Parameters

The request does not use any URI parameters.

### Request Body

The request accepts the following data in JSON format.

#### filters

Optional filters to limit results.

Type: Array of [Filter](#) objects

Array Members: Fixed number of 1 item.

Required: No

#### maxResults

When this parameter is used, `ListSimulationJobBatches` only returns `maxResults` results in a single page along with a `nextToken` response element. The remaining results of the initial

request can be seen by sending another `ListSimulationJobBatches` request with the returned `nextToken` value.

Type: Integer

Required: No

### nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListSimulationJobBatches` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "simulationJobBatchSummaries": [
    {
      "arn": "string",
      "createdAt": number,
      "createdRequestCount": number,
      "failedRequestCount": number,
      "lastUpdatedAt": number,
      "pendingRequestCount": number,
      "status": "string"
    }
  ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListSimulationJobBatches` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

### simulationJobBatchSummaries

A list of simulation job batch summaries.

Type: Array of [SimulationJobBatchSummary](#) objects

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

## ListSimulationJobs

Returns a list of simulation jobs. You can optionally provide filters to retrieve specific simulation jobs.

### Request Syntax

```
POST /listSimulationJobs HTTP/1.1
Content-type: application/json
```

```
{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

### URI Request Parameters

The request does not use any URI parameters.

### Request Body

The request accepts the following data in JSON format.

#### filters

Optional filters to limit results.

The filter names `status` and `simulationApplicationName` and `robotApplicationName` are supported. When filtering, you must use the complete value of the filtered item. You can use up to three filters, but they must be for the same named item. For example, if you are looking for items with the status `Preparing` or the status `Running`.

Type: Array of [Filter](#) objects

Array Members: Fixed number of 1 item.



Required: No

### maxResults

When this parameter is used, `ListSimulationJobs` only returns `maxResults` results in a single page along with a `nextToken` response element. The remaining results of the initial request can be seen by sending another `ListSimulationJobs` request with the returned `nextToken` value. This value can be between 1 and 1000. If this parameter is not used, then `ListSimulationJobs` returns up to 1000 results and a `nextToken` value if applicable.

Type: Integer

Required: No

### nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListSimulationJobs` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "simulationJobSummaries": [
    {
      "arn": "string",
      "computeType": "string",
      "dataSourceNames": [ "string" ],
      "lastUpdatedAt": number,
```

```
    "name": "string",
    "robotApplicationNames": [ "string" ],
    "simulationApplicationNames": [ "string" ],
    "status": "string"
  }
]
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListSimulationJobs` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

### simulationJobSummaries

A list of simulation job summaries that meet the criteria of the request.

Type: Array of [SimulationJobSummary](#) objects

Array Members: Minimum number of 0 items. Maximum number of 100 items.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **ThrottlingException**

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

## ListTagsForResource

Lists all tags on a AWS RoboMaker resource.

### Request Syntax

```
GET /tags/resourceArn HTTP/1.1
```

### URI Request Parameters

The request uses the following URI parameters.

#### resourceArn

The AWS RoboMaker Amazon Resource Name (ARN) with tags to be listed.

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: Yes

### Request Body

The request does not have a request body.

### Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "tags": {
    "string" : "string"
  }
}
```

### Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

## tags

The list of all tags added to the specified resource.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# ListWorldExportJobs

Lists world export jobs.

## Request Syntax

```
POST /listWorldExportJobs HTTP/1.1
Content-type: application/json

{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### filters

Optional filters to limit results. You can use `generationJobId` and `templateId`.

Type: Array of [Filter](#) objects

Array Members: Fixed number of 1 item.

Required: No

### maxResults

When this parameter is used, `ListWorldExportJobs` only returns `maxResults` results in a single page along with a `nextToken` response element. The remaining results of the initial request can be seen by sending another `ListWorldExportJobs` request with the returned

nextToken value. This value can be between 1 and 100. If this parameter is not used, then ListWorldExportJobs returns up to 100 results and a nextToken value if applicable.

Type: Integer

Required: No

### nextToken

If the previous paginated request did not return all of the remaining results, the response object's nextToken parameter value is set to a token. To retrieve the next set of results, call ListWorldExportJobs again and assign that token to the request object's nextToken parameter. If there are no remaining results, the previous response object's NextToken parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: [a-zA-Z0-9\_.\-\/+=]\*

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "worldExportJobSummaries": [
    {
      "arn": "string",
      "createdAt": number,
      "outputLocation": {
        "s3Bucket": "string",
        "s3Prefix": "string"
      },
      "status": "string",
      "worlds": [ "string" ]
    }
  ]
}
```



```
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListWorldExportJobsRequest` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

### worldExportJobSummaries

Summary information for world export jobs.

Type: Array of [WorldExportJobSummary](#) objects

Array Members: Minimum number of 0 items. Maximum number of 100 items.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

## ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

## ListWorldGenerationJobs

Lists world generator jobs.

### Request Syntax

```
POST /listWorldGenerationJobs HTTP/1.1
Content-type: application/json
```

```
{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

### URI Request Parameters

The request does not use any URI parameters.

### Request Body

The request accepts the following data in JSON format.

#### [filters](#)

Optional filters to limit results. You can use status and templateId.

Type: Array of [Filter](#) objects

Array Members: Fixed number of 1 item.

Required: No

#### [maxResults](#)

When this parameter is used, ListWorldGeneratorJobs only returns maxResults results in a single page along with a nextToken response element. The remaining results of the initial

request can be seen by sending another `ListWorldGeneratorJobs` request with the returned `nextToken` value. This value can be between 1 and 100. If this parameter is not used, then `ListWorldGeneratorJobs` returns up to 100 results and a `nextToken` value if applicable.

Type: Integer

Required: No

### nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListWorldGenerationJobsRequest` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "worldGenerationJobSummaries": [
    {
      "arn": "string",
      "createdAt": number,
      "failedWorldCount": number,
      "status": "string",
      "succeededWorldCount": number,
      "template": "string",
      "worldCount": {
        "floorplanCount": number,
        "interiorCountPerFloorplan": number
      }
    }
  ]
}
```

```
    }  
  }  
]  
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListWorldGeneratorJobsRequest` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

### worldGenerationJobSummaries

Summary information for world generator jobs.

Type: Array of [WorldGenerationJobSummary](#) objects

Array Members: Minimum number of 0 items. Maximum number of 100 items.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

## InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

## ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# ListWorlds

Lists worlds.

## Request Syntax

```
POST /listWorlds HTTP/1.1
Content-type: application/json

{
  "filters": [
    {
      "name": "string",
      "values": [ "string" ]
    }
  ],
  "maxResults": number,
  "nextToken": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### filters

Optional filters to limit results. You can use status.

Type: Array of [Filter](#) objects

Array Members: Fixed number of 1 item.

Required: No

### maxResults

When this parameter is used, ListWorlds only returns maxResults results in a single page along with a nextToken response element. The remaining results of the initial request can be seen by sending another ListWorlds request with the returned nextToken value. This value

can be between 1 and 100. If this parameter is not used, then `ListWorlds` returns up to 100 results and a `nextToken` value if applicable.

Type: Integer

Required: No

### nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListWorlds` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "worldSummaries": [
    {
      "arn": "string",
      "createdAt": number,
      "generationJob": "string",
      "template": "string"
    }
  ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.



The following data is returned in JSON format by the service.

### nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListWorlds` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

### worldSummaries

Summary information for worlds.

Type: Array of [WorldSummary](#) objects

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **ThrottlingException**

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# ListWorldTemplates

Lists world templates.

## Request Syntax

```
POST /listWorldTemplates HTTP/1.1
Content-type: application/json
```

```
{
  "maxResults": number,
  "nextToken": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### maxResults

When this parameter is used, `ListWorldTemplates` only returns `maxResults` results in a single page along with a `nextToken` response element. The remaining results of the initial request can be seen by sending another `ListWorldTemplates` request with the returned `nextToken` value. This value can be between 1 and 100. If this parameter is not used, then `ListWorldTemplates` returns up to 100 results and a `nextToken` value if applicable.

Type: Integer

Required: No

### nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListWorldTemplates` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "nextToken": "string",
  "templateSummaries": [
    {
      "arn": "string",
      "createdAt": number,
      "lastUpdatedAt": number,
      "name": "string",
      "version": "string"
    }
  ]
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### nextToken

If the previous paginated request did not return all of the remaining results, the response object's `nextToken` parameter value is set to a token. To retrieve the next set of results, call `ListWorldTemplates` again and assign that token to the request object's `nextToken` parameter. If there are no remaining results, the previous response object's `NextToken` parameter is set to null.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: `[a-zA-Z0-9_.\-\/+=]*`

## [templateSummaries](#)

Summary information for templates.

Type: Array of [TemplateSummary](#) objects

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# RegisterRobot

*This action has been deprecated.*

Registers a robot with a fleet.

## Important

This API is no longer supported and will throw an error if used. For more information, see the January 31, 2022 update in the [Support policy](#) page.

## Request Syntax

```
POST /registerRobot HTTP/1.1
Content-type: application/json

{
  "fleet": "string",
  "robot": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### fleet

The Amazon Resource Name (ARN) of the fleet.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

## [robot](#)

The Amazon Resource Name (ARN) of the robot.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "fleet": "string",
  "robot": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### [fleet](#)

The Amazon Resource Name (ARN) of the fleet that the robot will join.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

### [robot](#)

Information about the robot registration.

Type: String



Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: a1n:.\*

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### LimitExceededException

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

## RestartSimulationJob

Restarts a running simulation job.

### Request Syntax

```
POST /restartSimulationJob HTTP/1.1
Content-type: application/json

{
  "job": "string"
}
```

### URI Request Parameters

The request does not use any URI parameters.

### Request Body

The request accepts the following data in JSON format.

#### job

The Amazon Resource Name (ARN) of the simulation job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

### Response Syntax

```
HTTP/1.1 200
```

### Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### LimitExceededException

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)

- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

## StartSimulationJobBatch

Starts a new simulation job batch. The batch is defined using one or more SimulationJobRequest objects.

### Request Syntax

```
POST /startSimulationJobBatch HTTP/1.1
```

```
Content-type: application/json
```

```
{
  "batchPolicy": {
    "maxConcurrency": number,
    "timeoutInSeconds": number
  },
  "clientRequestToken": "string",
  "createSimulationJobRequests": [
    {
      "compute": {
        "computeType": "string",
        "gpuUnitLimit": number,
        "simulationUnitLimit": number
      },
      "dataSources": [
        {
          "destination": "string",
          "name": "string",
          "s3Bucket": "string",
          "s3Keys": [ "string " ],
          "type": "string"
        }
      ],
      "failureBehavior": "string",
      "iamRole": "string",
      "loggingConfig": {
        "recordAllRosTopics": boolean
      },
      "maxJobDurationInSeconds": number,
      "outputLocation": {
        "s3Bucket": "string",
        "s3Prefix": "string"
      },
      "robotApplications": [
```

```

{
  "application": "string",
  "applicationVersion": "string",
  "launchConfig": {
    "command": [ "string" ],
    "environmentVariables": {
      "string" : "string"
    },
    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean
}
],
"simulationApplications": [
  {
    "application": "string",

```

```

    "applicationVersion": "string",
    "launchConfig": {
      "command": [ "string" ],
      "environmentVariables": {
        "string" : "string"
      },
      "launchFile": "string",
      "packageName": "string",
      "portForwardingConfig": {
        "portMappings": [
          {
            "applicationPort": number,
            "enableOnPublicIp": boolean,
            "jobPort": number
          }
        ]
      },
      "streamUI": boolean
    },
    "tools": [
      {
        "command": "string",
        "exitBehavior": "string",
        "name": "string",
        "streamOutputToCloudWatch": boolean,
        "streamUI": boolean
      }
    ],
    "uploadConfigurations": [
      {
        "name": "string",
        "path": "string",
        "uploadBehavior": "string"
      }
    ],
    "useDefaultTools": boolean,
    "useDefaultUploadConfigurations": boolean,
    "worldConfigs": [
      {
        "world": "string"
      }
    ]
  }
],

```



```
    "tags": {
      "string" : "string"
    },
    "useDefaultApplications": boolean,
    "vpcConfig": {
      "assignPublicIp": boolean,
      "securityGroups": [ "string" ],
      "subnets": [ "string" ]
    }
  ],
  "tags": {
    "string" : "string"
  }
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### batchPolicy

The batch policy.

