Documentation Language: **Swift**

Metal / MTLComputePipelineDescriptor

Class

MTLComputePipelineDescriptor

An instance describing the desired GPU state for a kernel call in a compute pass.

iOS 9.0+ | iPadOS 9.0+ | Mac Catalyst 13.1+ | macOS 10.11+ | tvOS 9.0+ | visionOS 1.0+

```
class MTLComputePipelineDescriptor
```

Mentioned in

Compiling and linking Metal dynamic libraries

Overview

> **Important**
>
> Before creating a pipeline state, set the `computeFunction` property on your descriptor instance. This property tells the GPU which kernel to run.

A pipeline descriptor provides information necessary for creating an `MTLComputePipelineState` instance.

Topics

Configuring the compute execution environment

```
var computeFunction: (any MTLFunction)?
```

The compute kernel the pipeline calls.

```
var threadGroupSizeIsMultipleOfThreadExecutionWidth: Bool
```

A Boolean value that indicates whether the threadgroup size is always a multiple of the thread execution width.

```
var maxTotalThreadsPerThreadgroup: Int
```

The maximum number of threads in a threadgroup that you can dispatch to the compute function.

```
var maxCallStackDepth: Int
```

The maximum recursive call depth for dynamic library, visible, and intersection functions.

Configuring compute pass inputs

```
var stageInputDescriptor: MTLStageInputOutputDescriptor?
```

The organization of input and output data for the next kernel call.

```
class MTLAttributeDescriptor
```

A descriptor of an argument's format and where its data is in memory.

```
class MTLAttributeDescriptorArray
```

An array of attribute descriptor objects.

```
class MTLBufferLayoutDescriptor
```

A description of how a compute function fetches input data for an attribute.

```
class MTLBufferLayoutDescriptorArray
```

An array of buffer layout descriptor objects.

Configuring buffer mutability

```
var buffers: MTLPipelineBufferDescriptorArray
```

The buffer mutability options to apply to the next kernel call.

Identifying the pipeline state object

```
var label: String?
```

A string that identifies the instance.

Configuring indirect command buffers

`var supportIndirectCommandBuffers: Bool`

A Boolean value that indicates whether you can encode commands that reference the pipeline state object into an indirect command buffer.

Configuring shader validation

`var shaderValidation: MTLShaderValidation`

A value that enables or disables shader validation for the pipeline.

Reset to defaults

`func reset()`

Resets all compute pipeline descriptor properties to their default values.

Loading dynamic libraries to link at runtime

`var preloadedLibraries: [any MTLDynamicLibrary]`

The dynamic libraries that contain precompiled shader functions you want to link.

~~`var insertLibraries: [any MTLDynamicLibrary]?`~~

The dynamic libraries that contain precompiled shader functions you want to link.

Deprecated

Setting callable functions

`var linkedFunctions: MTLLinkedFunctions?`

The functions with available function pointers for the next kernel call.

Loading binary archives

`var supportAddingBinaryFunctions: Bool`

A Boolean value that indicates whether you can use the pipeline to create new pipelines by adding binary functions to its callable functions list.

`var binaryArchives: [any MTLBinaryArchive]?`

The binary archives that contain any precompiled shader functions to link.

Instance Properties

`var requiredThreadsPerThreadgroup: MTLSize`

Relationships

Inherits From

`NSObject`

Conforms To

`CVarArg`
`CustomDebugStringConvertible`
`CustomStringConvertible`
`Equatable`
`Hashable`
`NSCopying`
`NSObjectProtocol`

See Also

Configuring a compute pipeline state

`class MTL4ComputePipelineDescriptor`

Describes a compute pipeline state.

`protocol MTLComputePipelineState`

An interface that represents a GPU pipeline configuration for running kernels in a compute pass.

`class MTLStageInputOutputDescriptor`

A description of the input and output data of a function.

`class MTLPipelineBufferDescriptor`

The mutability options for a buffer that a render or compute pipeline uses.

`class MTLPipelineBufferDescriptorArray`

An array of pipeline buffer descriptors.

`struct MTLPipelineOption`

Options that determine how Metal prepares the pipeline.