

# Namespace CatSweeper.Core

## Classes

[BootstrapEndState](#)

[GameEntry](#)

[GameEntry.MethodName](#)

Cached StringNames for the methods contained in this class, for fast lookup.

[GameEntry.PropertyName](#)

Cached StringNames for the properties and fields contained in this class, for fast lookup.

[GameEntry.SignalName](#)

Cached StringNames for the signals contained in this class, for fast lookup.

[GameState](#)

[GameStateMachine](#)

[GameplayInitState](#)

[InfoLoadState](#)

[LoadStageState](#)

[LoadStageState.InitParam](#)

[LocalLoadState](#)

[LoginState](#)

[LoginState.Param](#)

[UILoadState](#)

[UserDataLoadState](#)

## Enums

[GameStatId](#)

# Class BootstrapEndState

Namespace: [CatSweeper.Core](#)

Assembly: CatSweeper.dll

```
public class BootstrapEndState : GameState, IDisposable
```

## Inheritance

[object](#) ← [State<GameStatId, GameState, GameStateMachine>](#) ← [GameState](#) ← BootstrapEndState

## Implements

[IDisposable](#)

## Inherited Members

[State<GameStatId, GameState, GameStateMachine>.Id](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StateMachine](#) ,  
[State<GameStatId, GameState, GameStateMachine>.IsReady\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StartContext\(StateParam\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.OnEndContext\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.Dispose\(\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Id

```
public override GameStateId Id { get; }
```

Property Value

[GameStatId](#)

## Methods

## StartContext(StateParam)

```
public override void StartContext(StateParam stateParam)
```

### Parameters

stateParam [StateParam](#)

# Class GameEntry

Namespace: [CatSweeper.Core](#)

Assembly: CatSweeper.dll

```
[ScriptPath("res://Script/Core/GameEntry/GameEntry.cs")]
public class GameEntry : Node, IDisposable
```

## Inheritance

[object](#) ← GodotObject ← Node ← GameEntry

## Implements

[IDisposable](#)

## Inherited Members

Node.NotificationEnterTree , Node.NotificationExitTree , Node.NotificationMovedInParent ,  
Node.NotificationReady , Node.NotificationPaused , Node.NotificationUnpaused ,  
Node.NotificationPhysicsProcess , Node.NotificationProcess , Node.NotificationParented ,  
Node.NotificationUnparented , Node.NotificationSceneInstantiated ,  
Node.NotificationDragBegin , Node.NotificationDragEnd , Node.NotificationPathRenamed ,  
Node.NotificationChildOrderChanged , Node.NotificationInternalProcess ,  
Node.NotificationInternalPhysicsProcess , Node.NotificationPostEnterTree ,  
Node.NotificationDisabled , Node.NotificationEnabled ,  
Node.NotificationResetPhysicsInterpolation , Node.NotificationEditorPreSave ,  
Node.NotificationEditorPostSave , Node.NotificationWMMouseEnter ,  
Node.NotificationWMMouseExit , Node.NotificationWMWindowFocusIn ,  
Node.NotificationWMWindowFocusOut , Node.NotificationWMCloseRequest ,  
Node.NotificationWMGoBackRequest , Node.NotificationWMSizeChanged ,  
Node.NotificationWMDpiChange , Node.NotificationVpMouseEnter ,  
Node.NotificationVpMouseExit , Node.NotificationWMPositionChanged ,  
Node.NotificationOsMemoryWarning , Node.NotificationTranslationChanged ,  
Node.NotificationWMAbout , Node.NotificationCrash , Node.NotificationOsImageUpdate ,  
Node.NotificationApplicationResumed , Node.NotificationApplicationPaused ,  
Node.NotificationApplicationFocusIn , Node.NotificationApplicationFocusOut ,  
Node.NotificationTextServerChanged , Node.GetNode<T>(NodePath) ,  
Node.GetNodeOrNull<T>(NodePath) , [Node.GetChild<T>\(int, bool\)](#) ,  
[Node.GetChildOrNull<T>\(int, bool\)](#) , Node.GetOwner<T>() , Node.GetOwnerOrNull<T>() ,  
Node.GetParent<T>() , Node.GetParentOrNull<T>() , Node.\_EnterTree() , Node.\_ExitTree() ,  
Node.\_GetConfigurationWarnings() , Node.\_Input(InputEvent) ,

[Node.\\_PhysicsProcess\(double\)](#) , [Node.\\_Process\(double\)](#) , Node.\_Ready() ,  
Node.\_ShortcutInput(InputEvent) , Node.\_UnhandledInput(InputEvent) ,  
Node.\_UnhandledKeyInput(InputEvent) , Node.PrintOrphanNodes() ,  
[Node.AddSibling\(Node, bool\)](#) , [Node.AddChild\(Node, bool, Node.InternalMode\)](#) ,  
Node.RemoveChild(Node) , [Node.Reparent\(Node, bool\)](#) , [Node.GetChildCount\(bool\)](#) ,  
[Node.GetChildren\(bool\)](#) , [Node.GetChild\(int, bool\)](#) , Node.HasNode(NodePath) ,  
Node.GetNode(NodePath) , Node.GetNodeOrNull(NodePath) , Node.GetParent() ,  
[Node.FindChild\(string, bool, bool\)](#) , [Node.FindChildren\(string, string, bool, bool\)](#) ,  
[Node.FindParent\(string\)](#) , Node.HasNodeAndResource(NodePath) ,  
Node.GetNodeAndResource(NodePath) , Node.IsInsideTree() , Node.IsPartOfEditedScene() ,  
Node.IsAncestorOf(Node) , Node.IsGreaterThanOrEqual(Node) , Node.GetPath() ,  
[Node.GetPathTo\(Node, bool\)](#) , [Node.AddToGroup\(StringName, bool\)](#) ,  
Node.RemoveFromGroup(StringName) , Node.IsInGroup(StringName) ,  
[Node.MoveChild\(Node, int\)](#) , Node.GetGroups() , [Node.GetIndex\(bool\)](#) , Node.PrintTree() ,  
Node.PrintTreePretty() , Node.GetTreeString() , Node.GetTreeStringPretty() ,  
[Node.PropagateNotification\(int\)](#) , [Node.PropagateCall\(StringName, Array, bool\)](#) ,  
[Node.SetPhysicsProcess\(bool\)](#) , Node.GetPhysicsProcessDeltaTime() ,  
Node.IsPhysicsProcessing() , Node.GetProcessDeltaTime() , [Node.SetProcess\(bool\)](#) ,  
Node.IsProcessing() , [Node.SetProcessInput\(bool\)](#) , Node.IsProcessingInput() ,  
[Node.SetProcessShortcutInput\(bool\)](#) , Node.IsProcessingShortcutInput() ,  
[Node.SetProcessUnhandledInput\(bool\)](#) , Node.IsProcessingUnhandledInput() ,  
[Node.SetProcessUnhandledKeyInput\(bool\)](#) , Node.IsProcessingUnhandledKeyInput() ,  
Node.CanProcess() , [Node.SetDisplayFolded\(bool\)](#) , Node.IsDisplayedFolded() ,  
[Node.SetProcessInternal\(bool\)](#) , Node.IsProcessingInternal() ,  
[Node.SetPhysicsProcessInternal\(bool\)](#) , Node.IsPhysicsProcessingInternal() ,  
Node.IsPhysicsInterpolated() , Node.IsPhysicsInterpolatedAndEnabled() ,  
Node.ResetPhysicsInterpolation() , Node.SetTranslationDomainInherited() ,  
Node.GetWindow() , Node.GetLastExclusiveWindow() , Node.GetTree() ,  
Node.CreateTween() , [Node.Duplicate\(int\)](#) , [Node.ReplaceBy\(Node, bool\)](#) ,  
[Node.setSceneInstanceLoadPlaceholder\(bool\)](#) ,  
Node.GetSceneInstanceLoadPlaceholder() , [Node.setEditableInstance\(Node, bool\)](#) ,  
Node.setEditableInstance(Node) , Node.GetViewport() , Node.QueueFree() ,  
Node.RequestReady() , Node.IsNodeReady() , [Node.SetMultiplayerAuthority\(int, bool\)](#) ,  
Node.GetMultiplayerAuthority() , Node.IsMultiplayerAuthority() ,  
Node.RpcConfig(StringName, Variant) , Node.GetRpcConfig() ,  
[Node.Atr\(string, StringName\)](#) , [Node.AtrN\(string, StringName, int, StringName\)](#) ,  
Node.Rpc(StringName, params Variant[]) ,  
[Node.Rpc\(StringName, ReadOnlySpan<Variant>\)](#) ,  
[Node.Rpcld\(long, StringName, params Variant\[\]\)](#) ,  
[Node.Rpcld\(long, StringName, ReadOnlySpan<Variant>\)](#) ,

```
Node.UpdateConfigurationWarnings() ,  
Node.CallDeferredThreadGroup(StringName, params Variant[]) ,  
Node.CallDeferredThreadGroup\(StringName, ReadOnlySpan<Variant>\) ,  
Node.SetDeferredThreadGroup(StringName, Variant) ,  
Node.NotifyDeferredThreadGroup\(int\) ,  
Node.CallThreadSafe(StringName, params Variant[]) ,  
Node.CallThreadSafe\(StringName, ReadOnlySpan<Variant>\) ,  
Node.SetThreadSafe(StringName, Variant) , Node.NotifyThreadSafe\(int\) ,  
Node.EmitSignalReady() , Node.EmitSignalRenamed() , Node.EmitSignalTreeEntered() ,  
Node.EmitSignalTreeExiting() , Node.EmitSignalTreeExited() ,  
Node.EmitSignalChildEnteredTree(Node) , Node.EmitSignalChildExitingTree(Node) ,  
Node.EmitSignalChildOrderChanged() , Node.EmitSignalReplacingBy(Node) ,  
Node.EmitSignalEditorDescriptionChanged(Node) , Node.EmitSignalEditorStateChanged() ,  
Node.InvokeGodotClassMethod(in godot_string_name, NativeVariantPtrArgs, out godot_variant) ,  
Node.HasGodotClassMethod(in godot_string_name) ,  
Node.HasGodotClassSignal(in godot_string_name) , Node.Name ,  
Node.UniqueNameInOwner , Node.SceneFilePath , Node.Owner , Node.Multiplayer ,  
Node.ProcessMode , Node.ProcessPriority , Node.ProcessPhysicsPriority ,  
Node.ProcessThreadGroup , Node.ProcessThreadGroupOrder ,  
Node.ProcessThreadMessages , Node.PhysicsInterpolationMode , Node.AutoTranslateMode ,  
Node.EditorDescription , Node.Ready , Node.Renamed , Node.TreeEntered ,  
Node.TreeExiting , Node.TreeExited , Node.ChildEnteredTree , Node.ChildExitingTree ,  
Node.ChildOrderChanged , Node.ReplacingBy , Node.EditorDescriptionChanged ,  
Node.EditorStateChanged , GodotObject.NotificationPostinitialize ,  
GodotObject.NotificationPredelete , GodotObject.NotificationExtensionReloaded ,  
GodotObject.InstanceFromId\(ulong\) , GodotObject.IsInstanceIdValid\(ulong\) ,  
GodotObject.IsValid(GodotObject) , GodotObject.WeakRef(GodotObject) ,  
GodotObject.Dispose() , GodotObject.Dispose\(bool\) , GodotObject.ToString() ,  
GodotObject.ToSignal(GodotObject, StringName) , GodotObject._Get(StringName) ,  
GodotObject._GetPropertyList() , GodotObject._IterGet(Variant) ,  
GodotObject._IterInit(Array) , GodotObject._IterNext(Array) ,  
GodotObject.\_Notification\(int\) , GodotObject._PropertyCanRevert(StringName) ,  
GodotObject._PropertyGetRevert(StringName) , GodotObject._Set(StringName, Variant) ,  
GodotObject._ValidateProperty(Dictionary) , GodotObject.Free() , GodotObject.GetClass() ,  
GodotObject.IsClass\(string\) , GodotObject.Set(StringName, Variant) ,  
GodotObject.Get(StringName) , GodotObject.SetIndexed(NodePath, Variant) ,  
GodotObject.GetIndexed(NodePath) , GodotObject.GetPropertyList() ,  
GodotObject.GetMethodList() , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , GodotObject.Notification\(int, bool\) ,
```

GodotObject.GetInstanceID() , GodotObject.SetScript(Variant) , GodotObject.GetScript() ,  
GodotObject.SetMeta(StringName, Variant) , GodotObject.RemoveMeta(StringName) ,  
GodotObject.GetMeta(StringName, Variant) , GodotObject.HasMeta(StringName) ,  
GodotObject.GetMetaList() , [GodotObject.AddUserSignal\(string, Array\)](#) ,  
GodotObject.HasUserSignal(StringName) , GodotObject.RemoveUserSignal(StringName) ,  
GodotObject.EmitSignal(StringName, params Variant[]) ,  
[GodotObject.EmitSignal\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.Call(StringName, params Variant[]) ,  
[GodotObject.Call\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.CallDeferred(StringName, params Variant[]) ,  
[GodotObject.CallDeferred\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.SetDeferred(StringName, Variant) , GodotObject.Callv(StringName, Array) ,  
GodotObject.HasMethod(StringName) ,  
GodotObjectGetMethodArgumentCount(StringName) ,  
GodotObject.HasSignal(StringName) , GodotObject.GetSignalList() ,  
GodotObject.GetSignalConnectionList(StringName) ,  
GodotObject.GetIncomingConnections() ,  
[GodotObject.Connect\(StringName, Callable, uint\)](#) ,  
GodotObject.Disconnect(StringName, Callable) ,  
GodotObject.IsConnected(StringName, Callable) ,  
GodotObject.HasConnections(StringName) , [GodotObject.SetBlockSignals\(bool\)](#) ,  
GodotObject.IsBlockingSignals() , GodotObject.NotifyPropertyListChanged() ,  
[GodotObject.SetMessageTranslation\(bool\)](#) , GodotObject.CanTranslateMessages() ,  
GodotObject.Tr(StringName, StringName) ,  
[GodotObject.TrN\(StringName, StringName, int, StringName\)](#) ,  
GodotObject.GetTranslationDomain() , GodotObject.SetTranslationDomain(StringName) ,  
GodotObject.IsQueuedForDeletion() , GodotObject.CancelFree() ,  
GodotObject.EmitSignalScriptChanged() , GodotObject.EmitSignalPropertyListChanged() ,  
GodotObject.NativeInstance , GodotObject.ScriptChanged ,  
GodotObject.PropertyListChanged , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Extension Methods

[NodeUtil.DontDestroyOnLoad\(Node\)](#)

# Methods

## RegisterGenericService(Game)

```
public static Game RegisterGenericService(Game game)
```

Parameters

game [Game](#)

Returns

[Game](#)

# Class GameEntry.MethodName

Namespace: [CatSweeper.Core](#)

Assembly: CatSweeper.dll

Cached StringNames for the methods contained in this class, for fast lookup.

```
public class GameEntry.MethodName : Node.MethodName
```

## Inheritance

[object](#) ← GodotObject.MethodName ← Node.MethodName ← GameEntry.MethodName

## Inherited Members

Node.MethodName.\_EnterTree , Node.MethodName.\_ExitTree ,  
Node.MethodName.\_GetConfigurationWarnings , Node.MethodName.\_Input ,  
Node.MethodName.\_PhysicsProcess , Node.MethodName.\_Process ,  
Node.MethodName.\_Ready , Node.MethodName.\_ShortcutInput ,  
Node.MethodName.\_UnhandledInput , Node.MethodName.\_UnhandledKeyInput ,  
Node.MethodName.PrintOrphanNodes , Node.MethodName.AddSibling ,  
Node.MethodName.SetName , Node.MethodName.GetName , Node.MethodName.AddChild ,  
Node.MethodName.RemoveChild , Node.MethodName.Reparent ,  
Node.MethodName.GetChildCount , Node.MethodName.GetChildren ,  
Node.MethodName.GetChild , Node.MethodName.HasNode , Node.MethodName.GetNode ,  
Node.MethodName.GetNodeOrNull , Node.MethodName.GetParent ,  
Node.MethodName.FindChild , Node.MethodName.FindChildren ,  
Node.MethodName.FindParent , Node.MethodName.HasNodeAndResource ,  
Node.MethodName.GetNodeAndResource , Node.MethodName.IsInsideTree ,  
Node.MethodName.IsPartOfEditedScene , Node.MethodName.IsAncestorOf ,  
Node.MethodName.IsGreaterThanOrEqual , Node.MethodName.GetPath ,  
Node.MethodName.GetPathTo , Node.MethodName.AddToGroup ,  
Node.MethodName.RemoveFromGroup , Node.MethodName.IsInGroup ,  
Node.MethodName.MoveChild , Node.MethodName.GetGroups ,  
Node.MethodName.SetOwner , Node.MethodName.GetOwner ,  
Node.MethodName.GetIndex , Node.MethodName.PrintTree ,  
Node.MethodName.PrintTreePretty , Node.MethodName.GetTreeString ,  
Node.MethodName.GetTreeStringPretty , Node.MethodName.SetSceneFilePath ,  
Node.MethodName.GetSceneFilePath , Node.MethodName.PropagateNotification ,  
Node.MethodName.PropagateCall , Node.MethodName.SetPhysicsProcess ,  
Node.MethodName.GetPhysicsProcessDeltaTime , Node.MethodName.IsPhysicsProcessing ,

Node.MethodName.GetProcessDeltaTime , Node.MethodName.SetProcess ,  
Node.MethodName.SetProcessPriority , Node.MethodName.GetProcessPriority ,  
Node.MethodName.SetPhysicsProcessPriority ,  
Node.MethodName.GetPhysicsProcessPriority , Node.MethodName.IsProcessing ,  
Node.MethodName.SetProcessInput , Node.MethodName.IsProcessingInput ,  
Node.MethodName.SetProcessShortcutInput ,  
Node.MethodName.IsProcessingShortcutInput ,  
Node.MethodName.SetProcessUnhandledInput ,  
Node.MethodName.IsProcessingUnhandledInput ,  
Node.MethodName.SetProcessUnhandledKeyInput ,  
Node.MethodName.IsProcessingUnhandledKeyInput , Node.MethodName.SetProcessMode ,  
Node.MethodName.GetProcessMode , Node.MethodName.CanProcess ,  
Node.MethodName.SetProcessThreadGroup , Node.MethodName.GetProcessThreadGroup ,  
Node.MethodName.SetProcessThreadMessages ,  
Node.MethodName.GetProcessThreadMessages ,  
Node.MethodName.SetProcessThreadGroupOrder ,  
Node.MethodName.GetProcessThreadGroupOrder , Node.MethodName.SetDisplayFolded ,  
Node.MethodName.IsDisplayedFolded , Node.MethodName.SetProcessInternal ,  
Node.MethodName.IsProcessingInternal , Node.MethodName.SetPhysicsProcessInternal ,  
Node.MethodName.IsPhysicsProcessingInternal ,  
Node.MethodName.SetPhysicsInterpolationMode ,  
Node.MethodName.GetPhysicsInterpolationMode ,  
Node.MethodName.IsPhysicsInterpolated ,  
Node.MethodName.IsPhysicsInterpolatedAndEnabled ,  
Node.MethodName.ResetPhysicsInterpolation , Node.MethodName.SetAutoTranslateMode ,  
Node.MethodName.GetAutoTranslateMode ,  
Node.MethodName.SetTranslationDomainInherited , Node.MethodName.GetWindow ,  
Node.MethodName.GetLastExclusiveWindow , Node.MethodName.GetTree ,  
Node.MethodName.CreateTween , Node.MethodName.Duplicate ,  
Node.MethodName.ReplaceBy , Node.MethodName.SetSceneInstanceLoadPlaceholder ,  
Node.MethodName.GetSceneInstanceLoadPlaceholder ,  
Node.MethodName.SetEditableInstance , Node.MethodName.IsEditableInstance ,  
Node.MethodName.GetViewport , Node.MethodName.QueueFree ,  
Node.MethodName.RequestReady , Node.MethodName.IsNodeReady ,  
Node.MethodName.SetMultiplayerAuthority , Node.MethodName.GetMultiplayerAuthority ,  
Node.MethodName.IsMultiplayerAuthority , Node.MethodName.GetMultiplayer ,  
Node.MethodName.RpcConfig , Node.MethodName.GetRpcConfig ,  
Node.MethodName.SetEditorDescription , Node.MethodName.GetEditorDescription ,  
Node.MethodName.\_SetImportPath , Node.MethodName.\_GetImportPath ,  
Node.MethodName.SetUniqueNameInOwner , Node.MethodName.IsUniqueNameInOwner ,

Node.MethodName.Atr , Node.MethodName.AtrN , Node.MethodName.Rpc ,  
Node.MethodName.RpcId , Node.MethodName.UpdateConfigurationWarnings ,  
Node.MethodName.CallDeferredThreadGroup ,  
Node.MethodName.SetDeferredThreadGroup ,  
Node.MethodName.NotifyDeferredThreadGroup , Node.MethodName.CallThreadSafe ,  
Node.MethodName.SetThreadSafe , Node.MethodName.NotifyThreadSafe ,  
GodotObject.MethodName.\_Get , GodotObject.MethodName.\_GetPropertyList ,  
GodotObject.MethodName.\_IterGet , GodotObject.MethodName.\_IterInit ,  
GodotObject.MethodName.\_IterNext , GodotObject.MethodName.\_Notification ,  
GodotObject.MethodName.\_PropertyCanRevert ,  
GodotObject.MethodName.\_PropertyGetRevert , GodotObject.MethodName.\_Set ,  
GodotObject.MethodName.\_ValidateProperty , GodotObject.MethodName.Free ,  
GodotObject.MethodName.GetClass , GodotObject.MethodName.IsClass ,  
GodotObject.MethodName.Set , GodotObject.MethodName.Get ,  
GodotObject.MethodName.SetIndexed , GodotObject.MethodName.GetIndexed ,  
GodotObject.MethodName.GetPropertyList , GodotObject.MethodName.GetMethodList ,  
GodotObject.MethodName.PropertyCanRevert ,  
GodotObject.MethodName.PropertyGetRevert , GodotObject.MethodName.Notification ,  
GodotObject.MethodName.GetInstanceId , GodotObject.MethodName.SetScript ,  
GodotObject.MethodName.GetScript , GodotObject.MethodName.SetMeta ,  
GodotObject.MethodName.RemoveMeta , GodotObject.MethodName.GetMeta ,  
GodotObject.MethodName.HasMeta , GodotObject.MethodName.GetMetaList ,  
GodotObject.MethodName.AddUserSignal , GodotObject.MethodName.HasUserSignal ,  
GodotObject.MethodName.RemoveUserSignal , GodotObject.MethodName.EmitSignal ,  
GodotObject.MethodName.Call , GodotObject.MethodName.CallDeferred ,  
GodotObject.MethodName.SetDeferred , GodotObject.MethodName.Callv ,  
GodotObject.MethodName.HasMethod ,  
GodotObject.MethodName.GetMethodArgumentCount ,  
GodotObject.MethodName.HasSignal , GodotObject.MethodName.GetSignalList ,  
GodotObject.MethodName.GetSignalConnectionList ,  
GodotObject.MethodName.GetIncomingConnections , GodotObject.MethodName.Connect ,  
GodotObject.MethodName.Disconnect , GodotObject.MethodName.IsConnected ,  
GodotObject.MethodName.HasConnections , GodotObject.MethodName.SetBlockSignals ,  
GodotObject.MethodName.IsBlockingSignals ,  
GodotObject.MethodName.NotifyPropertyChanged ,  
GodotObject.MethodName.SetMessageTranslation ,  
GodotObject.MethodName.CanTranslateMessages , GodotObject.MethodName.Tr ,  
GodotObject.MethodName.TrN , GodotObject.MethodName.GetTranslationDomain ,  
GodotObject.MethodName.SetTranslationDomain ,  
GodotObject.MethodName.IsQueuedForDeletion , GodotObject.MethodName.CancelFree ,

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### InfoBuildByte

Cached name for the 'InfoBuildByte' method.

```
public static readonly StringName InfoBuildByte
```

#### Field Value

StringName

### RegisterJsonConverters

Cached name for the 'RegisterJsonConverters' method.

```
public static readonly StringName RegisterJsonConverters
```

#### Field Value

StringName

# Class GameEntry.PropertyName

Namespace: [CatSweeper.Core](#)

Assembly: CatSweeper.dll

Cached StringNames for the properties and fields contained in this class, for fast lookup.

```
public class GameEntry.PropertyName : Node.PropertyName
```

## Inheritance

[object](#) ← GodotObject.PropertyName ← Node.PropertyName ← GameEntry.PropertyName

## Inherited Members

Node.PropertyName.\_ImportPath , Node.PropertyName.Name ,  
Node.PropertyName.UniqueNameInOwner , Node.PropertyName.SceneFilePath ,  
Node.PropertyName.Owner , Node.PropertyName.Multiplayer ,  
Node.PropertyName.ProcessMode , Node.PropertyName.ProcessPriority ,  
Node.PropertyName.ProcessPhysicsPriority , Node.PropertyName.ProcessThreadGroup ,  
Node.PropertyName.ProcessThreadGroupOrder ,  
Node.PropertyName.ProcessThreadMessages ,  
Node.PropertyName.PhysicsInterpolationMode , Node.PropertyName.AutoTranslateMode ,  
Node.PropertyName.EditorDescription , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Fields

### \_logLevel

Cached name for the '\_logLevel' field.

```
public static readonly StringName _logLevel
```

### Field Value

StringName

# Class GameEntry.SignalName

Namespace: [CatSweeper.Core](#)

Assembly: CatSweeper.dll

Cached StringNames for the signals contained in this class, for fast lookup.

```
public class GameEntry.SignalName : Node.SignalName
```

## Inheritance

[object](#) ← GodotObject.SignalName ← Node.SignalName ← GameEntry.SignalName

## Inherited Members

Node.SignalName.Ready , Node.SignalName.Renamed , Node.SignalName.TreeEntered ,  
Node.SignalName.TreeExiting , Node.SignalName.TreeExited ,  
Node.SignalName.ChildEnteredTree , Node.SignalName.ChildExitingTree ,  
Node.SignalName.ChildOrderChanged , Node.SignalName.ReplacingBy ,  
Node.SignalName.EditorDescriptionChanged , Node.SignalName.EditorStateChanged ,  
GodotObject.SignalName.ScriptChanged , GodotObject.SignalName.PropertyListChanged ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class GameState

Namespace: [CatSweeper.Core](#)

Assembly: CatSweeper.dll

```
public abstract class GameState : State<GameStateId, GameState,  
GameStateMachine>, IDisposable
```

## Inheritance

[object](#) ← [State<GameStatId, GameState, GameStateMachine>](#) ← GameState

## Implements

[IDisposable](#)

## Derived

[BootstrapEndState](#), [GameplayInitState](#), [InfoLoadState](#), [LoadStageState](#), [LocalLoadState](#),  
[LoginState](#), [UILoadState](#), [UserDataLoadState](#)

## Inherited Members

[State<GameStatId, GameState, GameStateMachine>.Id](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StateMachine](#) ,  
[State<GameStatId, GameState, GameStateMachine>.IsReady\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StartContext\(StateParam\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.OnEndContext\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.Dispose\(\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Enum GameStatId

Namespace: [CatSweeper.Core](#)

Assembly: CatSweeper.dll

```
public enum GameStateId
```

## Extension Methods

[EnumExtension.hasFlag\(Enumeration, Enumeration\)](#)

## Fields

BootstrapEnd = 6

GameplayInit = 3

InfoLoad = 1

LoadStage = 7

LocalLoad = 0

Login = 2

UILoad = 5

UserDataTableLoad = 4

# Class GameStateMachine

Namespace: [CatSweeper.Core](#)

Assembly: CatSweeper.dll

```
public class GameStateMachine : StateMachine<GameStatId, GameState,  
GameStateMachine>, IStateMachine<GameStatId>, IService, IDisposable
```

## Inheritance

[object](#) ← [StateMachine<GameStatId, GameState, GameStateMachine>](#) ←  
GameStateMachine

## Implements

[IStateMachine<GameStatId>](#), [IService](#), [IDisposable](#)

## Inherited Members

[StateMachine<GameStatId, GameState, GameStateMachine>.lastStatId](#) ,  
[StateMachine<GameStatId, GameState, GameStateMachine>.currentStatId](#) ,  
[StateMachine<GameStatId, GameState,](#)  
[GameStateMachine>.SubscribeBeforeStateChange\(Action<StateChangeRecord<GameStatId>>\)](#) ,  
[StateMachine<GameStatId, GameState,](#)  
[GameStateMachine>.SubscribeAfterStateChange\(Action<StateChangeRecord<GameStatId>>\)](#) ,  
[StateMachine<GameStatId, GameState, GameStateMachine>.RegisterState\(GameState\)](#) ,  
[StateMachine<GameStatId, GameState,](#)  
[GameStateMachine>.CanGoToState\(GameStatId, StateParam\)](#) ,  
[StateMachine<GameStatId, GameState, GameStateMachine>.TryGoToState\(GameStatId, StateParam\)](#) ,  
[StateMachine<GameStatId, GameState,](#)  
[GameStateMachine>.ForceGoToState\(GameStatId, StateParam\)](#) ,  
[StateMachine<GameStatId, GameState,](#)  
[GameStateMachine>.GetStateUnsafe\(GameStatId\)](#) ,  
[StateMachine<GameStatId, GameState, GameStateMachine>.GetStateUnsafe<T>\(GameStatId\)](#) ,  
[StateMachine<GameStatId, GameState, GameStateMachine>.TryGetState\(GameStatId, out GameState\)](#) ,  
[StateMachine<GameStatId, GameState, GameStateMachine>.Dispose\(\)](#) ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,

[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### GameStateMachine()

```
public GameStateMachine()
```

# Class GameplayInitState

Namespace: [CatSweeper.Core](#)

Assembly: CatSweeper.dll

```
public class GameplayInitState : GameState, IDisposable
```

## Inheritance

[object](#) ← [State<GameStatId, GameState, GameStateMachine>](#) ← [GameState](#) ← [GameplayInitState](#)

## Implements

[IDisposable](#)

## Inherited Members

[State<GameStatId, GameState, GameStateMachine>.Id](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StateMachine](#) ,  
[State<GameStatId, GameState, GameStateMachine>.IsReady\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StartContext\(StateParam\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.OnEndContext\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.Dispose\(\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Id

```
public override GameStateId Id { get; }
```

### Property Value

[GameStatId](#)

## Methods

## StartContext(StateParam)

```
public override void StartContext(StateParam stateParam)
```

### Parameters

stateParam [StateParam](#)

# Class InfoLoadState

Namespace: [CatSweeper.Core](#)

Assembly: CatSweeper.dll

```
public class InfoLoadState : GameState, IDisposable
```

## Inheritance

[object](#) ← [State<GameStatId, GameState, GameStateMachine>](#) ← [GameState](#) ← [InfoLoadState](#)

## Implements

[IDisposable](#)

## Inherited Members

[State<GameStatId, GameState, GameStateMachine>.Id](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StateMachine](#) ,  
[State<GameStatId, GameState, GameStateMachine>.IsReady\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StartContext\(StateParam\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.OnEndContext\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.Dispose\(\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Id

```
public override GameStateId Id { get; }
```

### Property Value

[GameStatId](#)

## Methods

## StartContext(StateParam)

```
public override void StartContext(StateParam stateParam)
```

### Parameters

stateParam [StateParam](#)

# Class LoadStageState

Namespace: [CatSweeper.Core](#)

Assembly: CatSweeper.dll

```
public class LoadStageState : GameState, IDisposable
```

## Inheritance

[object](#) ← [State<GameStatId, GameState, GameStateMachine>](#) ← [GameState](#) ← LoadStageState

## Implements

[IDisposable](#)

## Inherited Members

[State<GameStatId, GameState, GameStateMachine>.Id](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StateMachine](#) ,  
[State<GameStatId, GameState, GameStateMachine>.IsReady\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StartContext\(StateParam\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.OnEndContext\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.Dispose\(\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Id

```
public override GameStateId Id { get; }
```

### Property Value

[GameStatId](#)

## Methods

## OnEndContext()

```
protected override void OnEndContext()
```

## StartContext(StateParam)

```
public override void StartContext(StateParam param)
```

### Parameters

param [StateParam](#)

# Class LoadStageState.InitParam

Namespace: [CatSweeper.Core](#)

Assembly: CatSweeper.dll

```
public class LoadStageState.InitParam : StateParam
```

## Inheritance

[object](#) ← [StateParam](#) ← LoadStageState.InitParam

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### stageId

```
public string stageId
```

#### Field Value

[string](#)

# Class LocalLoadState

Namespace: [CatSweeper.Core](#)

Assembly: CatSweeper.dll

```
public class LocalLoadState : GameState, IDisposable
```

## Inheritance

[object](#) ← [State<GameStatId, GameState, GameStateMachine>](#) ← [GameState](#) ← LocalLoadState

## Implements

[IDisposable](#)

## Inherited Members

[State<GameStatId, GameState, GameStateMachine>.Id](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StateMachine](#) ,  
[State<GameStatId, GameState, GameStateMachine>.IsReady\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StartContext\(StateParam\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.OnEndContext\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.Dispose\(\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Id

```
public override GameStateId Id { get; }
```

### Property Value

[GameStatId](#)

## Methods

## StartContext(StateParam)

```
public override void StartContext(StateParam param)
```

### Parameters

param [StateParam](#)

# Class LoginState

Namespace: [CatSweeper.Core](#)

Assembly: CatSweeper.dll

```
public class LoginState : GameState, IDisposable
```

## Inheritance

[object](#) ← [State<GameStatId, GameState, GameStateMachine>](#) ← [GameState](#) ← LoginState

## Implements

[IDisposable](#)

## Inherited Members

[State<GameStatId, GameState, GameStateMachine>.Id](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StateMachine](#) ,  
[State<GameStatId, GameState, GameStateMachine>.IsReady\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StartContext\(StateParam\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.OnEndContext\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.Dispose\(\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Id

```
public override GameStateId Id { get; }
```

### Property Value

[GameStatId](#)

## Methods

## StartContext(StateParam)

```
public override void StartContext(StateParam stateParam)
```

### Parameters

stateParam [StateParam](#)

# Class LoginState.Param

Namespace: [CatSweeper.Core](#)

Assembly: CatSweeper.dll

```
public class LoginState.Param : StateParam
```

## Inheritance

[object](#) ← [StateParam](#) ← LoginState.Param

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### Platform

```
public LoginPlatform Platform
```

#### Field Value

[LoginPlatform](#)

### Token

```
public LoginToken Token
```

#### Field Value

[LoginToken](#)

# Class UILoadState

Namespace: [CatSweeper.Core](#)

Assembly: CatSweeper.dll

```
public class UILoadState : GameState, IDisposable
```

## Inheritance

[object](#) ← [State<GameStatId, GameState, GameStateMachine>](#) ← [GameState](#) ← [UILoadState](#)

## Implements

[IDisposable](#)

## Inherited Members

[State<GameStatId, GameState, GameStateMachine>.Id](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StateMachine](#) ,  
[State<GameStatId, GameState, GameStateMachine>.IsReady\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StartContext\(StateParam\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.OnEndContext\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.Dispose\(\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Id

```
public override GameStateId Id { get; }
```

### Property Value

[GameStatId](#)

## Methods

## StartContext(StateParam)

```
public override void StartContext(StateParam param)
```

### Parameters

param [StateParam](#)

# Class UserDataLoadState

Namespace: [CatSweeper.Core](#)

Assembly: CatSweeper.dll

```
public class UserDataLoadState : GameState, IDisposable
```

## Inheritance

[object](#) ← [State<GameStatId, GameState, GameStateMachine>](#) ← [GameState](#) ←   
UserDataLoadState

## Implements

[IDisposable](#)

## Inherited Members

[State<GameStatId, GameState, GameStateMachine>.Id](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StateMachine](#) ,  
[State<GameStatId, GameState, GameStateMachine>.IsReady\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.StartContext\(StateParam\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.OnEndContext\(\)](#) ,  
[State<GameStatId, GameState, GameStateMachine>.Dispose\(\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Id

```
public override GameStateId Id { get; }
```

### Property Value

[GameStatId](#)

## Methods

## StartContext(StateParam)

```
public override void StartContext(StateParam stateParam)
```

### Parameters

stateParam [StateParam](#)

# Namespace CatSweeper.Info

## Classes

[GameSettingInfo](#)

[GameSettingInfoManager](#)

[StageInfo](#)

[StageInfoManager](#)

[StageMapInfo](#)

[StageMapInfoManager](#)

# Class GameSettingInfo

Namespace: [CatSweeper.Info](#)

Assembly: CatSweeper.dll

```
[Serializable]
public class GameSettingInfo
```

## Inheritance

[object](#) ← GameSettingInfo

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### key

```
public string key { get; set; }
```

Property Value

[string](#)

### value

```
public string value { get; set; }
```

Property Value

[string](#)

# Class GameSettingInfoManager

Namespace: [CatSweeper.Info](#)

Assembly: CatSweeper.dll

```
public class GameSettingInfoManager : ConfigInfoManager<string, GameSettingInfo>,  
IInfoManager, IDisposable
```

## Inheritance

[object](#) ← [InfoManager](#) ← [ConfigInfoManager<string, GameSettingInfo>](#) ←  
GameSettingInfoManager

## Implements

[IInfoManager](#), [IDisposable](#)

## Inherited Members

[ConfigInfoManager<string, GameSettingInfo>.valueMap](#) ,  
[ConfigInfoManager<string, GameSettingInfo>.GetAllValue\(\)](#) ,  
[ConfigInfoManager<string, GameSettingInfo>.keyFn](#) ,  
[ConfigInfoManager<string, GameSettingInfo>.infoType](#) ,  
[ConfigInfoManager<string, GameSettingInfo>.LoadInfo\(\)](#) ,  
[ConfigInfoManager<string, GameSettingInfo>.LoadInfoAsync\(CancellationToken\)](#) ,  
[ConfigInfoManager<string, GameSettingInfo>.AddValue\(GameSettingInfo\)](#) ,  
[ConfigInfoManager<string, GameSettingInfo>.TryGetValue\(string, out GameSettingInfo\)](#) ,  
[ConfigInfoManager<string, GameSettingInfo>.Dispose\(\)](#) , [InfoManager.infoType](#) ,  
[InfoManager.GetAllValue\(\)](#) , [InfoManager.LoadInfo\(\)](#) ,  
[InfoManager.LoadInfoAsync\(CancellationToken\)](#) , [InfoManager.OnLoadCompleted\(\)](#) ,  
[InfoManager.Dispose\(\)](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

**GameSettingInfoManager(IValueLoader<GameSettingInfo>)**

```
public GameSettingInfoManager(IValueLoader<GameSettingInfo> loader)
```

## Parameters

loader [IValueLoader<GameSettingInfo>](#)

## Properties

### keyFn

```
protected override Func<GameSettingInfo, string> keyFn { get; }
```

## Property Value

[Func<GameSettingInfo, string>](#)

## Methods

### GetBoolValue(string)

```
public Res<bool, Exception> GetBoolValue(string id)
```

## Parameters

id [string](#)

## Returns

[Res<bool, Exception>](#)

### GetFloatValue(string)

```
public Res<float, Exception> GetFloatValue(string id)
```

## Parameters

id [string](#)

Returns

[Res<float, Exception>](#)

## GetIntValue(string)

`public Res<int, Exception> GetIntValue(string id)`

Parameters

`id string`

Returns

[Res<int, Exception>](#)

## GetStringValue(string)

`public Res<string, Exception> GetStringValue(string id)`

Parameters

`id string`

Returns

[Res<string, Exception>](#)

# Class StageInfo

Namespace: [CatSweeper.Info](#)

Assembly: CatSweeper.dll

```
public class StageInfo
```

## Inheritance

[object](#) ← StageInfo

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### map

Support Direct Size, or StageMapInfo Id reference

1. **Size(10, 10)** - Creates a map of size 10x10
2. **StageMapInfo(map\_1)** - References a [StageMapInfo](#) with ID 'map\_1'

```
public string map { get; set; }
```

### Property Value

[string](#)

### scenePath

```
public string scenePath { get; set; }
```

### Property Value

[string](#)

## stageId

```
public string stageId { get; set; }
```

Property Value

[string](#)

## Methods

### GetMap()

```
public Res<StageMapInfo, Exception> GetMap()
```

Returns

[Res<StageMapInfo, Exception>](#)

# Class StageInfoManager

Namespace: [CatSweeper.Info](#)

Assembly: CatSweeper.dll

```
public class StageInfoManager : ConfigInfoManager<string, StageInfo>,
IInfoManager, IDisposable
```

## Inheritance

[object](#) ← [InfoManager](#) ← [ConfigInfoManager<string, StageInfo>](#) ← StageInfoManager

## Implements

[IInfoManager](#), [IDisposable](#)

## Inherited Members

[ConfigInfoManager<string, StageInfo>.valueMap](#) ,  
[ConfigInfoManager<string, StageInfo>.GetAllValue\(\)](#) ,  
[ConfigInfoManager<string, StageInfo>.keyFn](#) ,  
[ConfigInfoManager<string, StageInfo>.infoType](#) ,  
[ConfigInfoManager<string, StageInfo>.LoadInfo\(\)](#) ,  
[ConfigInfoManager<string, StageInfo>.LoadInfoAsync\(CancellationToken\)](#) ,  
[ConfigInfoManager<string, StageInfo>.AddValue\(StageInfo\)](#) ,  
[ConfigInfoManager<string, StageInfo>.TryGetValue\(string, out StageInfo\)](#) ,  
[ConfigInfoManager<string, StageInfo>.Dispose\(\)](#) , [InfoManager.infoType](#) ,  
[InfoManager.GetAllValue\(\)](#) , [InfoManager.LoadInfo\(\)](#) ,  
[InfoManager.LoadInfoAsync\(CancellationToken\)](#) , [InfoManager.OnLoadCompleted\(\)](#) ,  
[InfoManager.Dispose\(\)](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### StageInfoManager(IValueLoader<StageInfo>)

```
public StageInfoManager(IValueLoader<StageInfo> loader)
```

## Parameters

loader [IValueLoader<StageInfo>](#)

## Properties

### keyFn

```
protected override Func<StageInfo, string> keyFn { get; }
```

Property Value

[Func<StageInfo, string>](#)

# Class StageMapInfo

Namespace: [CatSweeper.Info](#)

Assembly: CatSweeper.dll

```
public class StageMapInfo
```

## Inheritance

[object](#) ← StageMapInfo

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### floorGrids

```
public Vector2I[] floorGrids { get; set; }
```

Property Value

Vector2I[]

### mapId

```
public string mapId { get; set; }
```

Property Value

[string](#)

# Class StageMapInfoManager

Namespace: [CatSweeper.Info](#)

Assembly: CatSweeper.dll

```
public class StageMapInfoManager : ConfigInfoManager<string, StageMapInfo>,  
IInfoManager, IDisposable
```

## Inheritance

[object](#) ← [InfoManager](#) ← [ConfigInfoManager<string, StageMapInfo>](#) ←  
StageMapInfoManager

## Implements

[IInfoManager](#), [IDisposable](#)

## Inherited Members

[ConfigInfoManager<string, StageMapInfo>.valueMap](#) ,  
[ConfigInfoManager<string, StageMapInfo>.GetAllValue\(\)](#) ,  
[ConfigInfoManager<string, StageMapInfo>.keyFn](#) ,  
[ConfigInfoManager<string, StageMapInfo>.infoType](#) ,  
[ConfigInfoManager<string, StageMapInfo>.LoadInfo\(\)](#) ,  
[ConfigInfoManager<string, StageMapInfo>.LoadInfoAsync\(CancellationToken\)](#) ,  
[ConfigInfoManager<string, StageMapInfo>.AddValue\(StageMapInfo\)](#) ,  
[ConfigInfoManager<string, StageMapInfo>.TryGetValue\(string, out StageMapInfo\)](#) ,  
[ConfigInfoManager<string, StageMapInfo>.Dispose\(\)](#) , [InfoManager.infoType](#) ,  
[InfoManager.GetAllValue\(\)](#) , [InfoManager.LoadInfo\(\)](#) ,  
[InfoManager.LoadInfoAsync\(CancellationToken\)](#) , [InfoManager.OnLoadCompleted\(\)](#) ,  
[InfoManager.Dispose\(\)](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### StageMapInfoManager(IValueLoader<StageMapInfo>)

```
public StageMapInfoManager(IValueLoader<StageMapInfo> loader)
```

## Parameters

loader [IValueLoader<StageMapInfo>](#)

## Properties

### keyFn

```
protected override Func<StageMapInfo, string> keyFn { get; }
```

## Property Value

[Func<StageMapInfo, string>](#)

## Methods

### GetMapBySize(int, int)

```
public StageMapInfo GetMapBySize(int width, int height)
```

## Parameters

width [int](#)

height [int](#)

## Returns

[StageMapInfo](#)

# Namespace CatSweeper.Service

## Classes

[StageService](#)

[StageServiceModel](#)

## Structs

[StageData](#)

# Struct StageData

Namespace: [CatSweeper.Service](#)

Assembly: CatSweeper.dll

```
public struct StageData
```

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Fields

### mapInfo

```
public StageMapInfo mapInfo
```

Field Value

[StageMapInfo](#)

### stageInfo

```
public StageInfo stageInfo
```

Field Value

[StageInfo](#)

# Class StageService

Namespace: [CatSweeper.Service](#)

Assembly: CatSweeper.dll

```
public class StageService : IModelService, IService, IDisposable
```

## Inheritance

[object](#) ← StageService

## Implements

[IModelService](#), [IService](#), [IDisposable](#)

## Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#),  
[object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#),  
[object.ToString\(\)](#)

## Constructors

StageService(StageServiceModel, StageInfoManager,  
StageMapInfoManager)

```
public StageService(StageServiceModel model, StageInfoManager infoManger,  
StageMapInfoManager mapInfoManager)
```

## Parameters

model [StageServiceModel](#)

infoManger [StageInfoManager](#)

mapInfoManager [StageMapInfoManager](#)

## Properties

# GetModel

```
public IServiceProvider GetModel { get; }
```

Property Value

[IServiceProvider](#)

## Methods

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

### GetStageData(string)

```
public Res<StageData, Exception> GetStageData(string stageId)
```

Parameters

stageId [string](#)

Returns

[Res<StageData, Exception>](#)

# Class StageServiceModel

Namespace: [CatSweeper.Service](#)

Assembly: CatSweeper.dll

```
public class StageServiceModel : IServiceProvider, IRuntimeSavable, IDisposable
```

## Inheritance

[object](#) ← StageServiceModel

## Implements

[IServiceProvider](#), [IRuntimeSavable](#), [IDisposable](#)

## Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#),  
[object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#),  
[object.ToString\(\)](#)

## Methods

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

### Initialize(IUserData)

```
public void Initialize(IUserData userData)
```

## Parameters

userData [IUserData](#)

## SetSaveData(Dictionary<string, object>)

```
public void SetSaveData(Dictionary<string, object> dataMap)
```

### Parameters

dataMap [Dictionary](#)<[string](#), [object](#)>

# Namespace CatSweeper.Util

## Classes

[ExportUtil](#)

[InfoUtil](#)

# Class ExportUtil

Namespace: [CatSweeper.Util](#)

Assembly: CatSweeper.dll

```
public static class ExportUtil
```

## Inheritance

[object](#) ← ExportUtil

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### MarkReadOnly(Dictionary, StringName)

```
public static void MarkReadOnly(this Dictionary property, StringName propertyName)
```

#### Parameters

property Dictionary

propertyName StringName

# Class InfoUtil

Namespace: [CatSweeper.Util](#)

Assembly: CatSweeper.dll

```
public static class InfoUtil
```

## Inheritance

[object](#) ← InfoUtil

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### CreateStorage(Type)

```
public static IStorage CreateStorage(Type infoType)
```

#### Parameters

infoType [Type](#)

#### Returns

[IStorage](#)

# Namespace CofyDev.Xml.Doc

## Classes

[CofyXmlDocParser](#)

[DataContainer](#)

[DataObject](#)

[DataObject.Decoder](#)

[DataObject.ListValueDecoder](#)

[DataObject.StringValueDecoder](#)

[DataObjectEncoder](#)

[DataObjectExtension](#)

## Structs

[DataObject.BooleanDecoder](#)

[DataObject.DoubleDecoder](#)

[DataObject.EnumDecoder](#)

[DataObject.FloatDecoder](#)

[DataObject.IntDecoder](#)

[DataObject.StringDecoder](#)

## Interfaces

[DataObject.IStringDecoder](#)

[DataObject.IValueDecoder](#)

# Class CofyXmlDocParser

Namespace: [CofyDev.Xml.Doc](#)

Assembly: CatSweeper.dll

```
public static class CofyXmlDocParser
```

## Inheritance

[object](#) ← CofyXmlDocParser

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### ParseExcel(byte[])

```
public static DataContainer ParseExcel(byte[] fileBytes)
```

#### Parameters

fileBytes [byte](#)[]

#### Returns

[DataContainer](#)

# Class DataContainer

Namespace: [CofyDev.Xml.Doc](#)

Assembly: CatSweeper.dll

```
public class DataContainer : List<DataObject>, IList<DataObject>,
ICollection<DataObject>, IReadOnlyList<DataObject>, IReadOnlyCollection<DataObject>,
IEnumerable<DataObject>, IList, ICollection, IEnumerable
```

## Inheritance

[object](#) ← [List](#)<[DataObject](#)> ← DataContainer

## Implements

[IList](#)<[DataObject](#)>, [ICollection](#)<[DataObject](#)>, [IReadOnlyList](#)<[DataObject](#)>,
[IReadOnlyCollection](#)<[DataObject](#)>, [IEnumerable](#)<[DataObject](#)>, [IList](#), [ICollection](#),
[IEnumerable](#)

## Inherited Members

[List<DataObject>.Add\(DataObject\)](#) ,  
[List<DataObject>.AddRange\(IEnumerable<DataObject>\)](#) ,  
[List<DataObject>.AsReadOnly\(\)](#) ,  
[List<DataObject>.BinarySearch\(int, int, DataObject, IComparer<DataObject>\)](#) ,  
[List<DataObject>.BinarySearch\(DataObject\)](#) ,  
[List<DataObject>.BinarySearch\(DataObject, IComparer<DataObject>\)](#) ,  
[List<DataObject>.Clear\(\)](#) , [List<DataObject>.Contains\(DataObject\)](#) ,  
[List<DataObject>.ConvertAll<TOutput>\(Converter<DataObject, TOutput>\)](#) ,  
[List<DataObject>.CopyTo\(int, DataObject\[\], int, int\)](#) ,  
[List<DataObject>.CopyTo\(DataObject\[\]\)](#) , [List<DataObject>.CopyTo\(DataObject\[\], int\)](#) ,  
[List<DataObject>.EnsureCapacity\(int\)](#) ,  
[List<DataObject>.Exists\(Predicate<DataObject>\)](#) ,  
[List<DataObject>.Find\(Predicate<DataObject>\)](#) ,  
[List<DataObject>.FindAll\(Predicate<DataObject>\)](#) ,  
[List<DataObject>.FindIndex\(int, int, Predicate<DataObject>\)](#) ,  
[List<DataObject>.FindIndex\(int, Predicate<DataObject>\)](#) ,  
[List<DataObject>.FindIndex\(Predicate<DataObject>\)](#) ,  
[List<DataObject>.FindLast\(Predicate<DataObject>\)](#) ,  
[List<DataObject>.FindLastIndex\(int, int, Predicate<DataObject>\)](#) ,  
[List<DataObject>.FindLastIndex\(int, Predicate<DataObject>\)](#) ,  
[List<DataObject>.FindLastIndex\(Predicate<DataObject>\)](#) ,

[List<DataObject>.ForEach\(Action<DataObject>\)](#) , [List<DataObject>.GetEnumerator\(\)](#) ,  
[List<DataObject>.GetRange\(int, int\)](#) , [List<DataObject>.IndexOf\(DataObject\)](#) ,  
[List<DataObject>.IndexOf\(DataObject, int\)](#) ,  
[List<DataObject>.IndexOf\(DataObject, int, int\)](#) ,  
[List<DataObject>.Insert\(int, DataObject\)](#) ,  
[List<DataObject>.InsertRange\(int, IEnumerable<DataObject>\)](#) ,  
[List<DataObject>.LastIndexOf\(DataObject\)](#) ,  
[List<DataObject>.LastIndexOf\(DataObject, int\)](#) ,  
[List<DataObject>.LastIndexOf\(DataObject, int, int\)](#) ,  
[List<DataObject>.Remove\(DataObject\)](#) ,  
[List<DataObject>.RemoveAll\(Predicate<DataObject>\)](#) ,  
[List<DataObject>.RemoveAt\(int\)](#) , [List<DataObject>.RemoveRange\(int, int\)](#) ,  
[List<DataObject>.Reverse\(\)](#) , [List<DataObject>.Reverse\(int, int\)](#) ,  
[List<DataObject>.Slice\(int, int\)](#) , [List<DataObject>.Sort\(\)](#) ,  
[List<DataObject>.Sort\(IComparer<DataObject>\)](#) ,  
[List<DataObject>.Sort\(Comparison<DataObject>\)](#) ,  
[List<DataObject>.Sort\(int, int, IComparer<DataObject>\)](#) , [List<DataObject>.ToArray\(\)](#) ,  
[List<DataObject>.TrimExcess\(\)](#) , [List<DataObject>.TrueForAll\(Predicate<DataObject>\)](#) ,  
[List<DataObject>.Capacity](#) , [List<DataObject>.Count](#) , [List<DataObject>.this\[int\]](#) ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Extension Methods

[ListExtension.EnsureCapacity<T>\(List<T>, int\)](#)

# Class DataObject

Namespace: [CofyDev.Xml.Doc](#)

Assembly: CatSweeper.dll

```
public class DataObject : Dictionary<string, object>, IDictionary<string, object>,  
ICollection<KeyValuePair<string, object>>, IReadOnlyDictionary<string, object>,  
IReadOnlyCollection<KeyValuePair<string, object>>, IEnumerable<KeyValuePair<string,  
object>>, IDictionary, ICollection, IEnumerable, IDeserializationCallback,  
ISerializable
```

## Inheritance

[object](#) ← [Dictionary](#)<[string](#), [object](#)> ← DataObject

## Implements

[IDictionary](#)<[string](#), [object](#)>, [ICollection](#)<[KeyValuePair](#)<[string](#), [object](#)>>,  
[IReadOnlyDictionary](#)<[string](#), [object](#)>,  
[IReadOnlyCollection](#)<[KeyValuePair](#)<[string](#), [object](#)>>,  
[IEnumerable](#)<[KeyValuePair](#)<[string](#), [object](#)>>, [IDictionary](#), [ICollection](#),  
[IEnumerable](#), [IDeserializationCallback](#), [ISerializable](#)

## Inherited Members

[Dictionary](#)<[string](#), [object](#)>.Add([string](#), [object](#)) , [Dictionary](#)<[string](#), [object](#)>.Clear() ,  
[Dictionary](#)<[string](#), [object](#)>.ContainsKey([string](#)) ,  
[Dictionary](#)<[string](#), [object](#)>.ContainsValue([object](#)) ,  
[Dictionary](#)<[string](#), [object](#)>.EnsureCapacity([int](#)) ,  
[Dictionary](#)<[string](#), [object](#)>.GetEnumerator() ,  
[Dictionary](#)<[string](#), [object](#)>.OnDeserialization([object](#)) ,  
[Dictionary](#)<[string](#), [object](#)>.Remove([string](#)) ,  
[Dictionary](#)<[string](#), [object](#)>.Remove([string](#), [out object](#)) ,  
[Dictionary](#)<[string](#), [object](#)>.TrimExcess() , [Dictionary](#)<[string](#), [object](#)>.TrimExcess([int](#)) ,  
[Dictionary](#)<[string](#), [object](#)>.TryAdd([string](#), [object](#)) ,  
[Dictionary](#)<[string](#), [object](#)>.TryGetValue([string](#), [out object](#)) ,  
[Dictionary](#)<[string](#), [object](#)>.Comparer , [Dictionary](#)<[string](#), [object](#)>.Count ,  
[Dictionary](#)<[string](#), [object](#)>.this[[string](#)] , [Dictionary](#)<[string](#), [object](#)>.Keys ,  
[Dictionary](#)<[string](#), [object](#)>.Values , [object.Equals](#)([object](#)) ,  
[object.Equals](#)([object](#), [object](#)) , [object.GetHashCode](#)() , [object.GetType](#)() ,  
[object.MemberwiseClone](#)() , [object.ReferenceEquals](#)([object](#), [object](#)) , [object.ToString](#)()

# Constructors

## DataObject()

```
public DataObject()
```

## DataObject(int)

```
public DataObject(int capacity)
```

### Parameters

capacity [int](#)

# Struct DataObject.BooleanDecoder

Namespace: [CofyDev.Xml.Doc](#)

Assembly: CatSweeper.dll

```
public struct DataObject.BooleanDecoder : DataObject.IStringDecoder
```

## Implements

[DataObject.IStringDecoder](#)

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Properties

### propertyType

```
public Type propertyType { get; }
```

## Property Value

[Type](#)

## Methods

### TryDecode(string, out object)

```
public bool TryDecode(string raw, out object decoded)
```

## Parameters

raw [string](#)

decoded [object](#)

Returns

[bool](#) ↗

# Class DataObject.Decoder

Namespace: [CofyDev.Xml.Doc](#)

Assembly: CatSweeper.dll

```
public static class DataObject.Decoder
```

## Inheritance

[object](#) ← DataObject.Decoder

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### stringDecoders

```
public static IReadOnlyDictionary<Type, DataObject.IValueDecoder> stringDecoders {  
    get; }
```

## Property Value

[IReadOnlyDictionary](#)<[Type](#) , [DataObject.IValueDecoder](#)>

## Methods

### RegisterDecoder(IValueDecoder)

```
public static void RegisterDecoder(DataObject.IValueDecoder decoder)
```

## Parameters

decoder [DataObject.IValueDecoder](#)

## TryDecode(object, Type, out object)

```
public static bool TryDecode(object raw, Type decodedType, out object decoded)
```

Parameters

raw [object](#)

decodedType [Type](#)

decoded [object](#)

Returns

[bool](#)

## TryGetDecoder(Type, out IValueDecoder)

```
public static bool TryGetDecoder(Type type, out DataObject.IValueDecoder decoder)
```

Parameters

type [Type](#)

decoder [DataObject.IValueDecoder](#)

Returns

[bool](#)

# Struct DataObject.DoubleDecoder

Namespace: [CofyDev.Xml.Doc](#)

Assembly: CatSweeper.dll

```
public struct DataObject.DoubleDecoder : DataObject.IStringDecoder
```

## Implements

[DataObject.IStringDecoder](#)

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Properties

### propertyType

```
public Type propertyType { get; }
```

## Property Value

[Type](#)

## Methods

### TryDecode(string, out object)

```
public bool TryDecode(string raw, out object decoded)
```

## Parameters

raw [string](#)

decoded [object](#)

Returns

bool ↗

# Struct DataObject.EnumDecoder

Namespace: [CofyDev.Xml.Doc](#)

Assembly: CatSweeper.dll

```
public struct DataObject.EnumDecoder : DataObject.IStringDecoder
```

## Implements

[DataObject.IStringDecoder](#)

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Constructors

### EnumDecoder(Type)

```
public EnumDecoder(Type enumType)
```

## Parameters

enumType [Type](#)

## Properties

### propertyType

```
public Type propertyType { get; }
```

## Property Value

[Type](#)

# Methods

## TryDecode(string, out object)

```
public bool TryDecode(string raw, out object decoded)
```

### Parameters

raw [string](#)

decoded [object](#)

### Returns

[bool](#)

# Struct DataObject.FloatDecoder

Namespace: [CofyDev.Xml.Doc](#)

Assembly: CatSweeper.dll

```
public struct DataObject.FloatDecoder : DataObject.IStringDecoder
```

## Implements

[DataObject.IStringDecoder](#)

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Properties

### propertyType

```
public Type propertyType { get; }
```

## Property Value

[Type](#)

## Methods

### TryDecode(string, out object)

```
public bool TryDecode(string raw, out object decoded)
```

## Parameters

raw [string](#)

decoded [object](#)

Returns

bool ↗

# Interface DataObject.IStringDecoder

Namespace: [CofyDev.Xml.Doc](#)

Assembly: CatSweeper.dll

```
public interface DataObject.IStringDecoder
```

## Properties

### propertyType

```
Type propertyType { get; }
```

## Property Value

[Type](#) ↗

## Methods

### TryDecode(string, out object)

```
bool TryDecode(string raw, out object decoded)
```

## Parameters

raw [string](#) ↗

decoded [object](#) ↗

## Returns

[bool](#) ↗

# Interface DataObject.IValueDecoder

Namespace: [CofyDev.Xml.Doc](#)

Assembly: CatSweeper.dll

```
public interface DataObject.IValueDecoder
```

## Properties

### valueType

```
Type valueType { get; }
```

## Property Value

[Type](#)

## Methods

### TryDecode(object, Type, out object)

```
bool TryDecode(object raw, Type decodedType, out object decoded)
```

## Parameters

raw [object](#)

decodedType [Type](#)

decoded [object](#)

## Returns

[bool](#)

# Struct DataObject.IntDecoder

Namespace: [CofyDev.Xml.Doc](#)

Assembly: CatSweeper.dll

```
public struct DataObject.IntDecoder : DataObject.IStringDecoder
```

## Implements

[DataObject.IStringDecoder](#)

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Properties

### propertyType

```
public Type propertyType { get; }
```

## Property Value

[Type](#)

## Methods

### TryDecode(string, out object)

```
public bool TryDecode(string raw, out object decoded)
```

## Parameters

raw [string](#)

decoded [object](#)

Returns

bool ↗

# Class DataObject.ListValueDecoder

Namespace: [CofyDev.Xml.Doc](#)

Assembly: CatSweeper.dll

```
public class DataObject.ListValueDecoder : DataObject.IValueDecoder
```

## Inheritance

[object](#) ← DataObject.ListValueDecoder

## Implements

[DataObject.IValueDecoder](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### valueType

```
public Type valueType { get; }
```

Property Value

[Type](#)

## Methods

### TryDecode(object, Type, out object)

```
public bool TryDecode(object raw, Type decodedType, out object decoded)
```

## Parameters

`raw` [object](#)

`decodedType` [Type](#)

`decoded` [object](#)

Returns

[bool](#)

# Struct DataObject.StringDecoder

Namespace: [CofyDev.Xml.Doc](#)

Assembly: CatSweeper.dll

```
public struct DataObject.StringDecoder : DataObject.IStringDecoder
```

## Implements

[DataObject.IStringDecoder](#)

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Properties

### propertyType

```
public Type propertyType { get; }
```

## Property Value

[Type](#)

## Methods

### TryDecode(string, out object)

```
public bool TryDecode(string raw, out object decoded)
```

## Parameters

raw [string](#)

decoded [object](#)

Returns

bool ↗

# Class DataObject.StringValueDecoder

Namespace: [CofyDev.Xml.Doc](#)

Assembly: CatSweeper.dll

```
public class DataObject.StringValueDecoder : DataObject.IValueDecoder
```

## Inheritance

[object](#) ← DataObject.StringValueDecoder

## Implements

[DataObject.IValueDecoder](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Constructors

StringValueDecoder()

```
public StringValueDecoder()
```

## Properties

stringDecoders

```
public static IReadOnlyDictionary<Type, DataObject.IStringDecoder> stringDecoders {  
    get; }
```

Property Value

[IReadOnlyDictionary](#)<[Type](#) , [DataObject.IStringDecoder](#)>

## valueType

```
public Type valueType { get; }
```

### Property Value

Type ↗

## Methods

### RegisterStringDecoder(IStringDecoder)

```
public static void RegisterStringDecoder(DataObject.IStringDecoder decoder)
```

### Parameters

decoder [DataObject.IStringDecoder](#)

### TryDecode(object, Type, out object)

```
public bool TryDecode(object raw, Type decodedType, out object decoded)
```

### Parameters

raw [object](#) ↗

decodedType [Type](#) ↗

decoded [object](#) ↗

### Returns

[bool](#) ↗

### TryGetStringDecoder(Type, out IStringDecoder)

```
public static bool TryGetStringDecoder(Type type, out DataObject.IStringDecoder decoder)
```

## Parameters

type [Type](#)

decoder [DataObject.IStringDecoder](#)

## Returns

[bool](#)

# Class DataObjectEncoder

Namespace: [CofyDev.Xml.Doc](#)

Assembly: CatSweeper.dll

```
public class DataObjectEncoder : IDisposable
```

## Inheritance

[object](#) ← DataObjectEncoder

## Implements

[IDisposable](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

DecodeAs<T>(DataObject, Action< PropertyInfo, object, KeyValuePair<string, object>>)

```
public virtual T DecodeAs<T>(DataObject dataObject, Action< PropertyInfo, object, KeyValuePair<string, object>> propertyDecodeSetter)
```

## Parameters

dataObject [DataObject](#)

propertyDecodeSetter [Action](#)< [PropertyInfo](#), [object](#), [KeyValuePair](#)<[string](#), [object](#)>>

## Returns

T

## Type Parameters

T

## Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

## Encode(object)

```
public virtual DataObject Encode(object obj)
```

### Parameters

[obj](#) [object](#) ↗

### Returns

[DataObject](#)

# Class DataObjectExtension

Namespace: [CofyDev.Xml.Doc](#)

Assembly: CatSweeper.dll

```
public static class DataObjectExtension
```

## Inheritance

[object](#) ← DataObjectExtension

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

**SetDecodePropertyValue(PropertyInfo, object,  
KeyValuePair<string, object>)**

```
public static void SetDecodePropertyValue(PropertyInfo propertyInfo, object  
propertyObject, KeyValuePair<string, object> kvp)
```

### Parameters

propertyInfo  [PropertyInfo](#)

propertyObject  [object](#)

kvp  [KeyValuePair<string, object>](#)

# Namespace cfEngine

## Classes

[Optional](#)

[Res](#)

[Validation](#)

## Structs

[Failure<T>](#)

[Optional<T>](#)

[Pending<T>](#)

[Res<TOk, TErr>](#)

[Success<T>](#)

## Interfaces

[Validation<T>](#)

## Enums

[ValidationState](#)

# Struct Failure<T>

Namespace: [cfEngine](#)

Assembly: CatSweeper.dll

```
public readonly struct Failure<T> : Validation<T>
```

## Type Parameters

T

### Implements

[Validation<T>](#)

### Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Constructors

### Failure(Exception)

```
public Failure(Exception exception)
```

## Parameters

exception [Exception](#)

## Properties

### state

```
public ValidationState state { get; }
```

Property Value

[ValidationState](#)

## Methods

### GetException()

```
public Exception GetException()
```

Returns

[Exception](#)

### GetValue()

```
public T GetValue()
```

Returns

T

# Class Optional

Namespace: [cfEngine](#)

Assembly: CatSweeper.dll

```
public static class Optional
```

## Inheritance

[object](#) ↗ ← Optional

## Inherited Members

[object.Equals\(object\)](#) ↗ , [object.Equals\(object, object\)](#) ↗ , [object.GetHashCode\(\)](#) ↗ ,  
[object.GetType\(\)](#) ↗ , [object.MemberwiseClone\(\)](#) ↗ , [object.ReferenceEquals\(object, object\)](#) ↗ ,  
[object.ToString\(\)](#) ↗

## Methods

### None<T>()

```
public static Optional<T> None<T>() where T : class
```

Returns

[Optional](#)<T>

Type Parameters

T

### Some<T>(T)

```
public static Optional<T> Some<T>(T value) where T : class
```

Parameters

**value** T

Returns

[Optional<T>](#)

Type Parameters

T

# Struct Optional<T>

Namespace: [cfEngine](#)

Assembly: CatSweeper.dll

```
public readonly struct Optional<T>
```

## Type Parameters

T

### Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

## Methods

### None()

```
public static Optional<T> None()
```

#### Returns

[Optional<T>](#)

### Some(T)

```
public static Optional<T> Some(T value)
```

#### Parameters

value T

#### Returns

## [Optional<T>](#)

### ToString()

Returns the fully qualified type name of this instance.

```
public override string ToString()
```

Returns

[string](#)

The fully qualified type name.