Type: [BatchPolicy](#) object

Required: No

### clientRequestToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: [a-zA-Z0-9\_\-=]\*

Required: No

## createSimulationJobRequests

A list of simulation job requests to create in the batch.

Type: Array of [SimulationJobRequest](#) objects

Array Members: Minimum number of 1 item. Maximum number of 1000 items.

Required: Yes

## tags

A map that contains tag keys and tag values that are attached to the deployment job batch.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "batchPolicy": {
    "maxConcurrency": number,
    "timeoutInSeconds": number
  },
  "clientRequestToken": "string",
  "createdAt": number,
  "createdRequests": [
    {
      "arn": "string",
      "computeType": "string",
```

```

    "dataSourceNames": [ "string" ],
    "lastUpdatedAt": number,
    "name": "string",
    "robotApplicationNames": [ "string" ],
    "simulationApplicationNames": [ "string" ],
    "status": "string"
  }
],
"failedRequests": [
  {
    "failedAt": number,
    "failureCode": "string",
    "failureReason": "string",
    "request": {
      "compute": {
        "computeType": "string",
        "gpuUnitLimit": number,
        "simulationUnitLimit": number
      },
      "dataSources": [
        {
          "destination": "string",
          "name": "string",
          "s3Bucket": "string",
          "s3Keys": [ "string" ],
          "type": "string"
        }
      ],
      "failureBehavior": "string",
      "iamRole": "string",
      "loggingConfig": {
        "recordAllRosTopics": boolean
      },
      "maxJobDurationInSeconds": number,
      "outputLocation": {
        "s3Bucket": "string",
        "s3Prefix": "string"
      },
      "robotApplications": [
        {
          "application": "string",
          "applicationVersion": "string",
          "launchConfig": {
            "command": [ "string" ],

```

```

    "environmentVariables": {
      "string" : "string"
    },
    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean
}
],
"simulationApplications": [
  {
    "application": "string",
    "applicationVersion": "string",
    "launchConfig": {
      "command": [ "string" ],
      "environmentVariables": {
        "string" : "string"
      }
    }
  }
]

```

```

    },
    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean,
  "worldConfigs": [
    {
      "world": "string"
    }
  ]
}
],
"tags": {
  "string" : "string"
},
"useDefaultApplications": boolean,
"vpcConfig": {

```

```

        "assignPublicIp": boolean,
        "securityGroups": [ "string" ],
        "subnets": [ "string" ]
    }
}
],
"failureCode": "string",
"failureReason": "string",
"pendingRequests": [
    {
        "compute": {
            "computeType": "string",
            "gpuUnitLimit": number,
            "simulationUnitLimit": number
        },
        "dataSources": [
            {
                "destination": "string",
                "name": "string",
                "s3Bucket": "string",
                "s3Keys": [ "string" ],
                "type": "string"
            }
        ],
        "failureBehavior": "string",
        "iamRole": "string",
        "loggingConfig": {
            "recordAllRosTopics": boolean
        },
        "maxJobDurationInSeconds": number,
        "outputLocation": {
            "s3Bucket": "string",
            "s3Prefix": "string"
        },
        "robotApplications": [
            {
                "application": "string",
                "applicationVersion": "string",
                "launchConfig": {
                    "command": [ "string" ],
                    "environmentVariables": {
                        "string" : "string"
                    }
                }
            }
        ],

```

```

    "launchFile": "string",
    "packageName": "string",
    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean
}
],
"simulationApplications": [
  {
    "application": "string",
    "applicationVersion": "string",
    "launchConfig": {
      "command": [ "string ],
      "environmentVariables": {
        "string": "string"
      },
    },
    "launchFile": "string",
    "packageName": "string",

```

```

    "portForwardingConfig": {
      "portMappings": [
        {
          "applicationPort": number,
          "enableOnPublicIp": boolean,
          "jobPort": number
        }
      ]
    },
    "streamUI": boolean
  },
  "tools": [
    {
      "command": "string",
      "exitBehavior": "string",
      "name": "string",
      "streamOutputToCloudWatch": boolean,
      "streamUI": boolean
    }
  ],
  "uploadConfigurations": [
    {
      "name": "string",
      "path": "string",
      "uploadBehavior": "string"
    }
  ],
  "useDefaultTools": boolean,
  "useDefaultUploadConfigurations": boolean,
  "worldConfigs": [
    {
      "world": "string"
    }
  ]
}
],
"tags": {
  "string" : "string"
},
"useDefaultApplications": boolean,
"vpcConfig": {
  "assignPublicIp": boolean,
  "securityGroups": [ "string" ],
  "subnets": [ "string" ]
}

```



```
    }  
  }  
],  
"status": "string",  
"tags": {  
  "string" : "string"  
}  
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (arn) of the batch.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### batchPolicy

The batch policy.

Type: [BatchPolicy](#) object

### clientRequestToken

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: [a-zA-Z0-9\_\-=]\*

### createdAt

The time, in milliseconds since the epoch, when the simulation job batch was created.

Type: Timestamp

## createdRequests

A list of created simulation job request summaries.

Type: Array of [SimulationJobSummary](#) objects

Array Members: Minimum number of 0 items. Maximum number of 100 items.

## failedRequests

A list of failed simulation job requests. The request failed to be created into a simulation job. Failed requests do not have a simulation job ID.

Type: Array of [FailedCreateSimulationJobRequest](#) objects

## failureCode

The failure code if the simulation job batch failed.

Type: String

Valid Values: InternalServiceError

## failureReason

The reason the simulation job batch failed.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

## pendingRequests

A list of pending simulation job requests. These requests have not yet been created into simulation jobs.

Type: Array of [SimulationJobRequest](#) objects

Array Members: Minimum number of 1 item. Maximum number of 1000 items.

## status

The status of the simulation job batch.

Pending

The simulation job batch request is pending.

## InProgress

The simulation job batch is in progress.

## Failed

The simulation job batch failed. One or more simulation job requests could not be completed due to an internal failure (like `InternalServerError`). See `failureCode` and `failureReason` for more information.

## Completed

The simulation batch job completed. A batch is complete when (1) there are no pending simulation job requests in the batch and none of the failed simulation job requests are due to `InternalServerError` and (2) when all created simulation jobs have reached a terminal state (for example, `Completed` or `Failed`).

## Canceled

The simulation batch job was cancelled.

## Canceling

The simulation batch job is being cancelled.

## Completing

The simulation batch job is completing.

## TimingOut

The simulation job batch is timing out.

If a batch timing out, and there are pending requests that were failing due to an internal failure (like `InternalServerError`), the batch status will be `Failed`. If there are no such failing request, the batch status will be `TimedOut`.

## TimedOut

The simulation batch job timed out.

## Type: String

Valid Values: `Pending` | `InProgress` | `Failed` | `Completed` | `Canceled` | `Canceling` | `Completing` | `TimingOut` | `TimedOut`

## tags

A map that contains tag keys and tag values that are attached to the deployment job batch.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **IdempotentParameterMismatchException**

The request uses the same client token as a previous, but non-identical request. Do not reuse a client token with different requests, unless the requests are identical.

HTTP Status Code: 400

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **LimitExceededException**

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

HTTP Status Code: 400

## ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# SyncDeploymentJob

*This action has been deprecated.*

## Important

This API is no longer supported. For more information, see the May 2, 2022 update in the [Support policy](#) page.

Synchronizes robots in a fleet to the latest deployment. This is helpful if robots were added after a deployment.

## Request Syntax

```
POST /syncDeploymentJob HTTP/1.1
Content-type: application/json

{
  "clientRequestToken": "string",
  "fleet": "string"
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### [clientRequestToken](#)

Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `[a-zA-Z0-9_\-=]*`

Required: Yes

## fleet

The target fleet for the synchronization.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: Yes

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "createdAt": number,
  "deploymentApplicationConfigs": [
    {
      "application": "string",
      "applicationVersion": "string",
      "launchConfig": {
        "environmentVariables": {
          "string": "string"
        },
        "launchFile": "string",
        "packageName": "string",
        "postLaunchFile": "string",
        "preLaunchFile": "string"
      }
    }
  ],
  "deploymentConfig": {
    "concurrentDeploymentPercentage": number,
    "downloadConditionFile": {
      "bucket": "string",
      "etag": "string",
      "key": "string"
    }
  }
}
```

```
    },  
    "failureThresholdPercentage": number,  
    "robotDeploymentTimeoutInSeconds": number  
  },  
  "failureCode": "string",  
  "failureReason": "string",  
  "fleet": "string",  
  "status": "string"  
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the synchronization request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

### createdAt

The time, in milliseconds since the epoch, when the fleet was created.

Type: Timestamp

### deploymentApplicationConfigs

Information about the deployment application configurations.

Type: Array of [DeploymentApplicationConfig](#) objects

Array Members: Fixed number of 1 item.

### deploymentConfig

Information about the deployment configuration.

Type: [DeploymentConfig](#) object



## **failureCode**

The failure code if the job fails:

**InternalServiceError**

Internal service error.

**RobotApplicationCrash**

Robot application exited abnormally.

**SimulationApplicationCrash**

Simulation application exited abnormally.

**BadPermissionsRobotApplication**

Robot application bundle could not be downloaded.

**BadPermissionsSimulationApplication**

Simulation application bundle could not be downloaded.

**BadPermissionsS3Output**

Unable to publish outputs to customer-provided S3 bucket.

**BadPermissionsCloudwatchLogs**

Unable to publish logs to customer-provided CloudWatch Logs resource.

**SubnetIpLimitExceeded**

Subnet IP limit exceeded.

**ENILimitExceeded**

ENI limit exceeded.

**BadPermissionsUserCredentials**

Unable to use the Role provided.

**InvalidBundleRobotApplication**

Robot bundle cannot be extracted (invalid format, bundling error, or other issue).

**InvalidBundleSimulationApplication**

Simulation bundle cannot be extracted (invalid format, bundling error, or other issue).

## RobotApplicationVersionMismatchedEtag

Etag for RobotApplication does not match value during version creation.

## SimulationApplicationVersionMismatchedEtag

Etag for SimulationApplication does not match value during version creation.

Type: String

Valid Values: ResourceNotFound | EnvironmentSetupError | EtagMismatch | FailureThresholdBreach | RobotDeploymentAborted | RobotDeploymentNoResponse | RobotAgentConnectionTimeout | GreengrassDeploymentFailed | InvalidGreengrassGroup | MissingRobotArchitecture | MissingRobotApplicationArchitecture | MissingRobotDeploymentResource | GreengrassGroupVersionDoesNotExist | LambdaDeleted | ExtractingBundleFailure | PreLaunchFileFailure | PostLaunchFileFailure | BadPermissionError | DownloadConditionFailed | BadLambdaAssociated | InternalServerError | RobotApplicationDoesNotExist | DeploymentFleetDoesNotExist | FleetDeploymentTimeout

## failureReason

The failure reason if the job fails.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

## fleet

The Amazon Resource Name (ARN) of the fleet.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

## status

The status of the synchronization job.

Type: String

Valid Values: Pending | Preparing | InProgress | Failed | Succeeded | Canceled

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **ConcurrentDeploymentException**

The failure percentage threshold percentage was met.

HTTP Status Code: 400

### **IdempotentParameterMismatchException**

The request uses the same client token as a previous, but non-identical request. Do not reuse a client token with different requests, unless the requests are identical.

HTTP Status Code: 400

### **InternalServerException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **LimitExceededException**

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

HTTP Status Code: 400

### **ResourceNotFoundException**

The specified resource does not exist.

HTTP Status Code: 400

## ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

## TagResource

Adds or edits tags for a AWS RoboMaker resource.

Each tag consists of a tag key and a tag value. Tag keys and tag values are both required, but tag values can be empty strings.

For information about the rules that apply to tag keys and tag values, see [User-Defined Tag Restrictions](#) in the *AWS Billing and Cost Management User Guide*.

### Request Syntax

```
POST /tags/resourceArn HTTP/1.1
Content-type: application/json

{
  "tags": {
    "string" : "string"
  }
}
```

### URI Request Parameters

The request uses the following URI parameters.

#### [resourceArn](#)

The Amazon Resource Name (ARN) of the AWS RoboMaker resource you are tagging.

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: Yes

### Request Body

The request accepts the following data in JSON format.

#### [tags](#)

A map that contains tag keys and tag values that are attached to the resource.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Required: Yes

## Response Syntax

```
HTTP/1.1 200
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

## ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# UntagResource

Removes the specified tags from the specified AWS RoboMaker resource.

