

[Metal](#) / MTL4Compiler

Protocol


MTL4Compiler

A abstraction for a pipeline state and shader function compiler.

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

```
protocol MTL4Compiler : NSObjectProtocol, Sendable
```

Mentioned in

 Using the Metal 4 compilation API

Topics

Instance Properties

`var device: any MTLDevice`

Returns the device that this compiler belongs to.

Required

`var label: String?`

Returns the optional label you specify at creation time.

Required

`var pipelineDataSetSerializer: (any MTL4PipelineDataSetSerializer)?`

Returns the pipeline data set serializer into which this compiler stores data for all pipelines it creates.

Required

Instance Methods

```
func makeBinaryFunction(descriptor: MTL4BinaryFunctionDescriptor,  
compilerTaskOptions: MTL4CompilerTaskOptions?) throws -> any MTL4Binary  
Function
```

Creates a new binary visible or intersection function synchronously.

```
func makeBinaryFunction(descriptor: MTL4BinaryFunctionDescriptor,  
compilerTaskOptions: MTL4CompilerTaskOptions?) async throws -> any  
MTL4BinaryFunction
```

Creates a new binary visible or intersection function asynchronously.

```
func makeComputePipelineState(descriptor: MTL4ComputePipelineDescriptor  
, dynamicLinkingDescriptor: MTL4PipelineStageDynamicLinkingDescriptor?,  
compilerTaskOptions: MTL4CompilerTaskOptions?) async throws -> any  
MTLComputePipelineState
```

Creates a new compute pipeline state asynchronously.

```
func makeComputePipelineState(descriptor: MTL4ComputePipelineDescriptor  
, dynamicLinkingDescriptor: MTL4PipelineStageDynamicLinkingDescriptor?,  
compilerTaskOptions: MTL4CompilerTaskOptions?) throws -> any MTLCompute  
PipelineState
```

Creates a new compute pipeline state object synchronously.

```
func makeDynamicLibrary(library: any MTLLibrary) throws -> any  
MTLDynamicLibrary
```

Creates a new dynamic library from a library containing Metal IR code synchronously.

Required Default implementation provided.

```
func makeDynamicLibrary(url: URL) throws -> any MTLDynamicLibrary
```

Creates a new dynamic library from the contents of a file at an URL location synchronously.

Required Default implementation provided.

```
func makeLibrary(descriptor: MTL4LibraryDescriptor) throws -> any  
MTLLibrary
```

Creates a new Metal library synchronously.

Required Default implementation provided.

```
func makeMachineLearningPipelineState(descriptor: MTL4MachineLearning  
PipelineDescriptor) async throws -> any MTL4MachineLearningPipeline  
State
```

Creates a new machine learning pipeline state asynchronously.

```
func makeMachineLearningPipelineState(descriptor: MTL4MachineLearningPipelineDescriptor) throws -> any MTL4MachineLearningPipelineState
```

Creates a new ML pipeline state with descriptor.

```
func makeRenderPipelineState(descriptor: MTL4PipelineDescriptor,
dynamicLinkingDescriptor: MTL4RenderPipelineDynamicLinkingDescriptor?,
compilerTaskOptions: MTL4CompilerTaskOptions?) async throws -> any
MTLRenderPipelineState
```

Creates a new render pipeline state asynchronously.

```
func makeRenderPipelineState(descriptor: MTL4PipelineDescriptor,
dynamicLinkingDescriptor: MTL4RenderPipelineDynamicLinkingDescriptor?,
compilerTaskOptions: MTL4CompilerTaskOptions?) throws -> any MTLRender
PipelineState
```

Creates a new render pipeline state synchronously.

```
func makeRenderPipelineStateBySpecialization(descriptor: MTL4Pipeline
Descriptor, pipeline: any MTLRenderPipelineState) throws -> any
MTLRenderPipelineState
```

Creates a new render pipeline state from another, previously unspecialized, pipeline state.

```
func makeRenderPipelineStateBySpecialization(descriptor: MTL4Pipeline
Descriptor, pipeline: any MTLRenderPipelineState) async throws -> any
MTLRenderPipelineState
```

Creates a new render pipeline state from another, previously unspecialized, pipeline state

Relationships

Inherits From

NSObjectProtocol, Sendable, SendableMetatype

See Also

Shader compilation

⌵ Metal libraries

Compile and manage Metal libraries from the command line.

⌵ Metal dynamic libraries

Create a single Metal library containing reusable code to reduce library size and avoid repeated shader compilation at runtime.

⌵ Metal binary archives

Distribute precompiled GPU-specific binaries as part of your app to avoid runtime compilation of Metal shaders.

`class MTL4CompilerDescriptor`

Groups together properties for creating a compiler context.

`class MTL4CompilerTaskOptions`

The configuration options that control the behavior of a compilation task for a Metal 4 compiler instance.

`enum MTL4CompilerTaskStatus`

Represents the status of a compiler task.

`protocol MTL4Archive`

A read-only container that stores pipeline states from a shader compiler.

`protocol MTL4BinaryFunction`

Represents a binary function.

`class MTL4BinaryFunctionDescriptor`

Base interface for other function-derived interfaces.

`struct MTL4BinaryFunctionOptions`

Options for configuring the creation of binary functions.

`class MTL4PipelineStageDynamicLinkingDescriptor`

Groups together properties to drive the dynamic linking process of a pipeline stage.