### TryGetValue(out T)

```
public bool TryGetValue(out T value)
```

Parameters

**value** T

Returns

[bool](#)

## Operators

### implicit operator Optional<T>(T)

```
public static implicit operator Optional<T>(T value)
```

Parameters

**value** T

Returns

[Optional<T>](#)

# Struct Pending<T>

Namespace: [cfEngine](#)

Assembly: CatSweeper.dll

```
public readonly struct Pending<T> : Validation<T>
```

## Type Parameters

T

### Implements

[Validation<T>](#)

### Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Constructors

### Pending(T)

```
public Pending(T value)
```

## Parameters

value T

## Properties

### state

```
public ValidationState state { get; }
```

Property Value

[ValidationState](#)

## Methods

### GetException()

```
public Exception GetException()
```

Returns

[Exception](#)

### GetValue()

```
public T GetValue()
```

Returns

T

# Class Res

Namespace: [cfEngine](#)

Assembly: CatSweeper.dll

```
public static class Res
```

## Inheritance

[object](#) ← Res

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### Err<T>(Exception)

```
public static Res<T, Exception> Err<T>(Exception error)
```

#### Parameters

error [Exception](#)

#### Returns

[Res<T, Exception>](#)

#### Type Parameters

T

### Ok<T>(T)

```
public static Res<T, Exception> Ok<T>(T value)
```

Parameters

value T

Returns

[Res<T, Exception>](#)

Type Parameters

T

# Struct Res<TOK, TERR>

Namespace: [cfEngine](#)

Assembly: CatSweeper.dll

```
public readonly struct Res<TOK, TERR>
```

## Type Parameters

TOK

TERR

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

## Properties

IsErr

```
public bool IsErr { get; }
```

Property Value

[bool](#)

IsOk

```
public bool IsOk { get; }
```

Property Value

[bool](#)

## error

```
public TErr error { get; }
```

### Property Value

TErr

## value

```
public TOk value { get; }
```

### Property Value

TOk

## Methods

### Err(TErr)

```
public static Res<TOk, TErr> Err(TErr error)
```

### Parameters

**error** TErr

### Returns

[Res](#)<TOk, TErr>

### MapErr<T>(Func<TErr, T>)

```
public Res<TOk, T> MapErr<T>(Func<TErr, T> mapper)
```

Parameters

`mapper` [Func](#)<TErr, T>

Returns

[Res](#)<TOk, T>

Type Parameters

T

Map<T>(Func<TOk, T>)

```
public Res<T, TErr> Map<T>(Func<TOk, T> mapper)
```

Parameters

`mapper` [Func](#)<TOk, T>

Returns

[Res](#)<T, TErr>

Type Parameters

T

Ok(TOk)

```
public static Res<TOk, TErr> Ok(TOk value)
```

Parameters

`value` TOk

Returns

[Res](#)<TOK, TERR>

## OnErr(Action<TERR>)

public Res<TOK, TERR> **OnErr**(Action<TERR> action)

Parameters

action [Action](#)<TERR>

Returns

[Res](#)<TOK, TERR>

## OnOk(Action<TOK>)

public Res<TOK, TERR> **OnOk**(Action<TOK> action)

Parameters

action [Action](#)<TOK>

Returns

[Res](#)<TOK, TERR>

## ToString()

Returns the fully qualified type name of this instance.

public override string **ToString**()

Returns

[string](#)

The fully qualified type name.

## TryGetError(out TErr)

```
public bool TryGetError(out TErr error)
```

Parameters

**error** TErr

Returns

bool ↗

## TryGetValue(out TOk)

```
public bool TryGetValue(out TOk value)
```

Parameters

**value** TOk

Returns

bool ↗

# Struct Success<T>

Namespace: [cfEngine](#)

Assembly: CatSweeper.dll

```
public readonly struct Success<T> : Validation<T>
```

## Type Parameters

T

### Implements

[Validation<T>](#)

### Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Constructors

### Success(T)

```
public Success(T value)
```

## Parameters

value T

## Properties

### state

```
public ValidationState state { get; }
```

Property Value

[ValidationState](#)

## Methods

### GetException()

```
public Exception GetException()
```

Returns

[Exception](#)

### GetValue()

```
public T GetValue()
```

Returns

T

# Class Validation

Namespace: [cfEngine](#)

Assembly: CatSweeper.dll

```
public static class Validation
```

## Inheritance

[object](#) ← Validation

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

ContinueWith<T>(Task<Validation<T>>,  
Action<Validation<T>>)

```
public static Task<Validation<T>> ContinueWith<T>(this Task<Validation<T>> task,  
Action<Validation<T>> continuation)
```

### Parameters

task [Task](#)<Validation<T>>

continuation [Action](#)<Validation<T>>

### Returns

[Task](#)<Validation<T>>

### Type Parameters

T

## Failure<T>(Exception)

```
public static Validation<T> Failure<T>(Exception exception)
```

Parameters

exception [Exception](#)

Returns

[Validation](#)<T>

Type Parameters

T

## Pending<T>(T)

```
public static Validation<T> Pending<T>(T value)
```

Parameters

value T

Returns

[Validation](#)<T>

Type Parameters

T

## Success<T>(T)

```
public static Validation<T> Success<T>(T value)
```

Parameters

**value** T

Returns

[Validation](#)<T>

Type Parameters

T

# Enum ValidationState

Namespace: [cfEngine](#)

Assembly: CatSweeper.dll

```
public enum ValidationState : byte
```

## Extension Methods

[EnumExtension.hasFlag\(Enum, Enum\)](#)

## Fields

Failure = 2

Pending = 0

Success = 1

# Interface Validation<T>

Namespace: [cfEngine](#)

Assembly: CatSweeper.dll

```
public interface Validation<out T>
```

Type Parameters

T

## Properties

state

```
ValidationState state { get; }
```

Property Value

[ValidationState](#)

## Methods

GetException()

```
Exception GetException()
```

Returns

[Exception](#)

GetValue()

T `GetValue()`

Returns

T

# Namespace cfEngine.Asset

## Classes

[AssetHandle](#)

[AssetHandle<T>](#)

[AssetManager<TBaseObject>](#)

# Class AssetHandle

Namespace: [cfEngine.Asset](#)

Assembly: CatSweeper.dll

```
public abstract class AssetHandle
```

## Inheritance

[object](#) ← AssetHandle

## Derived

[AssetHandle<T>](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Constructors

### AssetHandle(Action)

```
public AssetHandle(Action releaseAction)
```

## Parameters

releaseAction [Action](#)

## Fields

### ReleaseAction

```
public readonly Action ReleaseAction
```

## Field Value

## Action ↗

# Class AssetHandle<T>

Namespace: [cfEngine.Asset](#)

Assembly: CatSweeper.dll

```
public class AssetHandle<T> : AssetHandle where T : class
```

## Type Parameters

T

## Inheritance

[object](#) ← [AssetHandle](#) ← AssetHandle<T>

## Inherited Members

[AssetHandle.ReleaseAction](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### AssetHandle(T, Action)

```
public AssetHandle(T asset, Action releaseAction)
```

## Parameters

asset T

releaseAction [Action](#)

## Fields

### Asset

```
public readonly WeakReference<T> Asset
```

## Field Value

[WeakReference](#) <T>

# Class AssetManager<TBaseObject>

Namespace: [cfEngine.Asset](#)

Assembly: CatSweeper.dll

```
public abstract class AssetManager<TBaseObject> : IService, IDisposable where  
    TBaseObject : class
```

## Type Parameters

**TBaseObject**

## Inheritance

[object](#) ← AssetManager<TBaseObject>

## Implements

[IService](#), [IDisposable](#)

## Derived

[ResourceAssetManager](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

### LoadAsync<T>(string, CancellationToken)

```
public Task<T> LoadAsync<T>(string path, CancellationToken token = default) where T : class, TBaseObject
```

## Parameters

path [string](#)

token [CancellationToken](#)

## Returns

[Task](#)<T>

## Type Parameters

T

## Load<T>(string)

```
public T Load<T>(string path) where T : class, TBaseObject
```

## Parameters

path [string](#)

## Returns

T

## Type Parameters

T

## TryGetAsset<T>(string, out T)

```
public bool TryGetAsset<T>(string path, out T asset) where T : class, TBaseObject
```

Parameters

path [string](#)

asset T

Returns

[bool](#)

Type Parameters

T

## \_LoadAsync<T>(string, CancellationToken)

```
protected abstract Task<AssetHandle<T>> _LoadAsync<T>(string path, CancellationToken  
token = default) where T : class, TBaseObject
```

Parameters

path [string](#)

token [CancellationToken](#)

Returns

[Task](#)<AssetHandle<T>>

Type Parameters

T

## \_Load<T>(string)

```
protected abstract AssetHandle<T> _Load<T>(string path) where T : class, TBaseObject
```

Parameters

path [string](#)

Returns

[AssetHandle](#)<T>

Type Parameters

T

# Namespace cfEngine.Command

## Classes

[CommandService](#)

[CommandService.RegisterOnInitializedAttribute](#)

[ICommand.HintAttribute](#)

## Interfaces

[ICommand](#)

# Class CommandService

Namespace: [cfEngine.Command](#)

Assembly: CatSweeper.dll

```
public static class CommandService
```

## Inheritance

[object](#) ← CommandService

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### commandMap

```
public static Dictionary<string[], ICommand> commandMap
```

## Field Value

[Dictionary](#)<[string](#)[], [ICommand](#)>

## Methods

### RegisterCommand(ICommand, params string[])

```
public static void RegisterCommand(ICommand resolver, params string[] commandKeys)
```

## Parameters

resolver [ICommand](#)

commandKeys [string](#)[]

## UnregisterCommand(params string[])

`public static void UnregisterCommand(params string[] command)`

### Parameters

command [string](#)[]

# Class CommandService.RegisterOnInitializedAttribute

Namespace: [cfEngine.Command](#)

Assembly: CatSweeper.dll

```
public class CommandService.RegisterOnInitializedAttribute : Attribute
```

## Inheritance

[object](#) ← [Attribute](#) ← CommandService.RegisterOnInitializedAttribute

## Inherited Members

[Attribute.Equals\(object\)](#) , [Attribute.GetCustomAttribute\(Assembly, Type\)](#) ,  
[Attribute.GetCustomAttribute\(Assembly, Type, bool\)](#) ,  
[Attribute.GetCustomAttribute\(MemberInfo, Type\)](#) ,  
[Attribute.GetCustomAttribute\(MemberInfo, Type, bool\)](#) ,  
[Attribute.GetCustomAttribute\(Module, Type\)](#) ,  
[Attribute.GetCustomAttribute\(Module, Type, bool\)](#) ,  
[Attribute.GetCustomAttribute\(ParameterInfo, Type\)](#) ,  
[Attribute.GetCustomAttribute\(ParameterInfo, Type, bool\)](#) ,  
[Attribute.GetCustomAttributes\(Assembly\)](#) ,  
[Attribute.GetCustomAttributes\(Assembly, bool\)](#) ,  
[Attribute.GetCustomAttributes\(Assembly, Type\)](#) ,  
[Attribute.GetCustomAttributes\(Assembly, Type, bool\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo, bool\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo, Type\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo, Type, bool\)](#) ,  
[Attribute.GetCustomAttributes\(Module\)](#) , [Attribute.GetCustomAttributes\(Module, bool\)](#) ,  
[Attribute.GetCustomAttributes\(Module, Type\)](#) ,  
[Attribute.GetCustomAttributes\(Module, Type, bool\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo, bool\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo, Type\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo, Type, bool\)](#) , [Attribute.GetHashCode\(\)](#) ,  
[Attribute.IsDefaultAttribute\(\)](#) , [Attribute.IsDefined\(Assembly, Type\)](#) ,

[Attribute.IsDefined\(Assembly, Type, bool\)](#) , [Attribute.IsDefined\(MemberInfo, Type\)](#) ,  
[Attribute.IsDefined\(MemberInfo, Type, bool\)](#) , [Attribute.IsDefined\(Module, Type\)](#) ,  
[Attribute.IsDefined\(Module, Type, bool\)](#) , [Attribute.IsDefined\(ParameterInfo, Type\)](#) ,  
[Attribute.IsDefined\(ParameterInfo, Type, bool\)](#) , [Attribute.Match\(object\)](#) ,  
[Attribute.TypeId](#) , [object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### RegisterOnInitializedAttribute(string)

```
public RegisterOnInitializedAttribute(string registerMethodName)
```

#### Parameters

registerMethodName [string](#)

## Fields

### registerMethodName

```
public readonly string registerMethodName
```

#### Field Value

[string](#)

# Interface ICommand

Namespace: [cfEngine.Command](#)

Assembly: CatSweeper.dll

```
public interface ICommand
```

## Methods

Execute(IReadOnlyDictionary<string, string>)

```
void Execute(IReadOnlyDictionary<string, string> args)
```

### Parameters

args [IReadOnlyDictionary<string, string>](#)

# Class ICommand.HintAttribute

Namespace: [cfEngine.Command](#)

Assembly: CatSweeper.dll

```
public class ICommand.HintAttribute : Attribute
```

## Inheritance

[object](#) ← [Attribute](#) ← ICommand.HintAttribute

## Inherited Members

[Attribute.Equals\(object\)](#) , [Attribute.GetCustomAttribute\(Assembly, Type\)](#) ,  
[Attribute.GetCustomAttribute\(Assembly, Type, bool\)](#) ,  
[Attribute.GetCustomAttribute\(MemberInfo, Type\)](#) ,  
[Attribute.GetCustomAttribute\(MemberInfo, Type, bool\)](#) ,  
[Attribute.GetCustomAttribute\(Module, Type\)](#) ,  
[Attribute.GetCustomAttribute\(Module, Type, bool\)](#) ,  
[Attribute.GetCustomAttribute\(ParameterInfo, Type\)](#) ,  
[Attribute.GetCustomAttribute\(ParameterInfo, Type, bool\)](#) ,  
[Attribute.GetCustomAttributes\(Assembly\)](#) ,  
[Attribute.GetCustomAttributes\(Assembly, bool\)](#) ,  
[Attribute.GetCustomAttributes\(Assembly, Type\)](#) ,  
[Attribute.GetCustomAttributes\(Assembly, Type, bool\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo, bool\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo, Type\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo, Type, bool\)](#) ,  
[Attribute.GetCustomAttributes\(Module\)](#) , [Attribute.GetCustomAttributes\(Module, bool\)](#) ,  
[Attribute.GetCustomAttributes\(Module, Type\)](#) ,  
[Attribute.GetCustomAttributes\(Module, Type, bool\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo, bool\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo, Type\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo, Type, bool\)](#) , [Attribute.GetHashCode\(\)](#) ,  
[Attribute.IsDefaultAttribute\(\)](#) , [Attribute.IsDefined\(Assembly, Type\)](#) ,  
[Attribute.IsDefined\(Assembly, Type, bool\)](#) , [Attribute.IsDefined\(MemberInfo, Type\)](#) ,  
[Attribute.IsDefined\(MemberInfo, Type, bool\)](#) , [Attribute.IsDefined\(Module, Type\)](#) ,  
[Attribute.IsDefined\(Module, Type, bool\)](#) , [Attribute.IsDefined\(ParameterInfo, Type\)](#) ,  
[Attribute.IsDefined\(ParameterInfo, Type, bool\)](#) , [Attribute.Match\(object\)](#) ,

[Attribute.TypeId](#) , [object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### HintAttribute(string)

```
public HintAttribute(string description)
```

## Parameters

description [string](#)

## Fields

### description

```
public readonly string description
```

## Field Value

[string](#)

# Namespace cfEngine.Core

## Classes

[Game](#)

[GameExtension](#)

[ServiceName](#)

[UserDataAdapter](#)

[UserDataManager](#)

[UserDataManager.JsonContextMap](#)

## Structs

[UserDataManager.DeleteSaveCommand](#)

## Interfaces

[IRuntimeSavable](#)

[IUserData](#)

# Class Game

Namespace: [cfEngine.Core](#)

Assembly: CatSweeper.dll

```
public class Game : ServiceLocator, IServiceLocator, IEnumerable<IService>,  
IEnumerable, IDisposable
```

## Inheritance

[object](#) ← [ServiceLocator](#) ← Game

## Implements

[IServiceLocator](#), [IEnumerable](#)<[IService](#)>, [IEnumerable](#), [IDisposable](#)

## Inherited Members

[ServiceLocator.Register<T>\(T, string\)](#) , [ServiceLocator.Unregister<T>\(T\)](#) ,  
[ServiceLocator.Unregister\(string\)](#) , [ServiceLocator.GetService<T>\(\)](#) ,  
[ServiceLocator.GetService<T>\(string\)](#) , [ServiceLocator.Dispose\(\)](#) ,  
[ServiceLocator.GetEnumerator\(\)](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Extension Methods

[GameExtension.GetStageService\(Game\)](#) ,  
[GameExtension.WithStage\(Game, StageService\)](#) , [GameExtension.GetAsset\(Game\)](#) ,  
[GameExtension.GetAuth\(Game\)](#) , [GameExtension.GetGameStateMachine\(Game\)](#) ,  
[GameExtension.GetInfo\(Game\)](#) , [GameExtension.GetInventory\(Game\)](#) ,  
[GameExtension.GetPoolManager\(Game\)](#) , [GameExtension.GetPool<T>\(Game, string\)](#) ,  
[GameExtension.GetSceneManager<TScene>\(Game\)](#) ,  
[GameExtension.GetUserData\(Game\)](#) ,  
[GameExtension.WithAsset\(Game, ResourceAssetManager\)](#) ,  
[GameExtension.WithAuthService\(Game, IAuthService\)](#) ,  
[GameExtension.WithGameStateMachine\(Game, GameStateMachine\)](#) ,  
[GameExtension.WithInfo\(Game, InfoLayer\)](#) ,  
[GameExtension.WithInventory\(Game, IInventoryService\)](#) ,  
[GameExtension.WithPoolManager\(Game, PoolManager\)](#) ,  
[GameExtension.WithSceneManager<TScene>\(Game, ISceneManager<TScene>\)](#) ,  
[GameExtension.WithUserData\(Game, UserDataManager\)](#)

# Properties

## Current

```
public static Game Current { get; }
```

### Property Value

[Game](#)

## TaskToken

```
public static CancellationToken TaskToken { get; }
```

### Property Value

[CancellationToken](#)

## Methods

### SetCurrent(Game)

```
public static void SetCurrent(Game game)
```

### Parameters

game [Game](#)

# Class GameExtension

Namespace: [cfEngine.Core](#)

Assembly: CatSweeper.dll

```
public static class GameExtension
```

## Inheritance

[object](#) ← GameExtension

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### GetAsset(Game)

```
public static AssetManager<Resource> GetAsset(this Game game)
```

#### Parameters

game [Game](#)

#### Returns

[AssetManager](#)<Resource>

### GetAuth(Game)

```
public static IAuthService GetAuth(this Game game)
```

#### Parameters

game [Game](#)

Returns

[IAuthService](#)

## GetGameStateMachine(Game)

```
public static GameStateMachine GetGameStateMachine(this Game game)
```

Parameters

game [Game](#)

Returns

[GameStateMachine](#)

## GetInfo(Game)

```
public static InfoLayer GetInfo(this Game game)
```

Parameters

game [Game](#)

Returns

[InfoLayer](#)

## GetInventory(Game)

```
public static InventoryService GetInventory(this Game game)
```

Parameters

game [Game](#)

Returns

[InventoryService](#)

## GetPoolManager(Game)

```
public static PoolManager GetPoolManager(this Game game)
```

Parameters

game [Game](#)

Returns

[PoolManager](#)

## GetPool<T>(Game, string)

```
public static Res<T, Exception> GetPool<T>(this Game game, string poolKey) where T : IObjectPool
```

Parameters

game [Game](#)

poolKey [string](#)

Returns

[Res<T, Exception>](#)

Type Parameters

T

## GetSceneManager<TScene>(Game)

```
public static ISceneManager<TScene> GetSceneManager<TScene>(this Game game)
```

Parameters

game [Game](#)

Returns

[ISceneManager](#)<TScene>

Type Parameters

TScene

## GetUserData(Game)

```
public static UserDataManager GetUserData(this Game game)
```

Parameters

game [Game](#)

Returns

[UserDataManager](#)

## WithAsset(Game, ResourceAssetManager)

```
public static Game WithAsset(this Game game, ResourceAssetManager assetManager)
```

Parameters

game [Game](#)

assetManager [ResourceAssetManager](#)

Returns

[Game](#)

## WithAuthService(Game, IAuthService)

```
public static Game WithAuthService(this Game game, IAuthService service)
```

Parameters

game [Game](#)

service [IAuthService](#)

Returns

[Game](#)

## WithGameStateMachine(Game, GameStateMachine)

```
public static Game WithGameStateMachine(this Game game, GameStateMachine service)
```

Parameters

game [Game](#)

service [GameStateMachine](#)

Returns

[Game](#)

## WithInfo(Game, InfoLayer)

```
public static Game WithInfo(this Game game, InfoLayer service)
```

Parameters

game [Game](#)

service [InfoLayer](#)

Returns

[Game](#)

## WithInventory(Game, IInventoryService)

```
public static Game WithInventory(this Game game, IInventoryService service)
```

Parameters

game [Game](#)

service [IInventoryService](#)

Returns

[Game](#)

## WithPoolManager(Game, PoolManager)

```
public static Game WithPoolManager(this Game game, PoolManager service)
```

Parameters

game [Game](#)

service [PoolManager](#)

Returns

[Game](#)

## WithSceneManager<TScene>(Game, ISceneManager<TScene>)

```
public static Game WithSceneManager<TScene>(this Game game,  
ISceneManager<TScene> sceneManager)
```

### Parameters

game [Game](#)

sceneManager [ISceneManager<TScene>](#)

### Returns

[Game](#)

### Type Parameters

TScene

## WithData(Game, UserDataManager)

```
public static Game WithUserData(this Game game, UserDataManager service)
```

### Parameters

game [Game](#)

service [UserDataManager](#)

### Returns

[Game](#)

# Interface IRuntimeSavable

Namespace: [cfEngine.Core](#)

Assembly: CatSweeper.dll

```
public interface IRuntimeSavable : IDisposable
```

## Inherited Members

[IDisposable.Dispose\(\)](#)

## Methods

### Initialize(IUserData)

```
void Initialize(IUserData userData)
```

#### Parameters

userData [IUserData](#)

### SetSaveData(Dictionary<string, object>)

```
void SetSaveData(Dictionary<string, object> dataMap)
```

#### Parameters

dataMap [Dictionary](#)<[string](#), [object](#)>

# Interface IUserData

Namespace: [cfEngine.Core](#)

Assembly: CatSweeper.dll

```
public interface IUserData
```

## Methods

### GetContext<T>(string)

```
T GetContext<T>(string contextKey)
```

#### Parameters

contextKey [string](#)

#### Returns

T

#### Type Parameters

T

### TryGetContext<T>(string, out T)

```
bool TryGetContext<T>(string contextKey, out T context)
```

#### Parameters

contextKey [string](#)

context T

Returns

[bool](#) ↴

Type Parameters

T

# Class ServiceName

Namespace: [cfEngine.Core](#)

Assembly: CatSweeper.dll

```
public static class ServiceName
```

## Inheritance

[object](#) ← ServiceName

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### Auth

```
public const string Auth = "Auth"
```

#### Field Value

[string](#)

### Info

```
public const string Info = "Info"
```

#### Field Value

[string](#)

### Inventory

```
public const string Inventory = "Inventory"
```

Field Value

[string](#)

Pool

```
public const string Pool = "Pool"
```

Field Value

[string](#)

UserDataManager

```
public const string UserDataManager = "UserDataManager"
```

Field Value

[string](#)

# Class UserDataKey

Namespace: [cfEngine.Core](#)

Assembly: CatSweeper.dll

```
public class UserDataKey
```

## Inheritance

[object](#) ← UserDataKey

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### Inventory

```
public const string Inventory = "Inventory"
```

Field Value

[string](#)

### SaveVersion

```
public const string SaveVersion = "SaveVersion"
```

Field Value

[string](#)

## Statistic

```
public const string Statistic = "Statistic"
```

## Field Value

[string](#) ↗

# Class UserDataManager

Namespace: [cfEngine.Core](#)

Assembly: CatSweeper.dll

```
public class UserDataManager : IService, IDisposable
```

## Inheritance

[object](#) ← UserDataManager

## Implements

[IService](#), [IDisposable](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Constructors

UserDataManager(IStorage, ISerializer)

```
public UserDataManager(IStorage storage, ISerializer serializer)
```

## Parameters

storage [IStorage](#)

serializer [ISerializer](#)

## Methods

DeleteSave()

```
public void DeleteSave()
```

## Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

## InitializeSavables()

```
public void InitializeSavables()
```

## LoadDataMap(CancellationToken)

```
public Task<IUserData> LoadDataMap(CancellationToken token = default)
```

Parameters

token  [CancellationToken](#)

Returns

[Task](#)<IUserData>

## Register(IRuntimeSavable)

```
public void Register(IRuntimeSavable savable)
```

Parameters

savable  [IRuntimeSavable](#)

## SaveAsync(IReadOnlyDictionary<string, object>, CancellationToken)

```
public Task SaveAsync(IReadOnlyDictionary<string, object> dataMap, CancellationToken token = default)
```

Parameters

dataMap [IReadOnlyDictionary](#)<[string](#), [object](#)>

token [CancellationToken](#)

Returns

[Task](#)

## SaveAsync(CancellationToken)

```
public Task SaveAsync(CancellationToken token = default)
```

Parameters

token [CancellationToken](#)

Returns

[Task](#)

## TriggerSave()

```
public void TriggerSave()
```

# Struct UserDataManager.DeleteSaveCommand

Namespace: [cfEngine.Core](#)

Assembly: CatSweeper.dll

```
[CommandService.RegisterOnInitialized("Register")]
public struct UserDataManager.DeleteSaveCommand : ICommand
```

## Implements

[ICommand](#)

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Methods

### Execute(IReadOnlyDictionary<string, string>)

```
public void Execute(IReadOnlyDictionary<string, string> args)
```

## Parameters

args [IReadOnlyDictionary<string, string>](#)

## Register()

```
public static void Register()
```

# Class UserDataManager.JsonContextMap

Namespace: [cfEngine.Core](#)

Assembly: CatSweeper.dll

```
public class UserDataManager.JsonContextMap : IUserData
```

## Inheritance

[object](#) ← UserDataManager.JsonContextMap

## Implements

[IUserData](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Constructors

JsonContextMap(IReadOnlyDictionary<string, JsonObject>)

```
public JsonContextMap(IReadOnlyDictionary<string, JsonObject> dataMap)
```

## Parameters

dataMap [IReadOnlyDictionary](#)<[string](#), [JsonObject](#)>

## Fields

Empty

```
public static UserDataManager.JsonContextMap Empty
```

## Field Value

[UserDataManager.JsonContextMap](#)

## Methods

### GetContext<T>(string)

```
public T GetContext<T>(string contextKey)
```

#### Parameters

contextKey [string](#)

#### Returns

T

#### Type Parameters

T

### TryGetContext<T>(string, out T)

```
public bool TryGetContext<T>(string contextKey, out T context)
```

#### Parameters

contextKey [string](#)

context T

#### Returns

[bool](#)

#### Type Parameters



# Namespace cfEngine.DataStructure

## Classes

[MemoryDictionary<TMemoryKey, TValue>](#)

[ReadOnlyMemoryComparer<T>](#)

[WeakReferenceListPool<T>](#)

[WeakReferenceList<T>](#)

# Class MemoryDictionary<TMemoryKey, TValue>

Namespace: [cfEngine.DataStructure](#)

Assembly: CatSweeper.dll

```
public class MemoryDictionary<TMemoryKey, TValue> :  
Dictionary<ReadOnlyMemory<TMemoryKey>, TValue>,  
IDictionary<ReadOnlyMemory<TMemoryKey>, TValue>,  
ICollection<KeyValuePair<ReadOnlyMemory<TMemoryKey>, TValue>>,  
IReadOnlyDictionary<ReadOnlyMemory<TMemoryKey>, TValue>,  
IReadOnlyCollection<KeyValuePair<ReadOnlyMemory<TMemoryKey>, TValue>>,  
IEnumerable<KeyValuePair<ReadOnlyMemory<TMemoryKey>, TValue>>, IDictionary,  
ICollection, IEnumerable, IDeserializationCallback, ISerializable
```

## Type Parameters

TMemoryKey

TValue

## Inheritance

[object](#) ← [Dictionary](#)<[ReadOnlyMemory](#)<TMemoryKey>, TValue> ←  
MemoryDictionary<TMemoryKey, TValue>

## Implements

[IDictionary](#)<[ReadOnlyMemory](#)<TMemoryKey>, TValue>,  
[ICollection](#)<[KeyValuePair](#)<[ReadOnlyMemory](#)<TMemoryKey>, TValue>>,  
[IReadOnlyDictionary](#)<[ReadOnlyMemory](#)<TMemoryKey>, TValue>,  
[IReadOnlyCollection](#)<[KeyValuePair](#)<[ReadOnlyMemory](#)<TMemoryKey>, TValue>>,  
[IEnumerable](#)<[KeyValuePair](#)<[ReadOnlyMemory](#)<TMemoryKey>, TValue>>,  
[IDictionary](#), [ICollection](#), [IEnumerable](#), [IDeserializationCallback](#), [ISerializable](#)

## Inherited Members

[Dictionary](#)<[ReadOnlyMemory](#)<TMemoryKey>,  
TValue>.Add([ReadOnlyMemory](#)<TMemoryKey>, TValue)>,  
[Dictionary](#)<[ReadOnlyMemory](#)<TMemoryKey>, TValue>.Clear()>,  
[Dictionary](#)<[ReadOnlyMemory](#)<TMemoryKey>,  
TValue>.ContainsKey([ReadOnlyMemory](#)<TMemoryKey>)>,  
[Dictionary](#)<[ReadOnlyMemory](#)<TMemoryKey>, TValue>.ContainsValue(TValue)>,

[Dictionary<ReadOnlyMemory<TMemoryKey>, TValue>.EnsureCapacity\(int\)](#) ,  
[Dictionary<ReadOnlyMemory<TMemoryKey>, TValue>.GetEnumerator\(\)](#) ,  
[Dictionary<ReadOnlyMemory<TMemoryKey>, TValue>.OnDeserialization\(object\)](#) ,  
[Dictionary<ReadOnlyMemory<TMemoryKey>, TValue>.Remove\(ReadOnlyMemory<TMemoryKey>\)](#) ,  
[Dictionary<ReadOnlyMemory<TMemoryKey>, TValue>.Remove\(ReadOnlyMemory<TMemoryKey>, out TValue\)](#) ,  
[Dictionary<ReadOnlyMemory<TMemoryKey>, TValue>.TrimExcess\(\)](#) ,  
[Dictionary<ReadOnlyMemory<TMemoryKey>, TValue>.TrimExcess\(int\)](#) ,  
[Dictionary<ReadOnlyMemory<TMemoryKey>, TValue>.TryAdd\(ReadOnlyMemory<TMemoryKey>, TValue\)](#) ,  
[Dictionary<ReadOnlyMemory<TMemoryKey>, TValue>.TryGetValue\(ReadOnlyMemory<TMemoryKey>, out TValue\)](#) ,  
[Dictionary<ReadOnlyMemory<TMemoryKey>, TValue>.Comparer](#) ,  
[Dictionary<ReadOnlyMemory<TMemoryKey>, TValue>.Count](#) ,  
[Dictionary<ReadOnlyMemory<TMemoryKey>, TValue>.this\[ReadOnlyMemory<TMemoryKey>\]](#) ,  
[Dictionary<ReadOnlyMemory<TMemoryKey>, TValue>.Keys](#) ,  
[Dictionary<ReadOnlyMemory<TMemoryKey>, TValue>.Values](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### MemoryDictionary()

```
public MemoryDictionary()
```

## Methods

### Create()

```
public static MemoryDictionary<TMemoryKey, TValue> Create()
```

## Returns

[MemoryDictionary<TMemoryKey, TValue>](#)



# Class ReadOnlyMemoryComparer<T>

Namespace: [cfEngine.DataStructure](#)

Assembly: CatSweeper.dll

```
public class ReadOnlyMemoryComparer<T> : IEqualityComparer<ReadOnlyMemory<T>>
```

## Type Parameters

T

### Inheritance

[object](#) ← `ReadOnlyMemoryComparer<T>`

### Implements

[IEqualityComparer](#)<[ReadOnlyMemory](#)<T>>

### Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### Instance

```
public static ReadOnlyMemoryComparer<T> Instance { get; }
```

### Property Value

[ReadOnlyMemoryComparer](#)<T>

## Methods

### Equals(ReadOnlyMemory<T>, ReadOnlyMemory<T>)

Determines whether the specified objects are equal.

```
public bool Equals(ReadOnlyMemory<T> x, ReadOnlyMemory<T> y)
```

## Parameters

x [ReadOnlyMemory](#)<T>

The first object of type T to compare.

y [ReadOnlyMemory](#)<T>

The second object of type T to compare.

## Returns

[bool](#)

[true](#) if the specified objects are equal; otherwise, [false](#).

## GetHashCode(ReadOnlyMemory<T>)

Returns a hash code for the specified object.

```
public int GetHashCode(ReadOnlyMemory<T> obj)
```

## Parameters

obj [ReadOnlyMemory](#)<T>

The [object](#) for which a hash code is to be returned.

## Returns

[int](#)

A hash code for the specified object.

## Exceptions

[ArgumentNullException](#)

The type of obj is a reference type and obj is [null](#).



# Class WeakReferenceListPool<T>

Namespace: [cfEngine.DataStructure](#)

Assembly: CatSweeper.dll

```
public class WeakReferenceListPool<T> : ObjectPool<WeakReferenceList<T>>,  
IObjectPool, IDisposable where T : class
```

## Type Parameters

T

## Inheritance

[object](#) ← [ObjectPool<WeakReferenceList<T>>](#) ← [WeakReferenceListPool<T>](#)

## Implements

[IObjectPool](#), [IDisposable](#)

## Inherited Members

[ObjectPool<WeakReferenceList<T>>.Queue](#) , [ObjectPool<WeakReferenceList<T>>.Get\(\)](#) ,  
[ObjectPool<WeakReferenceList<T>>.Get\(out WeakReferenceList<T>\)](#) ,  
[ObjectPool<WeakReferenceList<T>>.Release\(WeakReferenceList<T>\)](#) ,  
[ObjectPool<WeakReferenceList<T>>.Dispose\(\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### WeakReferenceListPool()

```
public WeakReferenceListPool()
```

# Class WeakReferenceList<T>

Namespace: [cfEngine.DataStructure](#)

Assembly: CatSweeper.dll

```
public class WeakReferenceList<T> : List<WeakReference<T>>, IList<WeakReference<T>>,  
ICollection<WeakReference<T>>, IReadOnlyList<WeakReference<T>>,  
IReadOnlyCollection<WeakReference<T>>, IEnumerable<WeakReference<T>>, IList,  
ICollection, IEnumerable where T : class
```

## Type Parameters

T

## Inheritance

[object](#) ← [List](#)<[WeakReference](#)<T>> ← WeakReferenceList<T>

## Implements

[IList](#)<[WeakReference](#)<T>>, [ICollection](#)<[WeakReference](#)<T>>,  
[IReadOnlyList](#)<[WeakReference](#)<T>>, [IReadOnlyCollection](#)<[WeakReference](#)<T>>,  
[IEnumerable](#)<[WeakReference](#)<T>>, [IList](#), [ICollection](#), [IEnumerable](#)

## Inherited Members

[List<WeakReference<T>>.Add\(WeakReference<T>\)](#) ,  
[List<WeakReference<T>>.AddRange\(IEnumerable<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.AsReadOnly\(\)](#) ,  
[List<WeakReference<T>>.BinarySearch\(int, int, WeakReference<T>, IComparer<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.BinarySearch\(WeakReference<T>\)](#) ,  
[List<WeakReference<T>>.BinarySearch\(WeakReference<T>, IComparer<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.Clear\(\)](#) ,  
[List<WeakReference<T>>.Contains\(WeakReference<T>\)](#) ,  
[List<WeakReference<T>>.ConvertAll<TOutput>\(Converter<WeakReference<T>, TOOutput>\)](#) ,  
[List<WeakReference<T>>.CopyTo\(int, WeakReference<T>\[\], int, int\)](#) ,  
[List<WeakReference<T>>.CopyTo\(WeakReference<T>\[\]\)](#) ,  
[List<WeakReference<T>>.CopyTo\(WeakReference<T>\[\], int\)](#) ,  
[List<WeakReference<T>>.EnsureCapacity\(int\)](#) ,  
[List<WeakReference<T>>.Exists\(Predicate<WeakReference<T>>\)](#) ,

[List<WeakReference<T>>.Find\(Predicate<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.FindAll\(Predicate<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.FindIndex\(int, int, Predicate<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.FindIndex\(int, Predicate<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.FindIndex\(Predicate<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.FindLast\(Predicate<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.FindLastIndex\(int, int, Predicate<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.FindLastIndex\(int, Predicate<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.FindLastIndex\(Predicate<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.ForEach\(Action<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.GetEnumerator\(\)](#) ,  
[List<WeakReference<T>>.GetRange\(int, int\)](#) ,  
[List<WeakReference<T>>.IndexOf\(WeakReference<T>\)](#) ,  
[List<WeakReference<T>>.IndexOf\(WeakReference<T>, int\)](#) ,  
[List<WeakReference<T>>.IndexOf\(WeakReference<T>, int, int\)](#) ,  
[List<WeakReference<T>>.Insert\(int, WeakReference<T>\)](#) ,  
[List<WeakReference<T>>.InsertRange\(int, IEnumerable<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.LastIndexOf\(WeakReference<T>\)](#) ,  
[List<WeakReference<T>>.LastIndexOf\(WeakReference<T>, int\)](#) ,  
[List<WeakReference<T>>.LastIndexOf\(WeakReference<T>, int, int\)](#) ,  
[List<WeakReference<T>>.Remove\(WeakReference<T>\)](#) ,  
[List<WeakReference<T>>.RemoveAll\(Predicate<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.RemoveAt\(int\)](#) ,  
[List<WeakReference<T>>.RemoveRange\(int, int\)](#) ,  
[List<WeakReference<T>>.Reverse\(\)](#) , [List<WeakReference<T>>.Reverse\(int, int\)](#) ,  
[List<WeakReference<T>>.Slice\(int, int\)](#) , [List<WeakReference<T>>.Sort\(\)](#) ,  
[List<WeakReference<T>>.Sort\(IComparer<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.Sort\(Comparison<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.Sort\(int, int, IComparer<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.ToArray\(\)](#) , [List<WeakReference<T>>.TrimExcess\(\)](#) ,  
[List<WeakReference<T>>.TrueForAll\(Predicate<WeakReference<T>>\)](#) ,  
[List<WeakReference<T>>.Capacity](#) , [List<WeakReference<T>>.Count](#) ,  
[List<WeakReference<T>>.this\[int\]](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Extension Methods

[ListExtension.EnsureCapacity<T>\(List<T>, int\)](#)

# Methods

## Add(T)

```
public void Add(T item)
```

### Parameters

item T

## Create()

```
public static WeakReferenceList<T> Create()
```

### Returns

[WeakReferenceList<T>](#)

## Remove(T)

```
public void Remove(T item)
```

### Parameters

item T

# Namespace cfEngine.DataStructure.test

## Classes

[TestWeakReferenceList](#)

[TestWeakReferenceList.TestObject](#)

# Class TestWeakReferenceList

Namespace: [cfEngine.DataStructure.test](#)

Assembly: CatSweeper.dll

```
[TestFixture]
public class TestWeakReferenceList
```

## Inheritance

[object](#) ← TestWeakReferenceList

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### Test\_Add()

```
[Test]
public void Test_Add()
```

### Test\_Create()

```
[Test]
public void Test_Create()
```

# Class TestWeakReferenceList.TestObject

Namespace: [cfEngine.DataStructure.test](#)

Assembly: CatSweeper.dll

```
public class TestWeakReferenceList.TestObject
```

## Inheritance

[object](#) ← TestWeakReferenceList.TestObject

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Namespace cfEngine.Extension

## Classes

[EnumExtension](#)

[ListExtension](#)

[TaskExtension](#)

# Class EnumExtension

Namespace: [cfEngine.Extension](#)

Assembly: CatSweeper.dll

```
public static class EnumExtension
```

## Inheritance

[object](#) ← EnumExtension

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### hasFlag(Enum, Enum)

```
public static bool hasFlag(this Enum target, Enum flag)
```

#### Parameters

target [Enum](#)

flag [Enum](#)

#### Returns

[bool](#)

# Class ListExtension

Namespace: [cfEngine.Extension](#)

Assembly: CatSweeper.dll

```
public static class ListExtension
```

## Inheritance

[object](#) ← ListExtension

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### EnsureCapacity<T>(List<T>, int)

```
public static void EnsureCapacity<T>(this List<T> list, int capacity)
```

#### Parameters

list [List](#)<T>

capacity [int](#)

#### Type Parameters

T

# Class TaskExtension

Namespace: [cfEngine.Extension](#)

Assembly: CatSweeper.dll

```
public static class TaskExtension
```

## Inheritance

[object](#) ← TaskExtension

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

ContinueWithSynchronized(Task, Action<Task>, CancellationToken)

```
public static Task ContinueWithSynchronized(this Task t, Action<Task> action,  
CancellationToken token = default)
```

### Parameters

t [Task](#)

action [Action](#)<[Task](#)>

token [CancellationToken](#)

### Returns

[Task](#)

## ContinueWithSynchronized<T>(Task<T>, Action<Task<T>>, CancellationToken)

```
public static Task ContinueWithSynchronized<T>(this Task<T> t, Action<Task<T>>
action, CancellationToken token = default)
```

### Parameters

t [Task](#)<T>

action [Action](#)<[Task](#)<T>>>

token [CancellationToken](#)

### Returns

[Task](#)

### Type Parameters

T

## ContinueWithSynchronized<T>(Task<T>, Func<Task<T>, T>, CancellationToken)

```
public static Task<T> ContinueWithSynchronized<T>(this Task<T> t, Func<Task<T>, T>
action, CancellationToken token = default)
```

### Parameters

t [Task](#)<T>

action [Func](#)<[Task](#)<T>, T>>

token [CancellationToken](#)

### Returns

[Task](#)<T>

Type Parameters

T

## DisposeIfCompleted(Task)

```
public static void DisposeIfCompleted(this Task t)
```

Parameters

t [Task](#)

## LogAggregateException(AggregateException)

```
public static void LogAggregateException(this AggregateException ex)
```

Parameters

ex [AggregateException](#)

## LogIfFaulted(Task)

```
public static Task LogIfFaulted(this Task t)
```

Parameters

t [Task](#)

Returns

[Task](#)

## LogIfFaulted<T>(Task<T>)

```
public static Task<T> LogIfFaulted<T>(this Task<T> t)
```

Parameters

t [Task](#)<T>

Returns

[Task](#)<T>

Type Parameters

T

# Namespace cfEngine.IO

## Classes

[LocalFileStorage](#)

## Interfaces

[IStorage](#)

# Interface IStorage

Namespace: [cfEngine.IO](#)

Assembly: CatSweeper.dll

```
public interface IStorage : IDisposable
```

## Inherited Members

[IDisposable.Dispose\(\)](#)

## Methods

### CopyFile(string, string, bool)

```
void CopyFile(string relativeFrom, string relativeTo, bool overwrite = false)
```

#### Parameters

relativeFrom [string](#)

relativeTo [string](#)

overwrite [bool](#)

### CreateStream(string, bool)

```
Stream CreateStream(string relativePath, bool useAsync)
```

#### Parameters

relativePath [string](#)

useAsync [bool](#)

#### Returns

## DeleteFile(string)

```
void DeleteFile(string relativePath)
```

### Parameters

relativePath [string](#)

## GetFiles(string)

```
string[] GetFiles(string searchPattern)
```

### Parameters

searchPattern [string](#)

### Returns

[string](#)[]

## GetFiles(string, string)

```
string[] GetFiles(string directory, string searchPattern)
```

### Parameters

directory [string](#)

searchPattern [string](#)

### Returns

[string](#)[]

## IsFileExist(string)

```
bool IsFileExist(string relativePath)
```

Parameters

relativePath [string](#)

Returns

[bool](#)

## IsStorageExist()

```
bool IsStorageExist()
```

Returns

[bool](#)

## LoadBytes(string)

```
byte[] LoadBytes(string relativePath)
```

Parameters

relativePath [string](#)

Returns

[byte](#)[]

## LoadBytesAsync(string, CancellationToken)

```
Task<byte[]> LoadBytesAsync(string relativePath, CancellationToken token = default)
```

Parameters

relativePath [string](#)

token [CancellationToken](#)

Returns

[Task](#)<[byte](#)[]>

## Save(string, byte[])

```
void Save(string relativeFilePath, byte[] data)
```

Parameters

relativeFilePath [string](#)

data [byte](#)[]

## Save(string, string)

```
void Save(string relativeFilePath, string data)
```

Parameters

relativeFilePath [string](#)

data [string](#)

## SaveAsync(string, byte[], CancellationToken)

```
Task SaveAsync(string relativeFilePath, byte[] data, CancellationToken token  
= default)
```

Parameters

`relativeFilePath` [string](#)

`data` [byte](#)[]

`token` [CancellationToken](#)

Returns

[Task](#)

## SaveAsync(string, string, CancellationToken)

```
Task SaveAsync(string relativeFilePath, string data, CancellationToken token  
= default)
```

Parameters

`relativeFilePath` [string](#)

`data` [string](#)

`token` [CancellationToken](#)

Returns

[Task](#)

# Class LocalFileStorage

Namespace: [cfEngine.IO](#)

Assembly: CatSweeper.dll

```
public class LocalFileStorage : IStorage, IDisposable
```

## Inheritance

[object](#) ← LocalFileStorage

## Implements

[IStorage](#), [IDisposable](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Constructors

### LocalFileStorage(string)

```
public LocalFileStorage(string storagePath)
```

## Parameters

storagePath [string](#)

## Methods

### CopyFile(string, string, bool)

```
public void CopyFile(string relativeFrom, string relativeTo, bool overwrite = false)
```

## Parameters

`relativeFrom` [string](#)

`relativeTo` [string](#)

`overwrite` [bool](#)

## CreateStream(string, bool)

```
public Stream CreateStream(string relativePath, bool useAsync)
```

### Parameters

`relativePath` [string](#)

`useAsync` [bool](#)

### Returns

[Stream](#)

## DeleteFile(string)

```
public void DeleteFile(string relativePath)
```

### Parameters

`relativePath` [string](#)

## Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

## GetFiles(string)

```
public string[] GetFiles(string searchPattern)
```

Parameters

searchPattern [string](#)

Returns

[string](#)[]

## GetFiles(string, string)

```
public string[] GetFiles(string directory, string searchPattern)
```

Parameters

directory [string](#)

searchPattern [string](#)

Returns

[string](#)[]

## IsFileExist(string)

```
public bool IsFileExist(string relativePath)
```

Parameters

relativePath [string](#)

Returns

[bool](#)

## IsStorageExist()

```
public bool IsStorageExist()
```

Returns

[bool](#)

## LoadBytes(string)

```
public byte[] LoadBytes(string relativePath)
```

Parameters

[relativePath](#) [string](#)

Returns

[byte](#)[]

## LoadBytesAsync(string, CancellationToken)

```
public Task<byte[]> LoadBytesAsync(string relativePath, CancellationToken  
cancellationToken = default)
```

Parameters

[relativePath](#) [string](#)

[cancellationToken](#) [CancellationToken](#)

Returns

[Task](#)<[byte](#)[]>

## Save(string, byte[])

```
public void Save(string relativeFilePath, byte[] data)
```

### Parameters

relativeFilePath [string](#)

data [byte](#)[]

## Save(string, string)

```
public void Save(string relativeFilePath, string data)
```

### Parameters

relativeFilePath [string](#)

data [string](#)

## SaveAsync(string, byte[], CancellationToken)

```
public Task SaveAsync(string relativeFilePath, byte[] data, CancellationToken token  
= default)
```

### Parameters

relativeFilePath [string](#)

data [byte](#)[]

token [CancellationToken](#)

### Returns

[Task](#)

## SaveAsync(string, string, CancellationToken)

```
public Task SaveAsync(string relativeFilePath, string data, CancellationToken token  
= default)
```

### Parameters

relativeFilePath [string](#)

data [string](#)

token [CancellationToken](#)

### Returns

[Task](#)

# Namespace cfEngine.Info

## Classes

[ConfigInfoManager<TKey, TInfo>](#)

[ExcelJsonLoader<TInfo>](#)

[InfoLayer](#)

[InfoManager](#)

[InfoUtil](#)

[JsonElementDecoder](#)

[SerializationLoader<TInfo>](#)

## Interfaces

[IInfoManager](#)

[IValueLoader<TInfo>](#)

# Class ConfigInfoManager<TKey, TInfo>

Namespace: [cfEngine.Info](#)

Assembly: CatSweeper.dll

```
public abstract class ConfigInfoManager<TKey, TInfo> : InfoManager, IInfoManager,  
IDisposable where TKey : notnull where TInfo : class
```

## Type Parameters

TKey

TInfo

## Inheritance

[object](#) ← [InfoManager](#) ← ConfigInfoManager<TKey, TInfo>

## Implements

[IInfoManager](#), [IDisposable](#)

## Derived

[GameSettingInfoManager](#), [StageInfoManager](#), [StageMapInfoManager](#), [InventoryInfoManager](#)

## Inherited Members

[InfoManager.OnLoadCompleted\(\)](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### ConfigInfoManager(IValueLoader<TInfo>)

```
protected ConfigInfoManager(IValueLoader<TInfo> loader)
```

## Parameters

loader [IValueLoader](#)<TInfo>

# Properties

## infoType

```
public override Type infoType { get; }
```

### Property Value

[Type](#)

## keyFn

```
protected abstract Func<TInfo, TKey> keyFn { get; }
```

### Property Value

[Func](#)<TInfo, TKey>

## valueMap

```
public IReadOnlyDictionary<TKey, TInfo> valueMap { get; }
```

### Property Value

[IReadOnlyDictionary](#)<TKey, TInfo>

# Methods

## AddValue(TInfo)

```
public void AddValue(TInfo value)
```

### Parameters

`value` `TInfo`

## Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public override void Dispose()
```

## GetAllValue()

```
public override IEnumerable<object> GetAllValue()
```

Returns

[`IEnumerable`](#)<[`object`](#)>

## LoadInfo()

```
public override void LoadInfo()
```

## LoadInfoAsync(CancellationToken)

```
public override Task LoadInfoAsync(CancellationToken cancellationToken)
```

Parameters

`cancellationToken` [`CancellationToken`](#)

Returns

[`Task`](#)

## TryGetValue(TKey, out TInfo)

```
public bool TryGetValue(TKey key, out TInfo value)
```

### Parameters

key TKey

value TInfo

### Returns

[bool](#)

# Class ExcelJsonLoader<TInfo>

Namespace: [cfEngine.Info](#)

Assembly: CatSweeper.dll