To remove a tag, specify the tag key. To change the tag value of an existing tag key, use [TagResource](#).

## Request Syntax

```
DELETE /tags/resourceArn?tagKeys=tagKeys HTTP/1.1
```

## URI Request Parameters

The request uses the following URI parameters.

### [resourceArn](#)

The Amazon Resource Name (ARN) of the AWS RoboMaker resource you are removing tags.

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: Yes

### [tagKeys](#)

A map that contains tag keys and tag values that will be unattached from the resource.

Length Constraints: Minimum length of 1. Maximum length of 128.

Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Required: Yes

## Request Body

The request does not have a request body.

## Response Syntax

```
HTTP/1.1 200
```



## Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# UpdateRobotApplication

Updates a robot application.

## Request Syntax

```
POST /updateRobotApplication HTTP/1.1
```

```
Content-type: application/json
```

```
{
  "application": "string",
  "currentRevisionId": "string",
  "environment": {
    "uri": "string"
  },
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
      "architecture": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ]
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### application

The application information for the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: Yes

### currentRevisionId

The revision id for the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 40.

Pattern: `[a-zA-Z0-9_.\-]*`

Required: No

### environment

The object that contains the Docker image URI for your robot application.

Type: [Environment](#) object

Required: No

### robotSoftwareSuite

The robot software suite used by the robot application.

Type: [RobotSoftwareSuite](#) object

Required: Yes

### sources

The sources of the robot application.

Type: Array of [SourceConfig](#) objects

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json
```

```
{
  "arn": "string",
  "environment": {
    "uri": "string"
  },
  "lastUpdatedAt": number,
  "name": "string",
  "revisionId": "string",
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
      "architecture": "string",
      "etag": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ],
  "version": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the updated robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### environment

The object that contains the Docker image URI for your robot application.

Type: [Environment](#) object

## lastUpdatedAt

The time, in milliseconds since the epoch, when the robot application was last updated.

Type: Timestamp

## name

The name of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

## revisionId

The revision id of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 40.

Pattern: [a-zA-Z0-9\_.\-]\*

## robotSoftwareSuite

The robot software suite used by the robot application.

Type: [RobotSoftwareSuite](#) object

## sources

The sources of the robot application.

Type: Array of [Source](#) objects

## version

The version of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: (\\$LATEST)|[0-9]\*

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### LimitExceededException

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

HTTP Status Code: 400

### ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

### ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)

- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)



# UpdateSimulationApplication

Updates a simulation application.

## Request Syntax

```
POST /updateSimulationApplication HTTP/1.1
```

```
Content-type: application/json
```

```
{
  "application": "string",
  "currentRevisionId": "string",
  "environment": {
    "uri": "string"
  },
  "renderingEngine": {
    "name": "string",
    "version": "string"
  },
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "simulationSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
      "architecture": "string",
      "s3Bucket": "string",
      "s3Key": "string"
    }
  ]
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

## application

The application information for the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: Yes

## currentRevisionId

The revision id for the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 40.

Pattern: `[a-zA-Z0-9_.\-]*`

Required: No

## environment

The object that contains the Docker image URI for your simulation application.

Type: [Environment](#) object

Required: No

## renderingEngine

The rendering engine for the simulation application.

Type: [RenderingEngine](#) object

Required: No

## robotSoftwareSuite

Information about the robot software suite.

Type: [RobotSoftwareSuite](#) object

Required: Yes

## simulationSoftwareSuite

The simulation software suite used by the simulation application.

Type: [SimulationSoftwareSuite](#) object

Required: Yes

## sources

The sources of the simulation application.

Type: Array of [SourceConfig](#) objects

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "environment": {
    "uri": "string"
  },
  "lastUpdatedAt": number,
  "name": "string",
  "renderingEngine": {
    "name": "string",
    "version": "string"
  },
  "revisionId": "string",
  "robotSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "simulationSoftwareSuite": {
    "name": "string",
    "version": "string"
  },
  "sources": [
    {
```

```
    "architecture": "string",
    "etag": "string",
    "s3Bucket": "string",
    "s3Key": "string"
  }
],
"version": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

### arn

The Amazon Resource Name (ARN) of the updated simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

### environment

The object that contains the Docker image URI used for your simulation application.

Type: [Environment](#) object

### lastUpdatedAt

The time, in milliseconds since the epoch, when the simulation application was last updated.

Type: Timestamp

### name

The name of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

## renderingEngine

The rendering engine for the simulation application.

Type: [RenderingEngine](#) object

## revisionId

The revision id of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 40.

Pattern: [a-zA-Z0-9\_.\-]\*

## robotSoftwareSuite

Information about the robot software suite.

Type: [RobotSoftwareSuite](#) object

## simulationSoftwareSuite

The simulation software suite used by the simulation application.

Type: [SimulationSoftwareSuite](#) object

## sources

The sources of the simulation application.

Type: Array of [Source](#) objects

## version

The version of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: (\\$LATEST) | [0-9]\*

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

## InternalServerErrorException

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

## InvalidParameterException

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

## LimitExceededException

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

HTTP Status Code: 400

## ResourceNotFoundException

The specified resource does not exist.

HTTP Status Code: 400

## ThrottlingException

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)

- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

# UpdateWorldTemplate

Updates a world template.

## Request Syntax

```
POST /updateWorldTemplate HTTP/1.1
Content-type: application/json

{
  "name": "string",
  "template": "string",
  "templateBody": "string",
  "templateLocation": {
    "s3Bucket": "string",
    "s3Key": "string"
  }
}
```

## URI Request Parameters

The request does not use any URI parameters.

## Request Body

The request accepts the following data in JSON format.

### name

The name of the template.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 255.

Pattern: .\*

Required: No

### template

The Amazon Resource Name (arn) of the world template to update.

Type: String



Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: Yes

### templateBody

The world template body.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 262144.

Pattern: `[\S\s]+`

Required: No

### templateLocation

The location of the world template.

Type: [TemplateLocation](#) object

Required: No

## Response Syntax

```
HTTP/1.1 200
Content-type: application/json

{
  "arn": "string",
  "createdAt": number,
  "lastUpdatedAt": number,
  "name": "string"
}
```

## Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

## arn

The Amazon Resource Name (arn) of the world template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

## createdAt

The time, in milliseconds since the epoch, when the world template was created.

Type: Timestamp

## lastUpdatedAt

The time, in milliseconds since the epoch, when the world template was last updated.

Type: Timestamp

## name

The name of the world template.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 255.

Pattern: .\*

## Errors

For information about the errors that are common to all actions, see [Common Errors](#).

### **InternalServerErrorException**

AWS RoboMaker experienced a service issue. Try your call again.

HTTP Status Code: 500

### **InvalidParameterException**

A parameter specified in a request is not valid, is unsupported, or cannot be used. The returned message provides an explanation of the error value.

HTTP Status Code: 400

### **ResourceNotFoundException**

The specified resource does not exist.

HTTP Status Code: 400

### **ThrottlingException**

AWS RoboMaker is temporarily unable to process the request. Try your call again.

HTTP Status Code: 400

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go v2](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript V3](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)

## **Data Types**

The following data types are supported:

- [BatchPolicy](#)
- [Compute](#)
- [ComputeResponse](#)
- [DataSource](#)

- [DataSourceConfig](#)
- [DeploymentApplicationConfig](#)
- [DeploymentConfig](#)
- [DeploymentJob](#)
- [DeploymentLaunchConfig](#)
- [Environment](#)
- [FailedCreateSimulationJobRequest](#)
- [FailureSummary](#)
- [Filter](#)
- [FinishedWorldsSummary](#)
- [Fleet](#)
- [LaunchConfig](#)
- [LoggingConfig](#)
- [NetworkInterface](#)
- [OutputLocation](#)
- [PortForwardingConfig](#)
- [PortMapping](#)
- [ProgressDetail](#)
- [RenderingEngine](#)
- [Robot](#)
- [RobotApplicationConfig](#)
- [RobotApplicationSummary](#)
- [RobotDeployment](#)
- [RobotSoftwareSuite](#)
- [S3KeyOutput](#)
- [S3Object](#)
- [SimulationApplicationConfig](#)
- [SimulationApplicationSummary](#)
- [SimulationJob](#)
- [SimulationJobBatchSummary](#)

- [SimulationJobRequest](#)
- [SimulationJobSummary](#)
- [SimulationSoftwareSuite](#)
- [Source](#)
- [SourceConfig](#)
- [TemplateLocation](#)
- [TemplateSummary](#)
- [Tool](#)
- [UploadConfiguration](#)
- [VPCConfig](#)
- [VPCConfigResponse](#)
- [WorldConfig](#)
- [WorldCount](#)
- [WorldExportJobSummary](#)
- [WorldFailure](#)
- [WorldGenerationJobSummary](#)
- [WorldSummary](#)

# BatchPolicy

Information about the batch policy.

## Contents

### maxConcurrency

The number of active simulation jobs create as part of the batch that can be in an active state at the same time.

Active states include: Pending,Preparing, Running, Restarting, RunningFailed and Terminating. All other states are terminal states.

Type: Integer

Required: No

### timeoutInSeconds

The amount of time, in seconds, to wait for the batch to complete.

If a batch times out, and there are pending requests that were failing due to an internal failure (like `InternalServiceError`), they will be moved to the failed list and the batch status will be `Failed`. If the pending requests were failing for any other reason, the failed pending requests will be moved to the failed list and the batch status will be `TimedOut`.

Type: Long

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# Compute

Compute information for the simulation job.

## Contents

### **computeType**

Compute type information for the simulation job.

Type: String

Valid Values: CPU | GPU\_AND\_CPU

Required: No

### **gpuUnitLimit**

Compute GPU unit limit for the simulation job. It is the same as the number of GPUs allocated to the SimulationJob.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 1.

Required: No

### **simulationUnitLimit**

The simulation unit limit. Your simulation is allocated CPU and memory proportional to the supplied simulation unit limit. A simulation unit is 1 vcpu and 2GB of memory. You are only billed for the SU utilization you consume up to the maximum value provided. The default is 15.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 15.

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)



# ComputeResponse

Compute information for the simulation job

## Contents

### computeType

Compute type response information for the simulation job.

Type: String

Valid Values: CPU | GPU\_AND\_CPU

Required: No

### gpuUnitLimit

Compute GPU unit limit for the simulation job. It is the same as the number of GPUs allocated to the SimulationJob.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 1.

Required: No

### simulationUnitLimit

The simulation unit limit. Your simulation is allocated CPU and memory proportional to the supplied simulation unit limit. A simulation unit is 1 vcpu and 2GB of memory. You are only billed for the SU utilization you consume up to the maximum value provided. The default is 15.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 15.

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# DataSource

Information about a data source.

## Contents

### destination

The location where your files are mounted in the container image.

If you've specified the type of the data source as an `Archive`, you must provide an Amazon S3 object key to your archive. The object key must point to either a `.zip` or `.tar.gz` file.

If you've specified the type of the data source as a `Prefix`, you provide the Amazon S3 prefix that points to the files that you are using for your data source.

If you've specified the type of the data source as a `File`, you provide the Amazon S3 path to the file that you're using as your data source.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `.*`

Required: No

### name

The name of the data source.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[a-zA-Z0-9_\-\ ]*`

Required: No

### s3Bucket

The S3 bucket where the data files are located.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 63.

Pattern: `[a-z0-9][a-z0-9.\-]*[a-z0-9]`

Required: No

### s3Keys

The list of S3 keys identifying the data source files.

Type: Array of [S3KeyOutput](#) objects

Required: No

### type

The data type for the data source that you're using for your container image or simulation job. You can use this field to specify whether your data source is an Archive, an Amazon S3 prefix, or a file.

