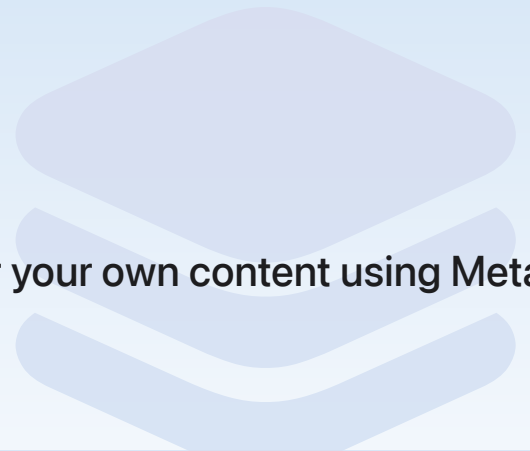


Framework

Compositor Services

Take control of the drawing environment and render your own content using Metal.

visionOS 1.0+



Overview

Compositor Services lets you draw directly to the device's displays using Metal and your own rendering engine. Use this framework to create a fully immersive scene in your app that doesn't require integration with the person's surroundings.

When you present an immersive space with [CompositorLayer](#) content from your app, you receive a [LayerRenderer](#) with the information you need to set up your Metal drawing environment. Use the layer to start your rendering loop and deliver successive frames of content. The layer provides the timing information you need to deliver frames at the refresh rate of the display. It also provides the Metal textures and other information that you need to draw content on the device displays.

For more information about how to draw your app's content using Metal, see [Metal](#).

Topics

App integration

 Drawing fully immersive content using Metal

Create a fully immersive experience in visionOS using a custom Metal-based rendering engine.

 Interacting with virtual content blended with passthrough

Present a mixed immersion style space to draw content in a person's surroundings, and choose how upper limbs appear with respect to rendered content.

`{}` Rendering hover effects in Metal immersive apps

Change the appearance of a rendered onscreen element when a player gazes at it.

`struct CompositorLayer`

A type that you use with an immersive space to display fully immersive content using Metal.

`protocol CompositorLayerConfiguration`

An interface for specifying the texture configurations and rendering behaviors to use with your Metal rendering engine.

`struct DefaultCompositorLayerConfiguration`

A type that configures the layer with the default texture configurations and rendering behaviors for the current device.

Render-loop setup

`class LayerRenderer`

A type that provides the Metal types and timing information you need to draw your content.

`struct Frame`

A type that provides access to the timing information and data types you need to render a single frame of content.

Drawing environment

`struct Drawable`

A type that provides the textures and information you need to draw a frame of content.

`struct View`

A type that provides information on how to render content into the frame's textures.

Errors

`enum LayerRendererConfigurationError`

Errors that can occur when configuring layers.

Articles

📄 Controlling Metal rendering immersion level

Enable flexible immersive rendering by supporting progressive immersion when rendering your Metal content.

Structures

`struct TextureTopology`

A type that specifies the organization of one of the drawable's textures.

Type Aliases

`typedef cp_drawable_array_t`

An opaque type that contains the drawable types and other information you need to set up your render pipeline.

`typedef cp_hover_effect_t`

An opaque type that describes a hover effect of the tracking area.