```
public class ExcelJsonLoader<TInfo> : IValueLoader<TInfo>
```

## Type Parameters

TInfo

### Inheritance

[object](#) ← ExcelJsonLoader<TInfo>

### Implements

[IValueLoader](#)<TInfo>

### Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Constructors

### ExcelJsonLoader(IStorage, DataObjectEncoder)

```
public ExcelJsonLoader(IStorage storage, DataObjectEncoder encoder)
```

## Parameters

storage [IStorage](#)

encoder [DataObjectEncoder](#)

## Methods

## Load(out List<TInfo>)

```
public ObjectPool<List<TInfo>>.Handle Load(out List<TInfo> values)
```

### Parameters

values [List](#)<TInfo>

### Returns

[ObjectPool](#)<[List](#)<TInfo>>.Handle

## LoadAsync(CancellationToken)

```
public Task<List<TInfo>> LoadAsync(CancellationToken cancellationToken)
```

### Parameters

cancellationToken  [CancellationToken](#)

### Returns

[Task](#)<[List](#)<TInfo>>

# Interface IInfoManager

Namespace: [cfEngine.Info](#)

Assembly: CatSweeper.dll

```
public interface IInfoManager : IDisposable
```

## Inherited Members

[IDisposable.Dispose\(\)](#)

## Properties

### infoType

```
Type infoType { get; }
```

Property Value

[Type](#)

## Methods

### GetAllValue()

```
IEnumerable<Object> GetAllValue()
```

Returns

[IEnumerable](#)<[Object](#)>

### LoadInfo()

```
void LoadInfo()
```

# LoadInfoAsync(CancellationToken)

Task `LoadInfoAsync(CancellationToken cancellationToken)`

## Parameters

`cancellationToken`  [CancellationToken ↗](#)

## Returns

[Task ↗](#)

# Interface IValueLoader<TInfo>

Namespace: [cfEngine.Info](#)

Assembly: CatSweeper.dll

```
public interface IValueLoader<TInfo>
```

## Type Parameters

TInfo

## Methods

### Load(out List<TInfo>)

```
ObjectPool<List<TInfo>>.Handle Load(out List<TInfo> values)
```

#### Parameters

values [List](#)<TInfo>

#### Returns

[ObjectPool](#)<[List](#)<TInfo>>.Handle

### LoadAsync(CancellationToken)

```
Task<List<TInfo>> LoadAsync(CancellationToken cancellationToken)
```

#### Parameters

cancellationToken  [CancellationToken](#)

#### Returns

[Task](#) <[List](#)<TInfo>>

# Class InfoLayer

Namespace: [cfEngine.Info](#)

Assembly: CatSweeper.dll

```
public class InfoLayer : IService, IDisposable
```

## Inheritance

[object](#) ← InfoLayer

## Implements

[IService](#), [IDisposable](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Properties

## InfoMap

```
public IReadOnlyDictionary<Type, IInfoManager> InfoMap { get; }
```

### Property Value

[IReadOnlyDictionary](#)<[Type](#), [IInfoManager](#)>

# Methods

## Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

## Get<TInfo>()

```
public TInfo Get<TInfo>() where TInfo : InfoManager
```

Returns

TInfo

Type Parameters

TInfo

## LoadInfoAsync()

```
public IEnumerable<Task> LoadInfoAsync()
```

Returns

IEnumerable<Task>

## RegisterInfo(IInfoManager)

```
public void RegisterInfo(IInfoManager infoManager)
```

Parameters

infoManager IInfoManager

## TryGetInfo<TInfo>(out TInfo)

```
public bool TryGetInfo<TInfo>(out TInfo infoManager) where TInfo : InfoManager
```

Parameters

infoManager TInfo

Returns

[bool](#) ↗

Type Parameters

[TInfo](#)

# Class InfoManager

Namespace: [cfEngine.Info](#)

Assembly: CatSweeper.dll

```
public abstract class InfoManager : IInfoManager, IDisposable
```

## Inheritance

[object](#) ← InfoManager

## Implements

[IInfoManager](#), [IDisposable](#)

## Derived

[ConfigInfoManager<TKey, TInfo>](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Properties

## infoType

```
public abstract Type infoType { get; }
```

Property Value

[Type](#)

# Methods

## Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public virtual void Dispose()
```

## GetAllValue()

```
public abstract IEnumerable<object> GetAllValue()
```

Returns

[IEnumerable](#)<[object](#)>

## LoadInfo()

```
public abstract void LoadInfo()
```

## LoadInfoAsync(CancellationToken)

```
public abstract Task LoadInfoAsync(CancellationToken cancellationToken)
```

Parameters

cancellationToken  [CancellationToken](#)

Returns

[Task](#)

## OnLoadCompleted()

```
protected virtual void OnLoadCompleted()
```

# Class InfoUtil

Namespace: [cfEngine.Info](#)

Assembly: CatSweeper.dll

```
public static class InfoUtil
```

## Inheritance

[object](#) ← InfoUtil

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

ParseCommand(ReadOnlySpan<char>, out  
IReadOnlyList<string>, char, char, char)

```
public static string ParseCommand(ReadOnlySpan<char> command, out  
IReadOnlyList<string> args, char separator = ',', char open = '(', char close = ')')
```

### Parameters

command [ReadOnlySpan](#)<char>

args [IReadOnlyList](#)<string>

separator [char](#)

open [char](#)

close [char](#)

### Returns

[string](#)

**ParseTypeCommand(ReadOnlySpan<char>, out  
IReadOnlyList<string>, char, char, char)**

```
public static Type ParseTypeCommand(ReadOnlySpan<char> command, out  
IReadOnlyList<string> args, char separator = ',', char open = '(', char close = ')')
```

Parameters

command [ReadOnlySpan<char>](#)

args [IReadOnlyList<string>](#)

separator [char](#)

open [char](#)

close [char](#)

Returns

[Type](#)

# Class JsonElementDecoder

Namespace: [cfEngine.Info](#)

Assembly: CatSweeper.dll

```
public class JsonElementDecoder : DataObject.IValueDecoder
```

## Inheritance

[object](#) ← JsonElementDecoder

## Implements

[DataObject.IValueDecoder](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### valueType

```
public Type valueType { get; }
```

Property Value

[Type](#)

## Methods

### TryDecode(object, Type, out object)

```
public bool TryDecode(object raw, Type decodedType, out object decoded)
```

## Parameters

`raw` [object](#)

`decodedType` [Type](#)

`decoded` [object](#)

Returns

[bool](#)

# Class SerializationLoader<TInfo>

Namespace: [cfEngine.Info](#)

Assembly: CatSweeper.dll

```
public class SerializationLoader<TInfo> : IValueLoader<TInfo>
```

## Type Parameters

TInfo

### Inheritance

[object](#) ← SerializationLoader<TInfo>

### Implements

[IValueLoader](#)<TInfo>

### Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Constructors

### SerializationLoader(IStorage, ISerializer)

```
public SerializationLoader(IStorage storage, ISerializer serializer)
```

## Parameters

storage [IStorage](#)

serializer [ISerializer](#)

## Methods

## Load(out List<TInfo>)

```
public ObjectPool<List<TInfo>>.Handle Load(out List<TInfo> values)
```

### Parameters

values [List](#)<TInfo>

### Returns

[ObjectPool](#)<[List](#)<TInfo>>.Handle

## LoadAsync(CancellationToken)

```
public Task<List<TInfo>> LoadAsync(CancellationToken cancellationToken)
```

### Parameters

cancellationToken  [CancellationToken](#)

### Returns

[Task](#)<[List](#)<TInfo>>

# Namespace cfEngine.Logging

## Classes

[Log](#)

## Interfaces

[ILogger](#)

## Enums

[LogLevel](#)

[LogType](#)

# Interface ILogger

Namespace: [cfEngine.Logging](#)

Assembly: CatSweeper.dll

```
public interface ILogger
```

## Methods

### Asset(bool, object)

```
void Asset(bool condition, object context = null)
```

#### Parameters

condition [bool](#)

context [object](#)

### LogDebug(string, object)

```
void LogDebug(string message, object context = null)
```

#### Parameters

message [string](#)

context [object](#)

### .LogError(string, object)

```
void LogError(string message, object context = null)
```

Parameters

message [string](#)

context [object](#)

## LogException(Exception, object)

```
void LogException(Exception ex, object message = null)
```

Parameters

ex [Exception](#)

message [object](#)

## LogInfo(string, object)

```
void LogInfo(string message, object context = null)
```

Parameters

message [string](#)

context [object](#)

## LogWarning(string, object)

```
void LogWarning(string message, object context = null)
```

Parameters

message [string](#)

context [object](#)

# Class Log

Namespace: [cfEngine.Logging](#)

Assembly: CatSweeper.dll

```
public static class Log
```

## Inheritance

[object](#) ← Log

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### Asset(bool, object)

```
public static void Asset(bool condition, object context = null)
```

#### Parameters

condition [bool](#)

context [object](#)

### LogDebug(string, object)

```
public static void LogDebug(string message, object context = null)
```

#### Parameters

message [string](#)

context [object](#)

## .LogError(string, object)

```
public static void LogError(string message, object context = null)
```

### Parameters

message [string](#)

context [object](#)

## LogException(Exception, string)

```
public static void LogException(Exception ex, string message = null)
```

### Parameters

ex [Exception](#)

message [string](#)

## LogInfo(string, object)

```
public static void LogInfo(string message, object context = null)
```

### Parameters

message [string](#)

context [object](#)

## LogWarning(string, object)

```
public static void LogWarning(string message, object context = null)
```

Parameters

message [string](#)

context [object](#)

## SetLogLevel(LogLevel)

```
public static void SetLogLevel(LogLevel logLevel)
```

Parameters

logLevel [LogLevel](#)

## SetLogger(ILogger)

```
public static void SetLogger(ILogger logger)
```

Parameters

logger [ILogger](#)

# Enum LogLevel

Namespace: [cfEngine.Logging](#)

Assembly: CatSweeper.dll

```
public enum LogLevel
```

## Extension Methods

[EnumExtension.hasFlag\(Enum, Enum\)](#)

## Fields

Debug = 63

Error = 3

Info = 15

Verbose = 15

Warn = 7

# Enum LogType

Namespace: [cfEngine.Logging](#)

Assembly: CatSweeper.dll

```
[Flags]  
public enum LogType
```

## Extension Methods

[EnumExtension.hasFlag\(Enum, Enum\)](#).

## Fields

Assert = 16

Debug = 32

Error = 1

Exception = 2

Info = 8

Warning = 4

# Namespace cfEngine.Pooling

## Classes

[ListPool<T>](#)

[ObjectPool<T>](#)

[PoolManager](#)

[StringBuilderPool](#)

## Structs

[ObjectPool<T>.Handle](#)

## Interfaces

[IObjectPool](#)

# Interface IObjectPool

Namespace: [cfEngine.Pooling](#)

Assembly: CatSweeper.dll

```
public interface IObjectPool : IDisposable
```

## Inherited Members

[IDisposable.Dispose\(\)](#) ↗

# Class ListPool<T>

Namespace: [cfEngine.Pooling](#)

Assembly: CatSweeper.dll

```
public class ListPool<T> : ObjectPool<List<T>>, IObjectPool, IDisposable
```

## Type Parameters

T

### Inheritance

[object](#) ← [ObjectPool<List>](#)<T> < ListPool<T>

### Implements

[IObjectPool](#), [IDisposable](#)

### Inherited Members

[ObjectPool<List>.Queue](#) , [ObjectPool<List>.Get\(\)](#) ,  
[ObjectPool<List>.Get\(out List\)](#) , [ObjectPool<List>.Release\(List\)](#) ,  
[ObjectPool<List>.Dispose\(\)](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### ListPool()

```
public ListPool()
```

## Properties

### Default

```
public static ListPool<T> Default { get; }
```

Property Value

[ListPool](#)<T>

# Class ObjectPool<T>

Namespace: [cfEngine.Pooling](#)

Assembly: CatSweeper.dll

```
public class ObjectPool<T> : IObjectPool, IDisposable where T : class
```

## Type Parameters

T

### Inheritance

[object](#) ← ObjectPool<T>

### Implements

[IObjectPool](#), [IDisposable](#)

### Derived

[WeakReferenceListPool<T>](#), [ListPool<T>](#), [StringBuilderPool](#)

### Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#),  
[object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#),  
[object.ToString\(\)](#)

## Constructors

### ObjectPool(Func<T>, Action<T>, Action<T>, int)

```
public ObjectPool(Func<T> createMethod, Action<T> getAction, Action<T>
releaseAction, int warmupSize = 0)
```

## Parameters

createMethod [Func](#)<T>

getAction [Action](#)<T>

`releaseAction` [Action](#)<T>

`warmupSize` [int](#)

## Fields

### Queue

`protected readonly Queue<T> Queue`

## Field Value

[Queue](#)<T>

## Methods

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

`public virtual void Dispose()`

### Get()

`public virtual T Get()`

## Returns

T

### Get(out T)

`public virtual ObjectPool<T>.Handle Get(out T value)`

Parameters

**value** T

Returns

[ObjectPool<T>.Handle](#)

**Release(T)**

```
public virtual void Release(T obj)
```

Parameters

**obj** T

# Struct ObjectPool<T>.Handle

Namespace: [cfEngine.Pooling](#)

Assembly: CatSweeper.dll

```
public struct ObjectPool<T>.Handle : IDisposable
```

## Implements

[IDisposable](#)

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Constructors

### Handle(Action<T>, T)

```
public Handle(Action<T> releaseAction, T obj)
```

## Parameters

releaseAction [Action](#)<T>

obj T

## Properties

### Empty

```
public static ObjectPool<T>.Handle Empty { get; }
```

## Property Value

[ObjectPool<T>.Handle](#)

# Methods

## Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

# Class PoolManager

Namespace: [cfEngine.Pooling](#)

Assembly: CatSweeper.dll

```
public class PoolManager : IService, IDisposable
```

## Inheritance

[object](#) ← PoolManager

## Implements

[IService](#), [IDisposable](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### AddPool<T>(string, T)

```
public void AddPool<T>(string key, T pool) where T : IObjectPool
```

#### Parameters

key [string](#)

pool [T](#)

#### Type Parameters

[T](#)

## Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

## GetOrCreatePool<T>(string, Func<T>)

```
public T GetOrCreatePool<T>(string key, Func<T> createFunc) where T :  
    class, IObjectPool
```

Parameters

key [string](#)

createFunc [Func](#)<T>

Returns

T

Type Parameters

T

## TryGetPool(string, out IObjectPool)

```
public bool TryGetPool(string key, out IObjectPool pool)
```

Parameters

key [string](#)

pool [IObjectPool](#)

Returns

[bool](#)

## TryGetPool<T>(string, out T)

```
public bool TryGetPool<T>(string key, out T pool) where T : class, IObjectPool
```

### Parameters

key [string](#)

pool [T](#)

### Returns

[bool](#)

### Type Parameters

[T](#)

# Class StringBuilderPool

Namespace: [cfEngine.Pooling](#)

Assembly: CatSweeper.dll

```
public class StringBuilderPool : ObjectPool<StringBuilder>, IObjectPool, IDisposable
```

## Inheritance

[object](#) ← [ObjectPool<StringBuilder>](#) ← StringBuilderPool

## Implements

[IObjectPool](#), [IDisposable](#)

## Inherited Members

[ObjectPool<StringBuilder>.Queue](#) , [ObjectPool<StringBuilder>.Get\(\)](#) ,  
[ObjectPool<StringBuilder>.Get\(out StringBuilder\)](#) ,  
[ObjectPool<StringBuilder>.Release\(StringBuilder\)](#) , [ObjectPool<StringBuilder>.Dispose\(\)](#) ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Constructors

### StringBuilderPool()

```
public StringBuilderPool()
```

## Properties

### Default

```
public static StringBuilderPool Default { get; }
```

## Property Value

## [StringBuilderPool](#)

# Namespace cfEngine.Rx

## Classes

[CollectionEventExtension](#)

[CollectionEvents<T>](#)

[Linq](#)

[Relay](#)

[RelayBase<TDelegate>](#)

[Relay<T>](#)

[Relay<T1, T2>](#)

[RtCollection<TEEventArgs>](#)

[RtCount<T>](#)

[RtDictionary< TKey, TValue >](#)

[RtFilteredDictionary< TKey, TValue >](#)

Represents a dictionary that filters its elements based on a provided function.

[RtGroup< TKey, TValue >](#)

[RtList<T>](#)

[RtMutatedDictionaryBase< TSourceKey, TSourceValue, TKey, TValue >](#)

Represents a base class for dictionaries that support mutation and event dispatching.

[RtMutatedListBase< TOrig, TNew >](#)

[RtMutatedLocalListBase< TOrig, TNew >](#)

[RtMutatedSingleBase< TOrig, TNew >](#)

[RtObserverList<T>](#)

[RtReadOnlyDictionary< TKey, TValue >](#)

[RtReadOnlyList<T>](#)

[RtSelectKeyDictionary< TOrigKey, TSelectKey, TValue >](#)

Represents a dictionary that selects keys based on a provided function.

## [RtSelectList<TOrig, TNew>](#)

Represents a list that projects each element of a source list into a new form.

## [RtSelectLocalList<T, TSelect>](#)

## [RtSelectValueDictionary< TKey, TOrigValue, TValue>](#)

Represents a dictionary that selects values based on a provided function.

## [Rt<T>](#)

## [Subscription](#)

## [SubscriptionBinding<TDelegate>](#)

## [SubscriptionGroup](#)

## [SubscriptionHandleExtension](#)

# Interfaces

## [ICollectionEvents<T>](#)

## [IRelay<TDelegate>](#)

# Class CollectionEventExtension

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public static class CollectionEventExtension
```

## Inheritance

[object](#) ← CollectionEventExtension

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### OnChange<T>(ICollectionEvents<T>, Action)

```
public static Subscription OnChange<T>(this ICollectionEvents<T> collectionEvents,  
Action action)
```

#### Parameters

collectionEvents [ICollectionEvents<T>](#)

action [Action](#)

#### Returns

[Subscription](#)

#### Type Parameters

T

# Subscribe<T>(ICollectionEvents<T>, Action<T>, Action<T>, Action<T, T>, Action)

```
public static Subscription Subscribe<T>(this ICollectionEvents<T> collectionEvents,  
Action<T> onAdd = null, Action<T> onRemove = null, Action<T, T> onUpdate = null,  
Action onDispose = null)
```

## Parameters

collectionEvents [ICollectionEvents<T>](#)

onAdd [Action](#)<T>

onRemove [Action](#)<T>

onUpdate [Action](#)<T, T>

onDispose [Action](#)

## Returns

[Subscription](#)

## Type Parameters

T

# Class CollectionEvents<T>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public class CollectionEvents<T> : ICollectionEvents<T>
```

## Type Parameters

T

### Inheritance

[object](#) ← CollectionEvents<T>

### Implements

[ICollectionEvents<T>](#)

### Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

### Extension Methods

[CollectionEventExtension.OnChange<T>\(ICollectionEvents<T>, Action\)](#) ,  
[CollectionEventExtension.Subscribe<T>\(ICollectionEvents<T>, Action<T>, Action<T>, Action<T, T>, Action\)](#)

## Constructors

### CollectionEvents(object)

```
public CollectionEvents(object owner)
```

## Parameters

owner [object](#)

# Fields

## OnAddRelay

```
public readonly Relay<T> OnAddRelay
```

### Field Value

[Relay](#)<T>

## OnDisposeRelay

```
public readonly Relay OnDisposeRelay
```

### Field Value

[Relay](#)

## OnRemoveRelay

```
public readonly Relay<T> OnRemoveRelay
```

### Field Value

[Relay](#)<T>

## OnUpdateRelay

```
public readonly Relay<T, T> OnUpdateRelay
```

### Field Value

[Relay](#)<T, T>

# Methods

## Dispose()

```
public void Dispose()
```

## SubscribeOnAdd(Action<T>)

```
public Subscription SubscribeOnAdd(Action<T> onAdd)
```

### Parameters

onAdd [Action](#)<T>

### Returns

[Subscription](#)

## SubscribeOnDispose(Action)

```
public Subscription SubscribeOnDispose(Action onDispose)
```

### Parameters

onDispose [Action](#)

### Returns

[Subscription](#)

## SubscribeOnRemove(Action<T>)

```
public Subscription SubscribeOnRemove(Action<T> onRemove)
```

Parameters

onRemove [Action](#)<T>

Returns

[Subscription](#)

## SubscribeOnUpdate(Action<T, T>)

```
public Subscription SubscribeOnUpdate(Action<T, T> onUpdate)
```

Parameters

onUpdate [Action](#)<T, T>

Returns

[Subscription](#)

# Interface ICollectionEvents<T>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public interface ICollectionEvents<out T>
```

## Type Parameters

T

## Extension Methods

[CollectionEventExtension.OnChange<T>\(ICollectionEvents<T>, Action\)](#),  
[CollectionEventExtension.Subscribe<T>\(ICollectionEvents<T>, Action<T>, Action<T>, Action<T, T>, Action\)](#)

## Methods

### Dispose()

```
void Dispose()
```

### SubscribeOnAdd(Action<T>)

```
Subscription SubscribeOnAdd(Action<out T> onAdd)
```

## Parameters

onAdd [Action](#)<T>

## Returns

[Subscription](#)

## SubscribeOnDispose(Action)

Subscription **SubscribeOnDispose**(Action onDispose)

Parameters

onDispose [Action](#)

Returns

[Subscription](#)

## SubscribeOnRemove(Action<T>)

Subscription **SubscribeOnRemove**(Action<**out** T> onRemove)

Parameters

onRemove [Action](#)<T>

Returns

[Subscription](#)

## SubscribeOnUpdate(Action<T, T>)

Subscription **SubscribeOnUpdate**(Action<**out** T, **out** T> onUpdate)

Parameters

onUpdate [Action](#)<T, T>

Returns

[Subscription](#)

# Interface IRelay<TDelegate>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public interface IRelay<TDelegate> where TDelegate : class
```

## Type Parameters

TDelegate

## Methods

### AddListener(TDelegate)

```
Subscription AddListener(TDelegate listener)
```

#### Parameters

listener TDelegate

#### Returns

[Subscription](#)

### Contains(TDelegate)

```
bool Contains(TDelegate d)
```

#### Parameters

d TDelegate

#### Returns

[bool](#)

## RemoveAll()

`void RemoveAll()`

## RemoveListener(TDelegate)

`bool RemoveListener(TDelegate listener)`

### Parameters

`listener` TDelegate

### Returns

[bool](#)

# Class Linq

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public static class Linq
```

## Inheritance

[object](#) ← Linq

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### count<T>(RtReadOnlyList<T>)

```
public static RtCount<T> count<T>(this RtReadOnlyList<T> source)
```

#### Parameters

source [RtReadOnlyList<T>](#)

#### Returns

[RtCount<T>](#)

#### Type Parameters

T

### groupBy< TKey, TValue >(RtReadOnlyList< TValue >, Func< TValue, TKey >)

```
public static RtGroup<TKey, TValue> groupBy<TKey, TValue>(this  
RtReadOnlyList<TValue> rtList, Func<TValue, TKey> keyFn)
```

## Parameters

rtList [RtReadOnlyList<TValue>](#)

keyFn [Func<TValue, TKey>](#)

## Returns

[RtGroup<TKey, TValue>](#)

## Type Parameters

TKey

TValue

**selectKey<TOrigKey, TSelectedKey, TValue>**  
(RtReadOnlyDictionary<TOrigKey, TValue>,  
Func<TOrigKey, TSelectedKey>)

```
public static RtSelectKeyDictionary<TOrigKey, TSelectedKey, TValue>  
selectKey<TOrigKey, TSelectedKey, TValue>(this RtReadOnlyDictionary<TOrigKey,  
TValue> source, Func<TOrigKey, TSelectedKey> selectFn)
```

## Parameters

source [RtReadOnlyDictionary<TOrigKey, TValue>](#)

selectFn [Func<TOrigKey, TSelectedKey>](#)

## Returns

[RtSelectKeyDictionary<TOrigKey, TSelectedKey, TValue>](#)

## Type Parameters

TOrigKey

TSelectedKey

TValue

selectNew<T, TSelect>(RtReadOnlyList<T>, Func<T, TSelect>)

```
public static RtSelectList<T, TSelect> selectNew<T, TSelect>(this RtReadOnlyList<T> source, Func<T, TSelect> selectFn)
```

Parameters

source [RtReadOnlyList<T>](#)

selectFn [Func<T, TSelect>](#)

Returns

[RtSelectList<T, TSelect>](#)

Type Parameters

T

TSelect

selectValue<TKey, TValue, TSelectValue>  
(RtReadOnlyDictionary<TKey, TValue>, Func<TValue, TSelectValue>)

```
public static RtSelectValueDictionary<TKey, TValue, TSelectValue> selectValue<TKey, TValue, TSelectValue>(this RtReadOnlyDictionary<TKey, TValue> source, Func<TValue, TSelectValue> selectFn)
```

Parameters

source [RtReadOnlyDictionary](#)<TKey, TValue>

selectFn [Func](#)<TValue, TSelectValue>

Returns

[RtSelectValueDictionary](#)<TKey, TValue, TSelectValue>

Type Parameters

TKey

TValue

TSelectValue

**select<T, TSelect>(RtReadOnlyList<T>, Func<T, TSelect>)**

```
public static RtSelectLocalList<T, TSelect> select<T, TSelect>(this  
RtReadOnlyList<T> source, Func<T, TSelect> selectFn)
```

Parameters

source [RtReadOnlyList](#)<T>

selectFn [Func](#)<T, TSelect>

Returns

[RtSelectLocalList](#)<T, TSelect>

Type Parameters

T

TSelect

where<TKey, TValue>(RtReadOnlyDictionary<TKey, TValue>, Func<KeyValuePair<TKey, TValue>, bool>)

```
public static RtFilteredDictionary<TKey, TValue> where<TKey, TValue>(this  
RtReadOnlyDictionary<TKey, TValue> source, Func<KeyValuePair<TKey, TValue>,  
bool> filterFn)
```

## Parameters

source [RtReadOnlyDictionary<TKey, TValue>](#)

filterFn [Func<KeyValuePair<TKey, TValue>, bool>](#)

## Returns

[RtFilteredDictionary<TKey, TValue>](#)

## Type Parameters

TKey

TValue

# Class Relay

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public class Relay : RelayBase<Action>, IRelay<Action>
```

## Inheritance

[object](#) ← [RelayBase<Action>](#) ← Relay

## Implements

[IRelay<Action>](#)

## Inherited Members

[RelayBase<Action>.\\_subscriptionRefList](#) , [RelayBase<Action>.\\_cap](#) ,  
[RelayBase<Action>.\\_count](#) , [RelayBase<Action>.listenerCount](#) ,  
[RelayBase<Action>.AddListener\(Action\)](#) , [RelayBase<Action>.RemoveListener\(Action\)](#) ,  
[RelayBase<Action>.RemoveAll\(\)](#) , [RelayBase<Action>.Contains\(Action\)](#) ,  
[RelayBase<Action>.Expand\(ref WeakReference<SubscriptionBinding<Action>>\[\]\)](#) ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Constructors

### Relay(object, int)

```
public Relay(object owner, int defaultSize = 1)
```

## Parameters

owner [object](#)

defaultSize [int](#)

## Methods

## Dispatch()

```
public void Dispatch()
```

# Class RelayBase<TDelegate>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public abstract class RelayBase<TDelegate> : IRelay<TDelegate> where TDelegate  
: class
```

## Type Parameters

**TDelegate**

### Inheritance

[object](#) ← RelayBase<TDelegate>

### Implements

[IRelay](#)<TDelegate>

### Derived

[Relay](#), [Relay<T>](#), [Relay<T1, T2>](#)

### Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#),  
[object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#),  
[object.ToString\(\)](#)

## Constructors

**RelayBase(object, int)**

```
public RelayBase(object owner, int defaultSize = 1)
```

## Parameters

**owner** [object](#)

**defaultSize** [int](#)

## Fields

### \_cap

```
protected int _cap
```

#### Field Value

[int](#)

### \_count

```
protected int _count
```

#### Field Value

[int](#)

### \_subscriptionRefList

```
protected WeakReference<SubscriptionBinding<TDelegate>>[] _subscriptionRefList
```

#### Field Value

[WeakReference](#)<[SubscriptionBinding](#)<TDelegate>>[]

## Properties

### listenerCount

```
public int listenerCount { get; }
```

#### Property Value

## Methods

### AddListener(TDelegate)

```
public Subscription AddListener(TDelegate listener)
```

#### Parameters

**listener** TDelegate

#### Returns

[Subscription](#)

### Contains(TDelegate)

```
public bool Contains(TDelegate d)
```

#### Parameters

**d** TDelegate

#### Returns

[bool](#)

### Expand(ref

### WeakReference<SubscriptionBinding<TDelegate>>[])

```
public int Expand(ref WeakReference<SubscriptionBinding<TDelegate>>[] bindings)
```

#### Parameters

bindings [WeakReference](#)<[SubscriptionBinding](#)<TDelegate>>[]

Returns

[int](#)

## RemoveAll()

`public void RemoveAll()`

## RemoveListener(TDelegate)

`public bool RemoveListener(TDelegate listener)`

Parameters

`listener` TDelegate

Returns

[bool](#)

# Class Relay<T>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public class Relay<T> : RelayBase<Action<T>>, IRelay<Action<T>>
```

## Type Parameters

T

### Inheritance

[object](#) ← [RelayBase<Action<T>>](#) ← [Relay<T>](#)

### Implements

[IRelay<Action<T>>](#)

### Inherited Members

[RelayBase<Action<T>>.\\_subscriptionRefList](#) , [RelayBase<Action<T>>.\\_cap](#) ,  
[RelayBase<Action<T>>.\\_count](#) , [RelayBase<Action<T>>.listenerCount](#) ,  
[RelayBase<Action<T>>.AddListener\(Action<T>\)](#) ,  
[RelayBase<Action<T>>.RemoveListener\(Action<T>\)](#) ,  
[RelayBase<Action<T>>.RemoveAll\(\)](#) , [RelayBase<Action<T>>.Contains\(Action<T>\)](#) ,  
[RelayBase<Action<T>>.Expand\(ref WeakReference<SubscriptionBinding<Action<T>>>\[\]\)](#) ,  
,

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Constructors

### Relay(object, int)

```
public Relay(object owner, int defaultSize = 1)
```

## Parameters

owner [object](#)

defaultSize [int](#)

## Methods

### Dispatch(T)

```
public void Dispatch(T value1)
```

#### Parameters

value1 T

# Class Relay<T1, T2>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public class Relay<T1, T2> : RelayBase<Action<T1, T2>>, IRelay<Action<T1, T2>>
```

## Type Parameters

T1

T2

## Inheritance

[object](#) ↗ ← [RelayBase<Action<T1, T2>>](#) ← [Relay<T1, T2>](#)

## Implements

[IRelay<Action<T1, T2>>](#)

## Inherited Members

[RelayBase<Action<T1, T2>>.\\_subscriptionRefList](#) , [RelayBase<Action<T1, T2>>.\\_cap](#) ,  
[RelayBase<Action<T1, T2>>.\\_count](#) , [RelayBase<Action<T1, T2>>.listenerCount](#) ,  
[RelayBase<Action<T1, T2>>.AddListener\(Action<T1, T2>\)](#) ,  
[RelayBase<Action<T1, T2>>.RemoveListener\(Action<T1, T2>\)](#) ,  
[RelayBase<Action<T1, T2>>.RemoveAll\(\)](#) ,  
[RelayBase<Action<T1, T2>>.Contains\(Action<T1, T2>\)](#) ,  
[RelayBase<Action<T1, T2>>.Expand\(ref WeakReference<SubscriptionBinding<Action<T1, T2>>>\[\]\)](#) ,  
[object.Equals\(object\)](#) ↗ , [object.Equals\(object, object\)](#) ↗ , [object.GetHashCode\(\)](#) ↗ ,  
[object.GetType\(\)](#) ↗ , [object.MemberwiseClone\(\)](#) ↗ , [object.ReferenceEquals\(object, object\)](#) ↗ ,  
[object.ToString\(\)](#) ↗

## Constructors

[Relay\(object, int\)](#)

```
public Relay(object owner, int defaultSize = 1)
```

## Parameters

owner [object](#)

defaultSize [int](#)

## Methods

### Dispatch(T1, T2)

```
public void Dispatch(T1 value1, T2 value2)
```

## Parameters

value1 T1

value2 T2

# Class RtCollection<TEventArgs>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public abstract class RtCollection<TEventArgs> : IDisposable
```

## Type Parameters

TEventArgs

### Inheritance

[object](#) ← RtCollection<TEventArgs>

### Implements

[IDisposable](#)

### Derived

[RtReadOnlyDictionary< TKey, TValue >](#), [RtReadOnlyList< T >](#)

### Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Constructors

RtCollection()

```
protected RtCollection()
```

## Properties

CollectionEvents

```
protected CollectionEvents<TEventArgs> CollectionEvents { get; }
```

Property Value

[CollectionEvents<TEventArgs>](#)

## Events

```
public ICollectionEvents<TEventArgs> Events { get; }
```

Property Value

[ICollectionEvents<TEventArgs>](#)

## Methods

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public virtual void Dispose()
```

# Class RtCount<T>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public class RtCount<T> : RtMutatedSingleBase<T, int>, IDisposable,  
 IReadOnlyList<int>, IReadOnlyCollection<int>, IEnumerable<int>, IEnumerable
```

## Type Parameters

T

## Inheritance

```
object ↪ RtCollection<(int index, int item)> ↪ RtReadOnlyList<int> ↪  
RtMutatedLocalListBase<T, int> ↪ RtMutatedSingleBase<T, int> ↪ RtCount<T>
```

## Implements

```
IDisposable, IReadOnlyList<int>, IReadOnlyCollection<int>, IEnumerable<int>,  
IEnumerable
```

## Inherited Members

```
RtMutatedSingleBase<T, int>.Value , RtMutatedSingleBase<T, int>.Dispose() ,  
RtMutatedSingleBase<T, int>.GetEnumerator() , RtMutatedSingleBase<T, int>.Count ,  
RtMutatedSingleBase<T, int>.this[int] , RtMutatedLocalListBase<T, int>.Dispose() ,  
RtMutatedLocalListBase<T, int>._OnSourceUpdate((int index, T item), (int index, T item)) ,  
RtMutatedLocalListBase<T, int>._OnSourceRemove((int index, T item)) ,  
RtMutatedLocalListBase<T, int>._OnSourceAdd((int index, T item)) ,  
RtReadOnlyList<int>.GetEnumerator() , RtReadOnlyList<int>.Count ,  
RtReadOnlyList<int>.this[int] , RtCollection<(int index, int item)>.CollectionEvents ,  
RtCollection<(int index, int item)>.Events , RtCollection<(int index, int item)>.Dispose() ,  
object.Equals(object) , object.Equals(object, object) , object.GetHashCode() ,  
object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) ,  
object.ToString()
```

## Extension Methods

```
Linq.count<T>(RtReadOnlyList<T>) ,  
Linq.groupBy< TKey, TValue >(RtReadOnlyList< TValue >, Func< TValue, TKey >) ,  
Linq.selectNew< T, TSelect >(RtReadOnlyList< T >, Func< T, TSelect >) ,  
Linq.select< T, TSelect >(RtReadOnlyList< T >, Func< T, TSelect >)
```

# Constructors

## RtCount(RtReadOnlyList<T>)

```
public RtCount(RtReadOnlyList<T> source)
```

### Parameters

source [RtReadOnlyList<T>](#)

## Methods

### \_OnSourceAdd((int index, T item))

```
protected override void _OnSourceAdd((int index, T item) item)
```

### Parameters

item ([int](#) [index](#), T [item](#))

### \_OnSourceRemove((int index, T item))

```
protected override void _OnSourceRemove((int index, T item) item)
```

### Parameters

item ([int](#) [index](#), T [item](#))

### \_OnSourceUpdate((int index, T item), (int index, T item))

```
protected override void _OnSourceUpdate((int index, T item) oldItem, (int index, T item) newItem)
```

### Parameters

`oldItem (int index, T item)`

`newItem (int index, T item)`

# Class RtDictionary<TKey, TValue>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public class RtDictionary<TKey, TValue> : RtReadOnlyDictionary<TKey, TValue>,  
IDisposable, IReadOnlyDictionary<TKey, TValue>,  
IReadOnlyCollection<KeyValuePair<TKey, TValue>>, IEnumerable<KeyValuePair<TKey,  
TValue>>, IEnumerable
```

## Type Parameters

TKey

TValue

## Inheritance

```
object ← RtCollection<KeyValuePair<TKey, TValue>> ←  
RtReadOnlyDictionary<TKey, TValue> ← RtDictionary<TKey, TValue>
```

## Implements

```
IDisposable, IReadOnlyDictionary<TKey, TValue>,  
IReadOnlyCollection<KeyValuePair<TKey, TValue>>,  
IEnumerable<KeyValuePair<TKey, TValue>>, IEnumerable
```

## Inherited Members

```
RtReadOnlyDictionary<TKey, TValue>.Count ,  
RtReadOnlyDictionary<TKey, TValue>.ContainsKey\(TKey\) ,  
RtReadOnlyDictionary<TKey, TValue>.TryGetValue\(TKey, out TValue\) ,  
RtReadOnlyDictionary<TKey, TValue>.this\[TKey\] ,  
RtReadOnlyDictionary<TKey, TValue>.Keys , RtReadOnlyDictionary<TKey, TValue>.Values ,  
RtReadOnlyDictionary<TKey, TValue>.RtPairs ,  
RtReadOnlyDictionary<TKey, TValue>.RtKeys ,  
RtReadOnlyDictionary<TKey, TValue>.RtValues ,  
RtReadOnlyDictionary<TKey, TValue>.GetEnumerator\(\) ,  
RtCollection<KeyValuePair<TKey, TValue>>.CollectionEvents ,  
RtCollection<KeyValuePair<TKey, TValue>>.Events ,  
RtCollection<KeyValuePair<TKey, TValue>>.Dispose\(\) , object.Equals\(object\) ,  
object.Equals\(object, object\) , object.GetHashCode\(\) , object.GetType\(\) ,  
object.MemberwiseClone\(\) , object.ReferenceEquals\(object, object\) , object.ToString\(\)
```

## Extension Methods

[Linq.selectKey<TOrigKey, TSelectedKey, TValue>\(RtReadOnlyDictionary<TOrigKey, TValue>, Func<TOrigKey, TSelectedKey>\)](#) ,  
[Linq.selectValue<TKey, TValue, TSelectValue>\(RtReadOnlyDictionary<TKey, TValue>, Func<TValue, TSelectValue>\)](#) ,  
[Linq.where<TKey, TValue>\(RtReadOnlyDictionary<TKey, TValue>, Func<KeyValuePair<TKey, TValue>, bool>\)](#).

# Properties

## Count

Gets the number of elements in the collection.

```
public override int Count { get; }
```

## Property Value

[int](#)

The number of elements in the collection.

## IsReadOnly

```
public bool IsReadOnly { get; }
```

## Property Value

[bool](#)

## this[TKey]

Gets the element that has the specified key in the read-only dictionary.

```
public override TValue this[TKey key] { get; }
```

## Parameters

### key TKey

The key to locate.

## Property Value

### TValue

The element that has the specified key in the read-only dictionary.

## Exceptions

### [ArgumentNullException](#)

key is [null](#).

### [KeyNotFoundException](#)

The property is retrieved and key is not found.

## Keys

Gets an enumerable collection that contains the keys in the read-only dictionary.

```
public override IEnumerable<TKey> Keys { get; }
```

## Property Value

### [IEnumerable](#)<TKey>

An enumerable collection that contains the keys in the read-only dictionary.

## Values

Gets an enumerable collection that contains the values in the read-only dictionary.

```
public override IEnumerable<TValue> Values { get; }
```

# Property Value

## [IEnumerable](#)<TValue>

An enumerable collection that contains the values in the read-only dictionary.

## Methods

### Add(KeyValuePair<TKey, TValue>)

Adds a key-value pair to the dictionary.

```
public void Add(KeyValuePair<TKey, TValue> kvp)
```

#### Parameters

##### kvp [KeyValuePair](#)<TKey, TValue>

The key-value pair to add.

#### Exceptions

##### [ArgumentException](#)

Thrown when the key already exists.

### Add(TKey, TValue)

```
public void Add(TKey key, TValue value)
```

#### Parameters

##### key TKey

##### value TValue

## Clear()

```
public void Clear()
```

## Contains(KeyValuePair<TKey, TValue>)

```
public bool Contains(KeyValuePair<TKey, TValue> kvp)
```

### Parameters

kvp [KeyValuePair](#)<TKey, TValue>

### Returns

[bool](#)

## ContainsKey(TKey)

Determines whether the read-only dictionary contains an element that has the specified key.

```
public override bool ContainsKey(TKey key)
```

### Parameters

key TKey

The key to locate.

### Returns

[bool](#)

[true](#) if the read-only dictionary contains an element that has the specified key; otherwise, [false](#).

### Exceptions

[ArgumentNullException](#)

`key` is [null](#).

## CopyTo(KeyValuePair<TKey, TValue>[], int)

```
public void CopyTo(KeyValuePair<TKey, TValue>[] array, int arrayIndex)
```

### Parameters

array [KeyValuePair](#)<TKey, TValue>[]

arrayIndex [int](#)

## Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public override void Dispose()
```

## EnsureCapacity(int)

```
public void EnsureCapacity(int capacity)
```

### Parameters

capacity [int](#)

## GetEnumerator()

Returns an enumerator that iterates through the collection.

```
public override IEnumerator<KeyValuePair<TKey, TValue>> GetEnumerator()
```

### Returns

## [IEnumerator](#) <[KeyValuePair](#) <TKey, TValue>>

An enumerator that can be used to iterate through the collection.

## Remove(KeyValuePair<TKey, TValue>)

```
public bool Remove(KeyValuePair<TKey, TValue> kvp)
```

Parameters

kvp [KeyValuePair](#) <TKey, TValue>

Returns

[bool](#)

## Remove(TKey)

```
public bool Remove(TKey key)
```

Parameters

key TKey

Returns

[bool](#)

## TryGetValue(TKey, out TValue)

Gets the value that is associated with the specified key.

```
public override bool TryGetValue(TKey key, out TValue value)
```

Parameters

**key** TKey

The key to locate.

**value** TValue

When this method returns, the value associated with the specified key, if the key is found; otherwise, the default value for the type of the **value** parameter. This parameter is passed uninitialized.

Returns

[bool](#)

[true](#) if the object that implements the [IReadOnlyDictionary<TKey, TValue>](#) interface contains an element that has the specified key; otherwise, [false](#).

Exceptions

[ArgumentNullException](#)

**key** is [null](#).

## Upsert(TKey, TValue)

```
public void Upsert(TKey key, TValue value)
```

Parameters

**key** TKey

**value** TValue

# Class RtFilteredDictionary<TKey, TValue>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

Represents a dictionary that filters its elements based on a provided function.

```
public class RtFilteredDictionary<TKey, TValue> : RtMutatedDictionaryBase<TKey, TValue, TKey, TValue>, IDisposable, IReadOnlyDictionary<TKey, TValue>, IReadOnlyCollection<KeyValuePair<TKey, TValue>>, IEnumerable<KeyValuePair<TKey, TValue>>, IEnumerable
```

## Type Parameters

### TKey

The type of keys in the dictionary.

### TValue

The type of values in the dictionary.

## Inheritance

```
object ↗ ← RtCollection<KeyValuePair ↗ <TKey, TValue>> ←  
RtReadOnlyDictionary<TKey, TValue> ←  
RtMutatedDictionaryBase<TKey, TValue, TKey, TValue> ←  
RtFilteredDictionary<TKey, TValue>
```

## Implements

```
IDisposable ↗ , IReadOnlyDictionary ↗ <TKey, TValue> ,  
IReadOnlyCollection ↗ <KeyValuePair ↗ <TKey, TValue>> ,  
IEnumerable ↗ <KeyValuePair ↗ <TKey, TValue>> , IEnumerable ↗
```

## Inherited Members

```
RtMutatedDictionaryBase<TKey, TValue, TKey, TValue>._mutated ,  
RtMutatedDictionaryBase<TKey, TValue, TKey, TValue>.Dispose() ,  
RtMutatedDictionaryBase<TKey, TValue, TKey, TValue>._OnSourceUpdate(in  
Dictionary<TKey, TValue> , KeyValuePair<TKey, TValue> , KeyValuePair<TKey, TValue>) ,  
RtMutatedDictionaryBase<TKey, TValue, TKey, TValue>._OnSourceRemove(in  
Dictionary<TKey, TValue> , KeyValuePair<TKey, TValue>) ,
```

[RtMutatedDictionaryBase<TKey, TValue, TKey, TValue>.OnSourceAdd\(in Dictionary<TKey, TValue>, KeyValuePair<TKey, TValue>\)](#) ,  
[RtMutatedDictionaryBase<TKey, TValue, TKey, TValue>.GetEnumerator\(\)](#) ,  
[RtMutatedDictionaryBase<TKey, TValue, TKey, TValue>.Count](#) ,  
[RtMutatedDictionaryBase<TKey, TValue, TKey, TValue>.ContainsKey\(TKey\)](#) ,  
[RtMutatedDictionaryBase<TKey, TValue, TKey, TValue>.TryGetValue\(TKey, out TValue\)](#) ,  
[RtMutatedDictionaryBase<TKey, TValue, TKey, TValue>.this\[TKey\]](#) ,  
[RtMutatedDictionaryBase<TKey, TValue, TKey, TValue>.Keys](#) ,  
[RtMutatedDictionaryBase<TKey, TValue, TKey, TValue>.Values](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.Count](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.ContainsKey\(TKey\)](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.TryGetValue\(TKey, out TValue\)](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.this\[TKey\]](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.Keys](#) , [RtReadOnlyDictionary<TKey, TValue>.Values](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.RtPairs](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.RtKeys](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.RtValues](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.GetEnumerator\(\)](#) ,  
[RtCollection<KeyValuePair<TKey, TValue>>.CollectionEvents](#) ,  
[RtCollection<KeyValuePair<TKey, TValue>>.Events](#) ,  
[RtCollection<KeyValuePair<TKey, TValue>>.Dispose\(\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Extension Methods

[Linq.selectKey<TOrigKey, TSelectedKey, TValue>\(RtReadOnlyDictionary<TOrigKey, TValue>, Func<TOrigKey, TSelectedKey>\)](#) ,  
[Linq.selectValue<TKey, TValue, TSelectValue>\(RtReadOnlyDictionary<TKey, TValue>, Func<TValue, TSelectValue>\)](#) ,  
[Linq.where<TKey, TValue>\(RtReadOnlyDictionary<TKey, TValue>, Func<KeyValuePair<TKey, TValue>, bool>\)](#)

## Constructors

**RtFilteredDictionary(RtReadOnlyDictionary<TKey, TValue>, Func<KeyValuePair<TKey, TValue>, bool>)**

```
public RtFilteredDictionary(RtReadOnlyDictionary<TKey, TValue> source,  
Func<KeyValuePair<TKey, TValue>, bool> filterFn)
```

## Parameters

source [RtReadOnlyDictionary<TKey, TValue>](#)

filterFn [Func<KeyValuePair<TKey, TValue>, bool>](#)

## Methods

\_OnSourceAdd(in Dictionary<TKey, TValue>, KeyValuePair<TKey, TValue>)

```
protected override void _OnSourceAdd(in Dictionary<TKey, TValue> mutated,  
KeyValuePair<TKey, TValue> kvp)
```

## Parameters

mutated [Dictionary<TKey, TValue>](#)

kvp [KeyValuePair<TKey, TValue>](#)

\_OnSourceRemove(in Dictionary<TKey, TValue>, KeyValuePair<TKey, TValue>)

```
protected override void _OnSourceRemove(in Dictionary<TKey, TValue> mutated,  
KeyValuePair<TKey, TValue> kvp)
```

## Parameters

mutated [Dictionary<TKey, TValue>](#)

kvp [KeyValuePair<TKey, TValue>](#)

\_OnSourceUpdate(in Dictionary<TKey, TValue>, KeyValuePair<TKey, TValue>, KeyValuePair<TKey, TValue>)

```
protected override void _OnSourceUpdate(in Dictionary< TKey, TValue> mutated,  
KeyValuePair< TKey, TValue> oldPair, KeyValuePair< TKey, TValue> newPair)
```

## Parameters

mutated [Dictionary](#)< TKey, TValue>

oldPair [KeyValuePair](#)< TKey, TValue>

newPair [KeyValuePair](#)< TKey, TValue>

# Class RtGroup<TKey, TValue>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public class RtGroup<TKey, TValue> : RtReadOnlyDictionary<TKey,  
RtReadOnlyList<TValue>>, IDisposable, IReadOnlyDictionary<TKey,  
RtReadOnlyList<TValue>>, IReadOnlyCollection<KeyValuePair<TKey,  
RtReadOnlyList<TValue>>>, IEnumerable<KeyValuePair<TKey, RtReadOnlyList<TValue>>>,  
IEnumerable
```

## Type Parameters

TKey

TValue

## Inheritance

```
object ↪ RtCollection<KeyValuePair<TKey, RtReadOnlyList<TValue>>> ↪  
RtReadOnlyDictionary<TKey, RtReadOnlyList<TValue>> ↪ RtGroup<TKey, TValue>
```

## Implements

```
IDisposable, IReadOnlyDictionary<TKey, RtReadOnlyList<TValue>>,  
IReadOnlyCollection<KeyValuePair<TKey, RtReadOnlyList<TValue>>>,  
IEnumerable<KeyValuePair<TKey, RtReadOnlyList<TValue>>>, IEnumerable
```

## Inherited Members

```
RtReadOnlyDictionary<TKey, RtReadOnlyList<TValue>>.Count ,  
RtReadOnlyDictionary<TKey, RtReadOnlyList<TValue>>.ContainsKey(TKey) ,  
RtReadOnlyDictionary<TKey, RtReadOnlyList<TValue>>.TryGetValue(TKey, out  
RtReadOnlyList<TValue>) ,  
RtReadOnlyDictionary<TKey, RtReadOnlyList<TValue>>.this[TKey] ,  
RtReadOnlyDictionary<TKey, RtReadOnlyList<TValue>>.Keys ,  
RtReadOnlyDictionary<TKey, RtReadOnlyList<TValue>>.Values ,  
RtReadOnlyDictionary<TKey, RtReadOnlyList<TValue>>.RtPairs ,  
RtReadOnlyDictionary<TKey, RtReadOnlyList<TValue>>.RtKeys ,  
RtReadOnlyDictionary<TKey, RtReadOnlyList<TValue>>.RtValues ,  
RtReadOnlyDictionary<TKey, RtReadOnlyList<TValue>>.GetEnumerator() ,  
RtCollection<KeyValuePair<TKey, RtReadOnlyList<TValue>>>.CollectionEvents ,  
RtCollection<KeyValuePair<TKey, RtReadOnlyList<TValue>>>.Events ,
```

[RtCollection<KeyValuePair< TKey, RtReadOnlyList< TValue>>>.Dispose\(\)](#) ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Extension Methods

[Linq.selectKey< TOrigKey, TSelectedKey, TValue>\(RtReadOnlyDictionary< TOrigKey, TValue>, Func< TOrigKey, TSelectedKey>\)](#) ,  
[Linq.selectValue< TKey, TValue, TSelectValue>\(RtReadOnlyDictionary< TKey, TValue>, Func< TValue, TSelectValue>\)](#) ,  
[Linq.where< TKey, TValue>\(RtReadOnlyDictionary< TKey, TValue>, Func< KeyValuePair< TKey, TValue>, bool>\)](#).

## Constructors

RtGroup(RtReadOnlyList< TValue>, Func< TValue, TKey>)

```
public RtGroup(RtReadOnlyList< TValue> source, Func< TValue, TKey> keyFn)
```

## Parameters

source [RtReadOnlyList< TValue>](#)

keyFn [Func< TValue, TKey>](#)

## Properties

### Count

Gets the number of elements in the collection.

```
public override int Count { get; }
```

## Property Value

[int](#)

The number of elements in the collection.

## this[TKey]

Gets the element that has the specified key in the read-only dictionary.

```
public override RtReadOnlyList< TValue> this[TKey key] { get; }
```

### Parameters

**key** TKey

The key to locate.

### Property Value

[RtReadOnlyList< TValue>](#)

The element that has the specified key in the read-only dictionary.

### Exceptions

[ArgumentNullException](#)

**key** is [null](#).

[KeyNotFoundException](#)

The property is retrieved and **key** is not found.

## Keys

Gets an enumerable collection that contains the keys in the read-only dictionary.

```
public override IEnumerable< TKey> Keys { get; }
```

### Property Value

[IEnumerable](#)< TKey>

An enumerable collection that contains the keys in the read-only dictionary.

## Values

Gets an enumerable collection that contains the values in the read-only dictionary.

```
public override IEnumerable<RtReadOnlyList<TValue>> Values { get; }
```

### Property Value

[IEnumerable](#)<RtReadOnlyList<TValue>>

An enumerable collection that contains the values in the read-only dictionary.

## Methods

### ContainsKey(TKey)

Determines whether the read-only dictionary contains an element that has the specified key.

```
public override bool ContainsKey(TKey key)
```

### Parameters

**key** TKey

The key to locate.

### Returns

[bool](#)

[true](#) if the read-only dictionary contains an element that has the specified key; otherwise, [false](#).

### Exceptions

[ArgumentNullException](#)

`key` is [null](#).

## Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public override void Dispose()
```

## GetEnumerator()

Returns an enumerator that iterates through the collection.

```
public override IEnumerator<KeyValuePair<TKey, RtReadOnlyList< TValue >>>
GetEnumerator()
```

Returns

[IEnumerator](#)<[KeyValuePair](#)<TKey, [RtReadOnlyList](#)< TValue >>>

An enumerator that can be used to iterate through the collection.

## TryGetValue(TKey, out RtReadOnlyList< TValue >)

Gets the value that is associated with the specified key.

```
public override bool TryGetValue(TKey key, out RtReadOnlyList< TValue > value)
```

Parameters

`key` TKey

The key to locate.

`value` [RtReadOnlyList](#)< TValue >

When this method returns, the value associated with the specified key, if the key is found; otherwise, the default value for the type of the `value` parameter. This parameter is passed uninitialized.

Returns

`bool`

`true` if the object that implements the [IReadOnlyDictionary<TKey, TValue>](#) interface contains an element that has the specified key; otherwise, `false`.

Exceptions

[ArgumentNullException](#)

`key` is `null`.

# Class RtList<T>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public class RtList<T> : RtReadOnlyList<T>, IDisposable, IReadOnlyList<T>,  
IReadOnlyCollection<T>, IEnumerable<T>, IEnumerable
```

## Type Parameters

T

### Inheritance

[object](#) ← [RtCollection<\(int index, T item\)>](#) ← [RtReadOnlyList<T>](#) ← [RtList<T>](#)

### Implements

[IDisposable](#) ,  [IReadOnlyList](#) <T> ,  [IReadOnlyCollection](#) <T> ,  [IEnumerable](#) <T> ,  
 [IEnumerable](#)

### Inherited Members

[RtCollection<\(int index, T item\)>.CollectionEvents](#) ,  
[RtCollection<\(int index, T item\)>.Events](#) , [RtCollection<\(int index, T item\)>.Dispose\(\)](#) ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

### Extension Methods

[Linq.count<T>\(RtReadOnlyList<T>\)](#) ,  
[Linq.groupBy< TKey, TValue >\(RtReadOnlyList< TValue >, Func< TValue, TKey >\)](#) ,  
[Linq.selectNew< T, TSelect >\(RtReadOnlyList< T >, Func< T, TSelect >\)](#) ,  
[Linq.select< T, TSelect >\(RtReadOnlyList< T >, Func< T, TSelect >\)](#)

## Constructors

### RtList()

```
public RtList()
```

## RtList(IEnumerable<T>)

```
public RtList(IEnumerable<T> defaultItems)
```

### Parameters

defaultItems [IEnumerable](#)<T>

## RtList(int)

```
public RtList(int capacity)
```

### Parameters

capacity [int](#)

## Properties

### Count

Gets the number of elements in the collection.

```
public override int Count { get; }
```

### Property Value

[int](#)

The number of elements in the collection.

### IsReadOnly

```
public bool IsReadOnly { get; }
```

### Property Value

[bool](#)

## this[int]

Gets the element at the specified index in the read-only list.

```
public override T this[int index] { get; }
```

### Parameters

[index int](#)

The zero-based index of the element to get.

### Property Value

T

The element at the specified index in the read-only list.

## Methods

### Add(T)

```
public void Add(T item)
```

### Parameters

[item T](#)

### Clear()

```
public void Clear()
```

### Contains(T)

```
public bool Contains(T item)
```

Parameters

item T

Returns

[bool](#)

## CopyTo(T[], int)

```
public void CopyTo(T[] array, int arrayIndex)
```

Parameters

array T[]

arrayIndex [int](#)

## Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public override void Dispose()
```

## GetEnumerator()

Returns an enumerator that iterates through the collection.

```
public override IEnumerator<T> GetEnumerator()
```

Returns

## [IEnumerator](#) <T>

An enumerator that can be used to iterate through the collection.

### IndexOf(T)

```
public int IndexOf(T item)
```

Parameters

item T

Returns

[int](#)

### Insert(int, T)

```
public void Insert(int index, T item)
```

Parameters

index [int](#)

item T

### Remove(T)

```
public bool Remove(T item)
```

Parameters

item T

Returns

[bool](#)

## RemoveAt(int)

```
public void RemoveAt(int index)
```

### Parameters

[index](#) [int](#)

## Update(int, T)

```
public void Update(int index, T item)
```

### Parameters

[index](#) [int](#)

[item](#) [T](#)

# Class

## RtMutatedDictionaryBase<TSourceKey, TValue>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

Represents a base class for dictionaries that support mutation and event dispatching.

```
public abstract class RtMutatedDictionaryBase<TSourceKey, TValue, TKey, TValue> : RtReadOnlyDictionary<TKey, TValue>, IDisposable, IReadOnlyDictionary<TKey, TValue>, IReadOnlyCollection<KeyValuePair<TKey, TValue>>, IEnumerable<KeyValuePair<TKey, TValue>>, IEnumerable
```

### Type Parameters

#### TSourceKey

The type of keys in the source dictionary.

#### TSourceValue

The type of values in the source dictionary.

#### TKey

The type of keys in the mutated dictionary.

#### TValue

The type of values in the mutated dictionary.