If you don't specify a field, the default value is `File`.

Type: String

Valid Values: `Prefix` | `Archive` | `File`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# DataSourceConfig

Information about a data source.

## Contents

### name

The name of the data source.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

Required: Yes

### s3Bucket

The S3 bucket where the data files are located.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 63.

Pattern: [a-z0-9][a-z0-9.\-]\*[a-z0-9]

Required: Yes

### s3Keys

The list of S3 keys identifying the data source files.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

Required: Yes

## destination

The location where your files are mounted in the container image.

If you've specified the type of the data source as an `Archive`, you must provide an Amazon S3 object key to your archive. The object key must point to either a `.zip` or `.tar.gz` file.

If you've specified the type of the data source as a `Prefix`, you provide the Amazon S3 prefix that points to the files that you are using for your data source.

If you've specified the type of the data source as a `File`, you provide the Amazon S3 path to the file that you're using as your data source.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `.*`

Required: No

## type

The data type for the data source that you're using for your container image or simulation job. You can use this field to specify whether your data source is an `Archive`, an Amazon S3 prefix, or a file.

If you don't specify a field, the default value is `File`.

Type: String

Valid Values: `Prefix` | `Archive` | `File`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)

- [AWS SDK for Ruby V3](#)

# DeploymentApplicationConfig

Information about a deployment application configuration.

## Contents

### application

The Amazon Resource Name (ARN) of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: Yes

### applicationVersion

The version of the application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [0-9]\*

Required: Yes

### launchConfig

The launch configuration.

Type: [DeploymentLaunchConfig](#) object

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)



- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# DeploymentConfig

Information about a deployment configuration.

## Contents

### **concurrentDeploymentPercentage**

The percentage of robots receiving the deployment at the same time.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 100.

Required: No

### **downloadConditionFile**

The download condition file.

Type: [S3Object](#) object

Required: No

### **failureThresholdPercentage**

The percentage of deployments that need to fail before stopping deployment.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 100.

Required: No

### **robotDeploymentTimeoutInSeconds**

The amount of time, in seconds, to wait for deployment to a single robot to complete. Choose a time between 1 minute and 7 days. The default is 5 hours.

Type: Long

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# DeploymentJob

Information about a deployment job.

## Contents

### arn

The Amazon Resource Name (ARN) of the deployment job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: No

### createdAt

The time, in milliseconds since the epoch, when the deployment job was created.

Type: Timestamp

Required: No

### deploymentApplicationConfigs

The deployment application configuration.

Type: Array of [DeploymentApplicationConfig](#) objects

Array Members: Fixed number of 1 item.

Required: No

### deploymentConfig

The deployment configuration.

Type: [DeploymentConfig](#) object

Required: No

### failureCode

The deployment job failure code.

Type: String

Valid Values: ResourceNotFound | EnvironmentSetupError | EtagMismatch  
| FailureThresholdBreached | RobotDeploymentAborted |  
RobotDeploymentNoResponse | RobotAgentConnectionTimeout  
| GreengrassDeploymentFailed | InvalidGreengrassGroup |  
MissingRobotArchitecture | MissingRobotApplicationArchitecture |  
MissingRobotDeploymentResource | GreengrassGroupVersionDoesNotExist  
| LambdaDeleted | ExtractingBundleFailure | PreLaunchFileFailure |  
PostLaunchFileFailure | BadPermissionError | DownloadConditionFailed |  
BadLambdaAssociated | InternalServerError | RobotApplicationDoesNotExist  
| DeploymentFleetDoesNotExist | FleetDeploymentTimeout

Required: No

### **failureReason**

A short description of the reason why the deployment job failed.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

Required: No

### **fleet**

The Amazon Resource Name (ARN) of the fleet.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: No

### **status**

The status of the deployment job.

Type: String

Valid Values: Pending | Preparing | InProgress | Failed | Succeeded | Canceled

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# DeploymentLaunchConfig

Configuration information for a deployment launch.

## Contents

### launchFile

The launch file name.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `[a-zA-Z0-9_.\-]*`

Required: Yes

### packageName

The package name.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `[a-zA-Z0-9_.\-]*`

Required: Yes

### environmentVariables

An array of key/value pairs specifying environment variables for the robot application

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 20 items.

Key Length Constraints: Minimum length of 1. Maximum length of 1024.

Key Pattern: `[A-Z_][A-Z0-9_]*`

Value Length Constraints: Minimum length of 1. Maximum length of 1024.

Value Pattern: `.*`

Required: No

### **postLaunchFile**

The deployment post-launch file. This file will be executed after the launch file.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: .\*

Required: No

### **preLaunchFile**

The deployment pre-launch file. This file will be executed prior to the launch file.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: .\*

Required: No

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)



## Environment

The object that contains the Docker image URI for either your robot or simulation applications.

### Contents

#### uri

The Docker image URI for either your robot or simulation applications.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: .+

Required: No

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# FailedCreateSimulationJobRequest

Information about a failed create simulation job request.

## Contents

### failedAt

The time, in milliseconds since the epoch, when the simulation job batch failed.

Type: Timestamp

Required: No

### failureCode

The failure code.

Type: String

Valid Values: `InternalServerError` | `RobotApplicationCrash` | `SimulationApplicationCrash` | `RobotApplicationHealthCheckFailure` | `SimulationApplicationHealthCheckFailure` | `BadPermissionsRobotApplication` | `BadPermissionsSimulationApplication` | `BadPermissionsS3Object` | `BadPermissionsS3Output` | `BadPermissionsCloudwatchLogs` | `SubnetIpLimitExceeded` | `ENILimitExceeded` | `BadPermissionsUserCredentials` | `InvalidBundleRobotApplication` | `InvalidBundleSimulationApplication` | `InvalidS3Resource` | `ThrottlingError` | `LimitExceeded` | `MismatchedEtag` | `RobotApplicationVersionMismatchedEtag` | `SimulationApplicationVersionMismatchedEtag` | `ResourceNotFound` | `RequestThrottled` | `BatchTimedOut` | `BatchCanceled` | `InvalidInput` | `WrongRegionS3Bucket` | `WrongRegionS3Output` | `WrongRegionRobotApplication` | `WrongRegionSimulationApplication` | `UploadContentMismatchError`

Required: No

### failureReason

The failure reason of the simulation job request.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

Required: No

## request

The simulation job request.

Type: [SimulationJobRequest](#) object

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# FailureSummary

Information about worlds that failed.

## Contents

### failures

The worlds that failed.

Type: Array of [WorldFailure](#) objects

Array Members: Minimum number of 0 items. Maximum number of 100 items.

Required: No

### totalFailureCount

The total number of failures.

Type: Integer

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

## Filter

Information about a filter.

### Contents

#### name

The name of the filter.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

Required: No

#### values

A list of values.

Type: Array of strings

Array Members: Fixed number of 1 item.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

Required: No

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

## FinishedWorldsSummary

Information about worlds that finished.

### Contents

#### failureSummary

Information about worlds that failed.

Type: [FailureSummary](#) object

Required: No

#### finishedCount

The total number of finished worlds.

Type: Integer

Required: No

#### succeededWorlds

A list of worlds that succeeded.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: a1n:.\*

Required: No

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)

- [AWS SDK for Ruby V3](#)

# Fleet

Information about a fleet.

## Contents

### **arn**

The Amazon Resource Name (ARN) of the fleet.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: No

### **createdAt**

The time, in milliseconds since the epoch, when the fleet was created.

Type: Timestamp

Required: No

### **lastDeploymentJob**

The Amazon Resource Name (ARN) of the last deployment job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: No

### **lastDeploymentStatus**

The status of the last fleet deployment.

Type: String

Valid Values: Pending | Preparing | InProgress | Failed | Succeeded | Canceled



Required: No

### **lastDeploymentTime**

The time of the last deployment.

Type: Timestamp

Required: No

### **name**

The name of the fleet.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

Required: No

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# LaunchConfig

Information about a launch configuration.

## Contents

### command

If you've specified `General` as the value for your `RobotSoftwareSuite`, you can use this field to specify a list of commands for your container image.

If you've specified `SimulationRuntime` as the value for your `SimulationSoftwareSuite`, you can use this field to specify a list of commands for your container image.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `.+`

Required: No

### environmentVariables

The environment variables for the application launch.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 20 items.

Key Length Constraints: Minimum length of 1. Maximum length of 1024.

Key Pattern: `[A-Z_][A-Z0-9_]*`

Value Length Constraints: Minimum length of 1. Maximum length of 1024.

Value Pattern: `.*`

Required: No

### launchFile

The launch file name.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `[a-zA-Z0-9_.\-]*`

Required: No

### **packageName**

The package name.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `[a-zA-Z0-9_.\-]*`

Required: No

### **portForwardingConfig**

The port forwarding configuration.

Type: [PortForwardingConfig](#) object

Required: No

### **streamUI**

Boolean indicating whether a streaming session will be configured for the application. If `True`, AWS RoboMaker will configure a connection so you can interact with your application as it is running in the simulation. You must configure and launch the component. It must have a graphical user interface.

Type: Boolean

Required: No

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)

- [AWS SDK for Ruby V3](#)

# LoggingConfig

The logging configuration.

## Contents

### recordAllRosTopics

*This member has been deprecated.*

A boolean indicating whether to record all ROS topics.

#### Important

This API is no longer supported and will throw an error if used.

Type: Boolean

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# NetworkInterface

Describes a network interface.

## Contents

### **networkInterfaceId**

The ID of the network interface.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

Required: No

### **privateIpAddress**

The IPv4 address of the network interface within the subnet.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

Required: No

### **publicIpAddress**

The IPv4 public address of the network interface.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# OutputLocation

The output location.

## Contents

### s3Bucket

The S3 bucket for output.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 63.

Pattern: [a-z0-9][a-z0-9.\-]\*[a-z0-9]

Required: No

### s3Prefix

The S3 folder in the s3Bucket where output files will be placed.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: .\*

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)



# PortForwardingConfig

Configuration information for port forwarding.

## Contents

### portMappings

The port mappings for the configuration.

Type: Array of [PortMapping](#) objects

Array Members: Minimum number of 0 items. Maximum number of 10 items.

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# PortMapping

An object representing a port mapping.

## Contents

### **applicationPort**

The port number on the application.

Type: Integer

Valid Range: Minimum value of 1024. Maximum value of 65535.

Required: Yes

### **jobPort**

The port number on the simulation job instance to use as a remote connection point.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 65535.

Required: Yes

### **enableOnPublicIp**

A Boolean indicating whether to enable this port mapping on public IP.

Type: Boolean

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)



## ProgressDetail

Information about the progress of a deployment job.

### Contents

#### currentProgress

The current progress status.

##### Validating

Validating the deployment.

##### DownloadingExtracting

Downloading and extracting the bundle on the robot.

##### ExecutingPreLaunch

Executing pre-launch script(s) if provided.

##### Launching

Launching the robot application.

##### ExecutingPostLaunch

Executing post-launch script(s) if provided.

##### Finished

Deployment is complete.

Type: String

Valid Values: Validating | DownloadingExtracting | ExecutingDownloadCondition | ExecutingPreLaunch | Launching | ExecutingPostLaunch | Finished

Required: No

#### estimatedTimeRemainingSeconds

Estimated amount of time in seconds remaining in the step. This currently only applies to the Downloading/Extracting step of the deployment. It is empty for other steps.

Type: Integer

Required: No

### **percentDone**

Percentage of the step that is done. This currently only applies to the Downloading/Extracting step of the deployment. It is empty for other steps.

Type: Float

Valid Range: Minimum value of 0.0. Maximum value of 100.0.

Required: No

### **targetResource**

The Amazon Resource Name (ARN) of the deployment job.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

Required: No

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# RenderingEngine

Information about a rendering engine.

## Contents

### name

The name of the rendering engine.

Type: String

Valid Values: OGRE

Required: No

### version

The version of the rendering engine.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 4.

Pattern: 1 . x

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# Robot

Information about a robot.

## Contents

### architecture

The architecture of the robot.

Type: String

Valid Values: X86\_64 | ARM64 | ARMHF

Required: No

### arn

The Amazon Resource Name (ARN) of the robot.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: No

### createdAt

The time, in milliseconds since the epoch, when the robot was created.

