Protocol

MTL4CommandBuffer

Records a sequence of GPU commands.

iOS 26.0+ | iPadOS 26.0+ | Mac Catalyst 26.0+ | macOS 26.0+ | tvOS 26.0+ | visionOS 26.0+

```
protocol MTL4CommandBuffer : NSObjectProtocol
```

Mentioned in

Understanding the Metal 4 core API

Topics

Instance Properties

`var device: any MTLDevice`

Returns the GPU device that this command buffer belongs to.

Required

`var label: String?`

Assigns an optional label with this command buffer.

Required

Instance Methods

`func beginCommandBuffer(allocator: any MTL4CommandAllocator)`

Prepares a command buffer for encoding.

Required

```
func beginCommandBuffer(allocator: any MTL4CommandAllocator, options: MTL4CommandBufferOptions)
```

Prepares a command buffer for encoding with additional options.

Required

```
func endCommandBuffer()
```

Closes a command buffer to prepare it for submission to a command queue.

Required

```
func makeComputeCommandEncoder() -> (any MTL4ComputeCommandEncoder)?
```

Creates a compute command encoder.

Required

```
func makeMachineLearningCommandEncoder() -> (any MTL4MachineLearningCommandEncoder)?
```

Creates a machine learning command encoder.

Required

```
func makeRenderCommandEncoder(descriptor: MTL4RenderPassDescriptor, options: MTL4RenderEncoderOptions) -> (any MTL4RenderCommandEncoder)?
```

Creates a render command encoder from a render pass descriptor with additional options.

Required

```
func popDebugGroup()
```

Pops the latest string from the stack of debug groups for this command buffer.

Required

```
func pushDebugGroup(String)
```

Pushes a string onto a stack of debug groups for this command buffer.

Required

```
func resolveCounterHeap(any MTL4CounterHeap, range: Range<Int>, buffer: MTL4BufferRange, fenceToWait: (any MTLFence)?, fenceToUpdate: (any MTLFence)?)
```

Encodes a command that resolves an opaque counter heap into a buffer.

```
func useResidencySet(any MTLResidencySet)
```

Applies a residency set to a command buffer.

Required

`func useResidencySets([any MTLResidencySet])`

Applies multiple residency sets to a command buffer.

`func writeTimestamp(counterHeap: any MTL4CounterHeap, index: Int)`

Writes a GPU timestamp into the given counter heap.

Required

Relationships

Inherits From

`NSObjectProtocol`

See Also

Submitting work to a GPU with Metal 4

`protocol MTL4CommandQueue`

An abstraction representing a command queue that you use commit and synchronize command buffers and to perform other GPU operations.

`class MTL4CommandQueueDescriptor`

Groups together parameters for the creation of a new command queue.

`struct MTL4CommandQueueError`

`enum Code`

Enumeration of kinds of errors that committing an array of command buffers instances can produce.

`let MTL4CommandQueueErrorDomain: String`

`class MTL4CommandBufferOptions`

Options to configure a command buffer before encoding work into it.

`protocol MTL4CommandEncoder`

An encoder that writes GPU commands into a command buffer.

`struct` MTL4RenderEncoderOptions

Custom render pass options you specify at encoder creation time.

`protocol` MTL4ArgumentTable

Provides a mechanism to manage and provide resource bindings for buffers, textures, sampler states and other Metal resources.

`class` MTL4ArgumentTableDescriptor

Groups parameters for the creation of a Metal argument table.

`protocol` MTL4CommandAllocator

Manages the memory backing the encoding of GPU commands into command buffers.

`class` MTL4CommandAllocatorDescriptor

Groups together parameters for creating a command allocator.

`class` MTL4CommitOptions

Represents options to configure a commit operation on a command queue.

`protocol` MTL4CommitFeedback

Describes an object containing debug information from Metal to your app after completing a workload.

`typealias` MTL4CommitFeedbackHandler

Defines the block signature for a callback Metal invokes to provide your app feedback after completing a workload.