### Inheritance

```
object ↵ ← RtCollection<KeyValuePair<TKey, TValue>> ←  
RtReadOnlyDictionary<TKey, TValue> ←  
RtMutatedDictionaryBase<TSourceKey, TValue>
```

### Implements

```
IDisposable, IReadOnlyDictionary<TKey, TValue>,  
IReadOnlyCollection<KeyValuePair<TKey, TValue>>,
```

[IEnumerable](#)<[KeyValuePair](#)<TKey, TValue>>, [IEnumerable](#)

## Derived

[RtFilteredDictionary](#)<TKey, TValue>,  
[RtSelectKeyDictionary](#)<TOrigKey, TSelectKey, TValue>,  
[RtSelectValueDictionary](#)<TKey, TOrigValue, TValue>

## Inherited Members

[RtReadOnlyDictionary](#)<TKey, TValue>.Count ,  
[RtReadOnlyDictionary](#)<TKey, TValue>.ContainsKey(TKey) ,  
[RtReadOnlyDictionary](#)<TKey, TValue>.TryGetValue(TKey, [out TValue](#)) ,  
[RtReadOnlyDictionary](#)<TKey, TValue>.this[TKey] ,  
[RtReadOnlyDictionary](#)<TKey, TValue>.Keys , [RtReadOnlyDictionary](#)<TKey, TValue>.Values ,  
[RtReadOnlyDictionary](#)<TKey, TValue>.RtPairs ,  
[RtReadOnlyDictionary](#)<TKey, TValue>.RtKeys ,  
[RtReadOnlyDictionary](#)<TKey, TValue>.RtValues ,  
[RtReadOnlyDictionary](#)<TKey, TValue>.GetEnumerator() ,  
[RtCollection](#)<KeyValuePair<TKey, TValue>>.CollectionEvents ,  
[RtCollection](#)<KeyValuePair<TKey, TValue>>.Events ,  
[RtCollection](#)<KeyValuePair<TKey, TValue>>.Dispose() , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Extension Methods

[Linq.selectKey](#)<TOrigKey, TSelectedKey, TValue>(RtReadOnlyDictionary<TOrigKey, TValue>, Func<TOrigKey, TSelectedKey>) ,  
[Linq.selectValue](#)<TKey, TValue, TSelectValue>(RtReadOnlyDictionary<TKey, TValue>, Func<TValue, TSelectValue>) ,  
[Linq.where](#)<TKey, TValue>(RtReadOnlyDictionary<TKey, TValue>, Func<KeyValuePair<TKey, TValue>, bool>)

## Constructors

RtMutatedDictionaryBase(ICollectionEvents<KeyValuePair<TSourceKey, TSourceValue>>)

```
protected RtMutatedDictionaryBase(ICollectionEvents<KeyValuePair<TSourceKey, TSourceValue>> sourceEvents)
```

## Parameters

sourceEvents [ICollectionEvents<KeyValuePair><TSourceKey, TValue>>](#)

## Fields

### \_mutated

`protected readonly Dictionary<TKey, TValue> _mutated`

## Field Value

[Dictionary<TKey, TValue>](#)

## Properties

### Count

Gets the number of elements in the collection.

`public override int Count { get; }`

## Property Value

[int](#)

The number of elements in the collection.

### this[TKey]

Gets the element that has the specified key in the read-only dictionary.

`public override TValue this[TKey key] { get; }`

## Parameters

**key** TKey

The key to locate.

Property Value

TValue

The element that has the specified key in the read-only dictionary.

Exceptions

[ArgumentNullException](#)

**key** is [null](#).

[KeyNotFoundException](#)

The property is retrieved and **key** is not found.

## Keys

Gets an enumerable collection that contains the keys in the read-only dictionary.

```
public override IEnumerable<TKey> Keys { get; }
```

Property Value

[IEnumerable](#)<TKey>

An enumerable collection that contains the keys in the read-only dictionary.

## Values

Gets an enumerable collection that contains the values in the read-only dictionary.

```
public override IEnumerable<TValue> Values { get; }
```

Property Value

## [IEnumerable](#)< TValue >

An enumerable collection that contains the values in the read-only dictionary.

## Methods

### ContainsKey(TKey)

Determines whether the read-only dictionary contains an element that has the specified key.

```
public override bool ContainsKey(TKey key)
```

#### Parameters

**key** TKey

The key to locate.

#### Returns

[bool](#)

[true](#) if the read-only dictionary contains an element that has the specified key; otherwise, [false](#).

#### Exceptions

[ArgumentNullException](#)

**key** is [null](#).

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public override void Dispose()
```

## GetEnumerator()

Returns an enumerator that iterates through the collection.

```
public override IEnumerator<KeyValuePair<TKey, TValue>> GetEnumerator()
```

Returns

[IEnumerator](#)<KeyValuePair<TKey, TValue>>

An enumerator that can be used to iterate through the collection.

## TryGetValue(TKey, out TValue)

Gets the value that is associated with the specified key.

```
public override bool TryGetValue(TKey key, out TValue value)
```

Parameters

**key** TKey

The key to locate.

**value** TValue

When this method returns, the value associated with the specified key, if the key is found; otherwise, the default value for the type of the **value** parameter. This parameter is passed uninitialized.

Returns

[bool](#)

[true](#) if the object that implements the [IReadOnlyDictionary<TKey, TValue>](#) interface contains an element that has the specified key; otherwise, [false](#).

Exceptions

[ArgumentNullException](#)

`key` is [null](#).

## \_OnSourceAdd(in Dictionary<TKey, TValue>, KeyValuePair<TSourceKey, TSourceValue>)

```
protected abstract void _OnSourceAdd(in Dictionary<TKey, TValue> mutated,  
KeyValuePair<TSourceKey, TSourceValue> kvp)
```

### Parameters

`mutated` [Dictionary](#)<TKey, TValue>

`kvp` [KeyValuePair](#)<TSourceKey, TSourceValue>

## \_OnSourceRemove(in Dictionary<TKey, TValue>, KeyValuePair<TSourceKey, TSourceValue>)

```
protected abstract void _OnSourceRemove(in Dictionary<TKey, TValue> mutated,  
KeyValuePair<TSourceKey, TSourceValue> kvp)
```

### Parameters

`mutated` [Dictionary](#)<TKey, TValue>

`kvp` [KeyValuePair](#)<TSourceKey, TSourceValue>

## \_OnSourceUpdate(in Dictionary<TKey, TValue>, KeyValuePair<TSourceKey, TSourceValue>, KeyValuePair<TSourceKey, TSourceValue>)

```
protected abstract void _OnSourceUpdate(in Dictionary<TKey, TValue> mutated,  
KeyValuePair<TSourceKey, TSourceValue> oldPair, KeyValuePair<TSourceKey,  
TSourceValue> newPair)
```

### Parameters

`mutated Dictionary<TKey, TValue>`

`oldPair KeyValuePair<TSourceKey, TValue>`

`newPair KeyValuePair<TSourceKey, TValue>`

# Class RtMutatedListBase<TOrig, TNew>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public abstract class RtMutatedListBase<TOrig, TNew> : RtReadOnlyList<TNew>,
IDisposable, IReadOnlyList<TNew>, IReadOnlyCollection<TNew>,
IEnumerable<TNew>, IEnumerable
```

## Type Parameters

TOrig

TNew

## Inheritance

```
object ↪ RtCollection<(int ↪ index ↪ , TNew item ↪ )> ↪ RtReadOnlyList<TNew> ↪
RtMutatedListBase<TOrig, TNew>
```

## Implements

```
IDisposable, IReadOnlyList<TNew>, IReadOnlyCollection<TNew>,
IEnumerable<TNew>, IEnumerable
```

## Derived

[RtSelectList<TOrig, TNew>](#)

## Inherited Members

```
RtCollection<(int index, TNew item)>.CollectionEvents ,
RtCollection<(int index, TNew item)>.Events ,
RtCollection<(int index, TNew item)>.Dispose() , object.Equals(object) ,
object.Equals(object, object) , object.GetHashCode() , object.GetType() ,
object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()
```

## Extension Methods

```
Linq.count<T>(RtReadOnlyList<T>) ,
Linq.groupBy< TKey, TValue >(RtReadOnlyList< TValue >, Func< TValue, TKey >) ,
Linq.selectNew< T, TSelect >(RtReadOnlyList< T >, Func< T, TSelect >) ,
Linq.select< T, TSelect >(RtReadOnlyList< T >, Func< T, TSelect >)
```

# Constructors

RtMutatedListBase(ICollectionEvents<(int index, TOrig item)>)

```
protected RtMutatedListBase(ICollectionEvents<(int index, TOrig item)> sourceEvents)
```

## Parameters

sourceEvents [ICollectionEvents<\(int index, TOrig item\)>](#)

# Fields

\_mutated

```
protected readonly List<TNew> _mutated
```

## Field Value

[List](#)<TNew>

# Properties

Count

Gets the number of elements in the collection.

```
public override int Count { get; }
```

## Property Value

[int](#)

The number of elements in the collection.

## this[int]

Gets the element at the specified index in the read-only list.

```
public override TNew this[int index] { get; }
```

### Parameters

index [int](#)

The zero-based index of the element to get.

### Property Value

TNew

The element at the specified index in the read-only list.

## Methods

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public override void Dispose()
```

### GetEnumerator()

Returns an enumerator that iterates through the collection.

```
public override IEnumerator<TNew> GetEnumerator()
```

### Returns

[IEnumerator](#) <TNew>

An enumerator that can be used to iterate through the collection.

## \_OnSourceAdd(List<TNew>, (int index, TOrig item))

```
protected abstract void _OnSourceAdd(List<TNew> mutated, (int index, TOrig item) item)
```

### Parameters

mutated [List](#)<TNew>

item ([int](#) [index](#), TOrig [item](#))

## \_OnSourceRemove(List<TNew>, (int index, TOrig item))

```
protected abstract void _OnSourceRemove(List<TNew> mutated, (int index, TOrig item) item)
```

### Parameters

mutated [List](#)<TNew>

item ([int](#) [index](#), TOrig [item](#))

## \_OnSourceUpdate(List<TNew>, (int index, TOrig item), (int index, TOrig item))

```
protected abstract void _OnSourceUpdate(List<TNew> mutated, (int index, TOrig item)  
oldItem, (int index, TOrig item) newItem)
```

### Parameters

mutated [List](#)<TNew>

oldItem ([int](#) [index](#), TOrig [item](#))

newItem ([int](#) [index](#), TOrig [item](#))

# Class RtMutatedLocalListBase<TOrig, TNew>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public abstract class RtMutatedLocalListBase<TOrig, TNew> : RtReadOnlyList<TNew>,  
IDisposable, IReadOnlyList<TNew>, IReadOnlyCollection<TNew>,  
IEnumerable<TNew>, IEnumerable
```

## Type Parameters

TOrig

TNew

## Inheritance

[object](#) ← [RtCollection<\(int index, TNew item\)>](#) ← [RtReadOnlyList<TNew>](#) ←  
RtMutatedLocalListBase<TOrig, TNew>

## Implements

[IDisposable](#), [IReadOnlyList](#)<TNew>, [IReadOnlyCollection](#)<TNew>,  
[IEnumerable](#)<TNew>, [IEnumerable](#)

## Derived

[RtMutatedSingleBase<TOrig, TNew>](#), [RtSelectLocalList<T, TSelect>](#)

## Inherited Members

[RtReadOnlyList<TNew>.GetEnumerator\(\)](#), [RtReadOnlyList<TNew>.Count](#) ,  
[RtReadOnlyList<TNew>.this\[int\]](#) , [RtCollection<\(int index, TNew item\)>.CollectionEvents](#) ,  
[RtCollection<\(int index, TNew item\)>.Events](#) ,  
[RtCollection<\(int index, TNew item\)>.Dispose\(\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Extension Methods

[Linq.count<T>\(RtReadOnlyList<T>\)](#) ,  
[Linq.groupBy< TKey, TValue >\(RtReadOnlyList< TValue >, Func< TValue, TKey >\)](#) ,  
[Linq.selectNew<T, TSelect>\(RtReadOnlyList<T>, Func<T, TSelect>\)](#) ,  
[Linq.select<T, TSelect>\(RtReadOnlyList<T>, Func<T, TSelect>\)](#)

# Constructors

## RtMutatedLocalListBase(ICollectionEvents<(int index, TOrig item)>)

```
protected RtMutatedLocalListBase(ICollectionEvents<(int index, TOrig item)> sourceEvents)
```

### Parameters

sourceEvents [ICollectionEvents<\(int index, TOrig item\)>](#)

# Methods

## Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public override void Dispose()
```

## \_OnSourceAdd((int index, TOrig item))

```
protected abstract void _OnSourceAdd((int index, TOrig item) item)
```

### Parameters

item ([int index](#), TOrig [item](#))

## \_OnSourceRemove((int index, TOrig item))

```
protected abstract void _OnSourceRemove((int index, TOrig item) item)
```

### Parameters

`item (int index, TOrig item)`

`_OnSourceUpdate((int index, TOrig item), (int index, TOrig item))`

```
protected abstract void _OnSourceUpdate((int index, TOrig item) oldItem, (int index, TOrig item) newItem)
```

Parameters

`oldItem (int index, TOrig item)`

`newItem (int index, TOrig item)`

# Class RtMutatedSingleBase<TOrig, TNew>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public abstract class RtMutatedSingleBase<TOrig, TNew> :  
    RtMutatedLocalListBase<TOrig, TNew>, IDisposable, IReadOnlyList<TNew>,  
    IReadOnlyCollection<TNew>, IEnumerable<TNew>, IEnumerable
```

## Type Parameters

TOrig

TNew

## Inheritance

```
object ↪ RtCollection<(int ↪ index ↪ , TNew item ↪ )> ↪ RtReadOnlyList<TNew> ↪  
RtMutatedLocalListBase<TOrig, TNew> ↪ RtMutatedSingleBase<TOrig, TNew>
```

## Implements

```
IDisposable, IReadOnlyList<TNew>, IReadOnlyCollection<TNew>,  
IEnumerable<TNew>, IEnumerable
```

## Derived

[RtCount<T>](#)

## Inherited Members

```
RtMutatedLocalListBase<TOrig, TNew>.Dispose(),  
RtMutatedLocalListBase<TOrig, TNew>._OnSourceUpdate((int index, TOrig item), (int  
index, TOrig item)),  
RtMutatedLocalListBase<TOrig, TNew>._OnSourceRemove((int index, TOrig item)),  
RtMutatedLocalListBase<TOrig, TNew>._OnSourceAdd((int index, TOrig item)),  
RtCollection<(int index, TNew item)>.CollectionEvents,  
RtCollection<(int index, TNew item)>.Events,  
RtCollection<(int index, TNew item)>.Dispose(), object.Equals(object) ↪ ,  
object.Equals(object, object) ↪ , object.GetHashCode() ↪ , object.GetType() ↪ ,  
object.MemberwiseClone() ↪ , object.ReferenceEquals(object, object) ↪ , object.ToString() ↪
```

## Extension Methods

```
Linq.count<T>\(RtReadOnlyList<T>\) ,  
Linq.groupBy< TKey, TValue >\(RtReadOnlyList< TValue >, Func< TValue, TKey >\) ,  
Linq.selectNew< T, TSelect >\(RtReadOnlyList< T >, Func< T, TSelect >\) ,  
Linq.select< T, TSelect >\(RtReadOnlyList< T >, Func< T, TSelect >\).
```

## Constructors

RtMutatedSingleBase(ICollectionEvents<(int index, TOrig item)>)

```
public RtMutatedSingleBase(ICollectionEvents<(int index, TOrig item)> sourceEvents)
```

## Parameters

sourceEvents [ICollectionEvents<\(int index, TOrig item\)>](#)

## Properties

### Count

Gets the number of elements in the collection.

```
public override int Count { get; }
```

### Property Value

[int](#)

The number of elements in the collection.

### this[int]

Gets the element at the specified index in the read-only list.

```
public override TNew this[int index] { get; }
```

## Parameters

index `int`

The zero-based index of the element to get.

## Property Value

TNew

The element at the specified index in the read-only list.

## Value

```
public TNew Value { get; protected set; }
```

## Property Value

TNew

## Methods

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public override void Dispose()
```

### GetEnumerator()

Returns an enumerator that iterates through the collection.

```
public override IEnumerator<TNew> GetEnumerator()
```

## Returns

## [IEnumerator](#) <TNew>

An enumerator that can be used to iterate through the collection.

# Class RtObserverList<T>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public class RtObserverList<T> : RtReadOnlyList<T>, IDisposable, IReadOnlyList<T>,  
IReadOnlyCollection<T>, IEnumerable<T>, IEnumerable
```

## Type Parameters

T

## Inheritance

```
object ↪ RtCollection<(int index, T item)> ↪ RtReadOnlyList<T> ↪  
RtObserverList<T>
```

## Implements

```
IDisposable, IReadOnlyList<T>, IReadOnlyCollection<T>, IEnumerable<T>,  
IEnumerable
```

## Inherited Members

```
RtCollection<(int index, T item)>.CollectionEvents ,  
RtCollection<(int index, T item)>.Events , RtCollection<(int index, T item)>.Dispose() ,  
object.Equals(object) , object.Equals(object, object) , object.GetHashCode() ,  
object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) ,  
object.ToString()
```

## Extension Methods

```
Linq.count<T>(RtReadOnlyList<T>) ,  
Linq.groupBy< TKey, TValue >(RtReadOnlyList< TValue >, Func< TValue, TKey >) ,  
Linq.selectNew< T, TSelect >(RtReadOnlyList< T >, Func< T, TSelect >) ,  
Linq.select< T, TSelect >(RtReadOnlyList< T >, Func< T, TSelect >)
```

## Constructors

```
RtObserverList(IEnumerable<T>,  
ICollectionEvents<T>)
```

```
public RtObserverList(IEnumerable<T> sourceItems, ICollectionEvents<T> sourceEvents)
```

## Parameters

sourceItems [IEnumerable](#)<T>

sourceEvents [ICollectionEvents](#)<T>

## Properties

### Count

Gets the number of elements in the collection.

```
public override int Count { get; }
```

### Property Value

[int](#)

The number of elements in the collection.

### this[int]

Gets the element at the specified index in the read-only list.

```
public override T this[int index] { get; }
```

## Parameters

index [int](#)

The zero-based index of the element to get.

### Property Value

T

The element at the specified index in the read-only list.

## Methods

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public override void Dispose()
```

### GetEnumerator()

Returns an enumerator that iterates through the collection.

```
public override IEnumerator<T> GetEnumerator()
```

Returns

[IEnumerator](#) <T>

An enumerator that can be used to iterate through the collection.

# Class RtReadOnlyDictionary<TKey, TValue>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public abstract class RtReadOnlyDictionary<TKey, TValue> :  
RtCollection<KeyValuePair<TKey, TValue>>, IDisposable, IReadOnlyDictionary<TKey,  
TValue>, IReadOnlyCollection<KeyValuePair<TKey, TValue>>,  
IEnumerable<KeyValuePair<TKey, TValue>>, IEnumerable
```

## Type Parameters

TKey

TValue

## Inheritance

```
object ← RtCollection<KeyValuePair<TKey, TValue>> ←  
RtReadOnlyDictionary<TKey, TValue>
```

## Implements

```
IDisposable, IReadOnlyDictionary<TKey, TValue>,  
IReadOnlyCollection<KeyValuePair<TKey, TValue>>,  
IEnumerable<KeyValuePair<TKey, TValue>>, IEnumerable
```

## Derived

```
RtDictionary<TKey, TValue>, RtGroup<TKey, TValue>,  
RtMutatedDictionaryBase<TSourceKey, TSourceValue, TKey, TValue>
```

## Inherited Members

```
RtCollection<KeyValuePair<TKey, TValue>>.CollectionEvents ,  
RtCollection<KeyValuePair<TKey, TValue>>.Events ,  
RtCollection<KeyValuePair<TKey, TValue>>.Dispose\(\) , object.Equals\(object\) ,  
object.Equals\(object, object\) , object.GetHashCode\(\) , object.GetType\(\) ,  
object.MemberwiseClone\(\) , object.ReferenceEquals\(object, object\) , object.ToString\(\)
```

## Extension Methods

```
Linq.selectKey<TOrigKey, TSelectedKey, TValue>(RtReadOnlyDictionary<TOrigKey, TValue>, Func<TOrigKey, TSelectedKey>),
Linq.selectValue<TKey, TValue, TSelectValue>(RtReadOnlyDictionary<TKey, TValue>, Func<TValue, TSelectValue>),
Linq.where<TKey, TValue>(RtReadOnlyDictionary<TKey, TValue>, Func<KeyValuePair<TKey, TValue>, bool>)
```

## Properties

### Count

Gets the number of elements in the collection.

```
public abstract int Count { get; }
```

### Property Value

[int](#)

The number of elements in the collection.

### this[TKey]

Gets the element that has the specified key in the read-only dictionary.

```
public abstract TValue this[TKey key] { get; }
```

### Parameters

**key** TKey

The key to locate.

### Property Value

TValue

The element that has the specified key in the read-only dictionary.

# Exceptions

## [ArgumentNullException](#)

`key` is `null`.

## [KeyNotFoundException](#)

The property is retrieved and `key` is not found.

# Keys

Gets an enumerable collection that contains the keys in the read-only dictionary.

```
public abstract IEnumerable<TKey> Keys { get; }
```

## Property Value

### [IEnumerable](#)<TKey>

An enumerable collection that contains the keys in the read-only dictionary.

# RtKeys

```
public RtReadOnlyList<TKey> RtKeys { get; }
```

## Property Value

### [RtReadOnlyList](#)<TKey>

# RtPairs

```
public RtReadOnlyList<KeyValuePair<TKey, TValue>> RtPairs { get; }
```

## Property Value

### [RtReadOnlyList](#)<[KeyValuePair](#)<TKey, TValue>>

## RtValues

```
public RtReadOnlyList<TValue> RtValues { get; }
```

Property Value

[RtReadOnlyList<TValue>](#)

## Values

Gets an enumerable collection that contains the values in the read-only dictionary.

```
public abstract IEnumerable<TValue> Values { get; }
```

Property Value

[IEnumerable<TValue>](#)

An enumerable collection that contains the values in the read-only dictionary.

## Methods

### ContainsKey(TKey)

Determines whether the read-only dictionary contains an element that has the specified key.

```
public abstract bool ContainsKey(TKey key)
```

Parameters

**key** TKey

The key to locate.

Returns

bool

true if the read-only dictionary contains an element that has the specified key; otherwise, false.

Exceptions

ArgumentNullException

`key` is null.

## GetEnumerator()

Returns an enumerator that iterates through the collection.

```
public abstract IEnumarator<KeyValuePair<TKey, TValue>> GetEnumerator()
```

Returns

IEnumerator<KeyValuePair<TKey, TValue>>

An enumerator that can be used to iterate through the collection.

## TryGetValue(TKey, out TValue)

Gets the value that is associated with the specified key.

```
public abstract bool TryGetValue(TKey key, out TValue value)
```

Parameters

`key` TKey

The key to locate.

`value` TValue

When this method returns, the value associated with the specified key, if the key is found; otherwise, the default value for the type of the `value` parameter. This parameter is passed uninitialized.

Returns

`bool`

`true` if the object that implements the [IReadOnlyDictionary<TKey, TValue>](#) interface contains an element that has the specified key; otherwise, `false`.

Exceptions

[ArgumentNullException](#)

`key` is `null`.

# Class RtReadOnlyList<T>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public abstract class RtReadOnlyList<T> : RtCollection<(int index, T item)>,
IDisposable, IReadOnlyList<T>, IReadOnlyCollection<T>, IEnumerable<T>, IEnumerable
```

## Type Parameters

T

## Inheritance

[object](#) ← [RtCollection<\(int index, T item\)>](#) ← [RtReadOnlyList<T>](#)

## Implements

[IDisposable](#), [IReadOnlyList](#)<T>, [IReadOnlyCollection](#)<T>, [IEnumerable](#)<T>, [IEnumerable](#)

## Derived

[RtList](#)<T>, [RtMutatedListBase](#)<TOrig, TNew>, [RtMutatedLocalListBase](#)<TOrig, TNew>, [RtObserverList](#)<T>, [Rt](#)<T>

## Inherited Members

[RtCollection<\(int index, T item\)>.CollectionEvents](#) ,  
[RtCollection<\(int index, T item\)>.Events](#) , [RtCollection<\(int index, T item\)>.Dispose\(\)](#) ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Extension Methods

[Linq.count](#)<T>([RtReadOnlyList](#)<T>) ,  
[Linq.groupBy](#)< TKey, TValue >([RtReadOnlyList](#)< TValue >, [Func](#)< TValue, TKey >) ,  
[Linq.selectNew](#)< T, TSelect >([RtReadOnlyList](#)< T >, [Func](#)< T, TSelect >) ,  
[Linq.select](#)< T, TSelect >([RtReadOnlyList](#)< T >, [Func](#)< T, TSelect >)

## Properties

# Count

Gets the number of elements in the collection.

```
public abstract int Count { get; }
```

## Property Value

[int](#)

The number of elements in the collection.

## this[int]

Gets the element at the specified index in the read-only list.

```
public abstract T this[int index] { get; }
```

## Parameters

[index](#) [int](#)

The zero-based index of the element to get.

## Property Value

T

The element at the specified index in the read-only list.

# Methods

## GetEnumerator()

Returns an enumerator that iterates through the collection.

```
public abstract IEnumerator<T> GetEnumerator()
```

Returns

[IEnumerator](#) <T>

An enumerator that can be used to iterate through the collection.

## Operators

implicit operator RtReadOnlyList<object>  
(RtReadOnlyList<T>)

```
public static implicit operator RtReadOnlyList<object>(RtReadOnlyList<T> list)
```

Parameters

list [RtReadOnlyList](#)<T>

Returns

[RtReadOnlyList](#)<object>

# Class RtSelectKeyDictionary<TOrigKey, TSelectKey, TValue>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

Represents a dictionary that selects keys based on a provided function.

```
public class RtSelectKeyDictionary<TOrigKey, TSelectKey, TValue> :  
    RtMutatedDictionaryBase<TOrigKey, TValue, TSelectKey, TValue>, IDisposable,  
    IReadOnlyDictionary<TSelectKey, TValue>,  
    IReadOnlyCollection<KeyValuePair<TSelectKey, TValue>>,  
    IEnumerable<KeyValuePair<TSelectKey, TValue>>, IEnumerable
```

## Type Parameters

### TOrigKey

The type of the original keys in the source dictionary.

### TSelectKey

The type of the selected keys in the mutated dictionary.

### TValue

The type of values in the dictionary.

## Inheritance

```
object ↵ ← RtCollection<KeyValuePair ↵ <TSelectKey, TValue>> ←  
RtReadOnlyDictionary<TSelectKey, TValue> ←  
RtMutatedDictionaryBase<TOrigKey, TValue, TSelectKey, TValue> ←  
RtSelectKeyDictionary<TOrigKey, TSelectKey, TValue>
```

## Implements

```
IDisposable ↵ , IReadOnlyDictionary ↵ <TSelectKey, TValue>,  
IReadOnlyCollection ↵ <KeyValuePair ↵ <TSelectKey, TValue>>,  
IEnumerable ↵ <KeyValuePair ↵ <TSelectKey, TValue>> , IEnumerable ↵
```

## Inherited Members

[RtMutatedDictionaryBase<TOrigKey, TValue, TSelectKey, TValue>.mutated](#) ,  
[RtMutatedDictionaryBase<TOrigKey, TValue, TSelectKey, TValue>.Dispose\(\)](#) ,  
[RtMutatedDictionaryBase<TOrigKey, TValue, TSelectKey, TValue>.OnSourceUpdate\(in Dictionary<TSelectKey, TValue>, KeyValuePair<TOrigKey, TValue>, KeyValuePair<TOrigKey, TValue>\)](#) ,  
[RtMutatedDictionaryBase<TOrigKey, TValue, TSelectKey, TValue>.OnSourceRemove\(in Dictionary<TSelectKey, TValue>, KeyValuePair<TOrigKey, TValue>\)](#) ,  
[RtMutatedDictionaryBase<TOrigKey, TValue, TSelectKey, TValue>.OnSourceAdd\(in Dictionary<TSelectKey, TValue>, KeyValuePair<TOrigKey, TValue>\)](#) ,  
[RtMutatedDictionaryBase<TOrigKey, TValue, TSelectKey, TValue>.GetEnumerator\(\)](#) ,  
[RtMutatedDictionaryBase<TOrigKey, TValue, TSelectKey, TValue>.Count](#) ,  
[RtMutatedDictionaryBase<TOrigKey, TValue, TSelectKey, TValue>.ContainsKey\(TSelectKey\)](#) ,  
  
[RtMutatedDictionaryBase<TOrigKey, TValue, TSelectKey, TValue>.TryGetValue\(TSelectKey, out TValue\)](#) ,  
[RtMutatedDictionaryBase<TOrigKey, TValue, TSelectKey, TValue>.this\[TSelectKey\]](#) ,  
[RtMutatedDictionaryBase<TOrigKey, TValue, TSelectKey, TValue>.Keys](#) ,  
[RtMutatedDictionaryBase<TOrigKey, TValue, TSelectKey, TValue>.Values](#) ,  
[RtReadOnlyDictionary<TSelectKey, TValue>.Count](#) ,  
[RtReadOnlyDictionary<TSelectKey, TValue>.ContainsKey\(TSelectKey\)](#) ,  
[RtReadOnlyDictionary<TSelectKey, TValue>.TryGetValue\(TSelectKey, out TValue\)](#) ,  
[RtReadOnlyDictionary<TSelectKey, TValue>.this\[TSelectKey\]](#) ,  
[RtReadOnlyDictionary<TSelectKey, TValue>.Keys](#) ,  
[RtReadOnlyDictionary<TSelectKey, TValue>.Values](#) ,  
[RtReadOnlyDictionary<TSelectKey, TValue>.RtPairs](#) ,  
[RtReadOnlyDictionary<TSelectKey, TValue>.RtKeys](#) ,  
[RtReadOnlyDictionary<TSelectKey, TValue>.RtValues](#) ,  
[RtReadOnlyDictionary<TSelectKey, TValue>.GetEnumerator\(\)](#) ,  
[RtCollection<KeyValuePair<TSelectKey, TValue>>.CollectionEvents](#) ,  
[RtCollection<KeyValuePair<TSelectKey, TValue>>.Events](#) ,  
[RtCollection<KeyValuePair<TSelectKey, TValue>>.Dispose\(\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Extension Methods

[Linq.selectKey<TOrigKey, TSelectedKey, TValue>\(RtReadOnlyDictionary<TOrigKey, TValue>, Func<TOrigKey, TSelectedKey>\)](#) ,  
[Linq.selectValue< TKey, TValue, TSelectValue>\(RtReadOnlyDictionary< TKey, TValue>, Func< TValue, TSelectValue>\)](#) ,  
[Linq.where< TKey, TValue>\(RtReadOnlyDictionary< TKey, TValue>, Func< KeyValuePair< TKey, TValue>, bool>\)](#)

# Constructors

## RtSelectKeyDictionary(RtReadOnlyDictionary<TOrigKey, TValue>, Func<TOrigKey, TSelectKey>)

Initializes a new instance of the [RtSelectKeyDictionary<TOrigKey, TSelectKey, TValue>](#) class.

```
public RtSelectKeyDictionary(RtReadOnlyDictionary<TOrigKey, TValue> source,
    Func<TOrigKey, TSelectKey> selectFn)
```

## Parameters

**source** [RtReadOnlyDictionary<TOrigKey, TValue>](#)

The source read-only dictionary.

**selectFn** [Func<TOrigKey, TSelectKey>](#)

The function to select keys.

## Methods

### \_OnSourceAdd(in Dictionary<TSelectKey, TValue>, KeyValuePair<TOrigKey, TValue>)

```
protected override void _OnSourceAdd(in Dictionary<TSelectKey, TValue> mutated,
    KeyValuePair<TOrigKey, TValue> kvp)
```

## Parameters

**mutated** [Dictionary<TSelectKey, TValue>](#)

**kvp** [KeyValuePair<TOrigKey, TValue>](#)

### \_OnSourceRemove(in Dictionary<TSelectKey, TValue>, KeyValuePair<TOrigKey, TValue>)

```
protected override void _OnSourceRemove(in Dictionary<TSelectKey, TValue> mutated,  
KeyValuePair<TOrigKey, TValue> kvp)
```

## Parameters

mutated [Dictionary](#)<TSelectKey, TValue>

kvp [KeyValuePair](#)<TOrigKey, TValue>

```
_OnSourceUpdate(in Dictionary<TSelectKey, TValue>,  
KeyValuePair<TOrigKey, TValue>,  
KeyValuePair<TOrigKey, TValue>)
```

```
protected override void _OnSourceUpdate(in Dictionary<TSelectKey, TValue> mutated,  
KeyValuePair<TOrigKey, TValue> oldPair, KeyValuePair<TOrigKey, TValue> newPair)
```

## Parameters

mutated [Dictionary](#)<TSelectKey, TValue>

oldPair [KeyValuePair](#)<TOrigKey, TValue>

newPair [KeyValuePair](#)<TOrigKey, TValue>

# Class RtSelectList<TOrig, TNew>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

Represents a list that projects each element of a source list into a new form.

```
public class RtSelectList<TOrig, TNew> : RtMutatedListBase<TOrig,
TNew>, IDisposable, IReadOnlyList<TNew>, IReadOnlyCollection<TNew>,
IEnumerable<TNew>, IEnumerable
```

## Type Parameters

### TOrig

The type of elements in the source list.

### TNew

The type of elements in the projected list.

## Inheritance

```
object ↵ ← RtCollection<(int ↵ index ↵ , TNew item ↵ )> ← RtReadOnlyList<TNew> ←
RtMutatedListBase<TOrig, TNew> ← RtSelectList<TOrig, TNew>
```

## Implements

```
IDisposable, IReadOnlyList<TNew>, IReadOnlyCollection<TNew>,
IEnumerable<TNew>, IEnumerable
```

## Inherited Members

```
RtMutatedListBase<TOrig, TNew>._mutated , RtMutatedListBase<TOrig, TNew>.Dispose() ,
RtMutatedListBase<TOrig, TNew>._OnSourceUpdate(List<TNew> , (int index, TOrig item),
(int index, TOrig item)) ,
RtMutatedListBase<TOrig, TNew>._OnSourceRemove(List<TNew> , (int index, TOrig item))
,
RtMutatedListBase<TOrig, TNew>._OnSourceAdd(List<TNew> , (int index, TOrig item)) ,
RtMutatedListBase<TOrig, TNew>.GetEnumerator() ,
RtMutatedListBase<TOrig, TNew>.Count , RtMutatedListBase<TOrig, TNew>.this[int] ,
RtReadOnlyList<TNew>.GetEnumerator() , RtReadOnlyList<TNew>.Count ,
RtReadOnlyList<TNew>.this[int] , RtCollection<(int index, TNew item)>.CollectionEvents ,
RtCollection<(int index, TNew item)>.Events ,
```

[RtCollection<\(int index, TNew item\)>.Dispose\(\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Extension Methods

[Linq.count<T>\(RtReadOnlyList<T>\)](#) ,  
[Linq.groupBy< TKey, TValue >\(RtReadOnlyList< TValue >, Func< TValue, TKey >\)](#) ,  
[Linq.selectNew< T, TSelect >\(RtReadOnlyList< T >, Func< T, TSelect >\)](#) ,  
[Linq.select< T, TSelect >\(RtReadOnlyList< T >, Func< T, TSelect >\)](#).

## Constructors

RtSelectList(RtReadOnlyList<TOrig>, Func<TOrig, TNew>)

```
public RtSelectList(RtReadOnlyList<TOrig> source, Func<TOrig, TNew> selectFn)
```

### Parameters

source [RtReadOnlyList<TOrig>](#)

selectFn [Func< TOrig, TNew >](#)

## Methods

\_OnSourceAdd(List<TNew>, (int index, TOrig item))

```
protected override void _OnSourceAdd(List<TNew> mutated, (int index, TOrig item) item)
```

### Parameters

mutated [List<TNew>](#)

item ([int](#) [index](#) , TOrig [item](#))

## \_OnSourceRemove(List<TNew>, (int index, TOrig item))

```
protected override void _OnSourceRemove(List<TNew> mutated, (int index, TOrig item) item)
```

### Parameters

mutated [List](#)<TNew>

item ([int](#) [index](#), TOrig [item](#))

## \_OnSourceUpdate(List<TNew>, (int index, TOrig item), (int index, TOrig item))

```
protected override void _OnSourceUpdate(List<TNew> mutated, (int index, TOrig item)  
oldItem, (int index, TOrig item) newItem)
```

### Parameters

mutated [List](#)<TNew>

oldItem ([int](#) [index](#), TOrig [item](#))

newItem ([int](#) [index](#), TOrig [item](#))

# Class RtSelectLocalList<T, TSelect>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public class RtSelectLocalList<T, TSelect> : RtMutatedLocalListBase<T, TSelect>,  
IDisposable, IReadOnlyList<TSelect>, IReadOnlyCollection<TSelect>,  
IEnumerable<TSelect>, IEnumerable
```

## Type Parameters

T

TSelect

## Inheritance

```
object ↪ RtCollection<(int ↪ index ↪ , TSelect item ↪ )> ↪ RtReadOnlyList<TSelect> ↪  
RtMutatedLocalListBase<T, TSelect> ↪ RtSelectLocalList<T, TSelect>
```

## Implements

```
IDisposable, IReadOnlyList<TSelect>, IReadOnlyCollection<TSelect>,  
IEnumerable<TSelect>, IEnumerable
```

## Inherited Members

```
RtMutatedLocalListBase<T, TSelect>.Dispose(),  
RtMutatedLocalListBase<T, TSelect>._OnSourceUpdate((int index, T item), (int index, T  
item)),  
RtMutatedLocalListBase<T, TSelect>._OnSourceRemove((int index, T item)),  
RtMutatedLocalListBase<T, TSelect>._OnSourceAdd((int index, T item)),  
RtCollection<(int index, TSelect item)>.CollectionEvents,  
RtCollection<(int index, TSelect item)>.Events,  
RtCollection<(int index, TSelect item)>.Dispose(), object.Equals(object) ↪ ,  
object.Equals(object, object) ↪ , object.GetHashCode() ↪ , object.GetType() ↪ ,  
object.MemberwiseClone() ↪ , object.ReferenceEquals(object, object) ↪ , object.ToString() ↪
```

## Extension Methods

```
Linq.count<T>(RtReadOnlyList<T>),  
Linq.groupBy< TKey, TValue >(RtReadOnlyList< TValue >, Func< TValue, TKey >),  
Linq.selectNew<T, TSelect>(RtReadOnlyList<T>, Func<T, TSelect>),  
Linq.select<T, TSelect>(RtReadOnlyList<T>, Func<T, TSelect>).
```

# Constructors

RtSelectLocalList(RtReadOnlyList<T>, Func<T, TSelect>)

```
public RtSelectLocalList(RtReadOnlyList<T> source, Func<T, TSelect> selectFn)
```

## Parameters

source [RtReadOnlyList<T>](#)

selectFn [Func<T, TSelect>](#)

# Properties

## Count

Gets the number of elements in the collection.

```
public override int Count { get; }
```

## Property Value

[int](#)

The number of elements in the collection.

## this[int]

Gets the element at the specified index in the read-only list.

```
public override TSelect this[int index] { get; }
```

## Parameters

index [int](#)

The zero-based index of the element to get.

## Property Value

### TSelect

The element at the specified index in the read-only list.

## Methods

### GetEnumerator()

Returns an enumerator that iterates through the collection.

```
public override IEnumerator<TSelect> GetEnumerator()
```

Returns

### IEnumerator<TSelect>

An enumerator that can be used to iterate through the collection.

### \_OnSourceAdd((int index, T item))

```
protected override void _OnSourceAdd((int index, T item) listItem)
```

Parameters

### listItem (int index, T item)

### \_OnSourceRemove((int index, T item))

```
protected override void _OnSourceRemove((int index, T item) listItem)
```

Parameters

`listItem (int index, T item)`

## \_OnSourceUpdate((int index, T item), (int index, T item))

```
protected override void _OnSourceUpdate((int index, T item) oldItem, (int index, T item) newItem)
```

### Parameters

`oldItem (int index, T item)`

`newItem (int index, T item)`

# Class RtSelectValueDictionary<TKey, TOrigValue, TValue>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

Represents a dictionary that selects values based on a provided function.

```
public class RtSelectValueDictionary<TKey, TOrigValue, TValue> :  
    RtMutatedDictionaryBase<TKey, TOrigValue, TKey, TValue>, IDisposable,  
    IReadOnlyDictionary<TKey, TValue>, IReadOnlyCollection<KeyValuePair<TKey, TValue>>,  
    IEnumerable<KeyValuePair<TKey, TValue>>, IEnumerable
```

## Type Parameters

### TKey

The type of keys in the dictionary.

### TOrigValue

The type of the original values in the source dictionary.

### TValue

The type of the selected values in the mutated dictionary.

## Inheritance

```
object ↵ ← RtCollection<KeyValuePair ↵ <TKey, TValue>> ←  
RtReadOnlyDictionary<TKey, TValue> ←  
RtMutatedDictionaryBase<TKey, TOrigValue, TKey, TValue> ←  
RtSelectValueDictionary<TKey, TOrigValue, TValue>
```

## Implements

```
IDisposable ↵ , IReadOnlyDictionary<TKey, TValue> ,  
IReadOnlyCollection<KeyValuePair<TKey, TValue>> ,  
IEnumerable<KeyValuePair<TKey, TValue>> , IEnumerable
```

## Inherited Members

```
RtMutatedDictionaryBase<TKey, TOrigValue, TKey, TValue>._mutated ,  
RtMutatedDictionaryBase<TKey, TOrigValue, TKey, TValue>.Dispose() ,
```

[RtMutatedDictionaryBase<TKey, TValue>.OnSourceUpdate\(in Dictionary<TKey, TValue>, KeyValuePair<TKey, TValue>, KeyValuePair<TKey, TValue>\)](#) ,  
[RtMutatedDictionaryBase<TKey, TValue>.OnSourceRemove\(in Dictionary<TKey, TValue>, KeyValuePair<TKey, TValue>\)](#) ,  
[RtMutatedDictionaryBase<TKey, TValue>.OnSourceAdd\(in Dictionary<TKey, TValue>, KeyValuePair<TKey, TValue>\)](#) ,  
[RtMutatedDictionaryBase<TKey, TValue>.GetEnumerator\(\)](#) ,  
[RtMutatedDictionaryBase<TKey, TValue>.Count](#) ,  
[RtMutatedDictionaryBase<TKey, TValue>.ContainsKey\(TKey\)](#) ,  
[RtMutatedDictionaryBase<TKey, TValue>.TryGetValue\(TKey, out TValue\)](#) ,  
[RtMutatedDictionaryBase<TKey, TValue>.this\[TKey\]](#) ,  
[RtMutatedDictionaryBase<TKey, TValue>.Keys](#) ,  
[RtMutatedDictionaryBase<TKey, TValue>.Values](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.Count](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.ContainsKey\(TKey\)](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.TryGetValue\(TKey, out TValue\)](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.this\[TKey\]](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.Keys](#) , [RtReadOnlyDictionary<TKey, TValue>.Values](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.RtPairs](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.RtKeys](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.RtValues](#) ,  
[RtReadOnlyDictionary<TKey, TValue>.GetEnumerator\(\)](#) ,  
[RtCollection<KeyValuePair<TKey, TValue>>.CollectionEvents](#) ,  
[RtCollection<KeyValuePair<TKey, TValue>>.Events](#) ,  
[RtCollection<KeyValuePair<TKey, TValue>>.Dispose\(\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Extension Methods

[Linq.selectKey<TOrigKey, TSelectedKey, TValue>\(RtReadOnlyDictionary<TOrigKey, TValue>, Func<TOrigKey, TSelectedKey>\)](#) ,  
[Linq.selectValue<TKey, TValue, TSelectValue>\(RtReadOnlyDictionary<TKey, TValue>, Func<TValue, TSelectValue>\)](#) ,  
[Linq.where<TKey, TValue>\(RtReadOnlyDictionary<TKey, TValue>, Func<KeyValuePair<TKey, TValue>, bool>\)](#)

## Constructors

# RtSelectValueDictionary(RtReadOnlyDictionary<TKey, TOrigValue>, Func<TOrigValue, TValue>)

Initializes a new instance of the [RtSelectValueDictionary<TKey, TOrigValue, TValue>](#) class.

```
public RtSelectValueDictionary(RtReadOnlyDictionary<TKey, TOrigValue> source,  
Func<TOrigValue, TValue> selectFn)
```

## Parameters

**source** [RtReadOnlyDictionary<TKey, TOrigValue>](#)

The source read-only dictionary.

**selectFn** [Func<TOrigValue, TValue>](#)

The function to select values.

## Methods

### \_OnSourceAdd(in Dictionary<TKey, TValue>, KeyValuePair<TKey, TOrigValue>)

```
protected override void _OnSourceAdd(in Dictionary<TKey, TValue> mutated,  
KeyValuePair<TKey, TOrigValue> kvp)
```

## Parameters

**mutated** [Dictionary<TKey, TValue>](#)

**kvp** [KeyValuePair<TKey, TOrigValue>](#)

### \_OnSourceRemove(in Dictionary<TKey, TValue>, KeyValuePair<TKey, TOrigValue>)

```
protected override void _OnSourceRemove(in Dictionary<TKey, TValue> mutated,  
KeyValuePair<TKey, TOrigValue> kvp)
```

## Parameters

**mutated** [Dictionary](#)<TKey, TValue>

**kvp** [KeyValuePair](#)<TKey, TOrigValue>

**\_OnSourceUpdate(in Dictionary<TKey, TValue>, KeyValuePair<TKey, TOrigValue>, KeyValuePair<TKey, TOrigValue>)**

```
protected override void _OnSourceUpdate(in Dictionary<TKey, TValue> mutated,
    KeyValuePair<TKey, TOrigValue> oldPair, KeyValuePair<TKey, TOrigValue> newPair)
```

## Parameters

**mutated** [Dictionary](#)<TKey, TValue>

**oldPair** [KeyValuePair](#)<TKey, TOrigValue>

**newPair** [KeyValuePair](#)<TKey, TOrigValue>

# Class Rt<T>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public class Rt<T> : RtReadOnlyList<T>, IDisposable, IReadOnlyList<T>,  
IReadOnlyCollection<T>, IEnumerable<T>, IEnumerable
```

## Type Parameters

T

### Inheritance

[object](#) ← [RtCollection<\(int index, T item\)>](#) ← [RtReadOnlyList<T>](#) ← [Rt<T>](#)

### Implements

[IDisposable](#) ,  [IReadOnlyList](#) <T> ,  [IReadOnlyCollection](#) <T> ,  [IEnumerable](#) <T> ,  
 [IEnumerable](#)

### Inherited Members

[RtCollection<\(int index, T item\)>.CollectionEvents](#) ,  
[RtCollection<\(int index, T item\)>.Events](#) , [RtCollection<\(int index, T item\)>.Dispose\(\)](#) ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

### Extension Methods

[Linq.count<T>\(RtReadOnlyList<T>\)](#) ,  
[Linq.groupBy< TKey, TValue >\(RtReadOnlyList< TValue >, Func< TValue, TKey >\)](#) ,  
[Linq.selectNew< T, TSelect >\(RtReadOnlyList< T >, Func< T, TSelect >\)](#) ,  
[Linq.select< T, TSelect >\(RtReadOnlyList< T >, Func< T, TSelect >\)](#)

## Constructors

### Rt()

```
public Rt()
```

# Rt(T)

```
public Rt(T defaultValue)
```

## Parameters

defaultValue T

## Properties

### Count

Gets the number of elements in the collection.

```
public override int Count { get; }
```

## Property Value

[int](#)

The number of elements in the collection.

### this[int]

Gets the element at the specified index in the read-only list.

```
public override T this[int index] { get; }
```

## Parameters

index [int](#)

The zero-based index of the element to get.

## Property Value

T

The element at the specified index in the read-only list.

## Value

```
public T Value { get; }
```

Property Value

T

## Methods

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public override void Dispose()
```

### GetEnumerator()

Returns an enumerator that iterates through the collection.

```
public override IEnumerator<T> GetEnumerator()
```

Returns

[IEnumerator](#) <T>

An enumerator that can be used to iterate through the collection.

### Set(T)

```
public void Set(T value)
```

Parameters

value T

## SetNoTrigger(T)

```
public void SetNoTrigger(T value)
```

Parameters

value T

## Operators

### implicit operator T(Rt<T>)

```
public static implicit operator T(Rt<T> rt)
```

Parameters

rt Rt<T>

Returns

T

# Class Subscription

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public abstract class Subscription
```

## Inheritance

[object](#) ← Subscription

## Derived

[SubscriptionBinding<TDelegate>](#), [SubscriptionGroup](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Extension Methods

[SubscriptionHandleExtension.UnsubscribeIfNotNull\(Subscription\)](#)

## Methods

### Unsubscribe()

```
public abstract void Unsubscribe()
```

# Class SubscriptionBinding<TDelegate>

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public class SubscriptionBinding<TDelegate> : Subscription where TDelegate : class
```

## Type Parameters

TDelegate

### Inheritance

[object](#) ← [Subscription](#) ← SubscriptionBinding<TDelegate>

### Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

### Extension Methods

[SubscriptionHandleExtension.UnsubscribeIfNotNull\(Subscription\)](#)

## Constructors

SubscriptionBinding(TDelegate,  
RelayBase<TDelegate>)

```
public SubscriptionBinding(TDelegate listener, RelayBase<TDelegate> relay)
```

## Parameters

listener TDelegate

relay [RelayBase](#)<TDelegate>

## Fields

# Listener

```
public readonly TDelegate Listener
```

## Field Value

TDelegate

## Methods

### IsListener(TDelegate)

```
public bool IsListener(TDelegate d)
```

## Parameters

d TDelegate

## Returns

[bool](#)

### Unsubscribe()

```
public override void Unsubscribe()
```

# Class SubscriptionGroup

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public class SubscriptionGroup : Subscription
```

## Inheritance

[object](#) ← [Subscription](#) ← SubscriptionGroup

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Extension Methods

[SubscriptionHandleExtension.UnsubscribeIfNotNull\(Subscription\)](#)

## Methods

### Add(Subscription)

```
public void Add(Subscription subscription)
```

#### Parameters

subscription [Subscription](#)

### Unsubscribe()

```
public override void Unsubscribe()
```

# Class SubscriptionHandleExtension

Namespace: [cfEngine.Rx](#)

Assembly: CatSweeper.dll

```
public static class SubscriptionHandleExtension
```

## Inheritance

[object](#) ← SubscriptionHandleExtension

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### UnsubscribeIfNotNull(Subscription)

```
public static void UnsubscribeIfNotNull(this Subscription binding)
```

#### Parameters

binding [Subscription](#)

# Namespace cfEngine.Rx.Test

## Classes

[RtDictionary\\_Test](#)

[RtList\\_Test](#)

# Class RtDictionary\_Test

Namespace: [cfEngine.Rx.Test](#)

Assembly: CatSweeper.dll

```
[TestFixture]
public class RtDictionary_Test
```

## Inheritance

[object](#) ← RtDictionary\_Test

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### RtDictionary\_AddRemove()

```
[Test]
public void RtDictionary_AddRemove()
```

### RtDictionary\_Dispose()

```
[Test]
public void RtDictionary_Dispose()
```

### RtDictionary\_NotCacheSubscription()

```
[Test]
public void RtDictionary_NotCacheSubscription()
```

## RtDictionary\_Remove()

```
[Test]  
public void RtDictionary_Remove()
```

## RtDictionary\_Upsert()

```
[Test]  
public void RtDictionary_Upsert()
```

## RtReadOnlyDictionary\_RtKeys()

```
[Test]  
public void RtReadOnlyDictionary_RtKeys()
```

## RtReadOnlyDictionary\_RtPairs()

```
[Test]  
public void RtReadOnlyDictionary_RtPairs()
```

## RtReadOnlyDictionary\_RtValues()

```
[Test]  
public void RtReadOnlyDictionary_RtValues()
```

# Class RtList\_Test

Namespace: [cfEngine.Rx.Test](#)

Assembly: CatSweeper.dll

```
[TestFixture]
public class RtList_Test
```

## Inheritance

[object](#) ← RtList\_Test

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### RtList\_Add()

```
[Test]
public void RtList_Add()
```

### RtList\_Dispose()

```
[Test]
public void RtList_Dispose()
```

# Namespace cfEngine.Serialize

## Classes

[JsonSerializer](#)

[Serializer](#)

## Interfaces

[ISerializer](#)

[ISerializer.IDeserializeParam](#)

[ISerializer.ISerializeParam](#)

# Interface ISerializer

Namespace: [cfEngine.Serialize](#)

Assembly: CatSweeper.dll

```
public interface ISerializer
```

## Methods

### Deserialize(byte[], IDeserializeParam)

```
object Deserialize(byte[] byteLoaded, ISerializer.IDeserializeParam param = null)
```

#### Parameters

byteLoaded [byte\[\]](#)

param [ISerializer.IDeserializeParam](#)

#### Returns

[object](#)

### DeserializeAsAsync<T>(byte[], IDeserializeParam, CancellationToken)

```
Task<T> DeserializeAsAsync<T>(byte[] byteLoaded, ISerializer.IDeserializeParam param = null, CancellationToken token = default)
```

#### Parameters

byteLoaded [byte\[\]](#)

param [ISerializer.IDeserializeParam](#)

token [CancellationToken](#)

Returns

[Task](#)<T>

Type Parameters

T

**DeserializeAs<T>(byte[], IDeserializeParam)**

T DeserializeAs<T>(byte[] byteLoaded, ISerializer.IDeserializeParam param = null)

Parameters

byteLoaded [byte](#)[]

param [ISerializer.IDeserializeParam](#)

Returns

T

Type Parameters

T

**DeserializeAsync(byte[], IDeserializeParam, CancellationToken)**

Task<object> DeserializeAsync(byte[] byteLoaded, ISerializer.IDeserializeParam param = null, CancellationToken token = default)

Parameters

byteLoaded [byte](#)[]

param [ISerializer.IDeserializeParam](#)

token [CancellationToken](#)

Returns

[Task](#)<[object](#)>

## Serialize(object, ISerializeParam)

```
string Serialize(object obj, ISerializer.ISerializeParam param = null)
```

Parameters

obj [object](#)

param [ISerializer.ISerializeParam](#)

Returns

[string](#)

## SerializeAsync(object, ISerializeParam, CancellationToken)

```
Task<string> SerializeAsync(object obj, ISerializer.ISerializeParam param = null, CancellationToken token = default)
```

Parameters

obj [object](#)

param [ISerializer.ISerializeParam](#)

token [CancellationToken](#)

Returns

[Task](#)<[string](#)>

# Interface ISerializer.IDeserializeParam

Namespace: [cfEngine.Serialize](#)

Assembly: CatSweeper.dll

```
public interface ISerializer.IDeserializeParam
```

# Interface ISerializer.ISerializeParam

Namespace: [cfEngine.Serialize](#)

Assembly: CatSweeper.dll

```
public interface ISerializer.ISerializeParam
```

# Class JsonSerializer

Namespace: [cfEngine.Serialize](#)

Assembly: CatSweeper.dll

```
public class JsonSerializer : Serializer, ISerializer
```

## Inheritance

[object](#) ← [Serializer](#) ← JsonSerializer

## Implements

[ISerializer](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### OPTIONS

```
public JsonSerializerOptions OPTIONS
```

## Field Value

[JsonSerializerOptions](#)

## Properties

### Instance

```
public static JsonSerializer Instance { get; }
```

## Property Value

## Methods

### Deserialize(byte[], IDeserializeParam)

```
public override object Deserialize(byte[] byteLoaded, ISerializer.IDeserializeParam param = null)
```

#### Parameters

byteLoaded [byte](#)[]

param [ISerializer.IDeserializeParam](#)

#### Returns

[object](#)

### DeserializeAsAsync<T>(byte[], IDeserializeParam, CancellationToken)

```
public override Task<T> DeserializeAsAsync<T>(byte[] byteLoaded, ISerializer.IDeserializeParam deserializeParam = null, CancellationToken token = default)
```

#### Parameters

byteLoaded [byte](#)[]

deserializeParam [ISerializer.IDeserializeParam](#)

token [CancellationToken](#)

#### Returns

[Task](#)<T>

## Type Parameters

T

## DeserializeAs<T>(byte[], IDeserializeParam)

```
public override T DeserializeAs<T>(byte[] byteLoaded, ISerializer.IDeserializeParam  
param = null)
```

## Parameters

byteLoaded [byte](#)[]

param [ISerializer.IDeserializeParam](#)

## Returns

T

## Type Parameters

T

## DeserializeAsync(byte[], IDeserializeParam, CancellationToken)

```
public override Task<object> DeserializeAsync(byte[] byteLoaded,  
ISerializer.IDeserializeParam deserializeParam = null, CancellationToken token  
= default)
```

## Parameters

byteLoaded [byte](#)[]

deserializeParam [ISerializer.IDeserializeParam](#)

token [CancellationToken](#)

Returns

[Task](#)<[object](#)>

## Serialize(object, ISerializeParam)

```
public override string Serialize(object obj, ISerializer.ISerializeParam param  
= null)
```

Parameters

obj [object](#)

param [ISerializer.ISerializeParam](#)

Returns

[string](#)

## SerializeAsync(object, ISerializeParam, CancellationToken)

```
public override Task<string> SerializeAsync(object obj, ISerializer.ISerializeParam  
param = null, CancellationToken token = default)
```

Parameters

obj [object](#)

param [ISerializer.ISerializeParam](#)

token [CancellationToken](#)

Returns

[Task](#)<[string](#)>

# Class Serializer

Namespace: [cfEngine.Serialize](#)

Assembly: CatSweeper.dll

```
public abstract class Serializer : ISerializer
```

## Inheritance

[object](#) ← Serializer

## Implements

[ISerializer](#)

## Derived

[JsonSerializer](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Methods

## Deserialize(byte[], IDeserializeParam)

```
public abstract object Deserialize(byte[] byteLoaded, ISerializer.IDeserializeParam  
param = null)
```

### Parameters

byteLoaded [byte](#)[]

param [ISerializer.IDeserializeParam](#)

### Returns

[object](#)

## DeserializeAsAsync<T>(byte[], IDeserializeParam, CancellationToken)

```
public abstract Task<T> DeserializeAsAsync<T>(byte[] byteLoaded,  
ISerializer.IDeserializeParam param = null, CancellationToken token = default)
```

### Parameters

byteLoaded [byte](#)[]

param [ISerializer.IDeserializeParam](#)

token [CancellationToken](#)

### Returns

[Task](#)<T>

### Type Parameters

T

## DeserializeAs<T>(byte[], IDeserializeParam)

```
public abstract T DeserializeAs<T>(byte[] byteLoaded, ISerializer.IDeserializeParam  
param = null)
```

### Parameters

byteLoaded [byte](#)[]

param [ISerializer.IDeserializeParam](#)

### Returns

T

### Type Parameters

## DeserializeAsync(byte[], IDeserializeParam, CancellationToken)

```
public abstract Task<object> DeserializeAsync(byte[] byteLoaded,  
ISerializer.IDeserializeParam param = null, CancellationToken token = default)
```

### Parameters

byteLoaded [byte](#)[]

param [ISerializer.IDeserializeParam](#)

token [CancellationToken](#)

### Returns

[Task](#)<[object](#)>

## Serialize(object, ISerializeParam)

```
public abstract string Serialize(object obj, ISerializer.ISerializeParam param  
= null)
```

### Parameters

obj [object](#)

param [ISerializer.ISerializeParam](#)

### Returns

[string](#)

## SerializeAsync(object, ISerializeParam, CancellationToken)

```
public abstract Task<string> SerializeAsync(object obj, ISerializer.ISerializeParam  
param = null, CancellationToken token = default)
```

### Parameters

obj [object](#)

param [ISerializer.ISerializeParam](#)

token [CancellationToken](#)

### Returns

[Task](#)<[string](#)>

# Namespace cfEngine.Service

## Classes

[InventoryInfo](#)

[InventoryInfoManager](#)

[ServiceLocator](#)

## Interfaces

[IModelService](#)

[IService](#)

[IServiceLocator](#)

[IServiceModel](#)

# Interface IModelService

Namespace: [cfEngine.Service](#)

Assembly: CatSweeper.dll

```
public interface IModelService : IService, IDisposable
```

## Inherited Members

[IDisposable.Dispose\(\)](#)

## Properties

### GetModel

```
IServiceModel GetModel { get; }
```

Property Value

[IServiceModel](#)

# Interface IService

Namespace: [cfEngine.Service](#)

Assembly: CatSweeper.dll

```
public interface IService : IDisposable
```

## Inherited Members

[IDisposable.Dispose\(\)](#) ↗

# Interface IServiceLocator

Namespace: [cfEngine.Service](#)

Assembly: CatSweeper.dll

```
public interface IServiceLocator : IEnumerable<IService>, IEnumerable, IDisposable
```

## Inherited Members

[IEnumerable<IService>.GetEnumerator\(\)](#) , [IDisposable.Dispose\(\)](#)

## Methods

### GetService<T>()

T `GetService<T>() where T : IService`

Returns

T

Type Parameters

T

### GetService<T>(string)

T `GetService<T>(string serviceName) where T : IService`

Parameters

serviceName [string](#)

Returns

T

## Type Parameters

T

## Register<T>(T, string)

```
void Register<T>(T service, string serviceName) where T : IService
```

### Parameters

service T

serviceName [string](#)

## Type Parameters

T

## Unregister(string)

```
void Unregister(string serviceName)
```

### Parameters

serviceName [string](#)

## Unregister<T>(T)

```
void Unregister<T>(T service) where T : IService
```

### Parameters

service T

## Type Parameters



# Interface IServiceModel

Namespace: [cfEngine.Service](#)

Assembly: CatSweeper.dll

```
public interface IServiceModel : IRuntimeSavable, IDisposable
```

## Inherited Members

[IRuntimeSavable.Initialize\(IUserData\)](#) ,  
[IRuntimeSavable.SetSaveData\(Dictionary<string, object>\)](#) , [IDisposable.Dispose\(\)](#) ↗

# Class InventoryInfo

Namespace: [cfEngine.Service](#)

Assembly: CatSweeper.dll

```
public class InventoryInfo
```

## Inheritance

[object](#) ← InventoryInfo

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### iconKey

```
public string iconKey { get; set; }
```

Property Value

[string](#)

### itemId

```
public string itemId { get; set; }
```

Property Value

[string](#)

### maxStackSize

```
public int maxStackSize { get; set; }
```

Property Value

[int](#) ↗

# Class InventoryInfoManager

Namespace: [cfEngine.Service](#)

Assembly: CatSweeper.dll

```
public class InventoryInfoManager : ConfigInfoManager<string, InventoryInfo>,  
IInfoManager, IDisposable
```

## Inheritance

[object](#) ← [InfoManager](#) ← [ConfigInfoManager<string, InventoryInfo>](#) ←  
InventoryInfoManager

## Implements

[IInfoManager](#), [IDisposable](#)

## Inherited Members

[ConfigInfoManager<string, InventoryInfo>.valueMap](#) ,  
[ConfigInfoManager<string, InventoryInfo>.GetAllValue\(\)](#) ,  
[ConfigInfoManager<string, InventoryInfo>.keyFn](#) ,  
[ConfigInfoManager<string, InventoryInfo>.infoType](#) ,  
[ConfigInfoManager<string, InventoryInfo>.LoadInfo\(\)](#) ,  
[ConfigInfoManager<string, InventoryInfo>.LoadInfoAsync\(CancellationToken\)](#) ,  
[ConfigInfoManager<string, InventoryInfo>.AddValue\(InventoryInfo\)](#) ,  
[ConfigInfoManager<string, InventoryInfo>.TryGetValue\(string, out InventoryInfo\)](#) ,  
[ConfigInfoManager<string, InventoryInfo>.Dispose\(\)](#) , [InfoManager.infoType](#) ,  
[InfoManager.GetAllValue\(\)](#) , [InfoManager.LoadInfo\(\)](#) ,  
[InfoManager.LoadInfoAsync\(CancellationToken\)](#) , [InfoManager.OnLoadCompleted\(\)](#) ,  
[InfoManager.Dispose\(\)](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### InventoryInfoManager(IValueLoader<InventoryInfo>)

```
public InventoryInfoManager(IValueLoader<InventoryInfo> loader)
```

## Parameters

loader [IValueLoader<InventoryInfo>](#)

## Properties

### keyFn

```
protected override Func<InventoryInfo, string> keyFn { get; }
```

## Property Value

[Func<InventoryInfo, string>](#)

## Methods

### GetOrDefault(string)

```
public InventoryInfo GetOrDefault(string itemId)
```

## Parameters

itemId [string](#)

## Returns

[InventoryInfo](#)

# Class ServiceLocator

Namespace: [cfEngine.Service](#)

Assembly: CatSweeper.dll

```
public class ServiceLocator : IServiceLocator, IEnumerable<IService>,
IEnumerable, IDisposable
```

## Inheritance

[object](#) ← ServiceLocator

## Implements

[IServiceLocator](#), [IEnumerable](#)<[IService](#)>, [IEnumerable](#), [IDisposable](#)

## Derived

[Game](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

### GetEnumerator()

Returns an enumerator that iterates through the collection.