Type: Timestamp

Required: No

### fleetArn

The Amazon Resource Name (ARN) of the fleet.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: No

### **greenGrassGroupId**

The Greengrass group associated with the robot.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: .\*

Required: No

### **lastDeploymentJob**

The Amazon Resource Name (ARN) of the last deployment job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: No

### **lastDeploymentTime**

The time of the last deployment.

Type: Timestamp

Required: No

### **name**

The name of the robot.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

Required: No



## status

The status of the robot.

Type: String

Valid Values: Available | Registered | PendingNewDeployment | Deploying | Failed | InSync | NoResponse

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# RobotApplicationConfig

Application configuration information for a robot.

## Contents

### application

The application information for the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `a1n:.*`

Required: Yes

### launchConfig

The launch configuration for the robot application.

Type: [LaunchConfig](#) object

Required: Yes

### applicationVersion

The version of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `(\\$LATEST)|[0-9]*`

Required: No

### tools

Information about tools configured for the robot application.

Type: Array of [Tool](#) objects

Array Members: Minimum number of 0 items. Maximum number of 10 items.

Required: No

## **uploadConfigurations**

The upload configurations for the robot application.

Type: Array of [UploadConfiguration](#) objects

Array Members: Minimum number of 0 items. Maximum number of 10 items.

Required: No

## **useDefaultTools**

*This member has been deprecated.*

A Boolean indicating whether to use default robot application tools. The default tools are rviz, rqt, terminal and rosbag record. The default is `False`.

### **Important**

This API is no longer supported and will throw an error if used.

Type: Boolean

Required: No

## **useDefaultUploadConfigurations**

*This member has been deprecated.*

A Boolean indicating whether to use default upload configurations. By default, `.ros` and `.gazebo` files are uploaded when the application terminates and all ROS topics will be recorded.

If you set this value, you must specify an `outputLocation`.

### **Important**

This API is no longer supported and will throw an error if used.

Type: Boolean

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# RobotApplicationSummary

Summary information for a robot application.

## Contents

### arn

The Amazon Resource Name (ARN) of the robot.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: No

### lastUpdatedAt

The time, in milliseconds since the epoch, when the robot application was last updated.

Type: Timestamp

Required: No

### name

The name of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

Required: No

### robotSoftwareSuite

Information about a robot software suite.

Type: [RobotSoftwareSuite](#) object

Required: No

## version

The version of the robot application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: (`\$LATEST`)|`[0-9]*`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# RobotDeployment

Information about a robot deployment.

## Contents

### arn

The robot deployment Amazon Resource Name (ARN).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: No

### deploymentFinishTime

The time, in milliseconds since the epoch, when the deployment finished.

Type: Timestamp

Required: No

### deploymentStartTime

The time, in milliseconds since the epoch, when the deployment was started.

Type: Timestamp

Required: No

### failureCode

The robot deployment failure code.

Type: String

Valid Values: `ResourceNotFound` | `EnvironmentSetupError` | `EtagMismatch` | `FailureThresholdBreached` | `RobotDeploymentAborted` | `RobotDeploymentNoResponse` | `RobotAgentConnectionTimeout` | `GreengrassDeploymentFailed` | `InvalidGreengrassGroup` | `MissingRobotArchitecture` | `MissingRobotApplicationArchitecture` |

MissingRobotDeploymentResource | GreengrassGroupVersionDoesNotExist  
| LambdaDeleted | ExtractingBundleFailure | PreLaunchFileFailure |  
PostLaunchFileFailure | BadPermissionError | DownloadConditionFailed |  
BadLambdaAssociated | InternalServerError | RobotApplicationDoesNotExist  
| DeploymentFleetDoesNotExist | FleetDeploymentTimeout

Required: No

### **failureReason**

A short description of the reason why the robot deployment failed.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

Required: No

### **progressDetail**

Information about how the deployment is progressing.

Type: [ProgressDetail](#) object

Required: No

### **status**

The status of the robot deployment.

Type: String

Valid Values: Available | Registered | PendingNewDeployment | Deploying |  
Failed | InSync | NoResponse

Required: No

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:



- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# RobotSoftwareSuite

Information about a robot software suite.

## Contents

### name

The name of the robot software suite. `General` is the only supported value.

Type: String

Valid Values: `ROS` | `ROS2` | `General`

Required: No

### version

The version of the robot software suite. Not applicable for `General` software suite.

Type: String

Valid Values: `Kinetic` | `Melodic` | `Dashing` | `Foxy`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# S3KeyOutput

Information about S3 keys.

## Contents

### etag

The etag for the object.

Type: String

Required: No

### s3Key

The S3 key.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# S3Object

Information about an S3 object.

## Contents

### bucket

The bucket containing the object.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 63.

Pattern: `[a-z0-9][a-z0-9.\-]*[a-z0-9]`

Required: Yes

### key

The key of the object.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `.*`

Required: Yes

### etag

The etag of the object.

Type: String

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)

- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# SimulationApplicationConfig

Information about a simulation application configuration.

## Contents

### application

The application information for the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `a1n:.*`

Required: Yes

### launchConfig

The launch configuration for the simulation application.

Type: [LaunchConfig](#) object

Required: Yes

### applicationVersion

The version of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `(\\$LATEST) | [0-9]*`

Required: No

### tools

Information about tools configured for the simulation application.

Type: Array of [Tool](#) objects

Array Members: Minimum number of 0 items. Maximum number of 10 items.

Required: No

## **uploadConfigurations**

Information about upload configurations for the simulation application.

Type: Array of [UploadConfiguration](#) objects

Array Members: Minimum number of 0 items. Maximum number of 10 items.

Required: No

## **useDefaultTools**

*This member has been deprecated.*

A Boolean indicating whether to use default simulation application tools. The default tools are rviz, rqt, terminal and rosbag record. The default is `False`.

### **Important**

This API is no longer supported and will throw an error if used.

Type: Boolean

Required: No

## **useDefaultUploadConfigurations**

*This member has been deprecated.*

A Boolean indicating whether to use default upload configurations. By default, `.ros` and `.gazebo` files are uploaded when the application terminates and all ROS topics will be recorded.

If you set this value, you must specify an `outputLocation`.

### **Important**

This API is no longer supported and will throw an error if used.

Type: Boolean

Required: No

## **worldConfigs**

A list of world configurations.

### **Important**

This API is no longer supported and will throw an error if used.

Type: Array of [WorldConfig](#) objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

Required: No

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)



# SimulationApplicationSummary

Summary information for a simulation application.

## Contents

### arn

The Amazon Resource Name (ARN) of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: No

### lastUpdatedAt

The time, in milliseconds since the epoch, when the simulation application was last updated.

Type: Timestamp

Required: No

### name

The name of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

Required: No

### robotSoftwareSuite

Information about a robot software suite.

Type: [RobotSoftwareSuite](#) object

Required: No

## simulationSoftwareSuite

Information about a simulation software suite.

Type: [SimulationSoftwareSuite](#) object

Required: No

### version

The version of the simulation application.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: (\\$LATEST) | [0-9]\*

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# SimulationJob

Information about a simulation job.

## Contents

### arn

The Amazon Resource Name (ARN) of the simulation job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: No

### clientRequestToken

A unique identifier for this `SimulationJob` request.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `[a-zA-Z0-9_\-=]*`

Required: No

### compute

Compute information for the simulation job

Type: [ComputeResponse](#) object

Required: No

### dataSources

The data sources for the simulation job.

Type: Array of [DataSource](#) objects

Required: No

## **failureBehavior**

The failure behavior the simulation job.

Continue

Leaves the host running for its maximum timeout duration after a 4XX error code.

Fail

Stop the simulation job and terminate the instance.

Type: String

Valid Values: Fail | Continue

Required: No

## **failureCode**

The failure code of the simulation job if it failed.

Type: String

Valid Values: InternalServiceError | RobotApplicationCrash | SimulationApplicationCrash | RobotApplicationHealthCheckFailure | SimulationApplicationHealthCheckFailure | BadPermissionsRobotApplication | BadPermissionsSimulationApplication | BadPermissionsS3Object | BadPermissionsS3Output | BadPermissionsCloudwatchLogs | SubnetIpLimitExceeded | ENILimitExceeded | BadPermissionsUserCredentials | InvalidBundleRobotApplication | InvalidBundleSimulationApplication | InvalidS3Resource | ThrottlingError | LimitExceeded | MismatchedEtag | RobotApplicationVersionMismatchedEtag | SimulationApplicationVersionMismatchedEtag | ResourceNotFound | RequestThrottled | BatchTimedOut | BatchCanceled | InvalidInput | WrongRegionS3Bucket | WrongRegionS3Output | WrongRegionRobotApplication | WrongRegionSimulationApplication | UploadContentMismatchError

Required: No

## **failureReason**

The reason why the simulation job failed.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

Required: No

### **iamRole**

The IAM role that allows the simulation instance to call the AWS APIs that are specified in its associated policies on your behalf. This is how credentials are passed in to your simulation job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: arn:aws:iam:\*\w+:role/.\*

Required: No

### **lastStartedAt**

The time, in milliseconds since the epoch, when the simulation job was last started.

Type: Timestamp

Required: No

### **lastUpdatedAt**

The time, in milliseconds since the epoch, when the simulation job was last updated.

Type: Timestamp

Required: No

### **loggingConfig**

The logging configuration.

Type: [LoggingConfig](#) object

Required: No

### **maxJobDurationInSeconds**

The maximum simulation job duration in seconds. The value must be 8 days (691,200 seconds) or less.

Type: Long

Required: No

### **name**

The name of the simulation job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

Required: No

### **networkInterface**

Information about a network interface.

Type: [NetworkInterface](#) object

Required: No

### **outputLocation**

Location for output files generated by the simulation job.

Type: [OutputLocation](#) object

Required: No

### **robotApplications**

A list of robot applications.

Type: Array of [RobotApplicationConfig](#) objects

Array Members: Fixed number of 1 item.

Required: No

### **simulationApplications**

A list of simulation applications.

Type: Array of [SimulationApplicationConfig](#) objects

Array Members: Fixed number of 1 item.

Required: No

### **simulationTimeMillis**

The simulation job execution duration in milliseconds.

Type: Long

Required: No

### **status**

Status of the simulation job.

Type: String

Valid Values: Pending | Preparing | Running | Restarting | Completed | Failed | RunningFailed | Terminating | Terminated | Canceled

Required: No

### **tags**

A map that contains tag keys and tag values that are attached to the simulation job.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: [a-zA-Z0-9 \_.\-\/+=:]\*

Required: No

### **vpcConfig**

VPC configuration information.

Type: [VPCConfigResponse](#) object

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)



# SimulationJobBatchSummary

Information about a simulation job batch.

## Contents

### **arn**

The Amazon Resource Name (ARN) of the batch.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn : . \*

Required: No

### **createdAt**

The time, in milliseconds since the epoch, when the simulation job batch was created.

Type: Timestamp

Required: No

### **createdRequestCount**

The number of created simulation job requests.

Type: Integer

Required: No

### **failedRequestCount**

The number of failed simulation job requests.

Type: Integer

Required: No

### **lastUpdatedAt**

The time, in milliseconds since the epoch, when the simulation job batch was last updated.

Type: Timestamp

Required: No

### **pendingRequestCount**

The number of pending simulation job requests.

Type: Integer

Required: No

### **status**

The status of the simulation job batch.

Pending

The simulation job batch request is pending.

InProgress

The simulation job batch is in progress.

Failed

The simulation job batch failed. One or more simulation job requests could not be completed due to an internal failure (like `InternalServerError`). See `failureCode` and `failureReason` for more information.

Completed

The simulation batch job completed. A batch is complete when (1) there are no pending simulation job requests in the batch and none of the failed simulation job requests are due to `InternalServerError` and (2) when all created simulation jobs have reached a terminal state (for example, `Completed` or `Failed`).

Canceled

The simulation batch job was cancelled.

Canceling

The simulation batch job is being cancelled.

Completing

The simulation batch job is completing.

## TimingOut

The simulation job batch is timing out.

If a batch timing out, and there are pending requests that were failing due to an internal failure (like `InternalServerError`), the batch status will be `Failed`. If there are no such failing request, the batch status will be `TimedOut`.

