Framework

RealityKit

Simulate and render 3D content for use in your augmented reality apps.

iOS 13.0+ | iPadOS 13.0+ | Mac Catalyst 13.1+ | macOS 10.15+ | tvOS 26.0+ | visionOS 1.0+

Overview

RealityKit provides high-performance 3D simulation and rendering capabilities you can use to create apps with 3D or augmented reality (AR) for iOS, iPadOS, macOS, tvOS, and visionOS. RealityKit is an AR-first 3D framework that leverages ARKit to seamlessly integrate virtual objects into the real world.

Use RealityKit's rich functionality to create compelling augmented reality (AR) experiences:

- Create and import full RealityKit scenes with models, animations, and Spatial Audio by using Reality Composer Pro for visionOS.

- Build or modify scenes at runtime by adding 3D models, shape primitives, and sounds from code.
- Have virtual objects interact with objects in the real world.
- Animate objects, both manually and with physics simulations.
- Respond to user input and changes in a person's surroundings.
- Synchronize across devices and use SharePlay to enable group AR experiences.

Topics

Essentials

Understanding the modular architecture of RealityKit

Learn how everything fits together in RealityKit.

{} Building an immersive experience with RealityKit

Use systems and postprocessing effects to create a realistic underwater scene.

`class Entity`

An element of a RealityKit scene to which you attach components that provide appearance and behavior characteristics for the entity.

`protocol Component`

A representation of a geometry or a behavior that you apply to an entity.

Presentation

Views and attachments

Bring RealityKit content into your app with views and renderers.

Presentation UI

Control your app's content and how people can interact with it.

Scene management and logic

Scenes

The context that holds all RealityKit entities.

Systems

Apply behaviors and physical effects to the entities in a RealityKit scene.

Events

Respond to things happening in your RealityKit scene by subscribing to specific event types.

Entity actions

Create simple, reusable actions that can change your app state, RealityKit scene, or animate an entity.

Asset creation

Swift Splash

Use RealityKit to create an interactive ride in visionOS.

Diorama

Design scenes for your visionOS app using Reality Composer Pro.

Composing interactive 3D content with RealityKit and Reality Composer Pro

Build an interactive scene using an animation timeline.

Presenting an artist's scene

Display a scene from Reality Composer Pro in visionOS.

Reality Composer

A visual editor for RealityKit AR scenes.

Object capture

Create 3D objects from a series of photographs using photogrammetry.

USD

An efficient and scalable way to represent 3D scenes.

Scene content

Hello World

Use windows, volumes, and immersive spaces to teach people about the Earth.

Enabling video reflections in an immersive environment

Create a more immersive experience by adding video reflections in a custom environment.

{} Creating a spatial drawing app with RealityKit

Use low-level mesh and texture APIs to achieve fast updates to a person's brush strokes by integrating RealityKit with ARKit and SwiftUI.

{} Generating interactive geometry with RealityKit

Create an interactive mesh with low-level mesh and low-level texture.

{} Combining 2D and 3D views in an immersive app

Use attachments to place 2D content relative to 3D content in your visionOS app.

{} Transforming RealityKit entities using gestures

Build a RealityKit component to support standard visionOS gestures on any entity.

{} Responding to gestures on an entity

Respond to gestures performed on RealityKit entities using input target and collision components.

Models and meshes

Display virtual objects in your scene with mesh-based models.

Materials, textures, and shaders

Apply textures to the surface of your scene's 3D objects to give each object a unique appearance.

Anchors

Lock virtual content to the real world.

Lights and cameras

Control the lighting and point of view for a scene.

Content synchronization

Synchronize the contents of entities locally or across the network.

Audio

Create personalized and realistic spatial audio experiences.

Videos

Present videos in your RealityKit experiences.

Images

Present images and spatial scenes in your RealityKit experiences.

Game development

- Gaming sample code projects
  Explore a collection of projects relating to game development.
- Entity animations
  Dynamically move, rotate, and scale entities at runtime.
- Character control, skeletons, and inverse kinematics
  Direct the movements and animation of models.

Physics simulation

- Collision detection
  Determine when entities collide with each other or the environment.
- Simulations and motion
  Simulate physical interactions between entities or systems.
- Force effects
  Control the movement of virtual objects with forces.
- Physics joints and pins
  Simulate joint physics that connect virtual objects.

Performance improvements

- Improving the Performance of a RealityKit App
  Measure CPU and GPU utilization to find ways to improve your app's performance.
- Reducing GPU Utilization in Your RealityKit App
  Prevent the GPU from limiting your app's frame rate by reducing the complexity of your render.
- Reducing CPU Utilization in Your RealityKit App
  Target specific CPU metrics with adjustments to your app and its content.
- Construct an immersive environment for visionOS
  Build efficient custom worlds for your app.

Passing Metal command objects around your application

Build a system that creates and passes Metal command objects to entities dispatching Metal compute shaders.

`protocol` Resource

A shared resource you use to configure a component, like a material, mesh, or texture.

Articles

{} Rendering stereoscopic video with RealityKit

Render stereoscopic video in visionOS with RealityKit.