```
public IEnumerator<IService> GetEnumerator()
```

Returns

[IEnumerator](#)<[IService](#)>

An enumerator that can be used to iterate through the collection.

## GetService<T>()

```
public T GetService<T>() where T : IService
```

Returns

T

Type Parameters

T

## GetService<T>(string)

```
public T GetService<T>(string serviceName) where T : IService
```

Parameters

serviceName [string](#)

Returns

T

Type Parameters

T

## Register<T>(T, string)

```
public void Register<T>(T service, string serviceName) where T : IService
```

Parameters

service T

serviceName [string](#)

Type Parameters

T

## Unregister(string)

```
public void Unregister(string serviceName)
```

Parameters

serviceName [string](#)

## Unregister<T>(T)

```
public void Unregister<T>(T service) where T : IService
```

Parameters

service T

Type Parameters

T

# Namespace cfEngine.Service.Auth

## Classes

[AuthService](#)

[AuthService.Builder](#)

[LocalAuthService](#)

[LocalPlatform](#)

[LoginToken](#)

[PlatformAuth](#)

## Interfaces

[IAuthService](#)

## Enums

[LoginPlatform](#)

# Class AuthService

Namespace: [cfEngine.Service.Auth](#)

Assembly: CatSweeper.dll

```
public abstract class AuthService : IAuthService, IService, IDisposable
```

## Inheritance

[object](#) ← AuthService

## Implements

[IAuthService](#), [IService](#), [IDisposable](#)

## Derived

[LocalAuthService](#)

## Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#),  
[object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#),  
[object.ToString\(\)](#)

# Properties

## PlatformAuths

```
public IReadOnlyDictionary<LoginPlatform, PlatformAuth> PlatformAuths { get; }
```

### Property Value

[IReadOnlyDictionary](#)<[LoginPlatform](#), [PlatformAuth](#)>

# Methods

## Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public virtual void Dispose()
```

## GetUserId()

```
public abstract string GetUserId()
```

Returns

[string](#)

## InitAsync(CancellationToken)

```
public abstract Task InitAsync(CancellationToken token)
```

Parameters

token  [CancellationToken](#)

Returns

[Task](#)

## IsSessionUserExist()

```
public bool IsSessionUserExist()
```

Returns

[bool](#)

## LinkAsync(LoginPlatform, LoginToken)

```
public Task LinkAsync(LoginPlatform platform, LoginToken token)
```

Parameters

platform [LoginPlatform](#)

token [LoginToken](#)

Returns

[Task](#)

## RegisterPlatform(PlatformAuth)

```
public void RegisterPlatform(PlatformAuth platform)
```

Parameters

platform [PlatformAuth](#)

## SignInAsync(LoginPlatform, LoginToken)

```
public Task SignInAsync(LoginPlatform platform, LoginToken token)
```

Parameters

platform [LoginPlatform](#)

token [LoginToken](#)

Returns

[Task](#)

## SignUpAsync(LoginPlatform, LoginToken)

```
public Task SignUpAsync(LoginPlatform platform, LoginToken token)
```

Parameters

platform [LoginPlatform](#)

token [LoginToken](#)

Returns

[Task](#)

## TryLoginCachedUserAsync(CancellationToken)

```
public abstract Task<bool> TryLoginCachedUserAsync(CancellationToken token)
```

Parameters

token [CancellationToken](#)

Returns

[Task](#)<[bool](#)>

# Class AuthService.Builder

Namespace: [cfEngine.Service.Auth](#)

Assembly: CatSweeper.dll

```
public class AuthService.Builder
```

## Inheritance

[object](#) ← AuthService.Builder

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### Build()

```
public IAuthService Build()
```

Returns

[IAuthService](#)

### RegisterPlatform(PlatformAuth)

```
public AuthService.Builder RegisterPlatform(PlatformAuth platform)
```

Parameters

platform [PlatformAuth](#)

Returns

## SetService(IAuthService)

```
public AuthService.Builder SetService(IAuthService authService)
```

### Parameters

authService [IAuthService](#)

### Returns

[AuthService.Builder](#)

# Interface IAuthService

Namespace: [cfEngine.Service.Auth](#)

Assembly: CatSweeper.dll

```
public interface IAuthService : IService, IDisposable
```

## Inherited Members

[IDisposable.Dispose\(\)](#)

## Methods

### GetUserId()

```
string GetUserId()
```

Returns

[string](#)

### InitAsync(CancellationToken)

```
Task InitAsync(CancellationToken token)
```

Parameters

token [CancellationToken](#)

Returns

[Task](#)

### IsSessionUserExist()

```
bool IsSessionUserExist()
```

Returns

[bool](#)

## LinkAsync(LoginPlatform, LoginToken)

```
Task LinkAsync(LoginPlatform platform, LoginToken token)
```

Parameters

platform [LoginPlatform](#)

token [LoginToken](#)

Returns

[Task](#)

## RegisterPlatform(PlatformAuth)

```
void RegisterPlatform(PlatformAuth platform)
```

Parameters

platform [PlatformAuth](#)

## SignInAsync(LoginPlatform, LoginToken)

```
Task SignInAsync(LoginPlatform platform, LoginToken token)
```

Parameters

platform [LoginPlatform](#)

token [LoginToken](#)

Returns

[Task](#)

## SignUpAsync(LoginPlatform, LoginToken)

Task [SignUpAsync](#)(LoginPlatform platform, LoginToken token)

Parameters

platform [LoginPlatform](#)

token [LoginToken](#)

Returns

[Task](#)

## TryLoginCachedUserAsync(CancellationToken)

Task<bool> [TryLoginCachedUserAsync](#)(CancellationToken token)

Parameters

token [CancellationToken](#)

Returns

[Task](#)<[bool](#)>

# Class LocalAuthService

Namespace: [cfEngine.Service.Auth](#)

Assembly: CatSweeper.dll

```
public class LocalAuthService : AuthService, IAuthService, IService, IDisposable
```

## Inheritance

[object](#) ← [AuthService](#) ← LocalAuthService

## Implements

[IAuthService](#), [IService](#), [IDisposable](#)

## Inherited Members

[AuthService.PlatformAuths](#) , [AuthService.IsSessionUserExist\(\)](#) ,  
[AuthService.RegisterPlatform\(PlatformAuth\)](#) ,  
[AuthService.SignInAsync\(LoginPlatform, LoginToken\)](#) ,  
[AuthService.SignUpAsync\(LoginPlatform, LoginToken\)](#) ,  
[AuthService.LinkAsync\(LoginPlatform, LoginToken\)](#) , [AuthService.Dispose\(\)](#) ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### GetUserId()

```
public override string GetUserId()
```

Returns

[string](#)

### InitAsync(CancellationToken)

```
public override Task InitAsync(CancellationToken token)
```

Parameters

token [CancellationToken](#)

Returns

[Task](#)

## TryLoginCachedUserAsync(CancellationToken)

```
public override Task<bool> TryLoginCachedUserAsync(CancellationToken token)
```

Parameters

token [CancellationToken](#)

Returns

[Task](#)<[bool](#)>

# Class LocalPlatform

Namespace: [cfEngine.Service.Auth](#)

Assembly: CatSweeper.dll

```
public class LocalPlatform : PlatformAuth
```

## Inheritance

[object](#) ← [PlatformAuth](#) ← LocalPlatform

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### Platform

```
public override LoginPlatform Platform { get; }
```

Property Value

[LoginPlatform](#)

## Methods

### LinkAsync(LoginToken)

```
public override Task LinkAsync(LoginToken token)
```

Parameters

token [LoginToken](#)

Returns

[Task](#)

## SignInAsync(LoginToken)

```
public override Task SignInAsync(LoginToken token)
```

Parameters

token [LoginToken](#)

Returns

[Task](#)

## SignUpAsync(LoginToken)

```
public override Task SignUpAsync(LoginToken token)
```

Parameters

token [LoginToken](#)

Returns

[Task](#)

# Enum LoginPlatform

Namespace: [cfEngine.Service.Auth](#)

Assembly: CatSweeper.dll

```
public enum LoginPlatform : byte
```

## Extension Methods

[EnumExtension.hasFlag\(Enum, Enum\)](#)

## Fields

Anonymous = 1

Apple = 4

AppleGame = 5

Facebook = 6

FromCached = 0

Google = 2

GooglePlay = 3

Local = 10

Oculus = 8

Steam = 7

Username = 9

# Class LoginToken

Namespace: [cfEngine.Service.Auth](#)

Assembly: CatSweeper.dll

```
public class LoginToken
```

## Inheritance

[object](#) ← LoginToken

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class PlatformAuth

Namespace: [cfEngine.Service.Auth](#)

Assembly: CatSweeper.dll

```
public abstract class PlatformAuth
```

## Inheritance

[object](#) ← PlatformAuth

## Derived

[LocalPlatform](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Properties

## Platform

```
public abstract LoginPlatform Platform { get; }
```

Property Value

[LoginPlatform](#)

# Methods

## LinkAsync(LoginToken)

```
public abstract Task LinkAsync(LoginToken token)
```

Parameters

`token` [LoginToken](#)

Returns

[Task](#) ↗

## SignInAsync(LoginToken)

```
public abstract Task SignInAsync(LoginToken token)
```

Parameters

`token` [LoginToken](#)

Returns

[Task](#) ↗

## SignUpAsync(LoginToken)

```
public abstract Task SignUpAsync(LoginToken token)
```

Parameters

`token` [LoginToken](#)

Returns

[Task](#) ↗

# Namespace cfEngine.Service.Inventory

## Classes

[InventoryModel](#)

[InventoryService](#)

## Interfaces

[IInventoryService](#)

# Interface IInventoryService

Namespace: [cfEngine.Service.Inventory](#)

Assembly: CatSweeper.dll

```
public interface IInventoryService : IModelService, IService, IDisposable
```

## Inherited Members

[IModelService.GetModel](#) , [IDisposable.Dispose\(\)](#) ↗

# Class InventoryModel

Namespace: [cfEngine.Service.Inventory](#)

Assembly: CatSweeper.dll

```
public class InventoryModel : IServiceModel, IRuntimeSavable, IDisposable
```

## Inheritance

[object](#) ← InventoryModel

## Implements

[IServiceModel](#), [IRuntimeSavable](#), [IDisposable](#)

## Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#),  
[object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#),  
[object.ToString\(\)](#)

## Methods

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

### Initialize(IUserData)

```
public void Initialize(IUserData userData)
```

## Parameters

userData [IUserData](#)

## SetSaveData(Dictionary<string, object>)

```
public void SetSaveData(Dictionary<string, object> dataMap)
```

### Parameters

dataMap [Dictionary](#)<[string](#), [object](#)>

# Class InventoryService

Namespace: [cfEngine.Service.Inventory](#)

Assembly: CatSweeper.dll

```
public class InventoryService : IInventoryService, IModelService,  
IService, IDisposable
```

## Inheritance

[object](#) ← InventoryService

## Implements

[IInventoryService](#), [IModelService](#), [IService](#), [IDisposable](#)

## Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#),  
[object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#),  
[object.ToString\(\)](#)

## Constructors

### InventoryService(InventoryModel)

```
public InventoryService(InventoryModel model)
```

## Parameters

model [InventoryModel](#)

## Methods

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

# Namespace cfEngine.Service.Statistic Classes

[Statistic](#)

[StatisticModel](#)

# Class Statistic

Namespace: [cfEngine.Service.Statistic](#)

Assembly: CatSweeper.dll

```
public class Statistic
```

## Inheritance

[object](#) ← Statistic

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Properties

## Value

```
public double Value { get; }
```

Property Value

[double](#)

# Methods

## RecordOnce()

```
public void RecordOnce()
```

# Events

## OnUpdate

```
public event Action<double> OnUpdate
```

Event Type

[Action](#) <[double](#)>

# Class StatisticModel

Namespace: [cfEngine.Service.Statistic](#)

Assembly: CatSweeper.dll

```
public class StatisticModel : IServiceProvider, IRuntimeSavable, IDisposable
```

## Inheritance

[object](#) ← StatisticModel

## Implements

[IServiceProvider](#), [IRuntimeSavable](#), [IDisposable](#)

## Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#),  
[object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#),  
[object.ToString\(\)](#)

## Properties

### StatisticMap

```
public Dictionary<string, Statistic> StatisticMap { get; }
```

#### Property Value

[Dictionary](#)<[string](#), [Statistic](#)>

## Methods

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

## GetMatchedStatistic(string)

```
public IEnumerable<KeyValuePair<string, Statistic>> GetMatchedStatistic(string  
regex)
```

Parameters

regex [string](#)

Returns

[IEnumerable](#)<[KeyValuePair](#)<[string](#), [Statistic](#)>>

## GetOrCreateStatistic(string)

```
public Statistic GetOrCreateStatistic(string statisticKey)
```

Parameters

statisticKey [string](#)

Returns

[Statistic](#)

## Initialize(IUserData)

```
public void Initialize(IUserData userData)
```

Parameters

userData [IUserData](#)

## SetSaveData(Dictionary<string, object>)

```
public void SetSaveData(Dictionary<string, object> dataMap)
```

Parameters

dataMap [Dictionary](#)<[string](#), [object](#)>

## TryGetStat(string, out Statistic)

```
public bool TryGetStat(string key, out Statistic statistic)
```

Parameters

key [string](#)

statistic [Statistic](#)

Returns

[bool](#)

## Events

### OnNewStatisticRecorded

```
public event Action<string> OnNewStatisticRecorded
```

Event Type

[Action](#)<[string](#)>

# Namespace cfEngine.Util

## Classes

[DirectoryUtil](#)

[JsonElementExtension](#)

[PathSegmentBuilder](#)

[SanityCheck](#)

[SanityCheckException](#)

[StateExecutionException<TStatId>](#)

[StateMachine<TStatId, TState, TStateMachine>](#)

[StateParam](#)

[State<TStatId, TState, TStateMachine>](#)

[TypeExtension](#)

## Structs

[PathSegment](#)

[StateChangeRecord<TStatId>](#)

## Interfaces

[IStateMachine<TStatId>](#)

# Class DirectoryUtil

Namespace: [cfEngine.Util](#)

Assembly: CatSweeper.dll

```
public static class DirectoryUtil
```

## Inheritance

[object](#) ← DirectoryUtil

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### CreateDirectoryIfNotExists(string, string)

```
public static DirectoryInfo CreateDirectoryIfNotExists(string directoryPath,  
string assetFolderPath)
```

#### Parameters

directoryPath [string](#)

assetFolderPath [string](#)

#### Returns

[DirectoryInfo](#)

# Interface IStateMachine<TStatId>

Namespace: [cfEngine.Util](#)

Assembly: CatSweeper.dll

```
public interface IStateMachine<TStateId>
```

## Type Parameters

TStateId

## Properties

### currentStatId

```
TStateId currentStateId { get; }
```

### Property Value

TStatId

### lastStatId

```
TStateId lastStateId { get; }
```

### Property Value

TStatId

## Methods

### CanGoToState(TStatId, StateParam)

```
bool CanGoToState(TStateId id, StateParam param = null)
```

Parameters

**id** TStateId

**param** [StateParam](#)

Returns

[bool](#)

## ForceGoToState(TStateId, StateParam)

```
void ForceGoToState(TStateId nextStateId, StateParam param = null)
```

Parameters

**nextStateId** TStateId

**param** [StateParam](#)

## SubscribeAfterStateChange(Action<StateChangeRecord<TStateId>>)

```
Subscription SubscribeAfterStateChange(Action<StateChangeRecord<TStateId>> listener)
```

Parameters

**listener** [Action](#)<[StateChangeRecord](#)<TStateId>>

Returns

[Subscription](#)

## SubscribeBeforeStateChange(Action<StateChangeRecord<TStatId>>)

Subscription `SubscribeBeforeStateChange(Action<StateChangeRecord<TStatId>> listener)`

### Parameters

`listener Action<StateChangeRecord<TStatId>>`

### Returns

[Subscription](#)

## TryGoToState(TStatId, StateParam)

`bool TryGoToState(TStatId nextStateId, StateParam param = null)`

### Parameters

`nextStateId TStatId`

`param StateParam`

### Returns

[bool](#)

# Class JsonElementExtension

Namespace: [cfEngine.Util](#)

Assembly: CatSweeper.dll

```
public static class JsonElementExtension
```

## Inheritance

[object](#) ← JsonElementExtension

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### ToObject(JsonElement)

```
public static object ToObject(this JsonElement jsonElement)
```

#### Parameters

jsonElement [JsonElement](#)

#### Returns

[object](#)

# Struct PathSegment

Namespace: [cfEngine.Util](#)

Assembly: CatSweeper.dll

```
public struct PathSegment
```

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

## Constructors

### PathSegment(ReadOnlyMemory<string>)

```
public PathSegment(ReadOnlyMemory<string> segments)
```

## Parameters

segments [ReadOnlyMemory](#)<[string](#)>

## Methods

### GetOsPath()

```
public readonly string GetOsPath()
```

## Returns

[string](#)

### GetPath()

```
public readonly string GetPath()
```

Returns

[string](#)

## GetSegments()

```
public ReadOnlyMemory<string> GetSegments()
```

Returns

[ReadOnlyMemory](#) <[string](#)>

## HasValue()

```
public readonly bool HasValue()
```

Returns

[bool](#)

## ToString()

Returns the fully qualified type name of this instance.

```
public override string ToString()
```

Returns

[string](#)

The fully qualified type name.

# Class PathSegmentBuilder

Namespace: [cfEngine.Util](#)

Assembly: CatSweeper.dll

```
public class PathSegmentBuilder : IDisposable
```

## Inheritance

[object](#) ← PathSegmentBuilder

## Implements

[IDisposable](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Constructors

### PathSegmentBuilder()

```
public PathSegmentBuilder()
```

## Methods

### AppendPath(string)

```
public PathSegmentBuilder AppendPath(string path)
```

## Parameters

path [string](#)

## Returns

## [PathSegmentBuilder](#)

### AppendPath(PathSegment)

```
public PathSegmentBuilder AppendPath(PathSegment pathSegment)
```

Parameters

pathSegment [PathSegment](#)

Returns

[PathSegmentBuilder](#)

### Build()

```
public PathSegment Build()
```

Returns

[PathSegment](#)

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

# Class SanityCheck

Namespace: [cfEngine.Util](#)

Assembly: CatSweeper.dll

```
public static class SanityCheck
```

## Inheritance

[object](#) ← SanityCheck

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### WhenNull<T>(T, string)

```
public static bool WhenNull<T>(T target, string message = "") where T : class
```

#### Parameters

target T

message [string](#)

#### Returns

[bool](#)

#### Type Parameters

T

### WhenTrue(bool, string)

```
public static bool WhenTrue(bool condition, string message = "")
```

Parameters

condition [bool](#)

message [string](#)

Returns

[bool](#)

# Class SanityCheckException

Namespace: [cfEngine.Util](#)

Assembly: CatSweeper.dll

```
public class SanityCheckException : Exception, ISerializable
```

## Inheritance

[object](#) ← [Exception](#) ← SanityCheckException

## Implements

[ISerializable](#)

## Inherited Members

[Exception.GetBaseException\(\)](#) , [Exception.GetType\(\)](#) , [Exception.ToString\(\)](#) ,  
[Exception.Data](#) , [Exception.HelpLink](#) , [Exception.HResult](#) , [Exception.InnerException](#) ,  
[Exception.Message](#) , [Exception.Source](#) , [Exception.StackTrace](#) , [Exception.TargetSite](#) ,  
[Exception.SerializeObjectState](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Constructors

### SanityCheckException()

```
public SanityCheckException()
```

### SanityCheckException(string)

```
public SanityCheckException(string message)
```

## Parameters

message [string](#)

## SanityCheckException(string, Exception)

```
public SanityCheckException(string message, Exception innerException)
```

### Parameters

message [string](#)

innerException [Exception](#)

# Struct StateChangeRecord<TStateId>

Namespace: [cfEngine.Util](#)

Assembly: CatSweeper.dll

```
public struct StateChangeRecord<TStateId>
```

## Type Parameters

TStateId

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Fields

### LastState

```
public TStateId LastState
```

### Field Value

TStateId

### NewState

```
public TStateId NewState
```

### Field Value

TStateId

# Class StateExecutionException<TStatId>

Namespace: [cfEngine.Util](#)

Assembly: CatSweeper.dll

```
public class StateExecutionException<TStateId> : Exception, ISerializable
```

## Type Parameters

TStateId

### Inheritance

[object](#) ↳ [Exception](#) ↳ StateExecutionException<TStatId>

### Implements

[ISerializable](#)

### Inherited Members

[Exception.GetBaseException\(\)](#) , [Exception.GetType\(\)](#) , [Exception.ToString\(\)](#) ,  
[Exception.Data](#) , [Exception.HelpLink](#) , [Exception.HResult](#) , [Exception.InnerException](#) ,  
[Exception.Message](#) , [Exception.Source](#) , [Exception.StackTrace](#) , [Exception.TargetSite](#) ,  
[Exception.SerializeObjectState](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Constructors

### StateExecutionException(TStatId, Exception)

```
public StateExecutionException(TStateId stateId, Exception innerException)
```

## Parameters

stateId TStatId

innerException [Exception](#)

# Class StateMachine<TStatId, TState, TStateMachine>

Namespace: [cfEngine.Util](#)

Assembly: CatSweeper.dll

```
public class StateMachine<TStateId, TState, TStateMachine> :  
    IStateMachine<TStateId>, IDisposable where TState : State<TStateId, TState,  
    TStateMachine> where TStateMachine : StateMachine<TStateId, TState, TStateMachine>
```

## Type Parameters

TStateId

TState

TStateMachine

## Inheritance

[object](#) ← StateMachine<TStatId, TState, TStateMachine>

## Implements

[IStateMachine](#)<TStatId>, [IDisposable](#)

## Derived

[GameStateMachine](#)

## Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#),  
[object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#),  
[object.ToString\(\)](#)

## Constructors

StateMachine()

```
public StateMachine()
```

# Properties

## currentStatId

```
public TStateId currentStateId { get; }
```

### Property Value

TStateId

## lastStatId

```
public TStateId lastStateId { get; }
```

### Property Value

TStateId

# Methods

## CanGoToState(TStatId, StateParam)

```
public bool CanGoToState(TStateId id, StateParam param)
```

### Parameters

**id** TStateId

**param** [StateParam](#)

### Returns

[bool](#)

## Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public void Dispose()
```

## ForceGoToState(TStatId, StateParam)

```
public void ForceGoToState(TStateId nextStateId, StateParam param = null)
```

### Parameters

nextStateId TStatId

param [StateParam](#)

## GetStateUnsafe(TStatId)

```
public TState GetStateUnsafe(TStateId id)
```

### Parameters

id TStatId

### Returns

TState

## GetStateUnsafe<T>(TStatId)

```
public T GetStateUnsafe<T>(TStateId id) where T : TState
```

### Parameters

id TStatId

Returns

T

Type Parameters

T

## RegisterState(TState)

```
public void RegisterState(TState state)
```

Parameters

state TState

## SubscribeAfterStateChange(Action<StateChangeRecord<TStatId>>)

```
public Subscription SubscribeAfterStateChange(Action<StateChangeRecord<TStatId>> listener)
```

Parameters

listener [Action](#)<[StateChangeRecord](#)<TStatId>>

Returns

[Subscription](#)

## SubscribeBeforeStateChange(Action<StateChangeRecord<TStatId>>)

```
public Subscription SubscribeBeforeStateChange(Action<StateChangeRecord<TStatId>> listener)
```

Parameters

`listener Action<StateChangeRecord<TStatId>>`

Returns

[Subscription](#)

## TryGetState(TStatId, out TState)

`public bool TryGetState(TStateId id, out TState state)`

Parameters

`id TStatId`

`state TState`

Returns

[bool](#)

## TryGoToState(TStatId, StateParam)

`public bool TryGoToState(TStateId nextStateId, StateParam param = null)`

Parameters

`nextStateId TStatId`

`param StateParam`

Returns

[bool](#)

# Class StateParam

Namespace: [cfEngine.Util](#)

Assembly: CatSweeper.dll

```
public class StateParam
```

## Inheritance

[object](#) ← StateParam

## Derived

[LoadStageState.InitParam](#), [LoginState.Param](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class State<TStatId, TState, TStateMachine>

Namespace: [cfEngine.Util](#)

Assembly: CatSweeper.dll

```
public abstract class State<TStateId, TState, TStateMachine> : IDisposable where
    TState : State<TStateId, TState, TStateMachine> where TStateMachine :
        StateMachine<TStateId, TState, TStateMachine>
```

## Type Parameters

TStateId

TState

TStateMachine

## Inheritance

[object](#) ← State<TStatId, TState, TStateMachine>

## Implements

[IDisposable](#)

## Derived

[GameState](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

Id

```
public abstract TStateId Id { get; }
```

Property Value

TStateId

## StateMachine

```
public TStateMachine StateMachine { get; }
```

Property Value

TStateMachine

## Methods

### Dispose()

Performs application-defined tasks associated with freeing, releasing, or resetting unmanaged resources.

```
public virtual void Dispose()
```

### IsReady()

```
public virtual bool IsReady()
```

Returns

bool ↗

### OnEndContext()

```
protected virtual void OnEndContext()
```

## StartContext(StateParam)

```
public abstract void StartContext(StateParam param)
```

### Parameters

param [StateParam](#)

# Class TypeExtension

Namespace: [cfEngine.Util](#)

Assembly: CatSweeper.dll

```
public static class TypeExtension
```

## Inheritance

[object](#) ← TypeExtension

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### FindDerivedTypes(Assembly, Type)

```
public static IEnumerable<Type> FindDerivedTypes(Assembly assembly, Type baseType)
```

#### Parameters

assembly [Assembly](#)

baseType [Type](#)

#### Returns

[IEnumerable](#)<[Type](#)>

### GetDefaultValue(Type)

```
public static object GetDefaultValue(this Type type)
```

Parameters

type [Type](#)

Returns

[object](#)

## GetFlattenMethods(Type)

```
public static MethodInfo[] GetFlattenMethods(this Type type)
```

Parameters

type [Type](#)

Returns

[MethodInfo](#)[]

## GetTypeNames(Type)

```
public static string GetTypeName(this Type type)
```

Parameters

type [Type](#)

Returns

[string](#)

# Namespace cfGodotEngine.Asset Classes

[AsyncResourceLoader](#)

[AsyncResourceLoader.MethodName](#)

Cached StringNames for the methods contained in this class, for fast lookup.

[AsyncResourceLoader.PropertyName](#)

Cached StringNames for the properties and fields contained in this class, for fast lookup.

[AsyncResourceLoader.SignalName](#)

Cached StringNames for the signals contained in this class, for fast lookup.

[GDAtlas](#)

[GDAtlas.MethodName](#)

Cached StringNames for the methods contained in this class, for fast lookup.

[GDAtlas.PropertyName](#)

Cached StringNames for the properties and fields contained in this class, for fast lookup.

[GDAtlas.SignalName](#)

Cached StringNames for the signals contained in this class, for fast lookup.

[GDAtlasPack](#)

[GDAtlasPack.MethodName](#)

Cached StringNames for the methods contained in this class, for fast lookup.

[GDAtlasPack.PropertyName](#)

Cached StringNames for the properties and fields contained in this class, for fast lookup.

[GDAtlasPack.SignalName](#)

Cached StringNames for the signals contained in this class, for fast lookup.

[GDAtlasTextureRef](#)

[GDAtlasTextureRef.MethodName](#)

Cached StringNames for the methods contained in this class, for fast lookup.

[GDAtlasTextureRef.PropertyName](#)

Cached StringNames for the properties and fields contained in this class, for fast lookup.

[GDAtlasTextureRef.SignalName](#)

Cached StringNames for the signals contained in this class, for fast lookup.

## [ResourceAssetManager](#)

# Class AsyncResourceLoader

Namespace: [cfGodotEngine.Asset](#)

Assembly: CatSweeper.dll

```
[ScriptPath("res://Modules/cfGodotEngine/asset/AsyncResourceLoader.cs")]
public class AsyncResourceLoader : MonoInstance<AsyncResourceLoader>, IDisposable
```

## Inheritance

[object](#) ← GodotObject ← Node ← [MonoInstance<AsyncResourceLoader>](#) ←  
AsyncResourceLoader

## Implements

[IDisposable](#)

## Inherited Members

[MonoInstance<AsyncResourceLoader>.Instance](#) , Node.NotificationEnterTree ,  
Node.NotificationExitTree , Node.NotificationMovedInParent , Node.NotificationReady ,  
Node.NotificationPaused , Node.NotificationUnpaused , Node.NotificationPhysicsProcess ,  
Node.NotificationProcess , Node.NotificationParented , Node.NotificationUnparented ,  
Node.NotificationSceneInstantiated , Node.NotificationDragBegin ,  
Node.NotificationDragEnd , Node.NotificationPathRenamed ,  
Node.NotificationChildOrderChanged , Node.NotificationInternalProcess ,  
Node.NotificationInternalPhysicsProcess , Node.NotificationPostEnterTree ,  
Node.NotificationDisabled , Node.NotificationEnabled ,  
Node.NotificationResetPhysicsInterpolation , Node.NotificationEditorPreSave ,  
Node.NotificationEditorPostSave , Node.NotificationWMMouseEnter ,  
Node.NotificationWMMouseExit , Node.NotificationWMWindowFocusIn ,  
Node.NotificationWMWindowFocusOut , Node.NotificationWMCloseRequest ,  
Node.NotificationWMGoBackRequest , Node.NotificationWMSizeChanged ,  
Node.NotificationWMDpiChange , Node.NotificationVpMouseEnter ,  
Node.NotificationVpMouseExit , Node.NotificationWMPositionChanged ,  
Node.NotificationOsMemoryWarning , Node.NotificationTranslationChanged ,  
Node.NotificationWMAbout , Node.NotificationCrash , Node.NotificationOslmeUpdate ,  
Node.NotificationApplicationResumed , Node.NotificationApplicationPaused ,  
Node.NotificationApplicationFocusIn , Node.NotificationApplicationFocusOut ,  
Node.NotificationTextServerChanged , Node.GetNode<T>(NodePath) ,  
Node.GetNodeOrNull<T>(NodePath) , [Node.GetChild<T>\(int, bool\)](#) ,  
[Node.GetChildOrNull<T>\(int, bool\)](#) , Node.GetOwner<T>() , Node.GetOwnerOrNull<T>()

Node.GetParent<T>() , Node.GetParentOrNull<T>() , Node.\_EnterTree() , Node.\_ExitTree() ,  
Node.\_GetConfigurationWarnings() , Node.\_Input(InputEvent) ,  
[Node.\\_PhysicsProcess\(double\)](#) , Node.\_Ready() , Node.\_ShortcutInput(InputEvent) ,  
Node.\_UnhandledInput(InputEvent) , Node.\_UnhandledKeyInput(InputEvent) ,  
Node.PrintOrphanNodes() , [Node.AddSibling\(Node, bool\)](#) ,  
[Node.AddChild\(Node, bool, Node.InternalMode\)](#) , Node.RemoveChild(Node) ,  
[Node.Reparent\(Node, bool\)](#) , [Node.GetChildCount\(bool\)](#) , [Node.GetChildren\(bool\)](#) ,  
[Node.GetChild\(int, bool\)](#) , Node.HasNode(NodePath) , Node.GetNode(NodePath) ,  
Node.GetNodeOrNull(NodePath) , Node.GetParent() , [Node.FindChild\(string, bool, bool\)](#) ,  
[Node.FindChildren\(string, string, bool, bool\)](#) , [Node.FindParent\(string\)](#) ,  
Node.HasNodeAndResource(NodePath) , Node.GetNodeAndResource(NodePath) ,  
Node.IsInsideTree() , Node.IsPartOfEditedScene() , Node.IsAncestorOf(Node) ,  
Node.IsGreater Than(Node) , Node.GetPath() , [Node.GetPathTo\(Node, bool\)](#) ,  
[Node.AddToGroup\(StringName, bool\)](#) , Node.RemoveFromGroup(StringName) ,  
Node.IsInGroup(StringName) , [Node.MoveChild\(Node, int\)](#) , Node.GetGroups() ,  
[Node.GetIndex\(bool\)](#) , Node.PrintTree() , Node.PrintTreePretty() , Node.GetTreeString() ,  
Node.GetTreeStringPretty() , [Node.PropagateNotification\(int\)](#) ,  
[Node.PropagateCall\(StringName, Array, bool\)](#) , [Node.SetPhysicsProcess\(bool\)](#) ,  
Node.GetPhysicsProcessDeltaTime() , Node.IsPhysicsProcessing() ,  
Node.GetProcessDeltaTime() , [Node.SetProcess\(bool\)](#) , Node.IsProcessing() ,  
[Node.SetProcessInput\(bool\)](#) , Node.IsProcessingInput() ,  
[Node.SetProcessShortcutInput\(bool\)](#) , Node.IsProcessingShortcutInput() ,  
[Node.SetProcessUnhandledInput\(bool\)](#) , Node.IsProcessingUnhandledInput() ,  
[Node.SetProcessUnhandledKeyInput\(bool\)](#) , Node.IsProcessingUnhandledKeyInput() ,  
Node.CanProcess() , [Node.SetDisplayFolded\(bool\)](#) , Node.IsDisplayedFolded() ,  
[Node.SetProcessInternal\(bool\)](#) , Node.IsProcessingInternal() ,  
[Node.SetPhysicsProcessInternal\(bool\)](#) , Node.IsPhysicsProcessingInternal() ,  
Node.IsPhysicsInterpolated() , Node.IsPhysicsInterpolatedAndEnabled() ,  
Node.ResetPhysicsInterpolation() , Node.SetTranslationDomainInherited() ,  
Node.GetWindow() , Node.GetLastExclusiveWindow() , Node.GetTree() ,  
Node.CreateTween() , [Node.Duplicate\(int\)](#) , [Node.ReplaceBy\(Node, bool\)](#) ,  
[Node.setSceneInstanceLoadPlaceholder\(bool\)](#) ,  
Node.GetSceneInstanceLoadPlaceholder() , [Node.setEditableInstance\(Node, bool\)](#) ,  
Node.setEditableInstance(Node) , Node.GetViewport() , Node.QueueFree() ,  
Node.RequestReady() , Node.IsNodeReady() , [Node.SetMultiplayerAuthority\(int, bool\)](#) ,  
Node.GetMultiplayerAuthority() , Node.IsMultiplayerAuthority() ,  
Node.RpcConfig(StringName, Variant) , Node.GetRpcConfig() ,  
[Node.Atr\(string, StringName\)](#) , [Node.AtrN\(string, StringName, int, StringName\)](#) ,  
Node.Rpc(StringName, params Variant[]) ,  
[Node.Rpc\(StringName, ReadOnlySpan<Variant>\)](#) ,

[Node.RpcId\(long, StringName, params Variant\[\]\)](#) ,  
[Node.RpcId\(long, StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.UpdateConfigurationWarnings() ,  
Node.CallDeferredThreadGroup(StringName, params Variant[]) ,  
[Node.CallDeferredThreadGroup\(StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.SetDeferredThreadGroup(StringName, Variant) ,  
[Node.NotifyDeferredThreadGroup\(int\)](#) ,  
Node.CallThreadSafe(StringName, params Variant[]) ,  
[Node.CallThreadSafe\(StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.SetThreadSafe(StringName, Variant) , [Node.NotifyThreadSafe\(int\)](#) ,  
Node.EmitSignalReady() , Node.EmitSignalRenamed() , Node.EmitSignalTreeEntered() ,  
Node.EmitSignalTreeExiting() , Node.EmitSignalTreeExited() ,  
Node.EmitSignalChildEnteredTree(Node) , Node.EmitSignalChildExitingTree(Node) ,  
Node.EmitSignalChildOrderChanged() , Node.EmitSignalReplacingBy(Node) ,  
Node.EmitSignalEditorDescriptionChanged(Node) , Node.EmitSignalEditorStateChanged() ,  
Node.InvokeGodotClassMethod(in godot\_string\_name, NativeVariantPtrArgs, out godot\_variant) ,  
Node.HasGodotClassMethod(in godot\_string\_name) ,  
Node.HasGodotClassSignal(in godot\_string\_name) , Node.Name ,  
Node.UniqueNameInOwner , Node.SceneFilePath , Node.Owner , Node.Multiplayer ,  
Node.ProcessMode , Node.ProcessPriority , Node.ProcessPhysicsPriority ,  
Node.ProcessThreadGroup , Node.ProcessThreadGroupOrder ,  
Node.ProcessThreadMessages , Node.PhysicsInterpolationMode , Node.AutoTranslateMode ,  
Node.EditorDescription , Node.Ready , Node.Renamed , Node.TreeEntered ,  
Node.TreeExiting , Node.TreeExited , Node.ChildEnteredTree , Node.ChildExitingTree ,  
Node.ChildOrderChanged , Node.ReplacingBy , Node.EditorDescriptionChanged ,  
Node.EditorStateChanged , GodotObject.NotificationPostinitialize ,  
GodotObject.NotificationPredelete , GodotObject.NotificationExtensionReloaded ,  
[GodotObject.InstanceFromId\(ulong\)](#) , [GodotObject.IsInstanceIdValid\(ulong\)](#) ,  
GodotObject.IsInstanceIdValid(GodotObject) , GodotObject.WeakRef(GodotObject) ,  
GodotObject.Dispose() , [GodotObject.Dispose\(bool\)](#) , GodotObject.ToString() ,  
GodotObject.ToSignal(GodotObject, StringName) , GodotObject.\_Get(StringName) ,  
GodotObject.\_GetPropertyList() , GodotObject.\_IterGet(Variant) ,  
GodotObject.\_IterInit(Array) , GodotObject.\_IterNext(Array) ,  
[GodotObject.\\_Notification\(int\)](#) , GodotObject.\_PropertyCanRevert(StringName) ,  
GodotObject.\_PropertyGetRevert(StringName) , GodotObject.\_Set(StringName, Variant) ,  
GodotObject.\_ValidateProperty(Dictionary) , GodotObject.Free() , GodotObject.GetClass() ,  
[GodotObject.IsClass\(string\)](#) , GodotObject.Set(StringName, Variant) ,  
GodotObject.Get(StringName) , GodotObject.SetIndexed(NodePath, Variant) ,  
GodotObject.GetIndexed(NodePath) , GodotObject.GetPropertyList() ,

GodotObject.GetMethodList() , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , [GodotObject.Notification\(int, bool\)](#) ,  
GodotObject.GetInstanceID() , GodotObject.SetScript(Variant) , GodotObject.GetScript() ,  
GodotObject.SetMeta(StringName, Variant) , GodotObject.RemoveMeta(StringName) ,  
GodotObject.GetMeta(StringName, Variant) , GodotObject.HasMeta(StringName) ,  
GodotObject.GetMetaList() , [GodotObject.AddUserSignal\(string, Array\)](#) ,  
GodotObject.HasUserSignal(StringName) , GodotObject.RemoveUserSignal(StringName) ,  
GodotObject.EmitSignal(StringName, params Variant[]) ,  
[GodotObject.EmitSignal\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.Call(StringName, params Variant[]) ,  
[GodotObject.Call\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.CallDeferred(StringName, params Variant[]) ,  
[GodotObject.CallDeferred\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.SetDeferred(StringName, Variant) , GodotObject.Callv(StringName, Array) ,  
GodotObject.HasMethod(StringName) ,  
GodotObjectGetMethodArgumentCount(StringName) ,  
GodotObject.HasSignal(StringName) , GodotObject.GetSignalList() ,  
GodotObject.GetSignalConnectionList(StringName) ,  
GodotObject.GetIncomingConnections() ,  
[GodotObject.Connect\(StringName, Callable, uint\)](#) ,  
GodotObject.Disconnect(StringName, Callable) ,  
GodotObject.IsConnected(StringName, Callable) ,  
GodotObject.HasConnections(StringName) , [GodotObject.SetBlockSignals\(bool\)](#) ,  
GodotObject.IsBlockingSignals() , GodotObject.NotifyPropertyListChanged() ,  
[GodotObject.SetMessageTranslation\(bool\)](#) , GodotObject.CanTranslateMessages() ,  
GodotObject.Tr(StringName, StringName) ,  
[GodotObject.TrN\(StringName, StringName, int, StringName\)](#) ,  
GodotObject.GetTranslationDomain() , GodotObject.SetTranslationDomain(StringName) ,  
GodotObject.IsQueuedForDeletion() , GodotObject.CancelFree() ,  
GodotObject.EmitSignalScriptChanged() , GodotObject.EmitSignalPropertyListChanged() ,  
GodotObject.NativeInstance , GodotObject.ScriptChanged ,  
GodotObject.PropertyListChanged , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Extension Methods

[NodeUtil.DontDestroyOnLoad\(Node\)](#)

# Methods

## Load(string, in IProgress<float>, string, bool, CacheMode)

```
public Task<Resource> Load(string path, in IProgress<float> progress, string typeHint = "", bool useSubThread = false, ResourceLoader.CacheMode cacheMode = CacheMode.Reuse)
```

### Parameters

path [string](#)

progress [IProgress](#)<[float](#)>

typeHint [string](#)

useSubThread [bool](#)

cacheMode [ResourceLoader.CacheMode](#)

### Returns

[Task](#)<Resource>

## LoadAsync(string, in IProgress<float>, string, bool, CacheMode)

```
public static Task<Resource> LoadAsync(string path, in IProgress<float> progress, string typeHint = "", bool useSubThread = false, ResourceLoader.CacheMode cacheMode = CacheMode.Reuse)
```

### Parameters

path [string](#)

progress [IProgress](#)<[float](#)>

typeHint [string](#)

useSubThread [bool](#)

`cacheMode` ResourceLoader.CacheMode

Returns

[Task](#)<Resource>

## SetLogger(ILogger)

```
public void SetLogger(ILogger logger)
```

Parameters

`logger` [ILogger](#)

## \_Process(double)

Called during the processing step of the main loop. Processing happens at every frame and as fast as possible, so the `delta` time since the previous frame is not constant. `delta` is in seconds.

It is only called if processing is enabled, which is done automatically if this method is overridden, and can be toggled with [SetProcess\(bool\)](#).

Processing happens in order of Godot.Node.ProcessPriority, lower priority values are called first. Nodes with the same priority are processed in tree order, or top to bottom as seen in the editor (also known as pre-order traversal).

Corresponds to the Godot.Node.NotificationProcess notification in [\\_Notification\(int\)](#).

**Note:** This method is only called if the node is present in the scene tree (i.e. if it's not an orphan).

**Note:** `delta` will be larger than expected if running at a framerate lower than Godot.Engine.PhysicsTicksPerSecond / Godot.Engine.MaxPhysicsStepsPerFrame FPS. This is done to avoid "spiral of death" scenarios where performance would plummet due to an ever-increasing number of physics steps per frame. This behavior affects both [\\_Process\(double\)](#) and [\\_PhysicsProcess\(double\)](#). As a result, avoid using `delta` for time measurements in real-world seconds. Use the Godot.Time singleton's methods for this purpose instead, such as Godot.Time.GetTicksUseC().

```
public override void _Process(double delta)
```

## Parameters

delta [double](#)

# Class AsyncResourceLoader.MethodName

Namespace: [cfGodotEngine.Asset](#)

Assembly: CatSweeper.dll

Cached StringNames for the methods contained in this class, for fast lookup.

```
public class AsyncResourceLoader.MethodName :  
MonoInstance<AsyncResourceLoader>.MethodName
```

## Inheritance

[object](#) ← GodotObject.MethodName ← Node.MethodName ←  
[MonoInstance<AsyncResourceLoader>.MethodName](#) ← AsyncResourceLoader.MethodName

## Inherited Members

Node.MethodName.\_EnterTree , Node.MethodName.\_ExitTree ,  
Node.MethodName.\_GetConfigurationWarnings , Node.MethodName.\_Input ,  
Node.MethodName.\_PhysicsProcess , Node.MethodName.\_Ready ,  
Node.MethodName.\_ShortcutInput , Node.MethodName.\_UnhandledInput ,  
Node.MethodName.\_UnhandledKeyInput , Node.MethodName.PrintOrphanNodes ,  
Node.MethodName.AddSibling , Node.MethodName.SetName ,  
Node.MethodName.GetName , Node.MethodName.AddChild ,  
Node.MethodName.RemoveChild , Node.MethodName.Reparent ,  
Node.MethodName.GetChildCount , Node.MethodName.GetChildren ,  
Node.MethodName.GetChild , Node.MethodName.HasNode , Node.MethodName.GetNode ,  
Node.MethodName.GetNodeOrNull , Node.MethodName.GetParent ,  
Node.MethodName.FindChild , Node.MethodName.FindChildren ,  
Node.MethodName.FindParent , Node.MethodName.HasNodeAndResource ,  
Node.MethodName.GetNodeAndResource , Node.MethodName.IsInsideTree ,  
Node.MethodName.IsPartOfEditedScene , Node.MethodName.IsAncestorOf ,  
Node.MethodName.IsGreater Than , Node.MethodName.GetPath ,  
Node.MethodName.GetPathTo , Node.MethodName.AddToGroup ,  
Node.MethodName.RemoveFromGroup , Node.MethodName.IsInGroup ,  
Node.MethodName.MoveChild , Node.MethodName.GetGroups ,  
Node.MethodName.SetOwner , Node.MethodName.GetOwner ,  
Node.MethodName.GetIndex , Node.MethodName.PrintTree ,  
Node.MethodName.PrintTreePretty , Node.MethodName.GetTreeString ,  
Node.MethodName.GetTreeStringPretty , Node.MethodName.SetSceneFilePath ,  
Node.MethodName.GetSceneFilePath , Node.MethodName.PropagateNotification ,

Node.MethodName.PropagateCall , Node.MethodName.SetPhysicsProcess ,  
Node.MethodName.GetPhysicsProcessDeltaTime , Node.MethodName.IsPhysicsProcessing ,  
Node.MethodName.GetProcessDeltaTime , Node.MethodName.SetProcess ,  
Node.MethodName.SetProcessPriority , Node.MethodName.GetProcessPriority ,  
Node.MethodName.SetPhysicsProcessPriority ,  
Node.MethodName.GetPhysicsProcessPriority , Node.MethodName.IsProcessing ,  
Node.MethodName.SetProcessInput , Node.MethodName.IsProcessingInput ,  
Node.MethodName.SetProcessShortcutInput ,  
Node.MethodName.IsProcessingShortcutInput ,  
Node.MethodName.SetProcessUnhandledInput ,  
Node.MethodName.IsProcessingUnhandledInput ,  
Node.MethodName.SetProcessUnhandledKeyInput ,  
Node.MethodName.IsProcessingUnhandledKeyInput , Node.MethodName.SetProcessMode ,  
Node.MethodName.GetProcessMode , Node.MethodName.CanProcess ,  
Node.MethodName.SetProcessThreadGroup , Node.MethodName.GetProcessThreadGroup ,  
Node.MethodName.SetProcessThreadMessages ,  
Node.MethodName.GetProcessThreadMessages ,  
Node.MethodName.SetProcessThreadGroupOrder ,  
Node.MethodName.GetProcessThreadGroupOrder , Node.MethodName.SetDisplayFolded ,  
Node.MethodName.IsDisplayedFolded , Node.MethodName.SetProcessInternal ,  
Node.MethodName.IsProcessingInternal , Node.MethodName.SetPhysicsProcessInternal ,  
Node.MethodName.IsPhysicsProcessingInternal ,  
Node.MethodName.SetPhysicsInterpolationMode ,  
Node.MethodName.GetPhysicsInterpolationMode ,  
Node.MethodName.IsPhysicsInterpolated ,  
Node.MethodName.IsPhysicsInterpolatedAndEnabled ,  
Node.MethodName.ResetPhysicsInterpolation , Node.MethodName.SetAutoTranslateMode ,  
Node.MethodName.GetAutoTranslateMode ,  
Node.MethodName.SetTranslationDomainInherited , Node.MethodName.GetWindow ,  
Node.MethodName.GetLastExclusiveWindow , Node.MethodName.GetTree ,  
Node.MethodName.CreateTween , Node.MethodName.Duplicate ,  
Node.MethodName.ReplaceBy , Node.MethodName.SetSceneInstanceLoadPlaceholder ,  
Node.MethodName.GetSceneInstanceLoadPlaceholder ,  
Node.MethodName.SetEditableInstance , Node.MethodName.IsEditableInstance ,  
Node.MethodName.GetViewport , Node.MethodName.QueueFree ,  
Node.MethodName.RequestReady , Node.MethodName.IsNodeReady ,  
Node.MethodName.SetMultiplayerAuthority , Node.MethodName.GetMultiplayerAuthority ,  
Node.MethodName.IsMultiplayerAuthority , Node.MethodName.GetMultiplayer ,  
Node.MethodName.RpcConfig , Node.MethodName.GetRpcConfig ,  
Node.MethodName.SetEditorDescription , Node.MethodName.GetEditorDescription ,

Node.MethodName.\_SetImportPath , Node.MethodName.\_GetImportPath ,  
Node.MethodName.SetUniqueNameInOwner , Node.MethodName.IsUniqueNameInOwner ,  
Node.MethodName.Atr , Node.MethodName.AtrN , Node.MethodName.Rpc ,  
Node.MethodName.RpcId , Node.MethodName.UpdateConfigurationWarnings ,  
Node.MethodName.CallDeferredThreadGroup ,  
Node.MethodName.SetDeferredThreadGroup ,  
Node.MethodName.NotifyDeferredThreadGroup , Node.MethodName.CallThreadSafe ,  
Node.MethodName.SetThreadSafe , Node.MethodName.NotifyThreadSafe ,  
GodotObject.MethodName.\_Get , GodotObject.MethodName.\_GetPropertyList ,  
GodotObject.MethodName.\_IterGet , GodotObject.MethodName.\_IterInit ,  
GodotObject.MethodName.\_IterNext , GodotObject.MethodName.\_Notification ,  
GodotObject.MethodName.\_PropertyCanRevert ,  
GodotObject.MethodName.\_PropertyGetRevert , GodotObject.MethodName.\_Set ,  
GodotObject.MethodName.\_ValidateProperty , GodotObject.MethodName.Free ,  
GodotObject.MethodName.GetClass , GodotObject.MethodName.IsClass ,  
GodotObject.MethodName.Set , GodotObject.MethodName.Get ,  
GodotObject.MethodName.SetIndexed , GodotObject.MethodName.GetIndexed ,  
GodotObject.MethodName.GetPropertyList , GodotObject.MethodName.GetMethodList ,  
GodotObject.MethodName.PropertyCanRevert ,  
GodotObject.MethodName.PropertyGetRevert , GodotObject.MethodName.Notification ,  
GodotObject.MethodName.GetInstanceId , GodotObject.MethodName.SetScript ,  
GodotObject.MethodName.GetScript , GodotObject.MethodName.SetMeta ,  
GodotObject.MethodName.RemoveMeta , GodotObject.MethodName.GetMeta ,  
GodotObject.MethodName.HasMeta , GodotObject.MethodName.GetMetaList ,  
GodotObject.MethodName.AddUserSignal , GodotObject.MethodName.HasUserSignal ,  
GodotObject.MethodName.RemoveUserSignal , GodotObject.MethodName.EmitSignal ,  
GodotObject.MethodName.Call , GodotObject.MethodName.CallDeferred ,  
GodotObject.MethodName.SetDeferred , GodotObject.MethodName.Callv ,  
GodotObject.MethodName.HasMethod ,  
GodotObject.MethodName.GetMethodArgumentCount ,  
GodotObject.MethodName.HasSignal , GodotObject.MethodName.GetSignalList ,  
GodotObject.MethodName.GetSignalConnectionList ,  
GodotObject.MethodName.GetIncomingConnections , GodotObject.MethodName.Connect ,  
GodotObject.MethodName.Disconnect , GodotObject.MethodName.IsConnected ,  
GodotObject.MethodName.HasConnections , GodotObject.MethodName.SetBlockSignals ,  
GodotObject.MethodName.IsBlockingSignals ,  
GodotObject.MethodName.NotifyPropertyChanged ,  
GodotObject.MethodName.SetMessageTranslation ,  
GodotObject.MethodName.CanTranslateMessages , GodotObject.MethodName.Tr ,  
GodotObject.MethodName.TrN , GodotObject.MethodName.GetTranslationDomain ,

GodotObject.MethodName.SetTranslationDomain ,  
GodotObject.MethodName.IsQueuedForDeletion , GodotObject.MethodName.CancelFree ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### \_Process

Cached name for the '\_Process' method.

```
public static readonly StringName _Process
```

### Field Value

StringName

# Class AsyncResourceLoader.PropertyName

Namespace: [cfGodotEngine.Asset](#)

Assembly: CatSweeper.dll

Cached StringNames for the properties and fields contained in this class, for fast lookup.

```
public class AsyncResourceLoader.PropertyName :  
MonoInstance<AsyncResourceLoader>.PropertyName
```

## Inheritance

```
object ← GodotObject.PropertyName ← Node.PropertyName ←  
MonoInstance<AsyncResourceLoader>.PropertyName ←  
AsyncResourceLoader.PropertyName
```

## Inherited Members

```
Node.PropertyName._ImportPath , Node.PropertyName.Name ,  
Node.PropertyName.UniqueNameInOwner , Node.PropertyName.SceneFilePath ,  
Node.PropertyName.Owner , Node.PropertyName.Multiplayer ,  
Node.PropertyName.ProcessMode , Node.PropertyName.ProcessPriority ,  
Node.PropertyName.ProcessPhysicsPriority , Node.PropertyName.ProcessThreadGroup ,  
Node.PropertyName.ProcessThreadGroupOrder ,  
Node.PropertyName.ProcessThreadMessages ,  
Node.PropertyName.PhysicsInterpolationMode , Node.PropertyName.AutoTranslateMode ,  
Node.PropertyName.EditorDescription , object.Equals\(object\) ,  
object.Equals\(object, object\) , object.GetHashCode\(\) , object.GetType\(\) ,  
object.MemberwiseClone\(\) , object.ReferenceEquals\(object, object\) , object.ToString\(\)
```

## Fields

### progressArray

Cached name for the 'progressArray' field.

```
public static readonly StringName progressArray
```

## Field Value

StringName

# Class AsyncResourceLoader.SignalName

Namespace: [cfGodotEngine.Asset](#)

Assembly: CatSweeper.dll

Cached StringNames for the signals contained in this class, for fast lookup.