## TimedOut

The simulation batch job timed out.

Type: String

Valid Values: Pending | InProgress | Failed | Completed | Canceled | Canceling | Completing | TimingOut | TimedOut

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# SimulationJobRequest

Information about a simulation job request.

## Contents

### maxJobDurationInSeconds

The maximum simulation job duration in seconds. The value must be 8 days (691,200 seconds) or less.

Type: Long

Required: Yes

### compute

Compute information for the simulation job

Type: [Compute](#) object

Required: No

### dataSources

Specify data sources to mount read-only files from S3 into your simulation. These files are available under `/opt/robomaker/datasources/data_source_name`.

#### Note

There is a limit of 100 files and a combined size of 25GB for all `DataSourceConfig` objects.

Type: Array of [DataSourceConfig](#) objects

Array Members: Minimum number of 1 item. Maximum number of 6 items.

Required: No

### failureBehavior

The failure behavior the simulation job.

## Continue

Leaves the host running for its maximum timeout duration after a 4XX error code.

## Fail

Stop the simulation job and terminate the instance.

Type: String

Valid Values: `Fail` | `Continue`

Required: No

## **iamRole**

The IAM role name that allows the simulation instance to call the AWS APIs that are specified in its associated policies on your behalf. This is how credentials are passed in to your simulation job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `arn:aws:iam::\w+:role/.*`

Required: No

## **loggingConfig**

The logging configuration.

Type: [LoggingConfig](#) object

Required: No

## **outputLocation**

The output location.

Type: [OutputLocation](#) object

Required: No

## **robotApplications**

The robot applications to use in the simulation job.

Type: Array of [RobotApplicationConfig](#) objects

Array Members: Fixed number of 1 item.

Required: No

### **simulationApplications**

The simulation applications to use in the simulation job.

Type: Array of [SimulationApplicationConfig](#) objects

Array Members: Fixed number of 1 item.

Required: No

### **tags**

A map that contains tag keys and tag values that are attached to the simulation job request.

Type: String to string map

Map Entries: Minimum number of 0 items. Maximum number of 50 items.

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Key Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Value Pattern: `[a-zA-Z0-9 _.\-\/+=:]*`

Required: No

### **useDefaultApplications**

A Boolean indicating whether to use default applications in the simulation job. Default applications include Gazebo, rqt, rviz and terminal access.

Type: Boolean

Required: No

### **vpcConfig**

If your simulation job accesses resources in a VPC, you provide this parameter identifying the list of security group IDs and subnet IDs. These must belong to the same VPC. You must provide at least one security group and two subnet IDs.

Type: [VPCConfig](#) object

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# SimulationJobSummary

Summary information for a simulation job.

## Contents

### arn

The Amazon Resource Name (ARN) of the simulation job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: No

### computeType

The compute type for the simulation job summary.

Type: String

Valid Values: CPU | GPU\_AND\_CPU

Required: No

### dataSourceNames

The names of the data sources.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[a-zA-Z0-9_\-]*`

Required: No

### lastUpdatedAt

The time, in milliseconds since the epoch, when the simulation job was last updated.

Type: Timestamp



Required: No

### **name**

The name of the simulation job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[a-zA-Z0-9_\-]*`

Required: No

### **robotApplicationNames**

A list of simulation job robot application names.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[a-zA-Z0-9_\-]*`

Required: No

### **simulationApplicationNames**

A list of simulation job simulation application names.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[a-zA-Z0-9_\-]*`

Required: No

### **status**

The status of the simulation job.

Type: String

Valid Values: Pending | Preparing | Running | Restarting | Completed | Failed | RunningFailed | Terminating | Terminated | Canceled

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# SimulationSoftwareSuite

Information about a simulation software suite.

## Contents

### name

The name of the simulation software suite. `SimulationRuntime` is the only supported value.

Type: String

Valid Values: `Gazebo` | `RosbagPlay` | `SimulationRuntime`

Required: No

### version

The version of the simulation software suite. Not applicable for `SimulationRuntime`.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: `7|9|11|Kinetic|Melodic|Dashing|Foxy`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

## Source

Information about a source.

### Contents

#### architecture

The target processor architecture for the application.

Type: String

Valid Values: X86\_64 | ARM64 | ARMHF

Required: No

#### etag

A hash of the object specified by s3Bucket and s3Key.

Type: String

Required: No

#### s3Bucket

The s3 bucket name.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 63.

Pattern: [a-z0-9][a-z0-9.\-]\*[a-z0-9]

Required: No

#### s3Key

The s3 object key.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: .\*

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# SourceConfig

Information about a source configuration.

## Contents

### architecture

The target processor architecture for the application.

Type: String

Valid Values: X86\_64 | ARM64 | ARMHF

Required: No

### s3Bucket

The Amazon S3 bucket name.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 63.

Pattern: [a-z0-9][a-z0-9.\-]\*[a-z0-9]

Required: No

### s3Key

The s3 object key.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: .\*

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# TemplateLocation

Information about a template location.

## Contents

### s3Bucket

The Amazon S3 bucket name.

Type: String

Length Constraints: Minimum length of 3. Maximum length of 63.

Pattern: `[a-z0-9][a-z0-9.\-]*[a-z0-9]`

Required: Yes

### s3Key

The list of S3 keys identifying the data source files.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `.*`

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)



# TemplateSummary

Summary information for a template.

## Contents

### arn

The Amazon Resource Name (ARN) of the template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: No

### createdAt

The time, in milliseconds since the epoch, when the template was created.

Type: Timestamp

Required: No

### lastUpdatedAt

The time, in milliseconds since the epoch, when the template was last updated.

Type: Timestamp

Required: No

### name

The name of the template.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 255.

Pattern: .\*

Required: No

## version

The version of the template that you're using.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# Tool

Information about a tool. Tools are used in a simulation job.

## Contents

### command

Command-line arguments for the tool. It must include the tool executable name.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: .\*

Required: Yes

### name

The name of the tool.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: [a-zA-Z0-9\_\-]\*

Required: Yes

### exitBehavior

Exit behavior determines what happens when your tool quits running. RESTART will cause your tool to be restarted. FAIL will cause your job to exit. The default is RESTART.

Type: String

Valid Values: FAIL | RESTART

Required: No

### streamOutputToCloudWatch

Boolean indicating whether logs will be recorded in CloudWatch for the tool. The default is False.

Type: Boolean

Required: No

### **streamUI**

Boolean indicating whether a streaming session will be configured for the tool. If `True`, AWS RoboMaker will configure a connection so you can interact with the tool as it is running in the simulation. It must have a graphical user interface. The default is `False`.

Type: Boolean

Required: No

### **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# UploadConfiguration

Provides upload configuration information. Files are uploaded from the simulation job to a location you specify.

## Contents

### name

A prefix that specifies where files will be uploaded in Amazon S3. It is appended to the simulation output location to determine the final path.

For example, if your simulation output location is `s3://my-bucket` and your upload configuration name is `robot-test`, your files will be uploaded to `s3://my-bucket/<simid>/<runid>/robot-test`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: `[a-zA-Z0-9_\-]*`

Required: Yes

### path

Specifies the path of the file(s) to upload. Standard Unix glob matching rules are accepted, with the addition of `**` as a *super asterisk*. For example, specifying `/var/log/**.log` causes all `.log` files in the `/var/log` directory tree to be collected. For more examples, see [Glob Library](#).

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1024.

Pattern: `.*`

Required: Yes

### uploadBehavior

Specifies when to upload the files:

## UPLOAD\_ON\_TERMINATE

Matching files are uploaded once the simulation enters the `TERMINATING` state. Matching files are not uploaded until all of your code (including tools) have stopped.

If there is a problem uploading a file, the upload is retried. If problems persist, no further upload attempts will be made.

## UPLOAD\_ROLLING\_AUTO\_REMOVE

Matching files are uploaded as they are created. They are deleted after they are uploaded. The specified path is checked every 5 seconds. A final check is made when all of your code (including tools) have stopped.

Type: String

Valid Values: `UPLOAD_ON_TERMINATE` | `UPLOAD_ROLLING_AUTO_REMOVE`

Required: Yes

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# VPCConfig

If your simulation job accesses resources in a VPC, you provide this parameter identifying the list of security group IDs and subnet IDs. These must belong to the same VPC. You must provide at least one security group and two subnet IDs.

## Contents

### subnets

A list of one or more subnet IDs in your VPC.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 16 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: . +

Required: Yes

### assignPublicIp

A boolean indicating whether to assign a public IP address.

Type: Boolean

Required: No

### securityGroups

A list of one or more security groups IDs in your VPC.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 5 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: . +

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)



# VPCConfigResponse

VPC configuration associated with your simulation job.

## Contents

### assignPublicIp

A boolean indicating if a public IP was assigned.

Type: Boolean

Required: No

### securityGroups

A list of security group IDs associated with the simulation job.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 5 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: .+

Required: No

### subnets

A list of subnet IDs associated with the simulation job.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 16 items.

Length Constraints: Minimum length of 1. Maximum length of 255.

Pattern: .+

Required: No

### vpcId

The VPC ID associated with your simulation job.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: .\*

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# WorldConfig

Configuration information for a world.

## Contents

### world

The world generated by Simulation WorldForge.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

## WorldCount

The number of worlds that will be created. You can configure the number of unique floorplans and the number of unique interiors for each floor plan. For example, if you want 1 world with 20 unique interiors, you set `floorplanCount = 1` and `interiorCountPerFloorplan = 20`. This will result in 20 worlds (`floorplanCount * interiorCountPerFloorplan`).

If you set `floorplanCount = 4` and `interiorCountPerFloorplan = 5`, there will be 20 worlds with 5 unique floor plans.

### Contents

#### **floorplanCount**

The number of unique floorplans.

Type: Integer

Required: No

#### **interiorCountPerFloorplan**

The number of unique interiors per floorplan.

Type: Integer

Required: No

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# WorldExportJobSummary

Information about a world export job.

## Contents

### arn

The Amazon Resource Name (ARN) of the world export job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: No

### createdAt

The time, in milliseconds since the epoch, when the world export job was created.

Type: Timestamp

Required: No

### outputLocation

The output location.

Type: [OutputLocation](#) object

Required: No

### status

The status of the world export job.

Pending

The world export job request is pending.

Running

The world export job is running.

## Completed

The world export job completed.

## Failed

The world export job failed. See `failureCode` for more information.

## Canceled

The world export job was cancelled.

## Canceling

The world export job is being cancelled.

Type: String

Valid Values: Pending | Running | Completed | Failed | Canceling | Canceled

Required: No

## worlds

A list of worlds.

Type: Array of strings

Array Members: Minimum number of 1 item. Maximum number of 100 items.

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)



# WorldFailure

Information about a failed world.

## Contents

### failureCode

The failure code of the world export job if it failed:

`InternalServerError`

Internal service error.

`LimitExceeded`

The requested resource exceeds the maximum number allowed, or the number of concurrent stream requests exceeds the maximum number allowed.

`ResourceNotFound`

The specified resource could not be found.

`RequestThrottled`

The request was throttled.

`InvalidInput`

An input parameter in the request is not valid.

Type: String

Valid Values: `InternalServerError` | `LimitExceeded` | `ResourceNotFound` | `RequestThrottled` | `InvalidInput` | `AllWorldGenerationFailed`

Required: No

### failureCount

The number of failed worlds.

Type: Integer

Required: No



## **sampleFailureReason**

The sample reason why the world failed. World errors are aggregated. A sample is used as the `sampleFailureReason`.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 1024.

Pattern: . \*

Required: No

## **See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# WorldGenerationJobSummary

Information about a world generator job.

## Contents

### arn

The Amazon Resource Name (ARN) of the world generator job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: arn:.\*

Required: No

### createdAt

The time, in milliseconds since the epoch, when the world generator job was created.

Type: Timestamp

Required: No

### failedWorldCount

The number of worlds that failed.

Type: Integer

Required: No

### status

The status of the world generator job:

Pending

The world generator job request is pending.

Running

The world generator job is running.

## Completed

The world generator job completed.

## Failed

The world generator job failed. See `failureCode` for more information.

## PartialFailed

Some worlds did not generate.