```
public class AsyncResourceLoader.SignalName :  
MonoInstance<AsyncResourceLoader>.SignalName
```

## Inheritance

[object](#) ← GodotObject.SignalName ← Node.SignalName ←  
[MonoInstance<AsyncResourceLoader>.SignalName](#) ← AsyncResourceLoader.SignalName

## Inherited Members

Node.SignalName.Ready , Node.SignalName.Renamed , Node.SignalName.TreeEntered ,  
Node.SignalName.TreeExiting , Node.SignalName.TreeExited ,  
Node.SignalName.ChildEnteredTree , Node.SignalName.ChildExitingTree ,  
Node.SignalName.ChildOrderChanged , Node.SignalName.ReplacingBy ,  
Node.SignalName.EditorDescriptionChanged , Node.SignalName.EditorStateChanged ,  
GodotObject.SignalName.ScriptChanged , GodotObject.SignalName.PropertyListChanged ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class GDAtlas

Namespace: [cfGodotEngine.Asset](#)

Assembly: CatSweeper.dll

```
[Tool]
[GlobalClass]
[ScriptPath("res://Modules/cfGodotEngine/atlaspack/GDAtlas.cs")]
public class GDAtlas : Resource, IDisposable
```

## Inheritance

[object](#) ← GodotObject ← RefCounted ← Resource ← GDAtlas

## Implements

[IDisposable](#)

## Inherited Members

Resource.\_GetRid() , Resource.\_ResetState() , [Resource.\\_SetPathCache\(string\)](#) ,  
Resource.\_SetupLocalToScene() , [Resource.TakeOverPath\(string\)](#) ,  
[Resource.SetPathCache\(string\)](#) , Resource.GetRid() , Resource.GetLocalScene() ,  
Resource.SetupLocalToScene() , Resource.ResetState() ,  
[Resource.SetIdForPath\(string, string\)](#) , [Resource.GetIdForPath\(string\)](#) ,  
Resource.IsBuiltIn() , Resource.GenerateSceneUniqueId() , Resource.EmitChanged() ,  
[Resource.Duplicate\(bool\)](#) , Resource.EmitSignalChanged() ,  
Resource.EmitSignalSetupLocalToSceneRequested() ,  
Resource.InvokeGodotClassMethod(in godot\_string\_name, NativeVariantPtrArgs, out godot\_variant) ,  
Resource.HasGodotClassMethod(in godot\_string\_name) ,  
Resource.HasGodotClassSignal(in godot\_string\_name) , Resource.ResourceLocalToScene ,  
Resource.ResourcePath , Resource.ResourceName , Resource.ResourceSceneUniqueId ,  
Resource.Changed , Resource.SetupLocalToSceneRequested , RefCounted.InitRef() ,  
RefCounted.Reference() , RefCounted.Unreference() , RefCounted.GetReferenceCount() ,  
GodotObject.NotificationPostinitialize , GodotObject.NotificationPredelete ,  
GodotObject.NotificationExtensionReloaded , [GodotObject.InstanceFromId\(ulong\)](#) ,  
[GodotObject.IsInstanceIdValid\(ulong\)](#) , GodotObject.IsInstanceIdValid(GodotObject) ,  
GodotObject.WeakRef(GodotObject) , GodotObject.Dispose() , [GodotObject.Dispose\(bool\)](#) ,  
GodotObject.ToString() , GodotObject.ToSignal(GodotObject, StringName) ,  
GodotObject.\_Get(StringName) , GodotObject.\_GetPropertyList() ,  
GodotObject.\_IterGet(Variant) , GodotObject.\_IterInit(Array) , GodotObject.\_IterNext(Array) ,

[GodotObject.Notification\(int\)](#) , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , GodotObject.Set(StringName, Variant) ,  
GodotObject.ValidateProperty(Dictionary) , GodotObject.Free() , GodotObject.GetClass() ,  
[GodotObject.IsClass\(string\)](#) , GodotObject.Set(StringName, Variant) ,  
GodotObject.Get(StringName) , GodotObject.SetIndexed(NodePath, Variant) ,  
GodotObject.GetIndexed(NodePath) , GodotObject.GetPropertyList() ,  
GodotObject.GetMethodList() , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , [GodotObject.Notification\(int, bool\)](#) ,  
GodotObject.GetInstanceId() , GodotObject.SetScript(Variant) , GodotObject.GetScript() ,  
GodotObject.SetMeta(StringName, Variant) , GodotObject.RemoveMeta(StringName) ,  
GodotObject.GetMeta(StringName, Variant) , GodotObject.HasMeta(StringName) ,  
GodotObject.GetMetaList() , [GodotObject.AddUserSignal\(string, Array\)](#) ,  
GodotObject.HasUserSignal(StringName) , GodotObject.RemoveUserSignal(StringName) ,  
GodotObject.EmitSignal(StringName, params Variant[]) ,  
[GodotObject.EmitSignal\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.Call(StringName, params Variant[]) ,  
[GodotObject.Call\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.CallDeferred(StringName, params Variant[]) ,  
[GodotObject.CallDeferred\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.SetDeferred(StringName, Variant) , GodotObject.Callv(StringName, Array) ,  
GodotObject.HasMethod(StringName) ,  
GodotObjectGetMethodArgumentCount(StringName) ,  
GodotObject.HasSignal(StringName) , GodotObject.GetSignalList() ,  
GodotObject.GetSignalConnectionList(StringName) ,  
GodotObject.GetIncomingConnections() ,  
[GodotObject.Connect\(StringName, Callable, uint\)](#) ,  
GodotObject.Disconnect(StringName, Callable) ,  
GodotObject.IsConnected(StringName, Callable) ,  
GodotObject.HasConnections(StringName) , [GodotObject.SetBlockSignals\(bool\)](#) ,  
GodotObject.IsBlockingSignals() , GodotObject.NotifyPropertyListChanged() ,  
[GodotObject.SetMessageTranslation\(bool\)](#) , GodotObject.CanTranslateMessages() ,  
GodotObject.Tr(StringName, StringName) ,  
[GodotObject.TrN\(StringName, StringName, int, StringName\)](#) ,  
GodotObject.GetTranslationDomain() , GodotObject.SetTranslationDomain(StringName) ,  
GodotObject.IsQueuedForDeletion() , GodotObject.CancelFree() ,  
GodotObject.EmitSignalScriptChanged() , GodotObject.EmitSignalPropertyListChanged() ,  
GodotObject.NativeInstance , GodotObject.ScriptChanged ,  
GodotObject.PropertyListChanged , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

# Fields

## atlasId

```
[Export(PropertyHint.None, "")]  
public string atlasId
```

### Field Value

[string](#) ↗

## atlasTexture

```
[Export(PropertyHint.None, "")]  
public Texture2D atlasTexture
```

### Field Value

Texture2D

## dimension

```
[Export(PropertyHint.None, "")]  
public Vector2 dimension
```

### Field Value

Vector2

## imageMap

```
[Export(PropertyHint.None, "")]  
public Dictionary<string, AtlasTexture> imageMap
```

### Field Value

Dictionary<[string](#), [AtlasTexture](#)>

# Class GDAtlas.MethodName

Namespace: [cfGodotEngine.Asset](#)

Assembly: CatSweeper.dll

Cached StringNames for the methods contained in this class, for fast lookup.

```
public class GDAtlas.MethodName : Resource.MethodName
```

## Inheritance

[object](#) ← GodotObject.MethodName ← RefCounted.MethodName ←

Resource.MethodName ← GDAtlas.MethodName

## Inherited Members

Resource.MethodName.\_GetRid , Resource.MethodName.\_ResetState ,  
Resource.MethodName.\_SetPathCache , Resource.MethodName.\_SetupLocalToScene ,  
Resource.MethodName.SetPath , Resource.MethodName.TakeOverPath ,  
Resource.MethodName.GetPath , Resource.MethodName.SetPathCache ,  
Resource.MethodNameSetName , Resource.MethodName.GetName ,  
Resource.MethodName.GetRid , Resource.MethodName.SetLocalToScene ,  
Resource.MethodName.IsLocalToScene , Resource.MethodName.GetLocalScene ,  
Resource.MethodName.SetupLocalToScene , Resource.MethodName.ResetState ,  
Resource.MethodName.SetIdForPath , Resource.MethodName.GetIdForPath ,  
Resource.MethodName.IsBuiltIn , Resource.MethodName.GenerateSceneUniqueId ,  
Resource.MethodName.SetSceneUniqueId , Resource.MethodName.GetSceneUniqueId ,  
Resource.MethodName.EmitChanged , Resource.MethodName.Duplicate ,  
RefCounted.MethodName.InitRef , RefCounted.MethodName.Reference ,  
RefCounted.MethodName.Unreference , RefCounted.MethodName.GetReferenceCount ,  
GodotObject.MethodName.\_Get , GodotObject.MethodName.\_GetPropertyList ,  
GodotObject.MethodName.\_IterGet , GodotObject.MethodName.\_IterInit ,  
GodotObject.MethodName.\_IterNext , GodotObject.MethodName.\_Notification ,  
GodotObject.MethodName.\_PropertyCanRevert ,  
GodotObject.MethodName.\_PropertyGetRevert , GodotObject.MethodName.\_Set ,  
GodotObject.MethodName.\_ValidateProperty , GodotObject.MethodName.Free ,  
GodotObject.MethodName.GetClass , GodotObject.MethodName.IsClass ,  
GodotObject.MethodName.Set , GodotObject.MethodName.Get ,  
GodotObject.MethodName.SetIndexed , GodotObject.MethodName.GetIndexed ,  
GodotObject.MethodName.GetPropertyList , GodotObject.MethodName.GetMethodList ,  
GodotObject.MethodName.PropertyCanRevert ,

GodotObject.MethodName.PropertyGetRevert , GodotObject.MethodName.Notification ,  
GodotObject.MethodName.GetInstanceID , GodotObject.MethodName.SetScript ,  
GodotObject.MethodName.GetScript , GodotObject.MethodName.SetMeta ,  
GodotObject.MethodName.RemoveMeta , GodotObject.MethodName.GetMeta ,  
GodotObject.MethodName.HasMeta , GodotObject.MethodName.GetMetaList ,  
GodotObject.MethodName.AddUserSignal , GodotObject.MethodName.HasUserSignal ,  
GodotObject.MethodName.RemoveUserSignal , GodotObject.MethodName.EmitSignal ,  
GodotObject.MethodName.Call , GodotObject.MethodName.CallDeferred ,  
GodotObject.MethodName.SetDeferred , GodotObject.MethodName.Callv ,  
GodotObject.MethodName.HasMethod ,  
GodotObject.MethodName.GetMethodArgumentCount ,  
GodotObject.MethodName.HasSignal , GodotObject.MethodName.GetSignalList ,  
GodotObject.MethodName.GetSignalConnectionList ,  
GodotObject.MethodName.GetIncomingConnections , GodotObject.MethodName.Connect ,  
GodotObject.MethodName.Disconnect , GodotObject.MethodName.IsConnected ,  
GodotObject.MethodName.HasConnections , GodotObject.MethodName.SetBlockSignals ,  
GodotObject.MethodName.IsBlockingSignals ,  
GodotObject.MethodName.NotifyPropertyListChanged ,  
GodotObject.MethodName.SetMessageTranslation ,  
GodotObject.MethodName.CanTranslateMessages , GodotObject.MethodName.Tr ,  
GodotObject.MethodName.TrN , GodotObject.MethodName.GetTranslationDomain ,  
GodotObject.MethodName.SetTranslationDomain ,  
GodotObject.MethodName.IsQueuedForDeletion , GodotObject.MethodName.CancelFree ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class GDAtlas.PropertyName

Namespace: [cfGodotEngine.Asset](#)

Assembly: CatSweeper.dll

Cached StringNames for the properties and fields contained in this class, for fast lookup.

```
public class GDAtlas.PropertyName : Resource.PropertyName
```

## Inheritance

[object](#) ← GodotObject.PropertyName ← RefCounted.PropertyName ← Resource.PropertyName ← GDAtlas.PropertyName

## Inherited Members

Resource.PropertyName.ResourceLocalToScene , Resource.PropertyName.ResourcePath ,  
Resource.PropertyName.ResourceName , Resource.PropertyName.ResourceSceneUniqueId ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### atlasId

Cached name for the 'atlasId' field.

```
public static readonly StringName atlasId
```

### Field Value

StringName

### atlasTexture

Cached name for the 'atlasTexture' field.

```
public static readonly StringName atlasTexture
```

## Field Value

StringName

## dimension

Cached name for the 'dimension' field.

```
public static readonly StringName dimension
```

## Field Value

StringName

## imageMap

Cached name for the 'imageMap' field.

```
public static readonly StringName imageMap
```

## Field Value

StringName

# Class GDAtlas.SignalName

Namespace: [cfGodotEngine.Asset](#)

Assembly: CatSweeper.dll

Cached StringNames for the signals contained in this class, for fast lookup.

```
public class GDAtlas.SignalName : Resource.SignalName
```

## Inheritance

[object](#) ← GodotObject.SignalName ← RefCounted.SignalName ← Resource.SignalName ← GDAtlas.SignalName

## Inherited Members

Resource.SignalName.Changed , Resource.SignalName.SetupLocalToSceneRequested ,  
GodotObject.SignalName.ScriptChanged , GodotObject.SignalName.PropertyListChanged ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class GDAtlasPack

Namespace: [cfGodotEngine.Asset](#)

Assembly: CatSweeper.dll

```
[Tool]
[GlobalClass]
[ScriptPath("res://Modules/cfGodotEngine/atlaspack/GDAtlasPack.cs")]
public class GDAtlasPack : Resource, IDisposable
```

## Inheritance

[object](#) ← GodotObject ← RefCounted ← Resource ← GDAtlasPack

## Implements

[IDisposable](#)

## Inherited Members

Resource.\_GetRid() , Resource.\_ResetState() , [Resource.\\_SetPathCache\(string\)](#) ,  
Resource.\_SetupLocalToScene() , [Resource.TakeOverPath\(string\)](#) ,  
[Resource.SetPathCache\(string\)](#) , Resource.GetRid() , Resource.GetLocalScene() ,  
Resource.SetupLocalToScene() , Resource.ResetState() ,  
[Resource.SetIdForPath\(string, string\)](#) , [Resource.GetIdForPath\(string\)](#) ,  
Resource.IsBuiltIn() , Resource.GenerateSceneUniqueId() , Resource.EmitChanged() ,  
[Resource.Duplicate\(bool\)](#) , Resource.EmitSignalChanged() ,  
Resource.EmitSignalSetupLocalToSceneRequested() ,  
Resource.InvokeGodotClassMethod(in godot\_string\_name, NativeVariantPtrArgs, out godot\_variant) ,  
Resource.HasGodotClassMethod(in godot\_string\_name) ,  
Resource.HasGodotClassSignal(in godot\_string\_name) , Resource.ResourceLocalToScene ,  
Resource.ResourcePath , Resource.ResourceName , Resource.ResourceSceneUniqueId ,  
Resource.Changed , Resource.SetupLocalToSceneRequested , RefCounted.InitRef() ,  
RefCounted.Reference() , RefCounted.Unreference() , RefCounted.GetReferenceCount() ,  
GodotObject.NotificationPostinitialize , GodotObject.NotificationPredelete ,  
GodotObject.NotificationExtensionReloaded , [GodotObject.InstanceFromId\(ulong\)](#) ,  
[GodotObject.IsInstanceIdValid\(ulong\)](#) , GodotObject.IsInstanceIdValid(GodotObject) ,  
GodotObject.WeakRef(GodotObject) , GodotObject.Dispose() , [GodotObject.Dispose\(bool\)](#) ,  
GodotObject.ToString() , GodotObject.ToSignal(GodotObject, StringName) ,  
GodotObject.\_Get(StringName) , GodotObject.\_GetPropertyList() ,  
GodotObject.\_IterGet(Variant) , GodotObject.\_IterInit(Array) , GodotObject.\_IterNext(Array) ,

[GodotObject.Notification\(int\)](#) , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , GodotObject.Set(StringName, Variant) ,  
GodotObject.ValidateProperty(Dictionary) , GodotObject.Free() , GodotObject.GetClass() ,  
[GodotObject.IsClass\(string\)](#) , GodotObject.Set(StringName, Variant) ,  
GodotObject.Get(StringName) , GodotObject.SetIndexed(NodePath, Variant) ,  
GodotObject.GetIndexed(NodePath) , GodotObject.GetPropertyList() ,  
GodotObject.GetMethodList() , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , [GodotObject.Notification\(int, bool\)](#) ,  
GodotObject.GetInstanceId() , GodotObject.SetScript(Variant) , GodotObject.GetScript() ,  
GodotObject.SetMeta(StringName, Variant) , GodotObject.RemoveMeta(StringName) ,  
GodotObject.GetMeta(StringName, Variant) , GodotObject.HasMeta(StringName) ,  
GodotObject.GetMetaList() , [GodotObject.AddUserSignal\(string, Array\)](#) ,  
GodotObject.HasUserSignal(StringName) , GodotObject.RemoveUserSignal(StringName) ,  
GodotObject.EmitSignal(StringName, params Variant[]) ,  
[GodotObject.EmitSignal\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.Call(StringName, params Variant[]) ,  
[GodotObject.Call\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.CallDeferred(StringName, params Variant[]) ,  
[GodotObject.CallDeferred\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.SetDeferred(StringName, Variant) , GodotObject.Callv(StringName, Array) ,  
GodotObject.HasMethod(StringName) ,  
GodotObjectGetMethodArgumentCount(StringName) ,  
GodotObject.HasSignal(StringName) , GodotObject.GetSignalList() ,  
GodotObject.GetSignalConnectionList(StringName) ,  
GodotObject.GetIncomingConnections() ,  
[GodotObject.Connect\(StringName, Callable, uint\)](#) ,  
GodotObject.Disconnect(StringName, Callable) ,  
GodotObject.IsConnected(StringName, Callable) ,  
GodotObject.HasConnections(StringName) , [GodotObject.SetBlockSignals\(bool\)](#) ,  
GodotObject.IsBlockingSignals() , GodotObject.NotifyPropertyListChanged() ,  
[GodotObject.SetMessageTranslation\(bool\)](#) , GodotObject.CanTranslateMessages() ,  
GodotObject.Tr(StringName, StringName) ,  
[GodotObject.TrN\(StringName, StringName, int, StringName\)](#) ,  
GodotObject.GetTranslationDomain() , GodotObject.SetTranslationDomain(StringName) ,  
GodotObject.IsQueuedForDeletion() , GodotObject.CancelFree() ,  
GodotObject.EmitSignalScriptChanged() , GodotObject.EmitSignalPropertyListChanged() ,  
GodotObject.NativeInstance , GodotObject.ScriptChanged ,  
GodotObject.PropertyListChanged , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

# Fields

## atlasList

```
[Export(PropertyHint.None, "")]  
public Array<GDAtlas> atlasList
```

### Field Value

Array<[GDAtlas](#)>

# Methods

## AddPack(string, AtlasContext, Texture2D)

```
public void AddPack(string atlasId, AtlasContext context, Texture2D atlasTexture)
```

### Parameters

atlasId [string](#)

context [AtlasContext](#)

atlasTexture [Texture2D](#)

# Class GDAtlasPack.MethodName

Namespace: [cfGodotEngine.Asset](#)

Assembly: CatSweeper.dll

Cached StringNames for the methods contained in this class, for fast lookup.

```
public class GDAtlasPack.MethodName : Resource.MethodName
```

## Inheritance

[object](#) ← GodotObject.MethodName ← RefCounted.MethodName ← Resource.MethodName ← GDAtlasPack.MethodName

## Inherited Members

Resource.MethodName.\_GetRid , Resource.MethodName.\_ResetState ,  
Resource.MethodName.\_SetPathCache , Resource.MethodName.\_SetupLocalToScene ,  
Resource.MethodName.SetPath , Resource.MethodName.TakeOverPath ,  
Resource.MethodName.GetPath , Resource.MethodName.SetPathCache ,  
Resource.MethodNameSetName , Resource.MethodName.GetName ,  
Resource.MethodName.GetRid , Resource.MethodName.SetLocalToScene ,  
Resource.MethodName.IsLocalToScene , Resource.MethodName.GetLocalScene ,  
Resource.MethodName.SetupLocalToScene , Resource.MethodName.ResetState ,  
Resource.MethodName.SetIdForPath , Resource.MethodName.GetIdForPath ,  
Resource.MethodName.IsBuiltIn , Resource.MethodName.GenerateSceneUniqueId ,  
Resource.MethodName.SetSceneUniqueId , Resource.MethodName.GetSceneUniqueId ,  
Resource.MethodName.EmitChanged , Resource.MethodName.Duplicate ,  
RefCounted.MethodName.InitRef , RefCounted.MethodName.Reference ,  
RefCounted.MethodName.Unreference , RefCounted.MethodName.GetReferenceCount ,  
GodotObject.MethodName.\_Get , GodotObject.MethodName.\_GetPropertyList ,  
GodotObject.MethodName.\_IterGet , GodotObject.MethodName.\_IterInit ,  
GodotObject.MethodName.\_IterNext , GodotObject.MethodName.\_Notification ,  
GodotObject.MethodName.\_PropertyCanRevert ,  
GodotObject.MethodName.\_PropertyGetRevert , GodotObject.MethodName.\_Set ,  
GodotObject.MethodName.\_ValidateProperty , GodotObject.MethodName.Free ,  
GodotObject.MethodName.GetClass , GodotObject.MethodName.IsClass ,  
GodotObject.MethodName.Set , GodotObject.MethodName.Get ,  
GodotObject.MethodName.SetIndexed , GodotObject.MethodName.GetIndexed ,  
GodotObject.MethodName.GetPropertyList , GodotObject.MethodName.GetMethodList ,  
GodotObject.MethodName.PropertyCanRevert ,

GodotObject.MethodName.PropertyGetRevert , GodotObject.MethodName.Notification ,  
GodotObject.MethodName.GetInstanceID , GodotObject.MethodName.SetScript ,  
GodotObject.MethodName.GetScript , GodotObject.MethodName.SetMeta ,  
GodotObject.MethodName.RemoveMeta , GodotObject.MethodName.GetMeta ,  
GodotObject.MethodName.HasMeta , GodotObject.MethodName.GetMetaList ,  
GodotObject.MethodName.AddUserSignal , GodotObject.MethodName.HasUserSignal ,  
GodotObject.MethodName.RemoveUserSignal , GodotObject.MethodName.EmitSignal ,  
GodotObject.MethodName.Call , GodotObject.MethodName.CallDeferred ,  
GodotObject.MethodName.SetDeferred , GodotObject.MethodName.Callv ,  
GodotObject.MethodName.HasMethod ,  
GodotObject.MethodName.GetMethodArgumentCount ,  
GodotObject.MethodName.HasSignal , GodotObject.MethodName.GetSignalList ,  
GodotObject.MethodName.GetSignalConnectionList ,  
GodotObject.MethodName.GetIncomingConnections , GodotObject.MethodName.Connect ,  
GodotObject.MethodName.Disconnect , GodotObject.MethodName.IsConnected ,  
GodotObject.MethodName.HasConnections , GodotObject.MethodName.SetBlockSignals ,  
GodotObject.MethodName.IsBlockingSignals ,  
GodotObject.MethodName.NotifyPropertyListChanged ,  
GodotObject.MethodName.SetMessageTranslation ,  
GodotObject.MethodName.CanTranslateMessages , GodotObject.MethodName.Tr ,  
GodotObject.MethodName.TrN , GodotObject.MethodName.GetTranslationDomain ,  
GodotObject.MethodName.SetTranslationDomain ,  
GodotObject.MethodName.IsQueuedForDeletion , GodotObject.MethodName.CancelFree ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class GDAtlasPack.PropertyName

Namespace: [cfGodotEngine.Asset](#)

Assembly: CatSweeper.dll

Cached StringNames for the properties and fields contained in this class, for fast lookup.

```
public class GDAtlasPack.PropertyName : Resource.PropertyName
```

## Inheritance

[object](#) ← GodotObject.PropertyName ← RefCounted.PropertyName ← Resource.PropertyName ← GDAtlasPack.PropertyName

## Inherited Members

Resource.PropertyName.ResourceLocalToScene , Resource.PropertyName.ResourcePath ,  
Resource.PropertyName.ResourceName , Resource.PropertyName.ResourceSceneUniqueId ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### atlasList

Cached name for the 'atlasList' field.

```
public static readonly StringName atlasList
```

### Field Value

StringName

# Class GDAtlasPack.SignalName

Namespace: [cfGodotEngine.Asset](#)

Assembly: CatSweeper.dll

Cached StringNames for the signals contained in this class, for fast lookup.

```
public class GDAtlasPack.SignalName : Resource.SignalName
```

## Inheritance

[object](#) ← GodotObject.SignalName ← RefCounted.SignalName ← Resource.SignalName ← GDAtlasPack.SignalName

## Inherited Members

Resource.SignalName.Changed , Resource.SignalName.SetupLocalToSceneRequested ,  
GodotObject.SignalName.ScriptChanged , GodotObject.SignalName.PropertyListChanged ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class GDAtlasTextureRef

Namespace: [cfGodotEngine.Asset](#)

Assembly: CatSweeper.dll

```
[Tool]
[GlobalClass]
[ScriptPath("res://Modules/cfGodotEngine/atlaspack/GDAtlasTextureRef.cs")]
public class GDAtlasTextureRef : Resource, IDisposable
```

## Inheritance

[object](#) ← GodotObject ← RefCounted ← Resource ← GDAtlasTextureRef

## Implements

[IDisposable](#)

## Inherited Members

Resource.\_GetRid() , Resource.\_ResetState() , [Resource.\\_SetPathCache\(string\)](#) ,  
Resource.\_SetupLocalToScene() , [Resource.TakeOverPath\(string\)](#) ,  
[Resource.SetPathCache\(string\)](#) , Resource.GetRid() , Resource.GetLocalScene() ,  
Resource.SetupLocalToScene() , Resource.ResetState() ,  
[Resource.SetIdForPath\(string, string\)](#) , [Resource.GetIdForPath\(string\)](#) ,  
Resource.IsBuiltIn() , Resource.GenerateSceneUniqueId() , Resource.EmitChanged() ,  
[Resource.Duplicate\(bool\)](#) , Resource.EmitSignalChanged() ,  
Resource.EmitSignalSetupLocalToSceneRequested() ,  
Resource.InvokeGodotClassMethod(in godot\_string\_name, NativeVariantPtrArgs, out godot\_variant) ,  
Resource.HasGodotClassMethod(in godot\_string\_name) ,  
Resource.HasGodotClassSignal(in godot\_string\_name) , Resource.ResourceLocalToScene ,  
Resource.ResourcePath , Resource.ResourceName , Resource.ResourceSceneUniqueId ,  
Resource.Changed , Resource.SetupLocalToSceneRequested , RefCounted.InitRef() ,  
RefCounted.Reference() , RefCounted.Unreference() , RefCounted.GetReferenceCount() ,  
GodotObject.NotificationPostinitialize , GodotObject.NotificationPredelete ,  
GodotObject.NotificationExtensionReloaded , [GodotObject.InstanceFromId\(ulong\)](#) ,  
[GodotObject.IsInstanceIdValid\(ulong\)](#) , GodotObject.IsInstanceIdValid(GodotObject) ,  
GodotObject.WeakRef(GodotObject) , GodotObject.Dispose() , [GodotObject.Dispose\(bool\)](#) ,  
GodotObject.ToString() , GodotObject.ToSignal(GodotObject, StringName) ,  
GodotObject.\_Get(StringName) , GodotObject.\_GetPropertyList() ,  
GodotObject.\_IterGet(Variant) , GodotObject.\_IterInit(Array) , GodotObject.\_IterNext(Array) ,

[GodotObject.Notification\(int\)](#) , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , GodotObject.Set(StringName, Variant) ,  
GodotObject.ValidateProperty(Dictionary) , GodotObject.Free() , GodotObject.GetClass() ,  
[GodotObject.IsClass\(string\)](#) , GodotObject.Set(StringName, Variant) ,  
GodotObject.Get(StringName) , GodotObject.SetIndexed(NodePath, Variant) ,  
GodotObject.GetIndexed(NodePath) , GodotObject.GetPropertyList() ,  
GodotObject.GetMethodList() , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , [GodotObject.Notification\(int, bool\)](#) ,  
GodotObject.GetInstanceId() , GodotObject.SetScript(Variant) , GodotObject.GetScript() ,  
GodotObject.SetMeta(StringName, Variant) , GodotObject.RemoveMeta(StringName) ,  
GodotObject.GetMeta(StringName, Variant) , GodotObject.HasMeta(StringName) ,  
GodotObject.GetMetaList() , [GodotObject.AddUserSignal\(string, Array\)](#) ,  
GodotObject.HasUserSignal(StringName) , GodotObject.RemoveUserSignal(StringName) ,  
GodotObject.EmitSignal(StringName, params Variant[]) ,  
[GodotObject.EmitSignal\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.Call(StringName, params Variant[]) ,  
[GodotObject.Call\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.CallDeferred(StringName, params Variant[]) ,  
[GodotObject.CallDeferred\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.SetDeferred(StringName, Variant) , GodotObject.Callv(StringName, Array) ,  
GodotObject.HasMethod(StringName) ,  
GodotObjectGetMethodArgumentCount(StringName) ,  
GodotObject.HasSignal(StringName) , GodotObject.GetSignalList() ,  
GodotObject.GetSignalConnectionList(StringName) ,  
GodotObject.GetIncomingConnections() ,  
[GodotObject.Connect\(StringName, Callable, uint\)](#) ,  
GodotObject.Disconnect(StringName, Callable) ,  
GodotObject.IsConnected(StringName, Callable) ,  
GodotObject.HasConnections(StringName) , [GodotObject.SetBlockSignals\(bool\)](#) ,  
GodotObject.IsBlockingSignals() , GodotObject.NotifyPropertyListChanged() ,  
[GodotObject.SetMessageTranslation\(bool\)](#) , GodotObject.CanTranslateMessages() ,  
GodotObject.Tr(StringName, StringName) ,  
[GodotObject.TrN\(StringName, StringName, int, StringName\)](#) ,  
GodotObject.GetTranslationDomain() , GodotObject.SetTranslationDomain(StringName) ,  
GodotObject.IsQueuedForDeletion() , GodotObject.CancelFree() ,  
GodotObject.EmitSignalScriptChanged() , GodotObject.EmitSignalPropertyListChanged() ,  
GodotObject.NativeInstance , GodotObject.ScriptChanged ,  
GodotObject.PropertyListChanged , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

# Properties

## atlasPack

```
[Export(PropertyHint.None, "")]  
public GDAtlasPack atlasPack { get; set; }
```

Property Value

[GDAtlasPack](#)

## imageName

```
[Export(PropertyHint.None, "")]  
public string imageName { get; set; }
```

Property Value

[string](#)

# Events

## OnAtlasTextureUpdated

```
public event Action<AtlasTexture> OnAtlasTextureUpdated
```

Event Type

[Action](#)<AtlasTexture>

# Class GDAtlasTextureRef.MethodName

Namespace: [cfGodotEngine.Asset](#)

Assembly: CatSweeper.dll

Cached StringNames for the methods contained in this class, for fast lookup.

```
public class GDAtlasTextureRef.MethodName : Resource.MethodName
```

## Inheritance

[object](#) ← GodotObject.MethodName ← RefCounted.MethodName ← Resource.MethodName ← GDAtlasTextureRef.MethodName

## Inherited Members

Resource.MethodName.\_GetRid , Resource.MethodName.\_ResetState ,  
Resource.MethodName.\_SetPathCache , Resource.MethodName.\_SetupLocalToScene ,  
Resource.MethodName.SetPath , Resource.MethodName.TakeOverPath ,  
Resource.MethodName.GetPath , Resource.MethodName.SetPathCache ,  
Resource.MethodNameSetName , Resource.MethodName.GetName ,  
Resource.MethodName.GetRid , Resource.MethodName.SetLocalToScene ,  
Resource.MethodName.IsLocalToScene , Resource.MethodName.GetLocalScene ,  
Resource.MethodName.SetupLocalToScene , Resource.MethodName.ResetState ,  
Resource.MethodName.SetIdForPath , Resource.MethodName.GetIdForPath ,  
Resource.MethodName.IsBuiltIn , Resource.MethodName.GenerateSceneUniqueId ,  
Resource.MethodName.SetSceneUniqueId , Resource.MethodName.GetSceneUniqueId ,  
Resource.MethodName.EmitChanged , Resource.MethodName.Duplicate ,  
RefCounted.MethodName.InitRef , RefCounted.MethodName.Reference ,  
RefCounted.MethodName.Unreference , RefCounted.MethodName.GetReferenceCount ,  
GodotObject.MethodName.\_Get , GodotObject.MethodName.\_GetPropertyList ,  
GodotObject.MethodName.\_IterGet , GodotObject.MethodName.\_IterInit ,  
GodotObject.MethodName.\_IterNext , GodotObject.MethodName.\_Notification ,  
GodotObject.MethodName.\_PropertyCanRevert ,  
GodotObject.MethodName.\_PropertyGetRevert , GodotObject.MethodName.\_Set ,  
GodotObject.MethodName.\_ValidateProperty , GodotObject.MethodName.Free ,  
GodotObject.MethodName.GetClass , GodotObject.MethodName.IsClass ,  
GodotObject.MethodName.Set , GodotObject.MethodName.Get ,  
GodotObject.MethodName.SetIndexed , GodotObject.MethodName.GetIndexed ,  
GodotObject.MethodName.GetPropertyList , GodotObject.MethodName.GetMethodList ,  
GodotObject.MethodName.PropertyCanRevert ,

GodotObject.MethodName.PropertyGetRevert , GodotObject.MethodName.Notification ,  
GodotObject.MethodName.GetInstanceID , GodotObject.MethodName.SetScript ,  
GodotObject.MethodName.GetScript , GodotObject.MethodName.SetMeta ,  
GodotObject.MethodName.RemoveMeta , GodotObject.MethodName.GetMeta ,  
GodotObject.MethodName.HasMeta , GodotObject.MethodName.GetMetaList ,  
GodotObject.MethodName.AddUserSignal , GodotObject.MethodName.HasUserSignal ,  
GodotObject.MethodName.RemoveUserSignal , GodotObject.MethodName.EmitSignal ,  
GodotObject.MethodName.Call , GodotObject.MethodName.CallDeferred ,  
GodotObject.MethodName.SetDeferred , GodotObject.MethodName.Callv ,  
GodotObject.MethodName.HasMethod ,  
GodotObject.MethodName.GetMethodArgumentCount ,  
GodotObject.MethodName.HasSignal , GodotObject.MethodName.GetSignalList ,  
GodotObject.MethodName.GetSignalConnectionList ,  
GodotObject.MethodName.GetIncomingConnections , GodotObject.MethodName.Connect ,  
GodotObject.MethodName.Disconnect , GodotObject.MethodName.IsConnected ,  
GodotObject.MethodName.HasConnections , GodotObject.MethodName.SetBlockSignals ,  
GodotObject.MethodName.IsBlockingSignals ,  
GodotObject.MethodName.NotifyPropertyListChanged ,  
GodotObject.MethodName.SetMessageTranslation ,  
GodotObject.MethodName.CanTranslateMessages , GodotObject.MethodName.Tr ,  
GodotObject.MethodName.TrN , GodotObject.MethodName.GetTranslationDomain ,  
GodotObject.MethodName.SetTranslationDomain ,  
GodotObject.MethodName.IsQueuedForDeletion , GodotObject.MethodName.CancelFree ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### OnTextureUpdate

Cached name for the 'OnTextureUpdate' method.

```
public static readonly StringName OnTextureUpdate
```

### Field Value

StringName

# Class GDAtlasTextureRef.PropertyName

Namespace: [cfGodotEngine.Asset](#)

Assembly: CatSweeper.dll

Cached StringNames for the properties and fields contained in this class, for fast lookup.

```
public class GDAtlasTextureRef.PropertyName : Resource.PropertyName
```

## Inheritance

[object](#) ← GodotObject(PropertyName) ← RefCounted(PropertyName) ← Resource(PropertyName) ← GDAtlasTextureRef(PropertyName)

## Inherited Members

Resource.PropertyName.ResourceLocalToScene , Resource.PropertyName.ResourcePath ,  
Resource.PropertyName.ResourceName , Resource.PropertyName.ResourceSceneUniqueId ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### \_atlasPack

Cached name for the '\_atlasPack' field.

```
public static readonly StringName _atlasPack
```

#### Field Value

StringName

### \_imageName

Cached name for the '\_imageName' field.

```
public static readonly StringName _imageName
```

## Field Value

StringName

## atlasPack

Cached name for the 'atlasPack' property.

```
public static readonly StringName atlasPack
```

## Field Value

StringName

## imageName

Cached name for the 'imageName' property.

```
public static readonly StringName imageName
```

## Field Value

StringName

# Class GDAtlasTextureRef.SignalName

Namespace: [cfGodotEngine.Asset](#)

Assembly: CatSweeper.dll

Cached StringNames for the signals contained in this class, for fast lookup.

```
public class GDAtlasTextureRef.SignalName : Resource.SignalName
```

## Inheritance

[object](#) ← GodotObject.SignalName ← RefCounted.SignalName ← Resource.SignalName ← GDAtlasTextureRef.SignalName

## Inherited Members

Resource.SignalName.Changed , Resource.SignalName.SetupLocalToSceneRequested ,  
GodotObject.SignalName.ScriptChanged , GodotObject.SignalName.PropertyListChanged ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class ResourceAssetManager

Namespace: [cfGodotEngine.Asset](#)

Assembly: CatSweeper.dll

```
public class ResourceAssetManager : AssetManager<Resource>, IService, IDisposable
```

## Inheritance

[object](#) ← [AssetManager](#)<Resource> ← ResourceAssetManager

## Implements

[IService](#), [IDisposable](#)

## Inherited Members

[AssetManager<Resource>.Load<T>\(string\)](#) ,  
[AssetManager<Resource>.LoadAsync<T>\(string, CancellationToken\)](#) ,  
[AssetManager<Resource>.TryGetAsset<T>\(string, out T\)](#) ,  
[AssetManager<Resource>.Dispose\(\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Methods

### \_LoadAsync<T>(string, CancellationToken)

```
protected override Task<AssetHandle<T>> _LoadAsync<T>(string path, CancellationToken  
token = default) where T : class, Resource
```

## Parameters

path [string](#)

token [CancellationToken](#)

## Returns

[Task](#)<[AssetHandle](#)<T>>

## Type Parameters

T

## \_Load<T>(string)

```
protected override AssetHandle<T> _Load<T>(string path) where T : class, Resource
```

## Parameters

path [string](#)

## Returns

[AssetHandle](#)<T>

## Type Parameters

T

# Namespace cfGodotEngine.Controls

## Classes

[AtlasTextureButton](#)

[AtlasTextureButton.MethodName](#)

Cached StringNames for the methods contained in this class, for fast lookup.

[AtlasTextureButton.PropertyName](#)

Cached StringNames for the properties and fields contained in this class, for fast lookup.

[AtlasTextureButton.SignalName](#)

Cached StringNames for the signals contained in this class, for fast lookup.

[AtlasTextureRect](#)

[AtlasTextureRect.MethodName](#)

Cached StringNames for the methods contained in this class, for fast lookup.

[AtlasTextureRect.PropertyName](#)

Cached StringNames for the properties and fields contained in this class, for fast lookup.

[AtlasTextureRect.SignalName](#)

Cached StringNames for the signals contained in this class, for fast lookup.

# Class AtlasTextureButton

Namespace: [cfGodotEngine.Controls](#)

Assembly: CatSweeper.dll

```
[Tool]
[GlobalClass]
[ScriptPath("res://Modules/cfGodotEngine/atlaspack/controls/AtlasTextureButton.cs")]
public class AtlasTextureButton : TextureButton, IDisposable
```

## Inheritance

[object](#) ← GodotObject ← Node ← CanvasItem ← Control ← BaseButton ← TextureButton ← AtlasTextureButton

## Implements

[IDisposable](#)

## Inherited Members

TextureButton.InvokeGodotClassMethod(in godot\_string\_name, NativeVariantPtrArgs, out godot\_variant) ,  
TextureButton.HasGodotClassMethod(in godot\_string\_name) ,  
TextureButton.HasGodotClassSignal(in godot\_string\_name) , TextureButton.TextureNormal ,  
TextureButton.TexturePressed , TextureButton.TextureHover ,  
TextureButton.TextureDisabled , TextureButton.TextureFocused ,  
TextureButton.TextureClickMask , TextureButton.IgnoreTextureSize ,  
TextureButton.StretchMode , TextureButton.FlipH , TextureButton.FlipV ,  
BaseButton.\_Pressed() , [BaseButton.\\_Toggled\(bool\)](#) ,  
[BaseButton.SetPressedNoSignal\(bool\)](#) , BaseButton.IsHovered() ,  
BaseButton.GetDrawMode() , BaseButton.EmitSignalPressed() ,  
BaseButton.EmitSignalButtonUp() , BaseButton.EmitSignalButtonDown() ,  
[BaseButton.EmitSignalToggled\(bool\)](#) , BaseButton.Disabled , BaseButton.ToggleMode ,  
BaseButton.ButtonPressed , BaseButton.ActionMode , BaseButton.ButtonMask ,  
BaseButton.KeepPressedOutside , BaseButton.ButtonGroup , BaseButton.Shortcut ,  
BaseButton.ShortcutFeedback , BaseButton.ShortcutInTooltip , BaseButton.Pressed ,  
BaseButton.ButtonUp , BaseButton.ButtonDown , BaseButton.Toggled ,  
Control.NotificationResized , Control.NotificationMouseEnter ,  
Control.NotificationMouseExit , Control.NotificationMouseEnterSelf ,  
Control.NotificationMouseExitSelf , Control.NotificationFocusEnter ,  
Control.NotificationFocusExit , Control.NotificationThemeChanged ,

Control.NotificationScrollBegin , Control.NotificationScrollEnd ,  
Control.NotificationLayoutDirectionChanged , Control.\_CanDropData(Vector2, Variant) ,  
Control.\_DropData(Vector2, Variant) , Control.\_GetDragData(Vector2) ,  
Control.\_GetMinimumSize() , Control.\_GetTooltip(Vector2) , Control.\_GuIInput(InputEvent) ,  
Control.\_HasPoint(Vector2) , [Control.\\_MakeCustomTooltip\(string\)](#) ,  
[Control.\\_StructuredTextParser\(Array, string\)](#) , Control.AcceptEvent() ,  
Control.GetMinimumSize() , Control.GetCombinedMinimumSize() ,  
[Control.SetAnchorsPreset\(Control.LayoutPreset, bool\)](#) ,  
[Control.SetOffsetsPreset\(Control.LayoutPreset, Control.LayoutPresetMode, int\)](#) ,  
[Control.SetAnchorsAndOffsetsPreset\(Control.LayoutPreset, Control.LayoutPresetMode, int\)](#) ,  
,

[Control.SetAnchor\(Side, float, bool, bool\)](#) ,  
[Control.SetAnchorAndOffset\(Side, float, float, bool\)](#) , Control.SetBegin(Vector2) ,  
Control.SetEnd(Vector2) , [Control.SetPosition\(Vector2, bool\)](#) ,  
[Control.SetSize\(Vector2, bool\)](#) , Control.ResetSize() ,  
[Control.SetGlobalPosition\(Vector2, bool\)](#) , Control.GetBegin() , Control.GetEnd() ,  
Control.GetParentAreaSize() , Control.GetScreenPosition() , Control.GetRect() ,  
Control.GetGlobalRect() , Control.HasFocus() , Control.GrabFocus() ,  
Control.ReleaseFocus() , Control.FindPrevValidFocus() , Control.FindNextValidFocus() ,  
Control.FindValidFocusNeighbor(Side) , Control.BeginBulkThemeOverride() ,  
Control.EndBulkThemeOverride() ,  
Control.AddThemelconOverride(StringName, Texture2D) ,  
Control.AddThemeStyleboxOverride(StringName, StyleBox) ,  
Control.AddThemeFontOverride(StringName, Font) ,  
[Control.AddThemeFontSizeOverride\(StringName, int\)](#) ,  
Control.AddThemeColorOverride(StringName, Color) ,  
[Control.AddThemeConstantOverride\(StringName, int\)](#) ,  
Control.RemoveThemelconOverride(StringName) ,  
Control.RemoveThemeStyleboxOverride(StringName) ,  
Control.RemoveThemeFontOverride(StringName) ,  
Control.RemoveThemeFontSizeOverride(StringName) ,  
Control.RemoveThemeColorOverride(StringName) ,  
Control.RemoveThemeConstantOverride(StringName) ,  
Control.GetThemelcon(StringName, StringName) ,  
Control.GetThemeStylebox(StringName, StringName) ,  
Control.GetThemeFont(StringName, StringName) ,  
Control.GetThemeFontSize(StringName, StringName) ,  
Control.GetThemeColor(StringName, StringName) ,  
Control.GetThemeConstant(StringName, StringName) ,  
Control.HasThemelconOverride(StringName) ,

Control.HasThemeStyleboxOverride(StringName) ,  
Control.HasThemeFontOverride(StringName) ,  
Control.HasThemeFontSizeOverride(StringName) ,  
Control.HasThemeColorOverride(StringName) ,  
Control.HasThemeConstantOverride(StringName) ,  
Control.HasThemelcon(StringName, StringName) ,  
Control.HasThemeStylebox(StringName, StringName) ,  
Control.HasThemeFont(StringName, StringName) ,  
Control.HasThemeFontSize(StringName, StringName) ,  
Control.HasThemeColor(StringName, StringName) ,  
Control.HasThemeConstant(StringName, StringName) ,  
Control.GetThemeDefaultBaseScale() , Control.GetThemeDefaultFont() ,  
Control.GetThemeDefaultFontSize() , Control.GetParentControl() ,  
Control.GetTooltip(Vector2?) , Control.GetCursorShape(Vector2?) ,  
Control.ForceDrag(Variant, Control) , Control.GrabClickFocus() ,  
Control.SetDragForwarding(Callable, Callable, Callable) , Control.SetDragPreview(Control) ,  
Control.IsDragSuccessful() , Control.WarpMouse(Vector2) , Control.UpdateMinimumSize() ,  
Control.IsLayoutRtl() , Control.EmitSignalResized() ,  
Control.EmitSignalGuilinput(InputEvent) , Control.EmitSignalMouseEntered() ,  
Control.EmitSignalMouseExited() , Control.EmitSignalFocusEntered() ,  
Control.EmitSignalFocusExited() , Control.EmitSignalSizeFlagsChanged() ,  
Control.EmitSignalMinimumSizeChanged() , Control.EmitSignalThemeChanged() ,  
Control.ClipContents , Control.CustomMinimumSize , Control.LayoutDirection ,  
Control.AnchorLeft , Control.AnchorTop , Control.AnchorRight , Control.AnchorBottom ,  
Control.OffsetLeft , Control.OffsetTop , Control.OffsetRight , Control.OffsetBottom ,  
Control.GrowHorizontal , Control.GrowVertical , Control.Size , Control.Position ,  
Control.GlobalPosition , Control.Rotation , Control.RotationDegrees , Control.Scale ,  
Control.PivotOffset , Control.SizeFlagsHorizontal , Control.SizeFlagsVertical ,  
Control.SizeFlagsStretchRatio , Control.LocalizeNumeralSystem , Control.AutoTranslate ,  
Control.TooltipText , Control.TooltipAutoTranslateMode , Control.FocusNeighborLeft ,  
Control.FocusNeighborTop , Control.FocusNeighborRight , Control.FocusNeighborBottom ,  
Control.FocusNext , Control.FocusPrevious , Control.FocusMode , Control.MouseFilter ,  
Control.MouseForcePassScrollEvents , Control.MouseDefaultCursorShape ,  
Control.ShortcutContext , Control.Theme , Control.ThemeTypeVariation , Control.Resized ,  
Control.Guilinput , Control.MouseEntered , Control.MouseExited , Control.FocusEntered ,  
Control.FocusExited , Control.SizeFlagsChanged , Control.MinimumSizeChanged ,  
Control.ThemeChanged , CanvasItem.NotificationTransformChanged ,  
CanvasItem.NotificationLocalTransformChanged , CanvasItem.NotificationDraw ,  
CanvasItem.NotificationVisibilityChanged , CanvasItem.NotificationEnterCanvas ,  
CanvasItem.NotificationExitCanvas , CanvasItem.NotificationWorld2DChanged ,

CanvasItem.\_Draw() , CanvasItem.GetCanvasItem() , CanvasItem.IsVisibleInTree() ,  
CanvasItem.Show() , CanvasItem.Hide() , CanvasItem.QueueRedraw() ,  
CanvasItem.MoveToFront() , [CanvasItem.DrawLine\(Vector2, Vector2, Color, float, bool\)](#) ,  
[CanvasItem.DrawDashedLine\(Vector2, Vector2, Color, float, float, bool, bool\)](#) ,  
[CanvasItem.DrawPolyline\(Vector2\[\], Color, float, bool\)](#) ,  
[CanvasItem.DrawPolyline\(ReadOnlySpan<Vector2>, Color, float, bool\)](#) ,  
[CanvasItem.DrawPolylineColors\(Vector2\[\], Color\[\], float, bool\)](#) ,  
[CanvasItem.DrawPolylineColors\(ReadOnlySpan<Vector2>, ReadOnlySpan<Color>, float, bool\)](#) ,  
[CanvasItem.DrawArc\(Vector2, float, float, float, int, Color, float, bool\)](#) ,  
[CanvasItem.DrawMultiline\(Vector2\[\], Color, float, bool\)](#) ,  
[CanvasItem.DrawMultiline\(ReadOnlySpan<Vector2>, Color, float, bool\)](#) ,  
[CanvasItem.DrawMultilineColors\(Vector2\[\], Color\[\], float, bool\)](#) ,  
[CanvasItem.DrawMultilineColors\(ReadOnlySpan<Vector2>, ReadOnlySpan<Color>, float, bool\)](#) ,  
[CanvasItem.DrawRect\(Rect2, Color, bool, float, bool\)](#) ,  
[CanvasItem.DrawCircle\(Vector2, float, Color, bool, float, bool\)](#) ,  
CanvasItem.DrawTexture(Texture2D, Vector2, Color?) ,  
[CanvasItem.DrawTextureRect\(Texture2D, Rect2, bool, Color?, bool\)](#) ,  
[CanvasItem.DrawTextureRectRegion\(Texture2D, Rect2, Rect2, Color?, bool, bool\)](#) ,  
[CanvasItem.DrawMsdfTextureRectRegion\(Texture2D, Rect2, Rect2, Color?, double, double, double\)](#) ,  
CanvasItem.DrawLcdTextureRectRegion(Texture2D, Rect2, Rect2, Color?) ,  
CanvasItem.DrawStyleBox(StyleBox, Rect2) ,  
CanvasItem.DrawPrimitive(Vector2[], Color[], Vector2[], Texture2D) ,  
[CanvasItem.DrawPrimitive\(ReadOnlySpan<Vector2>, ReadOnlySpan<Color>, ReadOnlySpan<Vector2>, Texture2D\)](#) ,  
CanvasItem.DrawPolygon(Vector2[], Color[], Vector2[], Texture2D) ,  
[CanvasItem.DrawPolygon\(ReadOnlySpan<Vector2>, ReadOnlySpan<Color>, ReadOnlySpan<Vector2>, Texture2D\)](#) ,  
CanvasItem.DrawColoredPolygon(Vector2[], Color, Vector2[], Texture2D) ,  
[CanvasItem.DrawColoredPolygon\(ReadOnlySpan<Vector2>, Color, ReadOnlySpan<Vector2>, Texture2D\)](#) ,  
[CanvasItem.DrawString\(Font, Vector2, string, HorizontalAlignment, float, int, Color?, TextServer.JustificationFlag, TextServer.Direction, TextServer.Orientation\)](#) ,  
[CanvasItem.DrawMultilineString\(Font, Vector2, string, HorizontalAlignment, float, int, int, Color?, TextServer.LineBreakFlag, TextServer.JustificationFlag, TextServer.Direction, TextServer.Orientation\)](#) ,  
[CanvasItem.DrawStringOutline\(Font, Vector2, string, HorizontalAlignment, float, int, int, Color?, TextServer.JustificationFlag, TextServer.Direction, TextServer.Orientation\)](#) ,

[CanvasItem.DrawMultilineStringOutline\(Font, Vector2, string, HorizontalAlignment, float, int, int, int, Color?, TextServer.LineBreakFlag, TextServer.JustificationFlag, TextServer.Direction, TextServer.Orientation\)](#) ,  
[CanvasItem.DrawChar\(Font, Vector2, string, int, Color?\)](#) ,  
[CanvasItem.DrawCharOutline\(Font, Vector2, string, int, int, Color?\)](#) ,  
CanvasItem.DrawMesh(Mesh, Texture2D, Transform2D?, Color?) ,  
CanvasItem.DrawMultimesh(MultiMesh, Texture2D) ,  
[CanvasItem.DrawSetTransform\(Vector2, float, Vector2?\)](#) ,  
CanvasItem.DrawSetTransformMatrix(Transform2D) ,  
[CanvasItem.DrawAnimationSlice\(double, double, double, double\)](#) ,  
CanvasItem.DrawEndAnimation() , CanvasItem.GetTransform() ,  
CanvasItem.GetGlobalTransform() , CanvasItem.GetGlobalTransformWithCanvas() ,  
CanvasItem.GetViewportTransform() , CanvasItem.GetViewportRect() ,  
CanvasItem.GetCanvasTransform() , CanvasItem.GetScreenTransform() ,  
CanvasItem.GetLocalMousePosition() , CanvasItem.GetGlobalMousePosition() ,  
CanvasItem.GetCanvas() , CanvasItem.GetCanvasLayerNode() , CanvasItem.GetWorld2D() ,  
CanvasItem.SetInstanceShaderParameter(StringName, Variant) ,  
CanvasItem.GetInstanceShaderParameter(StringName) ,  
[CanvasItem.SetNotifyLocalTransform\(bool\)](#) ,  
CanvasItem.IsLocalTransformNotificationEnabled() ,  
[CanvasItem.SetNotifyTransform\(bool\)](#) , CanvasItem.IsTransformNotificationEnabled() ,  
CanvasItem.ForceUpdateTransform() , CanvasItem.MakeCanvasPositionLocal(Vector2) ,  
CanvasItem.MakeInputLocal(InputEvent) , [CanvasItem.SetVisibilityLayerBit\(uint, bool\)](#) ,  
[CanvasItem.GetVisibilityLayerBit\(uint\)](#) , CanvasItem.EmitSignalDraw() ,  
CanvasItem.EmitSignalVisibilityChanged() , CanvasItem.EmitSignalHidden() ,  
CanvasItem.EmitSignalItemRectChanged() , CanvasItem.Visible , CanvasItem.Modulate ,  
CanvasItem.SelfModulate , CanvasItem.ShowBehindParent , CanvasItem.TopLevel ,  
CanvasItem.ClipChildren , CanvasItem.LightMask , CanvasItem.VisibilityLayer ,  
CanvasItem.ZIndex , CanvasItem.ZAsRelative , CanvasItem.YSortEnabled ,  
CanvasItem.TextureFilter , CanvasItem.TextureRepeat , CanvasItem.Material ,  
CanvasItem.UseParentMaterial , CanvasItem.Draw , CanvasItem.VisibilityChanged ,  
CanvasItem.Hidden , CanvasItem.ItemRectChanged , Node.NotificationEnterTree ,  
Node.NotificationExitTree , Node.NotificationMovedInParent , Node.NotificationReady ,  
Node.NotificationPaused , Node.NotificationUnpaused , Node.NotificationPhysicsProcess ,  
Node.NotificationProcess , Node.NotificationParented , Node.NotificationUnparented ,  
Node.NotificationSceneInstantiated , Node.NotificationDragBegin ,  
Node.NotificationDragEnd , Node.NotificationPathRenamed ,  
Node.NotificationChildOrderChanged , Node.NotificationInternalProcess ,  
Node.NotificationInternalPhysicsProcess , Node.NotificationPostEnterTree ,  
Node.NotificationDisabled , Node.NotificationEnabled ,

Node.NotificationResetPhysicsInterpolation , Node.NotificationEditorPreSave ,  
Node.NotificationEditorPostSave , Node.NotificationWMMouseEnter ,  
Node.NotificationWMMouseExit , Node.NotificationWMWindowFocusIn ,  
Node.NotificationWMWindowFocusOut , Node.NotificationWMCloseRequest ,  
Node.NotificationWMGoBackRequest , Node.NotificationWMSizeChanged ,  
Node.NotificationWMDpiChange , Node.NotificationVpMouseEnter ,  
Node.NotificationVpMouseExit , Node.NotificationWMPositionChanged ,  
Node.NotificationOsMemoryWarning , Node.NotificationTranslationChanged ,  
Node.NotificationWMAbout , Node.NotificationCrash , Node.NotificationOsImeUpdate ,  
Node.NotificationApplicationResumed , Node.NotificationApplicationPaused ,  
Node.NotificationApplicationFocusIn , Node.NotificationApplicationFocusOut ,  
Node.NotificationTextServerChanged , Node.GetNode<T>(NodePath) ,  
Node.GetNodeOrNull<T>(NodePath) , [Node.GetChild<T>\(int, bool\)](#) ,  
[Node.GetChildOrNull<T>\(int, bool\)](#) , Node.GetOwner<T>() , Node.GetOwnerOrNull<T>() ,  
Node.GetParent<T>() , Node.GetParentOrNull<T>() , Node.\_EnterTree() , Node.\_ExitTree() ,  
Node.\_GetConfigurationWarnings() , Node.\_Input(InputEvent) ,  
[Node.\\_PhysicsProcess\(double\)](#) , [Node.\\_Process\(double\)](#) , Node.\_Ready() ,  
Node.\_ShortcutInput(InputEvent) , Node.\_UnhandledInput(InputEvent) ,  
Node.\_UnhandledKeyInput(InputEvent) , Node.PrintOrphanNodes() ,  
[Node.AddSibling\(Node, bool\)](#) , [Node.AddChild\(Node, bool, Node.InternalMode\)](#) ,  
Node.RemoveChild(Node) , [Node.Reparent\(Node, bool\)](#) , [Node.GetChildCount\(bool\)](#) ,  
[Node.GetChildren\(bool\)](#) , [Node.GetChild\(int, bool\)](#) , Node.HasNode(NodePath) ,  
Node.GetNode(NodePath) , Node.GetNodeOrNull(NodePath) , Node.GetParent() ,  
[Node.FindChild\(string, bool, bool\)](#) , [Node.FindChildren\(string, string, bool, bool\)](#) ,  
[Node.FindParent\(string\)](#) , Node.HasNodeAndResource(NodePath) ,  
Node.GetNodeAndResource(NodePath) , Node.IsInsideTree() , Node.IsPartOfEditedScene() ,  
Node.IsAncestorOf(Node) , Node.IsGreaterThanOrEqual(Node) , Node.GetPath() ,  
[Node.GetPathTo\(Node, bool\)](#) , [Node.AddToGroup\(StringName, bool\)](#) ,  
Node.RemoveFromGroup(StringName) , Node.IsInGroup(StringName) ,  
[Node.MoveChild\(Node, int\)](#) , Node.GetGroups() , [Node.GetIndex\(bool\)](#) , Node.PrintTree() ,  
Node.PrintTreePretty() , Node.GetTreeString() , Node.GetTreeStringPretty() ,  
[Node.PropagateNotification\(int\)](#) , [Node.PropagateCall\(StringName, Array, bool\)](#) ,  
[Node.SetPhysicsProcess\(bool\)](#) , Node.GetPhysicsProcessDeltaTime() ,  
Node.IsPhysicsProcessing() , Node.GetProcessDeltaTime() , [Node.SetProcess\(bool\)](#) ,  
Node.IsProcessing() , [Node.SetProcessInput\(bool\)](#) , Node.IsProcessingInput() ,  
[Node.SetProcessShortcutInput\(bool\)](#) , Node.IsProcessingShortcutInput() ,  
[Node.SetProcessUnhandledInput\(bool\)](#) , Node.IsProcessingUnhandledInput() ,  
[Node.SetProcessUnhandledKeyInput\(bool\)](#) , Node.IsProcessingUnhandledKeyInput() ,  
Node.CanProcess() , [Node.SetDisplayFolded\(bool\)](#) , Node.IsDisplayedFolded() ,  
[Node.SetProcessInternal\(bool\)](#) , Node.IsProcessingInternal() ,

[Node.SetPhysicsProcessInternal\(bool\)](#) , Node.IsPhysicsProcessingInternal() ,  
Node.IsPhysicsInterpolated() , Node.IsPhysicsInterpolatedAndEnabled() ,  
Node.ResetPhysicsInterpolation() , Node.SetTranslationDomainInherited() ,  
Node.GetWindow() , Node.GetLastExclusiveWindow() , Node.GetTree() ,  
Node.CreateTween() , [Node.Duplicate\(int\)](#) , [Node.ReplaceBy\(Node, bool\)](#) ,  
[Node.setSceneInstanceLoadPlaceholder\(bool\)](#) ,  
Node.setSceneInstanceLoadPlaceholder() , [Node.setEditableInstance\(Node, bool\)](#) ,  
Node.setEditableInstance(Node) , Node.setViewport() , Node.QueueFree() ,  
Node.RequestReady() , Node.isNodeReady() , [Node.setMultiplayerAuthority\(int, bool\)](#) ,  
Node.GetMultiplayerAuthority() , Node.IsMultiplayerAuthority() ,  
Node.RpcConfig(StringName, Variant) , Node.GetRpcConfig() ,  
[Node.Atr\(string, StringName\)](#) , [Node.AtrN\(string, StringName, int, StringName\)](#) ,  
Node.Rpc(StringName, params Variant[]) ,  
[Node.Rpc\(StringName, ReadOnlySpan<Variant>\)](#) ,  
[Node.RpcId\(long, StringName, params Variant\[\]\)](#) ,  
[Node.RpcId\(long, StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.UpdateConfigurationWarnings() ,  
Node.CallDeferredThreadGroup(StringName, params Variant[]) ,  
[Node.CallDeferredThreadGroup\(StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.SetDeferredThreadGroup(StringName, Variant) ,  
[Node.NotifyDeferredThreadGroup\(int\)](#) ,  
Node.CallThreadSafe(StringName, params Variant[]) ,  
[Node.CallThreadSafe\(StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.SetThreadSafe(StringName, Variant) , [Node.NotifyThreadSafe\(int\)](#) ,  
Node.EmitSignalReady() , Node.EmitSignalRenamed() , Node.EmitSignalTreeEntered() ,  
Node.EmitSignalTreeExiting() , Node.EmitSignalTreeExited() ,  
Node.EmitSignalChildEnteredTree(Node) , Node.EmitSignalChildExitingTree(Node) ,  
Node.EmitSignalChildOrderChanged() , Node.EmitSignalReplacingBy(Node) ,  
Node.EmitSignalEditorDescriptionChanged(Node) , Node.EmitSignalEditorStateChanged() ,  
Node.Name , Node.UniqueNameInOwner , Node.SceneFilePath , Node.Owner ,  
Node.Multiplayer , Node.ProcessMode , Node.ProcessPriority , Node.ProcessPhysicsPriority ,  
Node.ProcessThreadGroup , Node.ProcessThreadGroupOrder ,  
Node.ProcessThreadMessages , Node.PhysicsInterpolationMode , Node.AutoTranslateMode ,  
Node.EditorDescription , Node.Ready , Node.Renamed , Node.TreeEntered ,  
Node.TreeExiting , Node.TreeExited , Node.ChildEnteredTree , Node.ChildExitingTree ,  
Node.ChildOrderChanged , Node.ReplacingBy , Node.EditorDescriptionChanged ,  
Node.EditorStateChanged , GodotObject.NotificationPostinitialize ,  
GodotObject.NotificationPredelete , GodotObject.NotificationExtensionReloaded ,  
[GodotObject.InstanceFromId\(ulong\)](#) , [GodotObject.IsInstanceIdValid\(ulong\)](#) ,  
GodotObject.IsInstanceIdValid(GodotObject) , GodotObject.WeakRef(GodotObject) ,

GodotObject.Dispose() , [GodotObject.Dispose\(bool\)](#) , GodotObject.ToString() ,  
GodotObject.ToSignal(GodotObject, StringName) , GodotObject.\_Get(StringName) ,  
GodotObject.\_GetPropertyList() , GodotObject.\_IterGet(Variant) ,  
GodotObject.\_IterInit(Array) , GodotObject.\_IterNext(Array) ,  
[GodotObject.Notification\(int\)](#) , GodotObject.\_PropertyCanRevert(StringName) ,  
GodotObject.\_PropertyGetRevert(StringName) , GodotObject.\_Set(StringName, Variant) ,  
GodotObject.\_ValidateProperty(Dictionary) , GodotObject.Free() , GodotObject.GetClass() ,  
[GodotObject.IsClass\(string\)](#) , GodotObject.Set(StringName, Variant) ,  
GodotObject.Get(StringName) , GodotObject.SetIndexed(NodePath, Variant) ,  
GodotObject.GetIndexed(NodePath) , GodotObject.GetPropertyList() ,  
GodotObject.GetMethodList() , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , [GodotObject.Notification\(int, bool\)](#) ,  
GodotObject.GetInstanceId() , GodotObject.SetScript(Variant) , GodotObject.GetScript() ,  
GodotObject.SetMeta(StringName, Variant) , GodotObject.RemoveMeta(StringName) ,  
GodotObject.GetMeta(StringName, Variant) , GodotObject.HasMeta(StringName) ,  
GodotObject.GetMetaList() , [GodotObject.AddUserSignal\(string, Array\)](#) ,  
GodotObject.HasUserSignal(StringName) , GodotObject.RemoveUserSignal(StringName) ,  
GodotObject.EmitSignal(StringName, params Variant[]) ,  
[GodotObject.EmitSignal\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.Call(StringName, params Variant[]) ,  
[GodotObject.Call\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.CallDeferred(StringName, params Variant[]) ,  
[GodotObject.CallDeferred\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.SetDeferred(StringName, Variant) , GodotObject.Callv(StringName, Array) ,  
GodotObject.HasMethod(StringName) ,  
GodotObject.GetMethodArgumentCount(StringName) ,  
GodotObject.HasSignal(StringName) , GodotObject.GetSignalList() ,  
GodotObject.GetSignalConnectionList(StringName) ,  
GodotObject.GetIncomingConnections() ,  
[GodotObject.Connect\(StringName, Callable, uint\)](#) ,  
GodotObject.Disconnect(StringName, Callable) ,  
GodotObject.IsConnected(StringName, Callable) ,  
GodotObject.HasConnections(StringName) , [GodotObject.SetBlockSignals\(bool\)](#) ,  
GodotObject.IsBlockingSignals() , GodotObject.NotifyPropertyListChanged() ,  
[GodotObject.SetMessageTranslation\(bool\)](#) , GodotObject.CanTranslateMessages() ,  
GodotObject.Tr(StringName, StringName) ,  
[GodotObject.TrN\(StringName, StringName, int, StringName\)](#) ,  
GodotObject.GetTranslationDomain() , GodotObject.SetTranslationDomain(StringName) ,  
GodotObject.IsQueuedForDeletion() , GodotObject.CancelFree() ,  
GodotObject.EmitSignalScriptChanged() , GodotObject.EmitSignalPropertyListChanged() ,

GodotObject.NativeInstance , GodotObject.ScriptChanged ,  
GodotObject.PropertyListChanged , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Extension Methods

[NodeUtil.DontDestroyOnLoad\(Node\)](#)

# Class AtlasTextureButton.MethodName

Namespace: [cfGodotEngine.Controls](#)

Assembly: CatSweeper.dll

Cached StringNames for the methods contained in this class, for fast lookup.

```
public class AtlasTextureButton.MethodName : TextureButton.MethodName
```

## Inheritance

[object](#) ← GodotObject.MethodName ← Node.MethodName ← CanvasItem.MethodName ← Control.MethodName ← BaseButton.MethodName ← TextureButton.MethodName ← AtlasTextureButton.MethodName

## Inherited Members

TextureButton.MethodName.SetTextureNormal ,  
TextureButton.MethodName.SetTexturePressed ,  
TextureButton.MethodName.SetTextureHover ,  
TextureButton.MethodName.SetTextureDisabled ,  
TextureButton.MethodName.SetTextureFocused ,  
TextureButton.MethodName.SetClickMask ,  
TextureButton.MethodName.SetIgnoreTextureSize ,  
TextureButton.MethodName.SetStretchMode , TextureButton.MethodName.SetFlipH ,  
TextureButton.MethodName.IsFlippedH , TextureButton.MethodName.SetFlipV ,  
TextureButton.MethodName.IsFlippedV , TextureButton.MethodName.GetTextureNormal ,  
TextureButton.MethodName.GetTexturePressed ,  
TextureButton.MethodName.GetTextureHover ,  
TextureButton.MethodName.GetTextureDisabled ,  
TextureButton.MethodName.GetTextureFocused ,  
TextureButton.MethodName.GetClickMask ,  
TextureButton.MethodName.GetIgnoreTextureSize ,  
TextureButton.MethodName.GetStretchMode , BaseButton.MethodName.\_Pressed ,  
BaseButton.MethodName.\_Toggled , BaseButton.MethodName.SetPressed ,  
BaseButton.MethodName.IsPressed , BaseButton.MethodName.SetPressedNoSignal ,  
BaseButton.MethodName.IsHovered , BaseButton.MethodName.SetToggleMode ,  
BaseButton.MethodName.IsToggleMode , BaseButton.MethodName.SetShortcutInTooltip ,  
BaseButton.MethodName.IsShortcutInTooltipEnabled ,  
BaseButton.MethodName.SetDisabled , BaseButton.MethodName.IsEnabled ,  
BaseButton.MethodName.SetActionMode , BaseButton.MethodName.GetActionMode ,

BaseButton.MethodName.SetButtonMask , BaseButton.MethodName.GetButtonMask ,  
BaseButton.MethodName.GetDrawMode ,  
BaseButton.MethodName.SetKeepPressedOutside ,  
BaseButton.MethodName.IsKeepPressedOutside ,  
BaseButton.MethodName.SetShortcutFeedback ,  
BaseButton.MethodName.IsShortcutFeedback , BaseButton.MethodName.SetShortcut ,  
BaseButton.MethodName.GetShortcut , BaseButton.MethodName.SetButtonGroup ,  
BaseButton.MethodName.GetButtonGroup , Control.MethodName.\_CanDropData ,  
Control.MethodName.\_DropData , Control.MethodName.\_GetDragData ,  
Control.MethodName.\_GetMinimumSize , Control.MethodName.\_GetTooltip ,  
Control.MethodName.\_GuiInput , Control.MethodName.\_HasPoint ,  
Control.MethodName.\_MakeCustomTooltip , Control.MethodName.\_StructuredTextParser ,  
Control.MethodName.AcceptEvent , Control.MethodName.GetMinimumSize ,  
Control.MethodName.GetCombinedMinimumSize , Control.MethodName.\_SetLayoutMode ,  
Control.MethodName.\_GetLayoutMode , Control.MethodName.\_SetAnchorsLayoutPreset ,  
Control.MethodName.\_GetAnchorsLayoutPreset , Control.MethodName.SetAnchorsPreset ,  
Control.MethodName.SetOffsetsPreset , Control.MethodName.SetAnchorsAndOffsetsPreset ,  
Control.MethodName.\_SetAnchor , Control.MethodName.SetAnchor ,  
Control.MethodName.GetAnchor , Control.MethodName.SetOffset ,  
Control.MethodName.GetOffset , Control.MethodName.SetAnchorAndOffset ,  
Control.MethodName.SetBegin , Control.MethodName.SetEnd ,  
Control.MethodName.SetPosition , Control.MethodName.\_SetPosition ,  
Control.MethodName.GetSize , Control.MethodName.ResetSize ,  
Control.MethodName.\_GetSize , Control.MethodName.SetCustomMinimumSize ,  
Control.MethodName.SetGlobalPosition , Control.MethodName.\_SetGlobalPosition ,  
Control.MethodName.SetRotation , Control.MethodName.SetRotationDegrees ,  
Control.MethodName.setScale , Control.MethodName.SetPivotOffset ,  
Control.MethodName.GetBegin , Control.MethodName.GetEnd ,  
Control.MethodNameGetPosition , Control.MethodName.GetSize ,  
Control.MethodName.GetRotation , Control.MethodName.GetRotationDegrees ,  
Control.MethodName.GetScale , Control.MethodName.GetPivotOffset ,  
Control.MethodName.GetCustomMinimumSize , Control.MethodName.GetParentAreaSize ,  
Control.MethodName.GetGlobalPosition , Control.MethodName.GetScreenPosition ,  
Control.MethodName.GetRect , Control.MethodName.GetGlobalRect ,  
Control.MethodName.SetFocusMode , Control.MethodName.GetFocusMode ,  
Control.MethodName.HasFocus , Control.MethodName.GrabFocus ,  
Control.MethodName.ReleaseFocus , Control.MethodName.FindPrevValidFocus ,  
Control.MethodName.FindNextValidFocus , Control.MethodName.FindValidFocusNeighbor ,  
Control.MethodName.SetHSizeFlags , Control.MethodName.GetHSizeFlags ,  
Control.MethodName.SetStretchRatio , Control.MethodName.GetStretchRatio ,

Control.MethodName.SetVSizeFlags , Control.MethodName.GetVSizeFlags ,  
Control.MethodName.SetTheme , Control.MethodName.GetTheme ,  
Control.MethodName.SetThemeTypeVariation ,  
Control.MethodName.GetThemeTypeVariation ,  
Control.MethodName.BeginBulkThemeOverride ,  
Control.MethodName.EndBulkThemeOverride ,  
Control.MethodName.AddThemelconOverride ,  
Control.MethodName.AddThemeStyleboxOverride ,  
Control.MethodName.AddThemeFontOverride ,  
Control.MethodName.AddThemeFontSizeOverride ,  
Control.MethodName.AddThemeColorOverride ,  
Control.MethodName.AddThemeConstantOverride ,  
Control.MethodName.RemoveThemelconOverride ,  
Control.MethodName.RemoveThemeStyleboxOverride ,  
Control.MethodName.RemoveThemeFontOverride ,  
Control.MethodName.RemoveThemeFontSizeOverride ,  
Control.MethodName.RemoveThemeColorOverride ,  
Control.MethodName.RemoveThemeConstantOverride ,  
Control.MethodName.GetThemelcon , Control.MethodName.GetThemeStylebox ,  
Control.MethodName.GetThemeFont , Control.MethodName.GetThemeFontSize ,  
Control.MethodName.GetThemeColor , Control.MethodName.GetThemeConstant ,  
Control.MethodName.HasThemelconOverride ,  
Control.MethodName.HasThemeStyleboxOverride ,  
Control.MethodName.HasThemeFontOverride ,  
Control.MethodName.HasThemeFontSizeOverride ,  
Control.MethodName.HasThemeColorOverride ,  
Control.MethodName.HasThemeConstantOverride , Control.MethodName.HasThemelcon ,  
Control.MethodName.HasThemeStylebox , Control.MethodName.HasThemeFont ,  
Control.MethodName.HasThemeFontSize , Control.MethodName.HasThemeColor ,  
Control.MethodName.HasThemeConstant ,  
Control.MethodName.GetThemeDefaultBaseScale ,  
Control.MethodName.GetThemeDefaultFont ,  
Control.MethodName.GetThemeDefaultFontSize , Control.MethodName.GetParentControl ,  
Control.MethodName.SetHGrowDirection , Control.MethodName.GetHGrowDirection ,  
Control.MethodName.SetVGrowDirection , Control.MethodName.GetVGrowDirection ,  
Control.MethodName.SetToolTipAutoTranslateMode ,  
Control.MethodName.SetToolTipAutoTranslateMode , Control.MethodName.SetToolTipText ,  
Control.MethodName.SetToolTipText , Control.MethodName.GetTooltip ,  
Control.MethodName.SetDefaultCursorShape ,  
Control.MethodName.GetDefaultCursorShape , Control.MethodName.GetCursorShape ,

Control.MethodName.SetFocusNeighbor , Control.MethodName.GetFocusNeighbor ,  
Control.MethodName.SetFocusNext , Control.MethodName.GetFocusNext ,  
Control.MethodName.SetFocusPrevious , Control.MethodName.GetFocusPrevious ,  
Control.MethodName.ForceDrag , Control.MethodName.SetMouseFilter ,  
Control.MethodName.GetMouseFilter , Control.MethodName.SetForcePassScrollEvents ,  
Control.MethodName.IsForcePassScrollEvents , Control.MethodName.SetClipContents ,  
Control.MethodName.IsClippingContents , Control.MethodName.GrabClickFocus ,  
Control.MethodName.SetDragForwarding , Control.MethodName.SetDragPreview ,  
Control.MethodName.IsDragSuccessful , Control.MethodName.WarpMouse ,  
Control.MethodName.SetShortcutContext , Control.MethodName.GetShortcutContext ,  
Control.MethodName.UpdateMinimumSize , Control.MethodName.SetLayoutDirection ,  
Control.MethodName.GetLayoutDirection , Control.MethodName.IsLayoutRtl ,  
Control.MethodName.SetAutoTranslate , Control.MethodName.IsAutoTranslating ,  
Control.MethodName.SetLocalizeNumeralSystem ,  
Control.MethodName.IsLocalizingNumeralSystem , CanvasItem.MethodName.\_Draw ,  
CanvasItem.MethodName.GetCanvasItem , CanvasItem.MethodName.SetVisible ,  
CanvasItem.MethodName.IsVisible , CanvasItem.MethodName.IsVisibleInTree ,  
CanvasItem.MethodName.Show , CanvasItem.MethodName.Hide ,  
CanvasItem.MethodName.QueueRedraw , CanvasItem.MethodName.MoveToFront ,  
CanvasItem.MethodName.SetAsTopLevel , CanvasItem.MethodName.IsSetAsTopLevel ,  
CanvasItem.MethodName.SetLightMask , CanvasItem.MethodName.GetLightMask ,  
CanvasItem.MethodName.SetModulate , CanvasItem.MethodName.GetModulate ,  
CanvasItem.MethodName.SetSelfModulate , CanvasItem.MethodName.GetSelfModulate ,  
CanvasItem.MethodName.SetZIndex , CanvasItem.MethodName.GetZIndex ,  
CanvasItem.MethodName.SetZAsRelative , CanvasItem.MethodName.IsZRelative ,  
CanvasItem.MethodName.SetYSortEnabled , CanvasItem.MethodName.IsYSortEnabled ,  
CanvasItem.MethodName.SetDrawBehindParent ,  
CanvasItem.MethodName.IsDrawBehindParentEnabled ,  
CanvasItem.MethodName.DrawLine , CanvasItem.MethodName.DrawDashedLine ,  
CanvasItem.MethodName.DrawPolyline , CanvasItem.MethodName.DrawPolylineColors ,  
CanvasItem.MethodName.DrawArc , CanvasItem.MethodName.DrawMultiline ,  
CanvasItem.MethodName.DrawMultilineColors , CanvasItem.MethodName.DrawRect ,  
CanvasItem.MethodName.DrawCircle , CanvasItem.MethodName.DrawTexture ,  
CanvasItem.MethodName.DrawTextureRect ,  
CanvasItem.MethodName.DrawTextureRectRegion ,  
CanvasItem.MethodName.DrawMsdfTextureRectRegion ,  
CanvasItem.MethodName.DrawLcdTextureRectRegion ,  
CanvasItem.MethodName.DrawStyleBox , CanvasItem.MethodName.DrawPrimitive ,  
CanvasItem.MethodName.DrawPolygon , CanvasItem.MethodName.DrawColoredPolygon ,  
CanvasItem.MethodName.DrawString , CanvasItem.MethodName.DrawMultilineString ,

CanvasItem.MethodName.DrawStringOutline ,  
CanvasItem.MethodName.DrawStringOutline ,  
CanvasItem.MethodName.DrawChar , CanvasItem.MethodName.DrawCharOutline ,  
CanvasItem.MethodName.DrawMesh , CanvasItem.MethodName.DrawMultimesh ,  
CanvasItem.MethodName.DrawSetTransform ,  
CanvasItem.MethodName.DrawSetTransformMatrix ,  
CanvasItem.MethodName.DrawAnimationSlice ,  
CanvasItem.MethodName.DrawEndAnimation , CanvasItem.MethodName.GetTransform ,  
CanvasItem.MethodName.GetGlobalTransform ,  
CanvasItem.MethodName.GetGlobalTransformWithCanvas ,  
CanvasItem.MethodName.GetViewportTransform ,  
CanvasItem.MethodName.GetViewportRect ,  
CanvasItem.MethodName.GetCanvasTransform ,  
CanvasItem.MethodName.GetScreenTransform ,  
CanvasItem.MethodName.GetLocalMousePosition ,  
CanvasItem.MethodName.GetGlobalMousePosition , CanvasItem.MethodName.GetCanvas ,  
CanvasItem.MethodName.GetCanvasLayerNode , CanvasItem.MethodName.GetWorld2D ,  
CanvasItem.MethodName.SetMaterial , CanvasItem.MethodName.GetMaterial ,  
CanvasItem.MethodName.SetInstanceShaderParameter ,  
CanvasItem.MethodName.GetInstanceShaderParameter ,  
CanvasItem.MethodName.SetUseParentMaterial ,  
CanvasItem.MethodName.GetUseParentMaterial ,  
CanvasItem.MethodName.SetNotifyLocalTransform ,  
CanvasItem.MethodName.IsLocalTransformNotificationEnabled ,  
CanvasItem.MethodName.SetNotifyTransform ,  
CanvasItem.MethodName.IsTransformNotificationEnabled ,  
CanvasItem.MethodName.ForceUpdateTransform ,  
CanvasItem.MethodName.MakeCanvasPositionLocal ,  
CanvasItem.MethodName.MakeInputLocal , CanvasItem.MethodName.SetVisibilityLayer ,  
CanvasItem.MethodName.GetVisibilityLayer ,  
CanvasItem.MethodName.SetVisibilityLayerBit ,  
CanvasItem.MethodName.GetVisibilityLayerBit , CanvasItem.MethodName.SetTextureFilter ,  
CanvasItem.MethodName.GetTextureFilter , CanvasItem.MethodName.SetTextureRepeat ,  
CanvasItem.MethodName.GetTextureRepeat ,  
CanvasItem.MethodName.SetClipChildrenMode ,  
CanvasItem.MethodName.GetClipChildrenMode , Node.MethodName.\_EnterTree ,  
Node.MethodName.\_ExitTree , Node.MethodName.\_GetConfigurationWarnings ,  
Node.MethodName.\_Input , Node.MethodName.\_PhysicsProcess ,  
Node.MethodName.\_Process , Node.MethodName.\_Ready ,  
Node.MethodName.\_ShortcutInput , Node.MethodName.\_UnhandledInput ,

Node.MethodName.\_UnhandledKeyInput , Node.MethodName.PrintOrphanNodes ,  
Node.MethodName.AddSibling , Node.MethodName.SetName ,  
Node.MethodName.GetName , Node.MethodName.AddChild ,  
Node.MethodName.RemoveChild , Node.MethodName.Reparent ,  
Node.MethodName.GetChildCount , Node.MethodName.GetChildren ,  
Node.MethodName.GetChild , Node.MethodName.HasNode , Node.MethodName.GetNode ,  
Node.MethodName.GetNodeOrNull , Node.MethodName.GetParent ,  
Node.MethodName.FindChild , Node.MethodName.FindChildren ,  
Node.MethodName.FindParent , Node.MethodName.HasNodeAndResource ,  
Node.MethodName.GetNodeAndResource , Node.MethodName.IsInsideTree ,  
Node.MethodName.IsPartOfEditedScene , Node.MethodName.IsAncestorOf ,  
Node.MethodName.IsGreaterThan , Node.MethodName.GetPath ,  
Node.MethodName.GetPathTo , Node.MethodName.AddToGroup ,  
Node.MethodName.RemoveFromGroup , Node.MethodName.IsInGroup ,  
Node.MethodName.MoveChild , Node.MethodName.GetGroups ,  
Node.MethodName.SetOwner , Node.MethodName.GetOwner ,  
Node.MethodName.GetIndex , Node.MethodName.PrintTree ,  
Node.MethodName.PrintTreePretty , Node.MethodName.GetTreeString ,  
Node.MethodName.GetTreeStringPretty , Node.MethodName.SetSceneFilePath ,  
Node.MethodName.GetSceneFilePath , Node.MethodName.PropagateNotification ,  
Node.MethodName.PropagateCall , Node.MethodName.SetPhysicsProcess ,  
Node.MethodName.GetPhysicsProcessDeltaTime , Node.MethodName.IsPhysicsProcessing ,  
Node.MethodName.GetProcessDeltaTime , Node.MethodName.SetProcess ,  
Node.MethodName.SetProcessPriority , Node.MethodName.GetProcessPriority ,  
Node.MethodName.SetPhysicsProcessPriority ,  
Node.MethodName.GetPhysicsProcessPriority , Node.MethodName.IsProcessing ,  
Node.MethodName.SetProcessInput , Node.MethodName.IsProcessingInput ,  
Node.MethodName.SetProcessShortcutInput ,  
Node.MethodName.IsProcessingShortcutInput ,  
Node.MethodName.SetProcessUnhandledInput ,  
Node.MethodName.IsProcessingUnhandledInput ,  
Node.MethodName.SetProcessUnhandledKeyInput ,  
Node.MethodName.IsProcessingUnhandledKeyInput , Node.MethodName.SetProcessMode ,  
Node.MethodName.GetProcessMode , Node.MethodName.CanProcess ,  
Node.MethodName.SetProcessThreadGroup , Node.MethodName.GetProcessThreadGroup ,  
Node.MethodName.SetProcessThreadMessages ,  
Node.MethodName.GetProcessThreadMessages ,  
Node.MethodName.SetProcessThreadGroupOrder ,  
Node.MethodName.GetProcessThreadGroupOrder , Node.MethodName.SetDisplayFolded ,  
Node.MethodName.IsDisplayedFolded , Node.MethodName.SetProcessInternal ,

Node.MethodName.IsProcessingInternal , Node.MethodName.SetPhysicsProcessInternal ,  
Node.MethodName.IsPhysicsProcessingInternal ,  
Node.MethodName.SetPhysicsInterpolationMode ,  
Node.MethodName.GetPhysicsInterpolationMode ,  
Node.MethodName.IsPhysicsInterpolated ,  
Node.MethodName.IsPhysicsInterpolatedAndEnabled ,  
Node.MethodName.ResetPhysicsInterpolation , Node.MethodName.SetAutoTranslateMode ,  
Node.MethodName.GetAutoTranslateMode ,  
Node.MethodName.SetTranslationDomainInherited , Node.MethodName.GetWindow ,  
Node.MethodName.GetLastExclusiveWindow , Node.MethodName.GetTree ,  
Node.MethodName.CreateTween , Node.MethodName.Duplicate ,  
Node.MethodName.ReplaceBy , Node.MethodName.SetSceneInstanceLoadPlaceholder ,  
Node.MethodName.GetSceneInstanceLoadPlaceholder ,  
Node.MethodName.SetEditableInstance , Node.MethodName.IsEditableInstance ,  
Node.MethodName.GetViewport , Node.MethodName.QueueFree ,  
Node.MethodName.RequestReady , Node.MethodName.IsNodeReady ,  
Node.MethodName.SetMultiplayerAuthority , Node.MethodName.GetMultiplayerAuthority ,  
Node.MethodName.IsMultiplayerAuthority , Node.MethodName.GetMultiplayer ,  
Node.MethodName.RpcConfig , Node.MethodName.GetRpcConfig ,  
Node.MethodName.SetEditorDescription , Node.MethodName.GetEditorDescription ,  
Node.MethodName.\_SetImportPath , Node.MethodName.\_GetImportPath ,  
Node.MethodName.SetUniqueNameInOwner , Node.MethodName.IsUniqueNameInOwner ,  
Node.MethodName.Atr , Node.MethodName.AtrN , Node.MethodName.Rpc ,  
Node.MethodName.RpcId , Node.MethodName.UpdateConfigurationWarnings ,  
Node.MethodName.CallDeferredThreadGroup ,  
Node.MethodName.SetDeferredThreadGroup ,  
Node.MethodName.NotifyDeferredThreadGroup , Node.MethodName.CallThreadSafe ,  
Node.MethodName.SetThreadSafe , Node.MethodName.NotifyThreadSafe ,  
GodotObject.MethodName.\_Get , GodotObject.MethodName.\_GetPropertyList ,  
GodotObject.MethodName.\_IterGet , GodotObject.MethodName.\_IterInit ,  
GodotObject.MethodName.\_IterNext , GodotObject.MethodName.\_Notification ,  
GodotObject.MethodName.\_PropertyCanRevert ,  
GodotObject.MethodName.\_PropertyGetRevert , GodotObject.MethodName.\_Set ,  
GodotObject.MethodName.\_ValidateProperty , GodotObject.MethodName.Free ,  
GodotObject.MethodName.GetClass , GodotObject.MethodName.IsClass ,  
GodotObject.MethodName.Set , GodotObject.MethodName.Get ,  
GodotObject.MethodName.SetIndexed , GodotObject.MethodName.GetIndexed ,  
GodotObject.MethodName.GetPropertyList , GodotObject.MethodName.GetMethodList ,  
GodotObject.MethodName.PropertyCanRevert ,  
GodotObject.MethodName.PropertyGetRevert , GodotObject.MethodName.Notification ,

GodotObject.MethodName.GetInstanceID , GodotObject.MethodName.SetScript ,  
GodotObject.MethodName.GetScript , GodotObject.MethodName.SetMeta ,  
GodotObject.MethodName.RemoveMeta , GodotObject.MethodName.GetMeta ,  
GodotObject.MethodName.HasMeta , GodotObject.MethodName.GetMetaList ,  
GodotObject.MethodName.AddUserSignal , GodotObject.MethodName.HasUserSignal ,  
GodotObject.MethodName.RemoveUserSignal , GodotObject.MethodName.EmitSignal ,  
GodotObject.MethodName.Call , GodotObject.MethodName.CallDeferred ,  
GodotObject.MethodName.SetDeferred , GodotObject.MethodName.Callv ,  
GodotObject.MethodName.HasMethod ,  
GodotObject.MethodName.GetMethodArgumentCount ,  
GodotObject.MethodName.HasSignal , GodotObject.MethodName.GetSignalList ,  
GodotObject.MethodName.GetSignalConnectionList ,  
GodotObject.MethodName.GetIncomingConnections , GodotObject.MethodName.Connect ,  
GodotObject.MethodName.Disconnect , GodotObject.MethodName.IsConnected ,  
GodotObject.MethodName.HasConnections , GodotObject.MethodName.SetBlockSignals ,  
GodotObject.MethodName.IsBlockingSignals ,  
GodotObject.MethodName.NotifyPropertyListChanged ,  
GodotObject.MethodName.SetMessageTranslation ,  
GodotObject.MethodName.CanTranslateMessages , GodotObject.MethodName.Tr ,  
GodotObject.MethodName.TrN , GodotObject.MethodName.GetTranslationDomain ,  
GodotObject.MethodName.SetTranslationDomain ,  
GodotObject.MethodName.IsQueuedForDeletion , GodotObject.MethodName.CancelFree ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class AtlasTextureButton.PropertyName

Namespace: [cfGodotEngine.Controls](#)

Assembly: CatSweeper.dll

Cached StringNames for the properties and fields contained in this class, for fast lookup.

```
public class AtlasTextureButton.PropertyName : TextureButton(PropertyName
```

## Inheritance

[object](#) ← GodotObject(PropertyName ← Node(PropertyName ← CanvasItem(PropertyName ← Control(PropertyName ← BaseButton(PropertyName ← TextureButton(PropertyName ← AtlasTextureButton(PropertyName

## Inherited Members

TextureButton(PropertyName.TextureNormal , TextureButton(PropertyName.TexturePressed ,  
TextureButton(PropertyName.TextureHover , TextureButton(PropertyName.TextureDisabled ,  
TextureButton(PropertyName.TextureFocused ,  
TextureButton(PropertyName.TextureClickMask ,  
TextureButton(PropertyName.IgnoreTextureSize ,  
TextureButton(PropertyName.StretchMode , TextureButton(PropertyName.FlipH ,  
TextureButton(PropertyName.FlipV , BaseButton(PropertyName.Disabled ,  
BaseButton(PropertyName.ToggleMode , BaseButton(PropertyName.ButtonPressed ,  
BaseButton(PropertyName.ActionMode , BaseButton(PropertyName.ButtonMask ,  
BaseButton(PropertyName.KeepPressedOutside , BaseButton(PropertyName.ButtonGroup ,  
BaseButton(PropertyName.Shortcut , BaseButton(PropertyName.ShortcutFeedback ,  
BaseButton(PropertyName.ShortcutInTooltip , Control(PropertyName.ClipContents ,  
Control(PropertyName.CustomMinimumSize , Control(PropertyName.LayoutDirection ,  
Control(PropertyName.LayoutMode , Control(PropertyName.AnchorsPreset ,  
Control(PropertyName.AnchorLeft , Control(PropertyName.AnchorTop ,  
Control(PropertyName.AnchorRight , Control(PropertyName.AnchorBottom ,  
Control(PropertyName.OffsetLeft , Control(PropertyName.OffsetTop ,  
Control(PropertyName.OffsetRight , Control(PropertyName.OffsetBottom ,  
Control(PropertyName.GrowHorizontal , Control(PropertyName.GrowVertical ,  
Control(PropertyName.Size , Control(PropertyName.Position ,  
Control(PropertyName.GlobalPosition , Control(PropertyName.Rotation ,  
Control(PropertyName.RotationDegrees , Control(PropertyName.Scale ,  
Control(PropertyName.PivotOffset , Control(PropertyName.SizeTypeFlagsHorizontal ,  
Control(PropertyName.SizeTypeFlagsVertical , Control(PropertyName.SizeTypeFlagsStretchRatio ,

Control.PropertyName.LocalizeNumeralSystem , Control.PropertyName.AutoTranslate ,  
Control.PropertyName.ToolTipText , Control.PropertyName.ToolTipAutoTranslateMode ,  
Control.PropertyName.FocusNeighborLeft , Control.PropertyName.FocusNeighborTop ,  
Control.PropertyName.FocusNeighborRight , Control.PropertyName.FocusNeighborBottom ,  
Control.PropertyName.FocusNext , Control.PropertyName.FocusPrevious ,  
Control.PropertyName.FocusMode , Control.PropertyName.MouseFilter ,  
Control.PropertyName.MouseForcePassScrollEvents ,  
Control.PropertyName.MouseDefaultCursorShape , Control.PropertyName.ShortcutContext ,  
Control.PropertyName.Theme , Control.PropertyName.ThemeTypeVariation ,  
CanvasItem.PropertyName.Visible , CanvasItem.PropertyName.Modulate ,  
CanvasItem.PropertyName.SelfModulate , CanvasItem.PropertyName.ShowBehindParent ,  
CanvasItem.PropertyName.TopLevel , CanvasItem.PropertyName.ClipChildren ,  
CanvasItem.PropertyName.LightMask , CanvasItem.PropertyName.VisibilityLayer ,  
CanvasItem.PropertyName.ZIndex , CanvasItem.PropertyName.ZAsRelative ,  
CanvasItem.PropertyName.YSortEnabled , CanvasItem.PropertyName.TextureFilter ,  
CanvasItem.PropertyName.TextureRepeat , CanvasItem.PropertyName.Material ,  
CanvasItem.PropertyName.UseParentMaterial , Node.PropertyName.\_ImportPath ,  
Node.PropertyName.Name , Node.PropertyName.UniqueNameInOwner ,  
Node.PropertyName.SceneFilePath , Node.PropertyName.Owner ,  
Node.PropertyName.Multiplayer , Node.PropertyName.ProcessMode ,  
Node.PropertyName.ProcessPriority , Node.PropertyName.ProcessPhysicsPriority ,  
Node.PropertyName.ProcessThreadGroup , Node.PropertyName.ProcessThreadGroupOrder ,  
Node.PropertyName.ProcessThreadMessages ,  
Node.PropertyName.PhysicsInterpolationMode , Node.PropertyName.AutoTranslateMode ,  
Node.PropertyName.EditorDescription , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Class AtlasTextureButton.SignalName

Namespace: [cfGodotEngine.Controls](#)

Assembly: CatSweeper.dll

Cached StringNames for the signals contained in this class, for fast lookup.

```
public class AtlasTextureButton.SignalName : TextureButton.SignalName
```

## Inheritance

[object](#) ← GodotObject.SignalName ← Node.SignalName ← CanvasItem.SignalName ← Control.SignalName ← BaseButton.SignalName ← TextureButton.SignalName ← AtlasTextureButton.SignalName

## Inherited Members

BaseButton.SignalName.Pressed , BaseButton.SignalName.ButtonUp ,  
BaseButton.SignalNameButtonDown , BaseButton.SignalName.Toggled ,  
Control.SignalName.Resized , Control.SignalName.GuiInput ,  
Control.SignalName.MouseEntered , Control.SignalName.MouseExited ,  
Control.SignalName.FocusEntered , Control.SignalName.FocusExited ,  
Control.SignalName.SizeFlagsChanged , Control.SignalName.MinimumSizeChanged ,  
Control.SignalName.ThemeChanged , CanvasItem.SignalName.Draw ,  
CanvasItem.SignalName.VisibilityChanged , CanvasItem.SignalName.Hidden ,  
CanvasItem.SignalName.ItemRectChanged , Node.SignalName.Ready ,  
Node.SignalName.Renamed , Node.SignalName.TreeEntered ,  
Node.SignalName.TreeExiting , Node.SignalName.TreeExited ,  
Node.SignalName.ChildEnteredTree , Node.SignalName.ChildExitingTree ,  
Node.SignalName.ChildOrderChanged , Node.SignalName.ReplacingBy ,  
Node.SignalName.EditorDescriptionChanged , Node.SignalName.EditorStateChanged ,  
GodotObject.SignalName.ScriptChanged , GodotObject.SignalName.PropertyListChanged ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class AtlasTextureRect

Namespace: [cfGodotEngine.Controls](#)

Assembly: CatSweeper.dll

```
[Tool]
[GlobalClass]
[ScriptPath("res://Modules/cfGodotEngine/atlaspack/controls/AtlasTextureRect.cs")]
public class AtlasTextureRect : TextureRect, IDisposable
```

## Inheritance

[object](#) ← GodotObject ← Node ← CanvasItem ← Control ← TextureRect ← AtlasTextureRect

## Implements

[IDisposable](#)

## Inherited Members

TextureRect.InvokeGodotClassMethod(in godot\_string\_name, NativeVariantPtrArgs, out godot\_variant) ,  
TextureRect.HasGodotClassMethod(in godot\_string\_name) ,  
TextureRect.HasGodotClassSignal(in godot\_string\_name) , TextureRect.Texture ,  
TextureRect.ExpandMode , TextureRect.StretchMode , TextureRect.FlipH , TextureRect.FlipV ,  
Control.NotificationResized , Control.NotificationMouseEnter ,  
Control.NotificationMouseExit , Control.NotificationMouseEnterSelf ,  
Control.NotificationMouseExitSelf , Control.NotificationFocusEnter ,  
Control.NotificationFocusExit , Control.NotificationThemeChanged ,  
Control.NotificationScrollBegin , Control.NotificationScrollEnd ,  
Control.NotificationLayoutDirectionChanged , Control.\_CanDropData(Vector2, Variant) ,  
Control.\_DropData(Vector2, Variant) , Control.\_GetDragData(Vector2) ,  
Control.\_GetMinimumSize() , Control.\_GetTooltip(Vector2) , Control.\_GUILInput(InputEvent) ,  
Control.\_HasPoint(Vector2) , [Control.\\_MakeCustomTooltip\(string\)](#) ,  
[Control.\\_StructuredTextParser\(Array, string\)](#) , Control.AcceptEvent() ,  
Control.GetMinimumSize() , Control.GetCombinedMinimumSize() ,  
[Control.SetAnchorsPreset\(Control.LayoutPreset, bool\)](#) ,  
[Control.SetOffsetsPreset\(Control.LayoutPreset, Control.LayoutPresetMode, int\)](#) ,  
[Control.SetAnchorsAndOffsetsPreset\(Control.LayoutPreset, Control.LayoutPresetMode, int\)](#) ,  
,

[Control.SetAnchor\(Side, float, bool, bool\)](#) ,  
[Control.SetAnchorAndOffset\(Side, float, float, bool\)](#) , Control.SetBegin(Vector2) ,

Control.SetEnd(Vector2) , [Control.SetPosition\(Vector2, bool\)](#) ,  
[Control.SetSize\(Vector2, bool\)](#) , Control.ResetSize() ,  
[Control.SetGlobalPosition\(Vector2, bool\)](#) , Control.GetBegin() , Control.GetEnd() ,  
Control.GetParentAreaSize() , Control.GetScreenPosition() , Control.GetRect() ,  
Control.GetGlobalRect() , Control.HasFocus() , Control.GrabFocus() ,  
Control.ReleaseFocus() , Control.FindPrevValidFocus() , Control.FindNextValidFocus() ,  
Control.FindValidFocusNeighbor(Side) , Control.BeginBulkThemeOverride() ,  
Control.EndBulkThemeOverride() ,  
Control.AddThemelconOverride(StringName, Texture2D) ,  
Control.AddThemeStyleboxOverride(StringName, StyleBox) ,  
Control.AddThemeFontOverride(StringName, Font) ,  
[Control.AddThemeFontSizeOverride\(StringName, int\)](#) ,  
Control.AddThemeColorOverride(StringName, Color) ,  
[Control.AddThemeConstantOverride\(StringName, int\)](#) ,  
Control.RemoveThemelconOverride(StringName) ,  
Control.RemoveThemeStyleboxOverride(StringName) ,  
Control.RemoveThemeFontOverride(StringName) ,  
Control.RemoveThemeFontSizeOverride(StringName) ,  
Control.RemoveThemeColorOverride(StringName) ,  
Control.RemoveThemeConstantOverride(StringName) ,  
Control.GetThemelcon(StringName, StringName) ,  
Control.GetThemeStylebox(StringName, StringName) ,  
Control.GetThemeFont(StringName, StringName) ,  
Control.GetThemeFontSize(StringName, StringName) ,  
Control.GetThemeColor(StringName, StringName) ,  
Control.GetThemeConstant(StringName, StringName) ,  
Control.HasThemelconOverride(StringName) ,  
Control.HasThemeStyleboxOverride(StringName) ,  
Control.HasThemeFontOverride(StringName) ,  
Control.HasThemeFontSizeOverride(StringName) ,  
Control.HasThemeColorOverride(StringName) ,  
Control.HasThemeConstantOverride(StringName) ,  
Control.HasThemelcon(StringName, StringName) ,  
Control.HasThemeStylebox(StringName, StringName) ,  
Control.HasThemeFont(StringName, StringName) ,  
Control.HasThemeFontSize(StringName, StringName) ,  
Control.HasThemeColor(StringName, StringName) ,  
Control.HasThemeConstant(StringName, StringName) ,  
Control.GetThemeDefaultBaseScale() , Control.GetThemeDefaultFont() ,  
Control.GetThemeDefaultFontSize() , Control.GetParentControl() ,

Control.SetToolTip(Vector2?) , Control.GetCursorShape(Vector2?) ,  
Control.ForceDrag(Variant, Control) , Control.GrabClickFocus() ,  
Control.SetDragForwarding(Callable, Callable, Callable) , Control.SetDragPreview(Control) ,  
Control.IsDragSuccessful() , Control.WarpMouse(Vector2) , Control.UpdateMinimumSize() ,  
Control.IsLayoutRtl() , Control.EmitSignalResized() ,  
Control.EmitSignalGuilInput(InputEvent) , Control.EmitSignalMouseEntered() ,  
Control.EmitSignalMouseExited() , Control.EmitSignalFocusEntered() ,  
Control.EmitSignalFocusExited() , Control.EmitSignalSizeFlagsChanged() ,  
Control.EmitSignalMinimumSizeChanged() , Control.EmitSignalThemeChanged() ,  
Control.ClipContents , Control.CustomMinimumSize , Control.LayoutDirection ,  
Control.AnchorLeft , Control.AnchorTop , Control.AnchorRight , Control.AnchorBottom ,  
Control.OffsetLeft , Control.OffsetTop , Control.OffsetRight , Control.OffsetBottom ,  
Control.GrowHorizontal , Control.GrowVertical , Control.Size , Control.Position ,  
Control.GlobalPosition , Control.Rotation , Control.RotationDegrees , Control.Scale ,  
Control.PivotOffset , Control.SizeFlagsHorizontal , Control.SizeFlagsVertical ,  
Control.SizeFlagsStretchRatio , Control.LocalizeNumeralSystem , Control.AutoTranslate ,  
Control.TooltipText , Control.TooltipAutoTranslateMode , Control.FocusNeighborLeft ,  
Control.FocusNeighborTop , Control.FocusNeighborRight , Control.FocusNeighborBottom ,  
Control.FocusNext , Control.FocusPrevious , Control.FocusMode , Control.MouseFilter ,  
Control.MouseForcePassScrollEvents , Control.MouseDefaultCursorShape ,  
Control.ShortcutContext , Control.Theme , Control.ThemeTypeVariation , Control.Resized ,  
Control.GuilInput , Control.MouseEntered , Control.MouseExited , Control.FocusEntered ,  
Control.FocusExited , Control.SizeFlagsChanged , Control.MinimumSizeChanged ,  
Control.ThemeChanged , CanvasItem.NotificationTransformChanged ,  
CanvasItem.NotificationLocalTransformChanged , CanvasItem.NotificationDraw ,  
CanvasItem.NotificationVisibilityChanged , CanvasItem.NotificationEnterCanvas ,  
CanvasItem.NotificationExitCanvas , CanvasItem.NotificationWorld2DChanged ,  
CanvasItem.\_Draw() , CanvasItem.GetCanvasItem() , CanvasItem.IsVisibleInTree() ,  
CanvasItem.Show() , CanvasItem.Hide() , CanvasItem.QueueRedraw() ,  
CanvasItem.MoveToFront() , [CanvasItem.DrawLine\(Vector2, Vector2, Color, float, bool\)](#) ,  
[CanvasItem.DrawDashedLine\(Vector2, Vector2, Color, float, float, bool, bool\)](#) ,  
[CanvasItem.DrawPolyline\(Vector2\[\], Color, float, bool\)](#) ,  
[CanvasItem.DrawPolyline\(ReadOnlySpan<Vector2>, Color, float, bool\)](#) ,  
[CanvasItem.DrawPolylineColors\(Vector2\[\], Color\[\], float, bool\)](#) ,  
[CanvasItem.DrawPolylineColors\(ReadOnlySpan<Vector2>, ReadOnlySpan<Color>, float, bool\)](#) ,  
[CanvasItem.DrawArc\(Vector2, float, float, float, int, Color, float, bool\)](#) ,  
[CanvasItem.DrawMultiline\(Vector2\[\], Color, float, bool\)](#) ,  
[CanvasItem.DrawMultiline\(ReadOnlySpan<Vector2>, Color, float, bool\)](#) ,  
[CanvasItem.DrawMultilineColors\(Vector2\[\], Color\[\], float, bool\)](#) ,

[CanvasItem.DrawMultilineColors\(ReadOnlySpan<Vector2>, ReadOnlySpan<Color>, float, bool\)](#) ,  
[CanvasItem.DrawRect\(Rect2, Color, bool, float, bool\)](#) ,  
[CanvasItem.DrawCircle\(Vector2, float, Color, bool, float, bool\)](#) ,  
CanvasItem.DrawTexture(Texture2D, Vector2, Color?) ,  
[CanvasItem.DrawTextureRect\(Texture2D, Rect2, bool, Color?, bool\)](#) ,  
[CanvasItem.DrawTextureRectRegion\(Texture2D, Rect2, Rect2, Color?, bool, bool\)](#) ,  
[CanvasItem.DrawMsdfTextureRectRegion\(Texture2D, Rect2, Rect2, Color?, double, double, double\)](#) ,  
CanvasItem.DrawLcdTextureRectRegion(Texture2D, Rect2, Rect2, Color?) ,  
CanvasItem.DrawStyleBox(StyleBox, Rect2) ,  
CanvasItem.DrawPrimitive(Vector2[], Color[], Vector2[], Texture2D) ,  
[CanvasItem.DrawPrimitive\(ReadOnlySpan<Vector2>, ReadOnlySpan<Color>, ReadOnlySpan<Vector2>, Texture2D\)](#) ,  
CanvasItem.DrawPolygon(Vector2[], Color[], Vector2[], Texture2D) ,  
[CanvasItem.DrawPolygon\(ReadOnlySpan<Vector2>, ReadOnlySpan<Color>, ReadOnlySpan<Vector2>, Texture2D\)](#) ,  
CanvasItem.DrawColoredPolygon(Vector2[], Color, Vector2[], Texture2D) ,  
[CanvasItem.DrawColoredPolygon\(ReadOnlySpan<Vector2>, Color, ReadOnlySpan<Vector2>, Texture2D\)](#) ,  
CanvasItem.DrawString(Font, Vector2, string, HorizontalAlignment, float, int, Color?, TextServer.JustificationFlag, TextServer.Direction, TextServer.Orientation) ,  
[CanvasItem.DrawMultilineString\(Font, Vector2, string, HorizontalAlignment, float, int, int, Color?, TextServer.LineBreakFlag, TextServer.JustificationFlag, TextServer.Direction, TextServer.Orientation\)](#) ,  
[CanvasItem.DrawStringOutline\(Font, Vector2, string, HorizontalAlignment, float, int, int, Color?, TextServer.JustificationFlag, TextServer.Direction, TextServer.Orientation\)](#) ,  
[CanvasItem.DrawMultilineStringOutline\(Font, Vector2, string, HorizontalAlignment, float, int, int, int, Color?, TextServer.LineBreakFlag, TextServer.JustificationFlag, TextServer.Direction, TextServer.Orientation\)](#) ,  
CanvasItem.DrawChar(Font, Vector2, string, int, Color?) ,  
[CanvasItem.DrawCharOutline\(Font, Vector2, string, int, int, Color?\)](#) ,  
CanvasItem.DrawMesh(Mesh, Texture2D, Transform2D?, Color?) ,  
CanvasItem.DrawMultimesh(MultiMesh, Texture2D) ,  
[CanvasItem.DrawSetTransform\(Vector2, float, Vector2?\)](#) ,  
CanvasItem.DrawSetTransformMatrix(Transform2D) ,  
[CanvasItem.DrawAnimationSlice\(double, double, double, double\)](#) ,  
CanvasItem.DrawEndAnimation() , CanvasItem.GetTransform() ,  
CanvasItem.GetGlobalTransform() , CanvasItem.GetGlobalTransformWithCanvas() ,  
CanvasItem.GetViewportTransform() , CanvasItem.GetViewportRect() ,

CanvasItem.GetCanvasTransform() , CanvasItem.GetScreenTransform() ,  
CanvasItem.GetLocalMousePosition() , CanvasItem.GetGlobalMousePosition() ,  
CanvasItem.GetCanvas() , CanvasItem.GetCanvasLayerNode() , CanvasItem.GetWorld2D() ,  
CanvasItem.SetInstanceShaderParameter(StringName, Variant) ,  
CanvasItem.GetInstanceShaderParameter(StringName) ,  
[CanvasItem.SetNotifyLocalTransform\(bool\)](#) ,  
CanvasItem.IsLocalTransformNotificationEnabled() ,  
[CanvasItem.SetNotifyTransform\(bool\)](#) , CanvasItem.IsTransformNotificationEnabled() ,  
CanvasItem.ForceUpdateTransform() , CanvasItem.MakeCanvasPositionLocal(Vector2) ,  
CanvasItem.MakeInputLocal(InputEvent) , [CanvasItem.SetVisibilityLayerBit\(uint, bool\)](#) ,  
[CanvasItem.GetVisibilityLayerBit\(uint\)](#) , CanvasItem.EmitSignalDraw() ,  
CanvasItem.EmitSignalVisibilityChanged() , CanvasItem.EmitSignalHidden() ,  
CanvasItem.EmitSignalItemRectChanged() , CanvasItem.Visible , CanvasItem.Modulate ,  
CanvasItem.SelfModulate , CanvasItem.ShowBehindParent , CanvasItem.TopLevel ,  
CanvasItem.ClipChildren , CanvasItem.LightMask , CanvasItem.VisibilityLayer ,  
CanvasItem.ZIndex , CanvasItem.ZAsRelative , CanvasItem.YSortEnabled ,  
CanvasItem.TextureFilter , CanvasItem.TextureRepeat , CanvasItem.Material ,  
CanvasItem.UseParentMaterial , CanvasItem.Draw , CanvasItem.VisibilityChanged ,  
CanvasItem.Hidden , CanvasItem.ItemRectChanged , Node.NotificationEnterTree ,  
Node.NotificationExitTree , Node.NotificationMovedInParent , Node.NotificationReady ,  
Node.NotificationPaused , Node.NotificationUnpaused , Node.NotificationPhysicsProcess ,  
Node.NotificationProcess , Node.NotificationParented , Node.NotificationUnparented ,  
Node.NotificationSceneInstantiated , Node.NotificationDragBegin ,  
Node.NotificationDragEnd , Node.NotificationPathRenamed ,  
Node.NotificationChildOrderChanged , Node.NotificationInternalProcess ,  
Node.NotificationInternalPhysicsProcess , Node.NotificationPostEnterTree ,  
Node.NotificationDisabled , Node.NotificationEnabled ,  
Node.NotificationResetPhysicsInterpolation , Node.NotificationEditorPreSave ,  
Node.NotificationEditorPostSave , Node.NotificationWMMouseEnter ,  
Node.NotificationWMMouseExit , Node.NotificationWMWindowFocusIn ,  
Node.NotificationWMWindowFocusOut , Node.NotificationWMCloseRequest ,  
Node.NotificationWMGoBackRequest , Node.NotificationWMSizeChanged ,  
Node.NotificationWMDpiChange , Node.NotificationVpMouseEnter ,  
Node.NotificationVpMouseExit , Node.NotificationWMPositionChanged ,  
Node.NotificationOsMemoryWarning , Node.NotificationTranslationChanged ,  
Node.NotificationWMAbout , Node.NotificationCrash , Node.NotificationOslmeUpdate ,  
Node.NotificationApplicationResumed , Node.NotificationApplicationPaused ,  
Node.NotificationApplicationFocusIn , Node.NotificationApplicationFocusOut ,  
Node.NotificationTextServerChanged , Node.GetNode<T>(NodePath) ,  
Node.GetNodeOrNull<T>(NodePath) , [Node.GetChild<T>\(int, bool\)](#) ,

[Node.GetChildOrNull<T>\(int, bool\)](#) , Node.GetOwner<T>() , Node.GetOwnerOrNull<T>() ,  
Node.GetParent<T>() , Node.GetParentOrNull<T>() , Node.\_EnterTree() , Node.\_ExitTree() ,  
Node.\_GetConfigurationWarnings() , Node.\_Input(InputEvent) ,  
[Node.\\_PhysicsProcess\(double\)](#) , [Node.\\_Process\(double\)](#) , Node.\_Ready() ,  
Node.\_ShortcutInput(InputEvent) , Node.\_UnhandledInput(InputEvent) ,  
Node.\_UnhandledKeyInput(InputEvent) , Node.PrintOrphanNodes() ,  
[Node.AddSibling\(Node, bool\)](#) , [Node.AddChild\(Node, bool, Node.InternalMode\)](#) ,  
Node.RemoveChild(Node) , [Node.Reparent\(Node, bool\)](#) , [Node.GetChildCount\(bool\)](#) ,  
[Node.GetChildren\(bool\)](#) , [Node.GetChild\(int, bool\)](#) , Node.HasNode(NodePath) ,  
Node.GetNode(NodePath) , Node.GetNodeOrNull(NodePath) , Node.GetParent() ,  
[Node.FindChild\(string, bool, bool\)](#) , [Node.FindChildren\(string, string, bool, bool\)](#) ,  
[Node.FindParent\(string\)](#) , Node.HasNodeAndResource(NodePath) ,  
Node.GetNodeAndResource(NodePath) , Node.IsInsideTree() , Node.IsPartOfEditedScene() ,  
Node.IsAncestorOf(Node) , Node.IsGreaterThanOrEqual(Node) , Node.GetPath() ,  
[Node.GetPathTo\(Node, bool\)](#) , [Node.AddToGroup\(StringName, bool\)](#) ,  
Node.RemoveFromGroup(StringName) , Node.IsInGroup(StringName) ,  
[Node.MoveChild\(Node, int\)](#) , Node.GetGroups() , [Node.GetIndex\(bool\)](#) , Node.PrintTree() ,  
Node.PrintTreePretty() , Node.GetTreeString() , Node.GetTreeStringPretty() ,  
[Node.PropagateNotification\(int\)](#) , [Node.PropagateCall\(StringName, Array, bool\)](#) ,  
[Node.SetPhysicsProcess\(bool\)](#) , Node.GetPhysicsProcessDeltaTime() ,  
Node.IsPhysicsProcessing() , Node.GetProcessDeltaTime() , [Node.SetProcess\(bool\)](#) ,  
Node.IsProcessing() , [Node.SetProcessInput\(bool\)](#) , Node.IsProcessingInput() ,  
[Node.SetProcessShortcutInput\(bool\)](#) , Node.IsProcessingShortcutInput() ,  
[Node.SetProcessUnhandledInput\(bool\)](#) , Node.IsProcessingUnhandledInput() ,  
[Node.SetProcessUnhandledKeyInput\(bool\)](#) , Node.IsProcessingUnhandledKeyInput() ,  
Node.CanProcess() , [Node.SetDisplayFolded\(bool\)](#) , Node.IsDisplayedFolded() ,  
[Node.SetProcessInternal\(bool\)](#) , Node.IsProcessingInternal() ,  
[Node.SetPhysicsProcessInternal\(bool\)](#) , Node.IsPhysicsProcessingInternal() ,  
Node.IsPhysicsInterpolated() , Node.IsPhysicsInterpolatedAndEnabled() ,  
Node.ResetPhysicsInterpolation() , Node.SetTranslationDomainInherited() ,  
Node.GetWindow() , Node.GetLastExclusiveWindow() , Node.GetTree() ,  
Node.CreateTween() , [Node.Duplicate\(int\)](#) , [Node.ReplaceBy\(Node, bool\)](#) ,  
[Node.setSceneInstanceLoadPlaceholder\(bool\)](#) ,  
Node.GetSceneInstanceLoadPlaceholder() , [Node.setEditableInstance\(Node, bool\)](#) ,  
Node.setEditableInstance(Node) , Node.GetViewport() , Node.QueueFree() ,  
Node.RequestReady() , Node.IsNodeReady() , [Node.SetMultiplayerAuthority\(int, bool\)](#) ,  
Node.GetMultiplayerAuthority() , Node.IsMultiplayerAuthority() ,  
Node.RpcConfig(StringName, Variant) , Node.GetRpcConfig() ,  
[Node.Atr\(string, StringName\)](#) , [Node.AtrN\(string, StringName, int, StringName\)](#) ,  
Node.Rpc(StringName, params Variant[]) ,

[Node.Rpc\(StringName, ReadOnlySpan<Variant>\)](#) ,  
[Node.RpcId\(long, StringName, params Variant\[\]\)](#) ,  
[Node.RpcId\(long, StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.UpdateConfigurationWarnings() ,  
Node.CallDeferredThreadGroup(StringName, params Variant[]) ,  
[Node.CallDeferredThreadGroup\(StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.SetDeferredThreadGroup(StringName, Variant) ,  
[Node.NotifyDeferredThreadGroup\(int\)](#) ,  
Node.CallThreadSafe(StringName, params Variant[]) ,  
[Node.CallThreadSafe\(StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.SetThreadSafe(StringName, Variant) , [Node.NotifyThreadSafe\(int\)](#) ,  
Node.EmitSignalReady() , Node.EmitSignalRenamed() , Node.EmitSignalTreeEntered() ,  
Node.EmitSignalTreeExiting() , Node.EmitSignalTreeExited() ,  
Node.EmitSignalChildEnteredTree(Node) , Node.EmitSignalChildExitingTree(Node) ,  
Node.EmitSignalChildOrderChanged() , Node.EmitSignalReplacingBy(Node) ,  
Node.EmitSignalEditorDescriptionChanged(Node) , Node.EmitSignalEditorStateChanged() ,  
Node.Name , Node.UniqueNameInOwner , Node.SceneFilePath , Node.Owner ,  
Node.Multiplayer , Node.ProcessMode , Node.ProcessPriority , Node.ProcessPhysicsPriority ,  
Node.ProcessThreadGroup , Node.ProcessThreadGroupOrder ,  
Node.ProcessThreadMessages , Node.PhysicsInterpolationMode , Node.AutoTranslateMode ,  
Node.EditorDescription , Node.Ready , Node.Renamed , Node.TreeEntered ,  
Node.TreeExiting , Node.TreeExited , Node.ChildEnteredTree , Node.ChildExitingTree ,  
Node.ChildOrderChanged , Node.ReplacingBy , Node.EditorDescriptionChanged ,  
Node.EditorStateChanged , GodotObject.NotificationPostinitialize ,  
GodotObject.NotificationPredelete , GodotObject.NotificationExtensionReloaded ,  
[GodotObject.InstanceFromId\(ulong\)](#) , [GodotObject.IsInstanceIdValid\(ulong\)](#) ,  
GodotObject.IsValid(GodotObject) , GodotObject.WeakRef(GodotObject) ,  
GodotObject.Dispose() , [GodotObject.Dispose\(bool\)](#) , GodotObject.ToString() ,  
GodotObject.ToSignal(GodotObject, StringName) , GodotObject.\_Get(StringName) ,  
GodotObject.\_GetPropertyList() , GodotObject.\_IterGet(Variant) ,  
GodotObject.\_IterInit(Array) , GodotObject.\_IterNext(Array) ,  
[GodotObject.\\_Notification\(int\)](#) , GodotObject.\_PropertyCanRevert(StringName) ,  
GodotObject.\_PropertyGetRevert(StringName) , GodotObject.\_Set(StringName, Variant) ,  
GodotObject.\_ValidateProperty(Dictionary) , GodotObject.Free() , GodotObject.GetClass() ,  
[GodotObject.IsClass\(string\)](#) , GodotObject.Set(StringName, Variant) ,  
GodotObject.Get(StringName) , GodotObject.SetIndexed(NodePath, Variant) ,  
GodotObject.GetIndexed(NodePath) , GodotObject.GetPropertyList() ,  
GodotObject.GetMethodList() , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , [GodotObject.Notification\(int, bool\)](#) ,  
GodotObject.GetInstanceId() , GodotObject.SetScript(Variant) , GodotObject.GetScript() ,

GodotObject.SetMeta(StringName, Variant) , GodotObject.RemoveMeta(StringName) ,  
GodotObject.GetMeta(StringName, Variant) , GodotObject.HasMeta(StringName) ,  
GodotObject.GetMetaList() , [GodotObject.AddUserSignal\(string, Array\)](#) ,  
GodotObject.HasUserSignal(StringName) , GodotObject.RemoveUserSignal(StringName) ,  
GodotObject.EmitSignal(StringName, params Variant[]) ,  
[GodotObject.EmitSignal\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.Call(StringName, params Variant[]) ,  
[GodotObject.Call\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.CallDeferred(StringName, params Variant[]) ,  
[GodotObject.CallDeferred\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.SetDeferred(StringName, Variant) , GodotObject.Callv(StringName, Array) ,  
GodotObject.HasMethod(StringName) ,  
GodotObjectGetMethodArgumentCount(StringName) ,  
GodotObject.HasSignal(StringName) , GodotObject.GetSignalList() ,  
GodotObject.GetSignalConnectionList(StringName) ,  
GodotObject.GetIncomingConnections() ,  
[GodotObject.Connect\(StringName, Callable, uint\)](#) ,  
GodotObject.Disconnect(StringName, Callable) ,  
GodotObject.IsConnected(StringName, Callable) ,  
GodotObject.HasConnections(StringName) , [GodotObject.SetBlockSignals\(bool\)](#) ,  
GodotObject.IsBlockingSignals() , GodotObject.NotifyPropertyListChanged() ,  
[GodotObject.SetMessageTranslation\(bool\)](#) , GodotObject.CanTranslateMessages() ,  
GodotObject.Tr(StringName, StringName) ,  
[GodotObject.TrN\(StringName, StringName, int, StringName\)](#) ,  
GodotObject.GetTranslationDomain() , GodotObject.SetTranslationDomain(StringName) ,  
GodotObject.IsQueuedForDeletion() , GodotObject.CancelFree() ,  
GodotObject.EmitSignalScriptChanged() , GodotObject.EmitSignalPropertyListChanged() ,  
GodotObject.NativeInstance , GodotObject.ScriptChanged ,  
GodotObject.PropertyListChanged , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Extension Methods

[NodeUtil.DontDestroyOnLoad\(Node\)](#)

# Class AtlasTextureRect.MethodName

Namespace: [cfGodotEngine.Controls](#)

Assembly: CatSweeper.dll

Cached StringNames for the methods contained in this class, for fast lookup.