## Canceled

The world generator job was cancelled.

## Canceling

The world generator job is being cancelled.

Type: String

Valid Values: Pending | Running | Completed | Failed | PartialFailed | Canceling | Canceled

Required: No

## **succeededWorldCount**

The number of worlds that were generated.

Type: Integer

Required: No

## **template**

The Amazon Resource Name (arn) of the world template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: No

## worldCount

Information about the world count.

Type: [WorldCount](#) object

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

# WorldSummary

Information about a world.

## Contents

### **arn**

The Amazon Resource Name (ARN) of the world.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: No

### **createdAt**

The time, in milliseconds since the epoch, when the world was created.

Type: Timestamp

Required: No

### **generationJob**

The Amazon Resource Name (arn) of the world generation job.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: No

### **template**

The Amazon Resource Name (arn) of the world template.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 1224.

Pattern: `arn:.*`

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

## Common Errors

This section lists the errors common to the API actions of all AWS services. For errors specific to an API action for this service, see the topic for that API action.

### AccessDeniedException

You do not have sufficient access to perform this action.

HTTP Status Code: 400

### IncompleteSignature

The request signature does not conform to AWS standards.

HTTP Status Code: 400

### InternalFailure

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

### InvalidAction

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 400

### InvalidClientTokenId

The X.509 certificate or AWS access key ID provided does not exist in our records.

HTTP Status Code: 403

### **NotAuthorized**

You do not have permission to perform this action.

HTTP Status Code: 400

### **OptInRequired**

The AWS access key ID needs a subscription for the service.

HTTP Status Code: 403

### **RequestExpired**

The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.

HTTP Status Code: 400

### **ServiceUnavailable**

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

### **ThrottlingException**

The request was denied due to request throttling.

HTTP Status Code: 400

### **ValidationError**

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400

## **Common Parameters**

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action.

For more information about Signature Version 4, see [Signing AWS API requests](#) in the *IAM User Guide*.

### Action

The action to be performed.

Type: string

Required: Yes

### Version

The API version that the request is written for, expressed in the format YYYY-MM-DD.

Type: string

Required: Yes

### X-Amz-Algorithm

The hash algorithm that you used to create the request signature.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Valid Values: AWS4-HMAC-SHA256

Required: Conditional

### X-Amz-Credential

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4\_request"). The value is expressed in the following format: `access_key/YYYYMMDD/region/service/aws4_request`.

For more information, see [Create a signed AWS API request](#) in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.



Type: string

Required: Conditional

### **X-Amz-Date**

The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: 20120325T120000Z.

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see [Elements of an AWS API request signature](#) in the *IAM User Guide*.

Type: string

Required: Conditional

### **X-Amz-Security-Token**

The temporary security token that was obtained through a call to AWS Security Token Service (AWS STS). For a list of services that support temporary security credentials from AWS STS, see [AWS services that work with IAM](#) in the *IAM User Guide*.

Condition: If you're using temporary security credentials from AWS STS, you must include the security token.

Type: string

Required: Conditional

### **X-Amz-Signature**

Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

## X-Amz-SignedHeaders

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see [Create a signed AWS API request](#) in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

# AWS RoboMaker endpoints and quotas

The following are the service endpoints and service quotas for AWS RoboMaker. To connect programmatically to an AWS service, you use an endpoint. In addition to the standard AWS endpoints, some AWS services offer FIPS endpoints in selected Regions. For more information, see [AWS service endpoints](#).

*Service quotas*, also referred to as *limits*, are the maximum number of service resources or operations for your AWS account. For more information, see [AWS service quotas](#).

## Service endpoints

Region Name	Region	Endpoint	Protocol
US East (Ohio)	us-east-2	robomaker.us-east-2.amazonaws.com	HTTPS
US East (N. Virginia)	us-east-1	robomaker.us-east-1.amazonaws.com	HTTPS
US West (Oregon)	us-west-2	robomaker.us-west-2.amazonaws.com	HTTPS
Asia Pacific (Singapore)	ap-southeast-1	robomaker.ap-southeast-1.amazonaws.com	HTTPS
Asia Pacific (Tokyo)	ap-northeast-1	robomaker.ap-northeast-1.amazonaws.com	HTTPS
Europe (Frankfurt)	eu-central-1	robomaker.eu-central-1.amazonaws.com	HTTPS

Region Name	Region	Endpoint	Protocol
Europe (Ireland)	eu-west-1	robomaker.eu-west-1.amazonaws.com	HTTPS
AWS GovCloud (US-West)	us-gov-west-1	robomaker.us-gov-west-1.amazonaws.com	HTTPS

## Service quotas

Name	Default	Adjustable	Description
Batch timeout	Each supported Region: 14	No	The maximum timeout in days for a simulation job batch
Concurrent GPU simulation jobs	Each supported Region: 1	<a href="#">Yes</a>	The maximum number of concurrent GPU simulation jobs you can run in this account in the current Region.
Concurrent World Export Jobs	Each supported Region: 3	<a href="#">Yes</a>	The maximum number of concurrent world export jobs that you can run in this account in this region.
Concurrent World Generation Jobs	Each supported Region: 3	<a href="#">Yes</a>	The maximum number of concurrent world generation jobs that you

Name	Default	Adjustable	Description
			can run in this account in this region.
Concurrent deployment jobs	Each supported Region: 20	<a href="#">Yes</a>	The maximum number of concurrent deployment jobs you can run in this account in the current Region.
Concurrent simulation job batches	Each supported Region: 5	<a href="#">Yes</a>	The maximum number of concurrent simulation job batches you can run in this account in the current Region.
Concurrent simulation jobs	Each supported Region: 1	<a href="#">Yes</a>	The maximum number of concurrent simulation jobs you can run in this account in the current Region.
Fleets	Each supported Region: 20	<a href="#">Yes</a>	The maximum number of fleets you can create in this account in the current Region.
GPU Simulation Job Creation Rate Per Minute	Each supported Region: 2	No	The maximum number of GPU simulation job you can create in this account in the current Region per minute.

Name	Default	Adjustable	Description
Minimum batch timeout	Each supported Region: 5	No	The minimum timeout in minutes that you can specify for a simulation job batch.
Minimum simulation duration	Each supported Region: 5	No	The minimum duration in minutes that you can specify for a simulation job.
Robot applications	Each supported Region: 40	<a href="#">Yes</a>	The maximum number of robot applications you can create in this account in the current Region.
Robots	Each supported Region: 100	<a href="#">Yes</a>	The maximum number of robots you can create in this account in the current Region.
Robots per fleet	Each supported Region: 100	<a href="#">Yes</a>	The maximum number of robots you can register to a fleet.
Simulation Job Creation Rate Per Minute	us-east-1: 10 us-west-2: 10 Each of the other supported Regions: 5	No	The maximum number of simulation job you can create in this account in the current Region per minute.

Name	Default	Adjustable	Description
Simulation applications	Each supported Region: 40	<a href="#">Yes</a>	The maximum number of simulation applications you can create in this account in the current Region.
Simulation duration	Each supported Region: 14	No	The maximum duration in days that a simulation job can run for including restarts.
Simulation job requests per batch	Each supported Region: 20	<a href="#">Yes</a>	The maximum number of simulation job requests that can be submitted in a StartSimulationJob Batch call
Source size	Each supported Region: 5 Gigabytes	No	The maximum size (in GB) for any source of robot application or simulation application.
Versions per robot application	Each supported Region: 40	<a href="#">Yes</a>	The maximum number of versions you can create for a Robot Application.
Versions per simulation application	Each supported Region: 40	<a href="#">Yes</a>	The maximum number of versions you can create for a Simulation Application.
World Templates Per Account	Each supported Region: 40	<a href="#">Yes</a>	The maximum number of world templates that you can create in this account in this region.

Name	Default	Adjustable	Description
Worlds Per Export Job	Each supported Region: 1	No	The maximum number of worlds in a world export job request.
Worlds Per Generation Job	Each supported Region: 50	No	The maximum number of worlds in a world generation job request.



# Troubleshooting AWS RoboMaker

The following sections provide troubleshooting advice for errors and issues that you might encounter when using AWS RoboMaker simulation, IDEs, or Simulation WorldForge. If you find an issue that is not listed here, use the *Provide feedback* link at the bottom of this page to report it.

You can find logs for your simulation job in the [CloudWatch Logs console](#). By default, AWS RoboMaker uploads logs for simulation jobs as they are generated for your application. If `streamOutputToCloudWatch` is set to `True`, the same behavior is applied to tools. You can also configure uploads by [Adding a custom upload configuration](#) in your simulation job.

For more information, see [Logging and monitoring in AWS RoboMaker](#).

## Sections

- [Simulation jobs](#)
- [Simulation WorldForge](#)

## Simulation jobs

**Issue:** Your simulation job failed.

Use the questions below to help identify the root cause and then take the suggested action.

### Are your Amazon S3 resources in the same region as AWS RoboMaker?

Your robot application, simulation application, and output locations must be in the same Region as AWS RoboMaker. Verify your application sources and simulation job output locations.

### Did your robot application exit abnormally?

There was a problem setting up your robot application for simulation. Review the robot application logs for the simulation job in Amazon CloudWatch.

Logs are accessed from the simulation job detail screen. Select **Logs**, and then select a log stream. To look for specific issues, use the filter. For example, **WARNING** or **ERROR**.

### Is your application missing an .so file?

If your application crashed, it might be missing a dependent *shared object* (.so) file. Extract your application bundle in your environment and verify that the shared object libraries you need are in `/usr/lib` or `/usr/local/lib`. Make sure the dependency is added to your package .xml file.

### Did you use the ARN of your role with the AWS CLI?

When you call `create-simulation-job` from the AWS CLI, use the full Amazon Resource Name (ARN) of the role and not just the role name.

### Does your role have a trust policy for AWS RoboMaker?

If you are passing the full Amazon Resource Name (ARN) of the IAM role when you call `create-simulation-job` from the AWS CLI, your trust policy might have insufficient privileges. Check the role to make sure it has a trust relationship with `robomaker.amazonaws.com` as follows.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "robomaker.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "account#" // Account where
          the simulation job resource is created
        },
        "StringEquals": {
          "aws:SourceArn":
            "arn:aws:robomaker:region:account#:simulation-job/*"
        }
      }
    }
  ]
}
```

Condition keys prevent an AWS service from being used as a [confused deputy](#) during transactions between services. See [SourceAccount](#) and [SourceArn](#) for additional information about condition keys.

See [Modifying a role](#) for more information about viewing role access and adding a trust policy to an IAM role.

### Does your role have permissions to publish to Amazon S3?

If you specify an output Amazon S3 bucket for a simulation job, your role must have write permissions to the bucket. Update your trust policy to include write permissions. The following example trust policy adds read, list, and write permissions to an Amazon S3 bucket.

```
{
  "Action": "s3:ListBucket",
  "Resource": [
    "my-bucket/*"
  ],
  "Effect": "Allow"
}, {
  "Action": [
    "s3:Get*",
    "s3:List*"
  ],
  "Resource": [
    "my-bucket/*"
  ],
  "Effect": "Allow"
}, {
  "Action": "s3:Put*",
  "Resource": [
    "my-bucket/*"
  ],
  "Effect": "Allow"
}
```

### Does your role have permission to publish to CloudWatch?

Update the permissions policies of your IAM role with CloudWatch access.

```
{
  "Effect": "Allow",
  "Action": [
    "logs:CreateLogGroup",
    "logs:CreateLogStream",
    "logs:PutLogEvents",
    "logs:DescribeLogStreams"
  ],
  "Resource": "*"
}
```

### Does your application have a mismatched entity tag?

The entity tag (ETag) is a hash of the Amazon S3 object provided while creating the simulation. The ETag reflects changes only to the contents of an object, not its metadata. If you change the content of the robot application or simulation bundle in Amazon S3 before AWS RoboMaker has consumed it, there will be a version mismatch.

To resolve this, create a new robot application or simulation application version and provide the key location for the updated application bundle. For more information, see [Creating a robot application version](#) or [Creating a simulation application version](#).

### **Is your subnet elastic network interface (ENI) limit exceeded?**

AWS RoboMaker uses one ENI for each concurrent simulation job in the subnet in which the simulation job is run. Each of these must be assigned an IP address. To resolve this, you can:

Delete unused ENIs to free up IP addresses in the subnet. To delete an unused ENI, see [Deleting a network interface](#).

Using the AWS Management Console, request a [service limit increase](#) for ENIs in a specific AWS Region.

### **Is the launch command properly configured?**

A simulation job can take a few minutes to launch if your simulation is complex or if the container image is large. If AWS RoboMaker spends more than 25 minutes preparing the simulation job, there might be a problem with the launch command. Cancel the job and then create a new simulation job. If the problem persists, contact AWS support.

You can use CloudWatch Logs to check the simulation and robot application execution logs for errors. You can also add a customized tool of terminal to connect and troubleshoot the running simulation job.

### **Are your subnets in zones that AWS RoboMaker supports?**

Provide subnets in two of the AWS availability zones supported by AWS RoboMaker. API response contains a list of supported AWS availability zones.

### **Are your world file model references correct?**

Use CloudWatch Logs to verify all of the models in your world file are correct. If a model cannot be located, you will see the following errors.

```
[WIn] [ModelDatabase.cc:340] Getting models from[http://models.gazebosim.org/]. This
may take a few seconds.
[WIn] [ModelDatabase.cc:212] Unable to connect to model database using [http://
models.gazebosim.org//database.config]. Only locally installed models will be
available.
```

```
[Err] [ModelDatabase.cc:414] Unable to download model[model://model_name]
[Err] [SystemPaths.cc:429] File or path does not exist[""]
Error [parser.cc:581] Unable to find uri[model://model_name]
```

## Simulation WorldForge

**Issue:** *My world generation job failed.*

If your world generation job did not complete, make sure your world count, `floorplanCount` \* `interiorCountPerFloorplan`, is greater than 1 and less than 50.

**Issue:** *Why did my world export job fail?*

Use the questions below to help identify the root cause and then take the suggested action.

### Do you have a trust policy for AWS RoboMaker?

If you are passing the full Amazon Resource Name (ARN) of the IAM role when you call `create-world-export-job` from the AWS CLI, your trust policy might have insufficient privileges. Check the role to make sure it has a trust relationship with `robomaker.amazonaws.com` as follows.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "robomaker.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "account#" // Account where
          the simulation job resource is created
        },
        "StringEquals": {
          "aws:SourceArn":
            "arn:aws:robomaker:region:account#:simulation-job/*"
        }
      }
    }
  ]
}
```

Condition keys prevent an AWS service from being used as a [confused deputy](#) during transactions between services. See [SourceAccount](#) and [SourceArn](#) for additional information about condition keys.

### Does your role have permissions to publish to Amazon S3?

If you specify an output Amazon S3 bucket for an export job, your role must have permissions to the bucket. Update your trust policy to include the following permissions:

```
{
  "Effect": "Allow",
  "Action": [
    "s3:AbortMultipartUpload",
    "s3:GetObject",
    "s3:PutObject"
  ],
  "Resource": "my-bucket"
}
```

### Did you modify or remove the bucket specified for the export job?

If you update your bucket during the export job, you may get a `ResourceNotFound` error from export job.

**Issue:** *There's a problem with the world image.*

Use the questions below to help identify the root cause and then take the suggested action.

### Why is there no door in my doorway?

You can only add doors using a Version 2 template or later. You can update a Version 1 template to a later version. For more information, see [Simulation world template versions, features, and changes](#).

Because AWS RoboMaker Simulation WorldForge creates worlds that are unique and random, the door configuration that you've specified might not exist in the world when you generate it. For example, you might specify a door between a living room and a kitchen in your template, but there might be an open wall between those rooms. Because there's an open wall instead of a doorway, you wouldn't be able to add a door there.

### Why does my door block the entrance to my room?

The door blocking the entrance to a room is a circumstance that you can use to challenge your robots. To create a world that doesn't present this challenge to your robots, you can do one of the following:

Generate another world from your world template. The door generated in the new world might not block the entrance.

Change the open percentage of the door in the world template.

### **Why are the walls in my world image shorter than the walls in my simulation job or exported world?**

AWS RoboMaker allows you to see your Simulation WorldForge worlds without it being obscured by walls by truncating the walls in the world image. The walls have the height that you specify in your world template in the worlds that you create.

For worlds generated by the Version 2 template or later, the door models are not truncated in the world images. The height of the doors in the world image are the same as the height of the doors in the worlds you create.

# Support policy

The following sections describe support changes for AWS RoboMaker.

## Support changes: December 15, 2022

On June 27, 2022, we migrated our development environment capabilities to AWS Cloud9 and ended support for the AWS RoboMaker development environment feature. As of December 15, 2022, you can't access previously created development environments in AWS RoboMaker.

### Why did we end support?

AWS Cloud9 provides an improved development experience with new functionality, flexibility, and expanded Region support. With AWS Cloud9, you have Amazon Linux and Ubuntu platform options, cost-savings settings, and the flexibility to use and configure any robot and simulation software of your choice. For more information about getting started with AWS Cloud9, see the [AWS Cloud9 User Guide](#).

### New development environments in AWS Cloud9

AWS Cloud9 provides you access to the same development environment features you had in AWS RoboMaker. Use the AWS Cloud9 console to create development environments and take advantage of new functionality. To learn how to configure your AWS Cloud9 development environments to build and simulate your robot and simulation applications, see [Build and Simulate Robotics Applications in AWS Cloud9](#) in the AWS Robotics blog.

### Existing AWS RoboMaker development environments

Access environments launched prior to June 27, 2022 through the AWS Cloud9 console. To retain NICE DCV functionality, see the [Configuring NICE DCV](#) section in *Build and Simulate Robotics Applications in AWS Cloud9*.

## Support changes: May 2, 2022

On May 2, 2022, we removed existing robots, fleets and deployments job resources from your account. The deletion of these AWS RoboMaker application deployment resources does not affect your physical hardware. You can continue to deploy your applications to physical robots through other means such as AWS IoT Greengrass Version 2.



The following API actions have been deprecated:

- CancelDeploymentJob
- DeleteFleet
- DeleteRobot
- DeregisterRobot
- DescribeDeploymentJob
- DescribeFleet
- DescribeRobot
- ListDeploymentJobs
- ListFleets
- ListRobots
- SyncDeploymentJob

## Recommended Actions

We recommend that you take the following actions.

1. If you haven't already done so, migrate your robot and simulation applications to a supported container image. To learn more about how to move your robot applications, simulation applications, and simulation jobs to a Docker-based workflow, see [Migrating ROS applications to containers](#).
2. Migrate to AWS IoT Greengrass Version 2. For additional information, see the [AWS IoT Greengrass Version 2 Developer Guide](#). To learn about AWS IoT Greengrass Version 2 deployments, see the following AWS Robotics blog posts:
  - [Deploy and Manage ROS Robots with AWS IoT Greengrass Version 2 and Docker](#)
  - [Deploying ROS applications as Snaps with AWS IoT Greengrass Version 2](#)

## Support changes: March 15, 2022

On March 15, 2022, we made two changes to AWS RoboMaker simulations which may have affected your simulation jobs.

1. We migrated AWS RoboMaker simulation jobs to container images. In other words, you must migrate your robot and simulation applications to a supported container image.

2. We stopped vending preinstalled Robot Operating Software (ROS), Gazebo, and Ubuntu base images in AWS RoboMaker simulations. You can continue to run ROS and Gazebo-based simulations, but the mechanism for doing so has changed. You must update to the **General** software suite for your robot applications and **Simulation runtime** software suite for your simulation applications.

### Why did we make these changes?

AWS RoboMaker now supports expanded configuration for any robot and simulation software, so you can use and configure any robot and simulation software of your choice while running simulations. You can still use ROS in your robot applications, including older versions such as ROS Kinetic or newer versions such as ROS2 Galactic, but you can also run custom robot applications without using ROS. Additionally, you can now use the simulation software of your choice while running simulations in AWS RoboMaker.

### What happens now?

Existing robot and simulation applications which have not been migrated are restricted from usage, but can be migrated to the supported software suites and container images. Any simulation jobs and simulation job batches launched prior to March 15, 2022, and whose duration extends past March 15, 2022, will continue to run until completed.

## Support ended: January 31, 2022

On January 31, 2022, we migrated deployment capabilities to AWS IoT Greengrass Version 2 and ended support for the AWS RoboMaker application deployment feature. AWS IoT Greengrass Version 2 supports existing application deployment capabilities and provides new functionality and an improved deployment experience. As of January 31, 2022, you can no longer create new application deployment resources (robots, fleets and deployments) in AWS RoboMaker.

The following API actions have been deprecated:

- `CreateDeploymentJob`
- `CreateFleet`
- `CreateRobot`
- `RegisterRobot`

# Support ended: April 30, 2021

Starting April 30, 2021, you can no longer create new ROS Kinetic, Gazebo 7.1, ROS Dashing or Ubuntu 16.04 resources in AWS RoboMaker. However, any existing AWS RoboMaker resources remain in your account. If you do not upgrade, functionality of your ROS Kinetic, Gazebo 7.1, ROS Dashing and Ubuntu 16.04 resources within AWS RoboMaker features might change or even break.

The following software suite combinations are deprecated:

- ROS Kinetic, Gazebo 7.1, Ubuntu 16.04
- ROS Kinetic, Gazebo 9, Ubuntu 16.04
- ROS Dashing, Gazebo 9, Ubuntu 16.04

The deprecation affected the following areas:

- **AWS Cloud9 integrated development environments (IDEs)**
  - You have access to all existing ROS Kinetic and ROS Dashing–based IDEs. You can continue to work within the IDE. The successful execution of the build and bundle process is not guaranteed.
  - You can't create new ROS Kinetic and ROS Dashing–based IDEs.
- **Robot and simulation applications**
  - You cannot create new ROS Kinetic and ROS Dashing–based robot applications.
  - You cannot create new simulation applications with ROS Kinetic with Gazebo 7.1, ROS Kinetic with Gazebo 9, or ROS Dashing with Gazebo 9.
  - You cannot create a new versions of existing robot or simulation applications using the deprecated ROS and Gazebo versions.
- **Simulation jobs and simulation batches**
  - You cannot create new simulation jobs with robot applications and simulation applications using Kinetic, Dashing or Gazebo 7.1.

Simulation jobs launched prior to the deprecation date and whose duration extends past the deprecation date continue to run successfully until completed. With a maximum simulation job duration of 14 days, these jobs can run for a maximum of 14 days post deprecation.

- **Deployment jobs**
  - You cannot create a deployment job for Kinetic or Dashing–based robot applications.

- **Sample applications and cloud extensions**

- Cloud extensions are no longer supported in ROS Kinetic and ROS Dashing–based applications. While you can install the cloud extensions into ROS Kinetic and ROS Dashing workspaces, they may or may not work.
- You can no longer select ROS Kinetic or ROS Dashing as the ROS distribution to launch a sample application. The sample applications may still be downloaded into existing ROS Kinetic and ROS Dashing IDEs. However, they are no longer supported and may break.

## Document history

The following table shows when features and deprecations were applied to the AWS RoboMaker service and documentation.

Change	Description	Date
Public ECR for AWS RoboMaker Simulation	Added support for the use of public ECR images. Create AWS RoboMaker robot and simulation applications without creating a private ECR repository.	1/26/2023
IDE deprecation	Deprecated the AWS RoboMaker IDE	12/15/2022
Preinstalled RUG deprecation	Deprecated preinstalled Robot Operating Software (ROS), Ubuntu, and Gazebo base images and migrated AWS RoboMaker simulation jobs to container images.	3/15/2022
Application deployment deprecation	Deprecated application deployment for AWS RoboMaker.	1/31/2022
Cloud extensions deprecation	Deprecated cloud extensions for AWS RoboMaker.	1/31/2022
Samples deprecation	Deprecated self-driving reinforcement, navigation, person detection, and voice command samples for AWS RoboMaker.	5/15/2020

Change	Description	Date
Support for tags	Added support for tags to many AWS RoboMaker resources.	1/24/2019
New service and guide	The initial release of AWS RoboMaker and the <i>AWS RoboMaker Developer Guide</i> .	11/27/2018