```
public class AtlasTextureRect.MethodName : TextureRect.MethodName
```

## Inheritance

[object](#) ← GodotObject.MethodName ← Node.MethodName ← CanvasItem.MethodName ← Control.MethodName ← TextureRect.MethodName ← AtlasTextureRect.MethodName

## Inherited Members

TextureRect.MethodName.SetTexture , TextureRect.MethodName.GetTexture ,  
TextureRect.MethodName.SetExpandMode , TextureRect.MethodName.GetExpandMode ,  
TextureRect.MethodName.SetFlipH , TextureRect.MethodName.IsFlippedH ,  
TextureRect.MethodName.SetFlipV , TextureRect.MethodName.IsFlippedV ,  
TextureRect.MethodName.SetStretchMode , TextureRect.MethodName.GetStretchMode ,  
Control.MethodName.\_CanDropData , Control.MethodName.\_DropData ,  
Control.MethodName.\_GetDragData , Control.MethodName.\_GetMinimumSize ,  
Control.MethodName.\_GetTooltip , Control.MethodName.\_GuiInput ,  
Control.MethodName.\_HasPoint , Control.MethodName.\_MakeCustomTooltip ,  
Control.MethodName.\_StructuredTextParser , Control.MethodName.AcceptEvent ,  
Control.MethodName.GetMinimumSize , Control.MethodName.GetCombinedMinimumSize ,  
Control.MethodName.\_SetLayoutMode , Control.MethodName.\_GetLayoutMode ,  
Control.MethodName.\_SetAnchorsLayoutPreset ,  
Control.MethodName.\_GetAnchorsLayoutPreset , Control.MethodName.SetAnchorsPreset ,  
Control.MethodName.SetOffsetsPreset , Control.MethodName.SetAnchorsAndOffsetsPreset ,  
Control.MethodName.\_SetAnchor , Control.MethodName.SetAnchor ,  
Control.MethodName.GetAnchor , Control.MethodName.SetOffset ,  
Control.MethodName.GetOffset , Control.MethodName.SetAnchorAndOffset ,  
Control.MethodName.SetBegin , Control.MethodName.SetEnd ,  
Control.MethodName.SetPosition , Control.MethodName.\_SetPosition ,  
Control.MethodName.SetSize , Control.MethodName.ResetSize ,  
Control.MethodName.\_SetSize , Control.MethodName.SetCustomMinimumSize ,  
Control.MethodName.SetGlobalPosition , Control.MethodName.\_SetGlobalPosition ,  
Control.MethodName.SetRotation , Control.MethodName.SetRotationDegrees ,  
Control.MethodName.setScale , Control.MethodName.SetPivotOffset ,

Control.MethodName.GetBegin , Control.MethodName.GetEnd ,  
Control.MethodNameGetPosition , Control.MethodName.GetSize ,  
Control.MethodName.GetRotation , Control.MethodName.GetRotationDegrees ,  
Control.MethodName.GetScale , Control.MethodName.GetPivotOffset ,  
Control.MethodName.GetCustomMinimumSize , Control.MethodName.GetParentAreaSize ,  
Control.MethodName.GetGlobalPosition , Control.MethodName.GetScreenPosition ,  
Control.MethodName.GetRect , Control.MethodName.GetGlobalRect ,  
Control.MethodName.SetFocusMode , Control.MethodName.GetFocusMode ,  
Control.MethodName.HasFocus , Control.MethodName.GrabFocus ,  
Control.MethodName.ReleaseFocus , Control.MethodName.FindPrevValidFocus ,  
Control.MethodName.FindNextValidFocus , Control.MethodName.FindValidFocusNeighbor ,  
Control.MethodName.SetHSizeFlags , Control.MethodName.GetHSizeFlags ,  
Control.MethodName.SetStretchRatio , Control.MethodName.GetStretchRatio ,  
Control.MethodName.SetVSizeFlags , Control.MethodName.GetVSizeFlags ,  
Control.MethodName.SetTheme , Control.MethodName.GetTheme ,  
Control.MethodName.SetThemeTypeVariation ,  
Control.MethodName.GetThemeTypeVariation ,  
Control.MethodName.BeginBulkThemeOverride ,  
Control.MethodName.EndBulkThemeOverride ,  
Control.MethodName.AddThemelconOverride ,  
Control.MethodName.AddThemeStyleboxOverride ,  
Control.MethodName.AddThemeFontOverride ,  
Control.MethodName.AddThemeFontSizeOverride ,  
Control.MethodName.AddThemeColorOverride ,  
Control.MethodName.AddThemeConstantOverride ,  
Control.MethodName.RemoveThemelconOverride ,  
Control.MethodName.RemoveThemeStyleboxOverride ,  
Control.MethodName.RemoveThemeFontOverride ,  
Control.MethodName.RemoveThemeFontSizeOverride ,  
Control.MethodName.RemoveThemeColorOverride ,  
Control.MethodName.RemoveThemeConstantOverride ,  
Control.MethodName.GetThemelcon , Control.MethodName.GetThemeStylebox ,  
Control.MethodName.GetThemeFont , Control.MethodName.GetThemeFontSize ,  
Control.MethodName.GetThemeColor , Control.MethodName.GetThemeConstant ,  
Control.MethodName.HasThemelconOverride ,  
Control.MethodName.HasThemeStyleboxOverride ,  
Control.MethodName.HasThemeFontOverride ,  
Control.MethodName.HasThemeFontSizeOverride ,  
Control.MethodName.HasThemeColorOverride ,  
Control.MethodName.HasThemeConstantOverride , Control.MethodName.HasThemelcon ,

Control.MethodName.HasThemeStylebox , Control.MethodName.HasThemeFont ,  
Control.MethodName.HasThemeFontSize , Control.MethodName.HasThemeColor ,  
Control.MethodName.HasThemeConstant ,  
Control.MethodName.GetThemeDefaultBaseScale ,  
Control.MethodName.GetThemeDefaultFont ,  
Control.MethodName.GetThemeDefaultFontSize , Control.MethodName.GetParentControl ,  
Control.MethodName.SetHGrowDirection , Control.MethodName.GetHGrowDirection ,  
Control.MethodName.SetVGrowDirection , Control.MethodName.GetVGrowDirection ,  
Control.MethodName.SetToolTipAutoTranslateMode ,  
Control.MethodName.GetTooltipAutoTranslateMode , Control.MethodName.SetToolTipText ,  
Control.MethodName.GetTooltipText , Control.MethodName.GetTooltip ,  
Control.MethodName.SetDefaultCursorShape ,  
Control.MethodName.GetDefaultCursorShape , Control.MethodName.GetCursorShape ,  
Control.MethodName.SetFocusNeighbor , Control.MethodName.GetFocusNeighbor ,  
Control.MethodName.SetFocusNext , Control.MethodName.GetFocusNext ,  
Control.MethodName.SetFocusPrevious , Control.MethodName.GetFocusPrevious ,  
Control.MethodName.ForceDrag , Control.MethodName.SetMouseFilter ,  
Control.MethodName.GetMouseFilter , Control.MethodName.SetForcePassScrollEvents ,  
Control.MethodName.IsForcePassScrollEvents , Control.MethodName.SetClipContents ,  
Control.MethodName.IsClippingContents , Control.MethodName.GrabClickFocus ,  
Control.MethodName.SetDragForwarding , Control.MethodName.SetDragPreview ,  
Control.MethodName.IsDragSuccessful , Control.MethodName.WarpMouse ,  
Control.MethodName.SetShortcutContext , Control.MethodName.GetShortcutContext ,  
Control.MethodName.UpdateMinimumSize , Control.MethodName.SetLayoutDirection ,  
Control.MethodName.GetLayoutDirection , Control.MethodName.IsLayoutRtl ,  
Control.MethodName.SetAutoTranslate , Control.MethodName.IsAutoTranslating ,  
Control.MethodName.SetLocalizeNumeralSystem ,  
Control.MethodName.IsLocalizingNumeralSystem , CanvasItem.MethodName.\_Draw ,  
CanvasItem.MethodName.GetCanvasItem , CanvasItem.MethodName.SetVisible ,  
CanvasItem.MethodName.IsVisible , CanvasItem.MethodName.IsVisibleInTree ,  
CanvasItem.MethodName.Show , CanvasItem.MethodName.Hide ,  
CanvasItem.MethodName.QueueRedraw , CanvasItem.MethodName.MoveToFront ,  
CanvasItem.MethodName.SetAsTopLevel , CanvasItem.MethodName.IsSetAsTopLevel ,  
CanvasItem.MethodName.SetLightMask , CanvasItem.MethodName.GetLightMask ,  
CanvasItem.MethodName.SetModulate , CanvasItem.MethodName.GetModulate ,  
CanvasItem.MethodName.SetSelfModulate , CanvasItem.MethodName.GetSelfModulate ,  
CanvasItem.MethodName.SetZIndex , CanvasItem.MethodName.GetZIndex ,  
CanvasItem.MethodName.SetZAsRelative , CanvasItem.MethodName.IsZRelative ,  
CanvasItem.MethodName.SetYSortEnabled , CanvasItem.MethodName.IsYSortEnabled ,  
CanvasItem.MethodName.SetDrawBehindParent ,

CanvasItem.MethodName.IsDrawBehindParentEnabled ,  
CanvasItem.MethodName.DrawLine , CanvasItem.MethodName.DrawDashedLine ,  
CanvasItem.MethodName.DrawPolyline , CanvasItem.MethodName.DrawPolylineColors ,  
CanvasItem.MethodName.DrawArc , CanvasItem.MethodName.DrawMultiline ,  
CanvasItem.MethodName.DrawMultilineColors , CanvasItem.MethodName.DrawRect ,  
CanvasItem.MethodName.DrawCircle , CanvasItem.MethodName.DrawTexture ,  
CanvasItem.MethodName.DrawTextureRect ,  
CanvasItem.MethodName.DrawTextureRectRegion ,  
CanvasItem.MethodName.DrawMsdfTextureRectRegion ,  
CanvasItem.MethodName.DrawLcdTextureRectRegion ,  
CanvasItem.MethodName.DrawStyleBox , CanvasItem.MethodName.DrawPrimitive ,  
CanvasItem.MethodName.DrawPolygon , CanvasItem.MethodName.DrawColoredPolygon ,  
CanvasItem.MethodName.DrawString , CanvasItem.MethodName.DrawMultilineString ,  
CanvasItem.MethodName.DrawStringOutline ,  
CanvasItem.MethodName.DrawMultilineStringOutline ,  
CanvasItem.MethodName.DrawChar , CanvasItem.MethodName.DrawCharOutline ,  
CanvasItem.MethodName.DrawMesh , CanvasItem.MethodName.DrawMultimesh ,  
CanvasItem.MethodName.DrawSetTransform ,  
CanvasItem.MethodName.DrawSetTransformMatrix ,  
CanvasItem.MethodName.DrawAnimationSlice ,  
CanvasItem.MethodName.DrawEndAnimation , CanvasItem.MethodName.GetTransform ,  
CanvasItem.MethodName.GetGlobalTransform ,  
CanvasItem.MethodName.GetGlobalTransformWithCanvas ,  
CanvasItem.MethodName.GetViewportTransform ,  
CanvasItem.MethodName.GetViewportRect ,  
CanvasItem.MethodName.GetCanvasTransform ,  
CanvasItem.MethodName.GetScreenTransform ,  
CanvasItem.MethodName.GetLocalMousePosition ,  
CanvasItem.MethodName.GetGlobalMousePosition , CanvasItem.MethodName.GetCanvas ,  
CanvasItem.MethodName.GetCanvasLayerNode , CanvasItem.MethodName.GetWorld2D ,  
CanvasItem.MethodName.SetMaterial , CanvasItem.MethodName.GetMaterial ,  
CanvasItem.MethodName.SetInstanceShaderParameter ,  
CanvasItem.MethodName.GetInstanceShaderParameter ,  
CanvasItem.MethodName.SetUseParentMaterial ,  
CanvasItem.MethodName.GetUseParentMaterial ,  
CanvasItem.MethodName.SetNotifyLocalTransform ,  
CanvasItem.MethodName.IsLocalTransformNotificationEnabled ,  
CanvasItem.MethodName.SetNotifyTransform ,  
CanvasItem.MethodName.IsTransformNotificationEnabled ,  
CanvasItem.MethodName.ForceUpdateTransform ,

CanvasItem.MethodName.MakeCanvasPositionLocal ,  
CanvasItem.MethodName.MakeInputLocal , CanvasItem.MethodName.SetVisibilityLayer ,  
CanvasItem.MethodName.GetVisibilityLayer ,  
CanvasItem.MethodName.SetVisibilityLayerBit ,  
CanvasItem.MethodName.GetVisibilityLayerBit , CanvasItem.MethodName.SetTextureFilter ,  
CanvasItem.MethodName.GetTextureFilter , CanvasItem.MethodName.SetTextureRepeat ,  
CanvasItem.MethodName.GetTextureRepeat ,  
CanvasItem.MethodName.SetClipChildrenMode ,  
CanvasItem.MethodName.GetClipChildrenMode , Node.MethodName.\_EnterTree ,  
Node.MethodName.\_ExitTree , Node.MethodName.\_GetConfigurationWarnings ,  
Node.MethodName.\_Input , Node.MethodName.\_PhysicsProcess ,  
Node.MethodName.\_Process , Node.MethodName.\_Ready ,  
Node.MethodName.\_ShortcutInput , Node.MethodName.\_UnhandledInput ,  
Node.MethodName.\_UnhandledKeyInput , Node.MethodName.PrintOrphanNodes ,  
Node.MethodName.AddSibling , Node.MethodName.SetName ,  
Node.MethodName.GetName , Node.MethodName.AddChild ,  
Node.MethodName.RemoveChild , Node.MethodName.Reparent ,  
Node.MethodName.GetChildCount , Node.MethodName.GetChildren ,  
Node.MethodName.GetChild , Node.MethodName.HasNode , Node.MethodName.GetNode ,  
Node.MethodName.GetNodeOrNull , Node.MethodName.GetParent ,  
Node.MethodName.FindChild , Node.MethodName.FindChildren ,  
Node.MethodName.FindParent , Node.MethodName.HasNodeAndResource ,  
Node.MethodName.GetNodeAndResource , Node.MethodName.IsInsideTree ,  
Node.MethodName.IsPartOfEditedScene , Node.MethodName.IsAncestorOf ,  
Node.MethodName.IsGreater Than , Node.MethodName.GetPath ,  
Node.MethodName.GetPathTo , Node.MethodName.AddToGroup ,  
Node.MethodName.RemoveFromGroup , Node.MethodName.IsInGroup ,  
Node.MethodName.MoveChild , Node.MethodName.GetGroups ,  
Node.MethodName.SetOwner , Node.MethodName.GetOwner ,  
Node.MethodName.GetIndex , Node.MethodName.PrintTree ,  
Node.MethodName.PrintTreePretty , Node.MethodName.GetTreeString ,  
Node.MethodName.GetTreeStringPretty , Node.MethodName.SetSceneFilePath ,  
Node.MethodName.GetSceneFilePath , Node.MethodName.PropagateNotification ,  
Node.MethodName.PropagateCall , Node.MethodName.SetPhysicsProcess ,  
Node.MethodName.GetPhysicsProcessDeltaTime , Node.MethodName.IsPhysicsProcessing ,  
Node.MethodName.GetProcessDeltaTime , Node.MethodName.SetProcess ,  
Node.MethodName.SetProcessPriority , Node.MethodName.GetProcessPriority ,  
Node.MethodName.SetPhysicsProcessPriority ,  
Node.MethodName.GetPhysicsProcessPriority , Node.MethodName.IsProcessing ,  
Node.MethodName.SetProcessInput , Node.MethodName.IsProcessingInput ,

Node.MethodName.SetProcessShortcutInput ,  
Node.MethodName.IsProcessingShortcutInput ,  
Node.MethodName.SetProcessUnhandledInput ,  
Node.MethodName.IsProcessingUnhandledInput ,  
Node.MethodName.SetProcessUnhandledKeyInput ,  
Node.MethodName.IsProcessingUnhandledKeyInput , Node.MethodName.SetProcessMode ,  
Node.MethodName.GetProcessMode , Node.MethodName.CanProcess ,  
Node.MethodName.SetProcessThreadGroup , Node.MethodName.GetProcessThreadGroup ,  
Node.MethodName.SetProcessThreadMessages ,  
Node.MethodName.GetProcessThreadMessages ,  
Node.MethodName.SetProcessThreadGroupOrder ,  
Node.MethodName.GetProcessThreadGroupOrder , Node.MethodName.SetDisplayFolded ,  
Node.MethodName.IsDisplayedFolded , Node.MethodName.SetProcessInternal ,  
Node.MethodName.IsProcessingInternal , Node.MethodName.SetPhysicsProcessInternal ,  
Node.MethodName.IsPhysicsProcessingInternal ,  
Node.MethodName.SetPhysicsInterpolationMode ,  
Node.MethodName.GetPhysicsInterpolationMode ,  
Node.MethodName.IsPhysicsInterpolated ,  
Node.MethodName.IsPhysicsInterpolatedAndEnabled ,  
Node.MethodName.ResetPhysicsInterpolation , Node.MethodName.SetAutoTranslateMode ,  
Node.MethodName.GetAutoTranslateMode ,  
Node.MethodName.SetTranslationDomainInherited , Node.MethodName.GetWindow ,  
Node.MethodName.GetLastExclusiveWindow , Node.MethodName.GetTree ,  
Node.MethodName.CreateTween , Node.MethodName.Duplicate ,  
Node.MethodName.ReplaceBy , Node.MethodName.SetSceneInstanceLoadPlaceholder ,  
Node.MethodName.GetSceneInstanceLoadPlaceholder ,  
Node.MethodName.SetEditableInstance , Node.MethodName.IsEditableInstance ,  
Node.MethodName.GetViewport , Node.MethodName.QueueFree ,  
Node.MethodName.RequestReady , Node.MethodName.IsNodeReady ,  
Node.MethodName.SetMultiplayerAuthority , Node.MethodName.GetMultiplayerAuthority ,  
Node.MethodName.IsMultiplayerAuthority , Node.MethodName.GetMultiplayer ,  
Node.MethodName.RpcConfig , Node.MethodName.GetRpcConfig ,  
Node.MethodName.SetEditorDescription , Node.MethodName.GetEditorDescription ,  
Node.MethodName.\_SetImportPath , Node.MethodName.\_GetImportPath ,  
Node.MethodName.SetUniqueNameInOwner , Node.MethodName.IsUniqueNameInOwner ,  
Node.MethodName.Atr , Node.MethodName.AtrN , Node.MethodName.Rpc ,  
Node.MethodName.RpcId , Node.MethodName.UpdateConfigurationWarnings ,  
Node.MethodName.CallDeferredThreadGroup ,  
Node.MethodName.SetDeferredThreadGroup ,  
Node.MethodName.NotifyDeferredThreadGroup , Node.MethodName.CallThreadSafe ,

Node.MethodName.SetThreadSafe , Node.MethodName.NotifyThreadSafe ,  
GodotObject.MethodName.\_Get , GodotObject.MethodName.\_GetPropertyList ,  
GodotObject.MethodName.\_IterGet , GodotObject.MethodName.\_IterInit ,  
GodotObject.MethodName.\_IterNext , GodotObject.MethodName.\_Notification ,  
GodotObject.MethodName.\_PropertyCanRevert ,  
GodotObject.MethodName.\_PropertyGetRevert , GodotObject.MethodName.\_Set ,  
GodotObject.MethodName.\_ValidateProperty , GodotObject.MethodName.Free ,  
GodotObject.MethodName.GetClass , GodotObject.MethodName.IsClass ,  
GodotObject.MethodName.Set , GodotObject.MethodName.Get ,  
GodotObject.MethodName.SetIndexed , GodotObject.MethodName.GetIndexed ,  
GodotObject.MethodName.GetPropertyList , GodotObject.MethodName.GetMethodList ,  
GodotObject.MethodName.PropertyCanRevert ,  
GodotObject.MethodName.PropertyGetRevert , GodotObject.MethodName.Notification ,  
GodotObject.MethodName.GetInstanceId , GodotObject.MethodName.SetScript ,  
GodotObject.MethodName.GetScript , GodotObject.MethodName.SetMeta ,  
GodotObject.MethodName.RemoveMeta , GodotObject.MethodName.GetMeta ,  
GodotObject.MethodName.HasMeta , GodotObject.MethodName.GetMetaList ,  
GodotObject.MethodName.AddUserSignal , GodotObject.MethodName.HasUserSignal ,  
GodotObject.MethodName.RemoveUserSignal , GodotObject.MethodName.EmitSignal ,  
GodotObject.MethodName.Call , GodotObject.MethodName.CallDeferred ,  
GodotObject.MethodName.SetDeferred , GodotObject.MethodName.Callv ,  
GodotObject.MethodName.HasMethod ,  
GodotObject.MethodName.GetMethodArgumentCount ,  
GodotObject.MethodName.HasSignal , GodotObject.MethodName.GetSignalList ,  
GodotObject.MethodName.GetSignalConnectionList ,  
GodotObject.MethodName.GetIncomingConnections , GodotObject.MethodName.Connect ,  
GodotObject.MethodName.Disconnect , GodotObject.MethodName.IsConnected ,  
GodotObject.MethodName.HasConnections , GodotObject.MethodName.SetBlockSignals ,  
GodotObject.MethodName.IsBlockingSignals ,  
GodotObject.MethodName.NotifyPropertyChanged ,  
GodotObject.MethodName.SetMessageTranslation ,  
GodotObject.MethodName.CanTranslateMessages , GodotObject.MethodName.Tr ,  
GodotObject.MethodName.TrN , GodotObject.MethodName.GetTranslationDomain ,  
GodotObject.MethodName.SetTranslationDomain ,  
GodotObject.MethodName.IsQueuedForDeletion , GodotObject.MethodName.CancelFree ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Fields

## UpdateTexture

Cached name for the 'UpdateTexture' method.

```
public static readonly StringName UpdateTexture
```

Field Value

StringName

# Class AtlasTextureRect.PropertyName

Namespace: [cfGodotEngine.Controls](#)

Assembly: CatSweeper.dll

Cached StringNames for the properties and fields contained in this class, for fast lookup.

```
public class AtlasTextureRect.PropertyName : TextureRect.PropertyName
```

## Inheritance

[object](#) ← GodotObject(PropertyName) ← Node(PropertyName) ← CanvasItem(PropertyName) ← Control(PropertyName) ← TextureRect(PropertyName) ← AtlasTextureRect(PropertyName)

## Inherited Members

TextureRect(PropertyName.Texture , TextureRect(PropertyName.ExpandMode ,  
TextureRect(PropertyName.StretchMode , TextureRect(PropertyName.FlipH ,  
TextureRect(PropertyName.FlipV , Control(PropertyName.ClipContents ,  
Control(PropertyName.CustomMinimumSize , Control(PropertyName.LayoutDirection ,  
Control(PropertyName.LayoutMode , Control(PropertyName.AnchorsPreset ,  
Control(PropertyName.AnchorLeft , Control(PropertyName.AnchorTop ,  
Control(PropertyName.AnchorRight , Control(PropertyName.AnchorBottom ,  
Control(PropertyName.OffsetLeft , Control(PropertyName.OffsetTop ,  
Control(PropertyName.OffsetRight , Control(PropertyName.OffsetBottom ,  
Control(PropertyName.GrowHorizontal , Control(PropertyName.GrowVertical ,  
Control(PropertyName.Size , Control(PropertyName.Position ,  
Control(PropertyName.GlobalPosition , Control(PropertyName.Rotation ,  
Control(PropertyName.RotationDegrees , Control(PropertyName.Scale ,  
Control(PropertyName.PivotOffset , Control(PropertyName.SizeFlagsHorizontal ,  
Control(PropertyName.SizeFlagsVertical , Control(PropertyName.SizeFlagsStretchRatio ,  
Control(PropertyName.LocalizeNumeralSystem , Control(PropertyName.AutoTranslate ,  
Control(PropertyName.TooltipText , Control(PropertyName.TooltipAutoTranslateMode ,  
Control(PropertyName.FocusNeighborLeft , Control(PropertyName.FocusNeighborTop ,  
Control(PropertyName.FocusNeighborRight , Control(PropertyName.FocusNeighborBottom ,  
Control(PropertyName.FocusNext , Control(PropertyName.FocusPrevious ,  
Control(PropertyName.FocusMode , Control(PropertyName.MouseFilter ,  
Control(PropertyName.MouseForcePassScrollEvents ,  
Control(PropertyName.MouseDefaultCursorShape , Control(PropertyName.ShortcutContext ,  
Control(PropertyName.Theme , Control(PropertyName.ThemeTypeVariation ,  
CanvasItem(PropertyName.Visible , CanvasItem(PropertyName.Modulate ,

CanvasItem.PropertyName.SelfModulate , CanvasItem.PropertyName.ShowBehindParent ,  
CanvasItem.PropertyName.TopLevel , CanvasItem.PropertyName.ClipChildren ,  
CanvasItem.PropertyName.LightMask , CanvasItem.PropertyName.VisibilityLayer ,  
CanvasItem.PropertyName.ZIndex , CanvasItem.PropertyName.ZAsRelative ,  
CanvasItem.PropertyName.YSortEnabled , CanvasItem.PropertyName.TextureFilter ,  
CanvasItem.PropertyName.TextureRepeat , CanvasItem.PropertyName.Material ,  
CanvasItem.PropertyName.UseParentMaterial , Node.PropertyName.\_ImportPath ,  
Node.PropertyName.Name , Node.PropertyName.UniqueNameInOwner ,  
Node.PropertyName.SceneFilePath , Node.PropertyName.Owner ,  
Node.PropertyName.Multiplayer , Node.PropertyName.ProcessMode ,  
Node.PropertyName.ProcessPriority , Node.PropertyName.ProcessPhysicsPriority ,  
Node.PropertyName.ProcessThreadGroup , Node.PropertyName.ProcessThreadGroupOrder ,  
Node.PropertyName.ProcessThreadMessages ,  
Node.PropertyName.PhysicsInterpolationMode , Node.PropertyName.AutoTranslateMode ,  
Node.PropertyName.EditorDescription , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Fields

### \_textureRef

Cached name for the '\_textureRef' field.

```
public static readonly StringName _textureRef
```

### Field Value

StringName

### textureRef

Cached name for the 'textureRef' property.

```
public static readonly StringName textureRef
```

### Field Value

StringName

# Class AtlasTextureRect.SignalName

Namespace: [cfGodotEngine.Controls](#)

Assembly: CatSweeper.dll

Cached StringNames for the signals contained in this class, for fast lookup.

```
public class AtlasTextureRect.SignalName : TextureRect.SignalName
```

## Inheritance

[object](#) ← GodotObject.SignalName ← Node.SignalName ← CanvasItem.SignalName ← Control.SignalName ← TextureRect.SignalName ← AtlasTextureRect.SignalName

## Inherited Members

Control.SignalName.Resized , Control.SignalName.GuiInput ,  
Control.SignalName.MouseEntered , Control.SignalName.MouseExited ,  
Control.SignalName.FocusEntered , Control.SignalName.FocusExited ,  
Control.SignalName.SizeFlagsChanged , Control.SignalName.MinimumSizeChanged ,  
Control.SignalName.ThemeChanged , CanvasItem.SignalName.Draw ,  
CanvasItem.SignalName.VisibilityChanged , CanvasItem.SignalName.Hidden ,  
CanvasItem.SignalName.ItemRectChanged , Node.SignalName.Ready ,  
Node.SignalName.Renamed , Node.SignalName.TreeEntered ,  
Node.SignalName.TreeExiting , Node.SignalName.TreeExited ,  
Node.SignalName.ChildEnteredTree , Node.SignalName.ChildExitingTree ,  
Node.SignalName.ChildOrderChanged , Node.SignalName.ReplacingBy ,  
Node.SignalName.EditorDescriptionChanged , Node.SignalName.EditorStateChanged ,  
GodotObject.SignalName.ScriptChanged , GodotObject.SignalName.PropertyListChanged ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Namespace cfGodotEngine.Core

## Classes

[GodotLogger](#)

# Class GodotLogger

Namespace: [cfGodotEngine.Core](#)

Assembly: CatSweeper.dll

```
public class GodotLogger : ILogger
```

## Inheritance

[object](#) ← GodotLogger

## Implements

[ILogger](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### Asset(bool, object)

```
public void Asset(bool condition, object context = null)
```

#### Parameters

condition [bool](#)

context [object](#)

### LogDebug(string, object)

```
public void LogDebug(string message, object context = null)
```

#### Parameters

message [string](#)

context [object](#)

## .LogError(string, object)

```
public void LogError(string message, object context = null)
```

Parameters

message [string](#)

context [object](#)

## LogException(Exception, object)

```
public void LogException(Exception ex, object message = null)
```

Parameters

ex [Exception](#)

message [object](#)

## LogInfo(string, object)

```
public void LogInfo(string message, object context = null)
```

Parameters

message [string](#)

context [object](#)

## .LogWarning(string, object)

```
public void LogWarning(string message, object context = null)
```

## Parameters

message [string](#)

context [object](#)

# Namespace cfGodotEngine.GoogleDrive

## Classes

[AssetDirectFileMirror](#)

[ChangeHandler](#)

[DriveMirror](#)

[DriveMirrorSetting](#)

[DriveMirrorSetting.MethodName](#)

Cached StringNames for the methods contained in this class, for fast lookup.

[DriveMirrorSetting.PropertyName](#)

Cached StringNames for the properties and fields contained in this class, for fast lookup.

[DriveMirrorSetting.SignalName](#)

Cached StringNames for the signals contained in this class, for fast lookup.

[DriveUtil](#)

[SettingItem](#)

[SettingItem.MethodName](#)

Cached StringNames for the methods contained in this class, for fast lookup.

[SettingItem.PropertyName](#)

Cached StringNames for the properties and fields contained in this class, for fast lookup.

[SettingItem.SignalName](#)

Cached StringNames for the signals contained in this class, for fast lookup.

## Structs

[ChangeInfo](#)

[FileHandler.DownloadRequest](#)

[FileHandler.FileItem](#)

[FolderMimeType](#)

[RefreshRequest](#)

[RefreshResult](#)

[RefreshStatus](#)

[SheetFileHandler](#)

[UrlInfo](#)

## Interfaces

[FileHandler](#)

[IChangeHandler](#)

[IFileMirrorHandler](#)

## Enums

[ChangeType](#)

[FileType](#)

# Class AssetDirectFileMirror

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public class AssetDirectFileMirror : IFileMirrorHandler
```

## Inheritance

[object](#) ← AssetDirectFileMirror

## Implements

[IFileMirrorHandler](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Constructors

### AssetDirectFileMirror(ILogger, string)

```
public AssetDirectFileMirror(ILogger logger, string assetDirectoryPath)
```

## Parameters

logger [ILogger](#)

assetDirectoryPath [string](#)

## Methods

### RefreshFiles(DriveService, in RefreshRequest)

```
public void RefreshFiles(DriveService driveService, in RefreshRequest request)
```

## Parameters

driveService [DriveService](#)

request [RefreshRequest](#)

## RefreshFilesAsync(DriveService, RefreshRequest)

```
public IEnumerable<Task<RefreshStatus>> RefreshFilesAsync(DriveService driveService,  
RefreshRequest request)
```

## Parameters

driveService [DriveService](#)

request [RefreshRequest](#)

## Returns

[IEnumerable](#)<[Task](#)<[RefreshStatus](#)>>

# Class ChangeHandler

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public class ChangeHandler : IChangeHandler
```

## Inheritance

[object](#) ← ChangeHandler

## Implements

[IChangeHandler](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Constructors

### ChangeHandler(ILogger)

```
public ChangeHandler(ILogger logger)
```

## Parameters

logger [ILogger](#)

## Properties

### ChangedFiles

```
public IReadOnlyList<Change> ChangedFiles { get; }
```

## Property Value

[IReadOnlyList](#)<[Change](#)>

## Methods

### IsFileChanged(File)

```
public bool IsFileChanged(File googleFile)
```

#### Parameters

googleFile [File](#)

#### Returns

[bool](#)

### LoadChanges(DriveService, string)

```
public string LoadChanges(DriveService driveService, string startPageToken)
```

#### Parameters

driveService [DriveService](#)

startPageToken [string](#)

#### Returns

[string](#)

### LoadChangesAsync(DriveService, string)

```
public Task<string> LoadChangesAsync(DriveService driveService,  
string startPageToken)
```

Parameters

driveService [DriveService](#)

startPageToken [string](#)

Returns

[Task](#)<[string](#)>

## TryGetFileChange(File, out ChangeInfo?)

```
public bool TryGetFileChange(File googleFile, out ChangeInfo? changeInfo)
```

Parameters

googleFile [File](#)

changeInfo [ChangeInfo?](#)

Returns

[bool](#)

# Struct ChangeInfo

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public struct ChangeInfo
```

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Fields

### File

```
public Optional<File> File
```

#### Field Value

[Optional<File>](#)

### type

```
public ChangeType type
```

#### Field Value

[ChangeType](#)

# Enum ChangeType

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public enum ChangeType
```

## Extension Methods

[EnumExtension.hasFlag\(Enum, Enum\)](#)

## Fields

Modified = 1

None = 0

Removed = 2

# Class DriveMirror

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public class DriveMirror
```

## Inheritance

[object](#) ← DriveMirror

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Constructors

DriveMirror(IFileMirrorHandler, ILogger)

```
public DriveMirror(IFileMirrorHandler mirrorHandler, ILogger logger)
```

## Parameters

mirrorHandler [IFileMirrorHandler](#)

logger [ILogger](#)

## Properties

### instance

```
public static DriveMirror instance { get; }
```

## Property Value

## Methods

### ClearAllAndRefreshAsync()

```
public Task<IEnumerable<RefreshStatus>> ClearAllAndRefreshAsync()
```

Returns

[Task](#)<[IEnumerable](#)<RefreshStatus>>

### ClearAllAndRefreshWithProgressBar()

```
public Task ClearAllAndRefreshWithProgressBar()
```

Returns

[Task](#)

### Refresh()

```
public void Refresh()
```

### RefreshAsync()

```
public Task<IEnumerable<RefreshStatus>> RefreshAsync()
```

Returns

[Task](#)<[IEnumerable](#)<RefreshStatus>>

## RefreshWithProgressBar()

```
public Task RefreshWithProgressBar()
```

Returns

[Task](#)

# Class DriveMirrorSetting

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
[SettingPath("res://Setting/GoogleDrive/DriveMirrorSetting.tres")]
[Tool]
[GlobalClass]
[ScriptPath("res://Modules/cfGodotEngine/google/drive/DriveMirrorSetting.cs")]
public class DriveMirrorSetting : Setting<DriveMirrorSetting>, IDisposable
```

## Inheritance

[object](#) ← GodotObject ← RefCounted ← Resource ← [Setting<DriveMirrorSetting>](#) ← DriveMirrorSetting

## Implements

[IDisposable](#)

## Inherited Members

[Setting<DriveMirrorSetting>.GetSetting\(\)](#) , Resource.\_GetRid() , Resource.\_ResetState() ,  
[Resource.\\_SetPathCache\(string\)](#) , Resource.\_SetupLocalToScene() ,  
[Resource.TakeOverPath\(string\)](#) , [Resource.SetPathCache\(string\)](#) , Resource.GetRid() ,  
Resource.GetLocalScene() , Resource.SetupLocalToScene() , Resource.ResetState() ,  
[Resource.SetIdForPath\(string, string\)](#) , [Resource.GetIdForPath\(string\)](#) ,  
Resource.IsBuiltIn() , Resource.GenerateSceneUniqueId() , Resource.EmitChanged() ,  
[Resource.Duplicate\(bool\)](#) , Resource.EmitSignalChanged() ,  
Resource.EmitSignalSetupLocalToSceneRequested() ,  
Resource.InvokeGodotClassMethod(in godot\_string\_name, NativeVariantPtrArgs, out godot\_variant) ,  
Resource.HasGodotClassMethod(in godot\_string\_name) ,  
Resource.HasGodotClassSignal(in godot\_string\_name) , Resource.ResourceLocalToScene ,  
Resource.ResourcePath , Resource.ResourceName , Resource.ResourceSceneUniqueId ,  
Resource.Changed , Resource.SetupLocalToSceneRequested , RefCounted.InitRef() ,  
RefCounted.Reference() , RefCounted.Unreference() , RefCounted.GetReferenceCount() ,  
GodotObject.NotificationPostinitialize , GodotObject.NotificationPredelete ,  
GodotObject.NotificationExtensionReloaded , [GodotObject.InstanceFromId\(ulong\)](#) ,  
[GodotObject.IsInstanceIdValid\(ulong\)](#) , GodotObject.IsInstanceIdValid(GodotObject) ,  
GodotObject.WeakRef(GodotObject) , GodotObject.Dispose() , [GodotObject.Dispose\(bool\)](#) ,  
GodotObject.ToString() , GodotObject.ToSignal(GodotObject, StringName) ,  
GodotObject.\_Get(StringName) , GodotObject.\_GetPropertyList() ,

GodotObject.\_IterGet(Variant) , GodotObject.\_IterInit(Array) , GodotObject.\_IterNext(Array) ,  
[GodotObject.Notification\(int\)](#) , GodotObject.\_PropertyCanRevert(StringName) ,  
GodotObject.\_PropertyGetRevert(StringName) , GodotObject.\_Set(StringName, Variant) ,  
GodotObject.\_ValidateProperty(Dictionary) , GodotObject.Free() , GodotObject.GetClass() ,  
[GodotObject.IsClass\(string\)](#) , GodotObject.Set(StringName, Variant) ,  
GodotObject.Get(StringName) , GodotObject.SetIndexed(NodePath, Variant) ,  
GodotObject.GetIndexed(NodePath) , GodotObject.GetPropertyList() ,  
GodotObject.GetMethodList() , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , [GodotObject.Notification\(int, bool\)](#) ,  
GodotObject.GetInstanceId() , GodotObject.SetScript(Variant) , GodotObject.GetScript() ,  
GodotObject.SetMeta(StringName, Variant) , GodotObject.RemoveMeta(StringName) ,  
GodotObject.GetMeta(StringName, Variant) , GodotObject.HasMeta(StringName) ,  
GodotObject.GetMetaList() , [GodotObject.AddUserSignal\(string, Array\)](#) ,  
GodotObject.HasUserSignal(StringName) , GodotObject.RemoveUserSignal(StringName) ,  
GodotObject.EmitSignal(StringName, params Variant[]) ,  
[GodotObject.EmitSignal\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.Call(StringName, params Variant[]) ,  
[GodotObject.Call\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.CallDeferred(StringName, params Variant[]) ,  
[GodotObject.CallDeferred\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.SetDeferred(StringName, Variant) , GodotObject.Callv(StringName, Array) ,  
GodotObject.HasMethod(StringName) ,  
GodotObject.GetMethodArgumentCount(StringName) ,  
GodotObject.HasSignal(StringName) , GodotObject.GetSignalList() ,  
GodotObject.GetSignalConnectionList(StringName) ,  
GodotObject.GetIncomingConnections() ,  
[GodotObject.Connect\(StringName, Callable, uint\)](#) ,  
GodotObject.Disconnect(StringName, Callable) ,  
GodotObject.IsConnected(StringName, Callable) ,  
GodotObject.HasConnections(StringName) , [GodotObject.SetBlockSignals\(bool\)](#) ,  
GodotObject.IsBlockingSignals() , GodotObject.NotifyPropertyListChanged() ,  
[GodotObject.SetMessageTranslation\(bool\)](#) , GodotObject.CanTranslateMessages() ,  
GodotObject.Tr(StringName, StringName) ,  
[GodotObject.TrN\(StringName, StringName, int, StringName\)](#) ,  
GodotObject.GetTranslationDomain() , GodotObject.SetTranslationDomain(StringName) ,  
GodotObject.IsQueuedForDeletion() , GodotObject.CancelFree() ,  
GodotObject.EmitSignalScriptChanged() , GodotObject.EmitSignalPropertyListChanged() ,  
GodotObject.NativeInstance , GodotObject.ScriptChanged ,  
GodotObject.PropertyListChanged , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,

[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Fields

### changeChecksumToken

```
[Export(PropertyHint.None, "")]  
public string changeChecksumToken
```

#### Field Value

[string](#)

### items

```
[Export(PropertyHint.None, "")]  
public Array<SettingItem> items
```

#### Field Value

[Array<SettingItem>](#)

### refreshOnEnterPlayMode

```
[Export(PropertyHint.None, "")]  
public bool refreshOnEnterPlayMode
```

#### Field Value

[bool](#)

## Properties

## ForceRefreshAllButton

```
[ExportToolButton("Force Refresh All")]
public Callable ForceRefreshAllButton { get; }
```

Property Value

Callable

## RefreshButton

```
[ExportToolButton("Refresh")]
public Callable RefreshButton { get; }
```

Property Value

Callable

## serviceAccountCredentialJson

```
public string serviceAccountCredentialJson { get; }
```

Property Value

[string](#)

## settingMap

```
public Dictionary<string, SettingItem> settingMap { get; }
```

Property Value

[Dictionary](#)<[string](#), [SettingItem](#)>

# Methods

## ForceRefreshAll()

```
public void ForceRefreshAll()
```

## Refresh()

```
public void Refresh()
```

# Class DriveMirrorSetting.MethodName

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

Cached StringNames for the methods contained in this class, for fast lookup.

```
public class DriveMirrorSetting.MethodName : Setting<DriveMirrorSetting>.MethodName
```

## Inheritance

```
object ← GodotObject.MethodName ← RefCounted.MethodName ←  
Resource.MethodName ← Setting<DriveMirrorSetting>.MethodName ←  
DriveMirrorSetting.MethodName
```

## Inherited Members

```
Resource.MethodName._GetRid , Resource.MethodName._ResetState ,  
Resource.MethodName._SetPathCache , Resource.MethodName._SetupLocalToScene ,  
Resource.MethodName.SetPath , Resource.MethodName.TakeOverPath ,  
Resource.MethodName.GetPath , Resource.MethodName.SetPathCache ,  
Resource.MethodNameSetName , Resource.MethodName.GetName ,  
Resource.MethodName.GetRid , Resource.MethodName.SetLocalToScene ,  
Resource.MethodName.IsLocalToScene , Resource.MethodName.GetLocalScene ,  
Resource.MethodName.SetupLocalToScene , Resource.MethodName.ResetState ,  
Resource.MethodName.SetIdForPath , Resource.MethodName.GetIdForPath ,  
Resource.MethodName.IsBuiltIn , Resource.MethodName.GenerateSceneUniqueId ,  
Resource.MethodName.SetSceneUniqueId , Resource.MethodName.GetSceneUniqueId ,  
Resource.MethodName.EmitChanged , Resource.MethodName.Duplicate ,  
RefCounted.MethodName.InitRef , RefCounted.MethodName.Reference ,  
RefCounted.MethodName.Unreference , RefCounted.MethodName.GetReferenceCount ,  
GodotObject.MethodName._Get , GodotObject.MethodName._GetPropertyList ,  
GodotObject.MethodName._IterGet , GodotObject.MethodName._IterInit ,  
GodotObject.MethodName._IterNext , GodotObject.MethodName._Notification ,  
GodotObject.MethodName._PropertyCanRevert ,  
GodotObject.MethodName._PropertyGetRevert , GodotObject.MethodName._Set ,  
GodotObject.MethodName._ValidateProperty , GodotObject.MethodName.Free ,  
GodotObject.MethodName.GetClass , GodotObject.MethodName.IsClass ,  
GodotObject.MethodName.Set , GodotObject.MethodName.Get ,  
GodotObject.MethodName.SetIndexed , GodotObject.MethodName.GetIndexed ,  
GodotObject.MethodName.GetPropertyList , GodotObject.MethodName.GetMethodList ,
```

GodotObject.MethodName.PropertyCanRevert ,  
GodotObject.MethodName.PropertyGetRevert , GodotObject.MethodName.Notification ,  
GodotObject.MethodName.GetInstanceID , GodotObject.MethodName.SetScript ,  
GodotObject.MethodName.GetScript , GodotObject.MethodName.SetMeta ,  
GodotObject.MethodName.RemoveMeta , GodotObject.MethodName.GetMeta ,  
GodotObject.MethodName.HasMeta , GodotObject.MethodName.GetMetaList ,  
GodotObject.MethodName.AddUserSignal , GodotObject.MethodName.HasUserSignal ,  
GodotObject.MethodName.RemoveUserSignal , GodotObject.MethodName.EmitSignal ,  
GodotObject.MethodName.Call , GodotObject.MethodName.CallDeferred ,  
GodotObject.MethodName.SetDeferred , GodotObject.MethodName.Callv ,  
GodotObject.MethodName.HasMethod ,  
GodotObject.MethodName.GetMethodArgumentCount ,  
GodotObject.MethodName.HasSignal , GodotObject.MethodName.GetSignalList ,  
GodotObject.MethodName.GetSignalConnectionList ,  
GodotObject.MethodName.GetIncomingConnections , GodotObject.MethodName.Connect ,  
GodotObject.MethodName.Disconnect , GodotObject.MethodName.IsConnected ,  
GodotObject.MethodName.HasConnections , GodotObject.MethodName.SetBlockSignals ,  
GodotObject.MethodName.IsBlockingSignals ,  
GodotObject.MethodName.NotifyPropertyListChanged ,  
GodotObject.MethodName.SetMessageTranslation ,  
GodotObject.MethodName.CanTranslateMessages , GodotObject.MethodName.Tr ,  
GodotObject.MethodName.TrN , GodotObject.MethodName.GetTranslationDomain ,  
GodotObject.MethodName.SetTranslationDomain ,  
GodotObject.MethodName.IsQueuedForDeletion , GodotObject.MethodName.CancelFree ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### ForceRefreshAll

Cached name for the 'ForceRefreshAll' method.

```
public static readonly StringName ForceRefreshAll
```

### Field Value

StringName

## Refresh

Cached name for the 'Refresh' method.

```
public static readonly StringName Refresh
```

Field Value

StringName

## UpdateSettingMap

Cached name for the 'UpdateSettingMap' method.

```
public static readonly StringName UpdateSettingMap
```

Field Value

StringName

# Class DriveMirrorSetting.PropertyName

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

Cached StringNames for the properties and fields contained in this class, for fast lookup.

```
public class DriveMirrorSetting.PropertyName :  
Setting<DriveMirrorSetting>.PropertyName
```

## Inheritance

```
object ← GodotObject(PropertyName) ← RefCounted(PropertyName) ←  
Resource(PropertyName) ← Setting<DriveMirrorSetting>.PropertyName ←  
DriveMirrorSetting(PropertyName)
```

## Inherited Members

```
Resource(PropertyName.ResourceLocalToScene , Resource(PropertyName.ResourcePath ,  
Resource(PropertyName.ResourceName , Resource(PropertyName.ResourceSceneUniqueID ,  
object.Equals\(object\) , object.Equals\(object, object\) , object.GetHashCode\(\) ,  
object.GetType\(\) , object.MemberwiseClone\(\) , object.ReferenceEquals\(object, object\) ,  
object.ToString\(\))
```

## Fields

### ForceRefreshAllButton

Cached name for the 'ForceRefreshAllButton' property.

```
public static readonly StringName ForceRefreshAllButton
```

#### Field Value

StringName

### RefreshButton

Cached name for the 'RefreshButton' property.

```
public static readonly StringName RefreshButton
```

## Field Value

StringName

## \_serviceAccountCredentialJsonPath

Cached name for the '\_serviceAccountCredentialJsonPath' field.

```
public static readonly StringName _serviceAccountCredentialJsonPath
```

## Field Value

StringName

## changeChecksumToken

Cached name for the 'changeChecksumToken' field.

```
public static readonly StringName changeChecksumToken
```

## Field Value

StringName

## items

Cached name for the 'items' field.

```
public static readonly StringName items
```

## Field Value

StringName

## refreshOnEnterPlayMode

Cached name for the 'refreshOnEnterPlayMode' field.

```
public static readonly StringName refreshOnEnterPlayMode
```

### Field Value

StringName

## serviceAccountCredentialJson

Cached name for the 'serviceAccountCredentialJson' property.

```
public static readonly StringName serviceAccountCredentialJson
```

### Field Value

StringName

## serviceAccountCredentialJsonPath

Cached name for the 'serviceAccountCredentialJsonPath' property.

```
public static readonly StringName serviceAccountCredentialJsonPath
```

### Field Value

StringName

# Class DriveMirrorSetting.SignalName

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

Cached StringNames for the signals contained in this class, for fast lookup.

```
public class DriveMirrorSetting.SignalName : Setting<DriveMirrorSetting>.SignalName
```

## Inheritance

[object](#) ← GodotObject.SignalName ← RefCounted.SignalName ← Resource.SignalName ← [Setting<DriveMirrorSetting>.SignalName](#) ← DriveMirrorSetting.SignalName

## Inherited Members

Resource.SignalName.Changed , Resource.SignalName.SetupLocalToSceneRequested ,  
GodotObject.SignalName.ScriptChanged , GodotObject.SignalName.PropertyListChanged ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class DriveUtil

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public static class DriveUtil
```

## Inheritance

[object](#) ← DriveUtil

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### MimeFileHandlers

```
public static IReadOnlyDictionary<string, FileHandler> MimeFileHandlers
```

#### Field Value

[IReadOnlyDictionary](#)<[string](#) , [FileHandler](#)>

### godotLogger

```
public static readonly ILogger godotLogger
```

#### Field Value

[ILogger](#)

## Methods

## FormLink(string)

```
public static string FormLink(string driveFileDialog)
```

Parameters

driveFileDialog [string](#)

Returns

[string](#)

## ParseSegments(ReadOnlyMemory<string>)

```
public static Res<UrlInfo, Exception> ParseSegments(ReadOnlyMemory<string> segments)
```

Parameters

segments [ReadOnlyMemory](#)<[string](#)>

Returns

[Res](#)<[UrlInfo](#), [Exception](#)>

## ParseUrl(string)

```
public static Res<UrlInfo, Exception> ParseUrl(string driveLink)
```

Parameters

driveLink [string](#)

Returns

[Res](#)<[UrlInfo](#), [Exception](#)>

# Interface FileHandler

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public interface FileHandler
```

## Methods

### DownloadAsync(FilesResource, DownloadRequest)

```
Task<IDownloadProgress> DownloadAsync(FilesResource filesResource,  
FileHandler.DownloadRequest downloadRequest)
```

#### Parameters

filesResource [FilesResource](#)

downloadRequest [FileHandler.DownloadRequest](#)

#### Returns

[Task](#)<[IDownloadProgress](#)>

### DownloadWithStatus(FilesResource, in DownloadRequest)

```
IDownloadProgress DownloadWithStatus(FilesResource filesResource, in  
FileHandler.DownloadRequest downloadRequest)
```

#### Parameters

filesResource [FilesResource](#)

downloadRequest [FileHandler.DownloadRequest](#)

## Returns

[IDownloadProgress](#)

# Struct FileHandler.DownloadRequest

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public struct FileHandler.DownloadRequest
```

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Fields

### changeHandler

```
public IChangeHandler changeHandler
```

Field Value

[IChangeHandler](#)

### googleFileDialog

```
public string googleFileDialog
```

Field Value

[string](#)

### localName

```
public string localName
```

Field Value

[string](#) ↗

## rootDirectoryInfo

`public DirectoryInfo rootDirectoryInfo`

Field Value

[DirectoryInfo](#) ↗

# Struct FileHandler.FileItem

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public struct FileHandler.FileItem
```

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Fields

### RelativePathSegment

```
public PathSegment RelativePathSegment
```

#### Field Value

[PathSegment](#)

### googleFile

```
public File googleFile
```

#### Field Value

[File](#)

# Enum FileType

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public enum FileType
```

## Extension Methods

[EnumExtension.hasFlag\(Enum, Enum\)](#)

## Fields

Folder = 1

None = 0

Sheet = 2

# Struct FolderMimeHandler

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public struct FolderMimeHandler : FileHandler
```

## Implements

[FileHandler](#)

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Constructors

### FolderMimeHandler(ILocator, string)

```
public FolderMimeHandler(ILocator logger, string assetDirectoryPath)
```

## Parameters

logger [ILocator](#)

assetDirectoryPath [string](#)

## Methods

### DownloadAsync(FilesResource, DownloadRequest)

```
public Task<IDownloadProgress> DownloadAsync(FilesResource filesResource,  
FileHandler.DownloadRequest downloadRequest)
```

## Parameters

`filesResource` [FilesResource](#)

`downloadRequest` [FileHandler.DownloadRequest](#)

Returns

[Task](#)<[IDownloadProgress](#)>

## DownloadWithStatus(FilesResource, in DownloadRequest)

```
public IDownloadProgress DownloadWithStatus(FilesResource filesResource, in  
FileHandler.DownloadRequest downloadRequest)
```

Parameters

`filesResource` [FilesResource](#)

`downloadRequest` [FileHandler.DownloadRequest](#)

Returns

[IDownloadProgress](#)

## GetFolderContent(FilesResource, string)

```
public IEnumerable<FileHandler.FileItem> GetFolderContent(FilesResource  
filesResource, string googleFileDialog)
```

Parameters

`filesResource` [FilesResource](#)

`googleFileDialog` [string](#)

Returns

[IEnumerable](#)<[FileHandler.FileItem](#)>

## GetFolderContentAsync(FilesResource, string)

```
public Task<IEnumerable<FileHandler.FileItem>> GetFolderContentAsync(FilesResource  
filesResource, string googleFileDialog)
```

### Parameters

filesResource [FilesResource](#)

googleFileDialog [string](#)

### Returns

[Task](#) <[IEnumerable](#) <[FileHandler](#).[FileItem](#)>>

# Interface IChangeHandler

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public interface IChangeHandler
```

## Methods

### IsFileChanged(File)

```
bool IsFileChanged(File googleFile)
```

#### Parameters

googleFile [File](#)

#### Returns

[bool](#)

### LoadChanges(DriveService, string)

```
string LoadChanges(DriveService driveService, string startPageToken)
```

#### Parameters

driveService [DriveService](#)

startPageToken [string](#)

#### Returns

[string](#)

## LoadChangesAsync(DriveService, string)

Task<string> **LoadChangesAsync**(DriveService driveService, **string** startPageToken)

Parameters

driveService [DriveService](#)

startPageToken [string](#)

Returns

[Task](#)<[string](#)>

## TryGetFileChange(File, out ChangeInfo?)

bool **TryGetFileChange**(File googleFile, **out** ChangeInfo? changeInfo)

Parameters

googleFile [File](#)

changeInfo [ChangeInfo?](#)

Returns

[bool](#)

# Interface IFileMirrorHandler

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public interface IFileMirrorHandler
```

## Methods

### RefreshFiles(DriveService, in RefreshRequest)

```
void RefreshFiles(DriveService driveService, in RefreshRequest request)
```

#### Parameters

driveService [DriveService](#)

request [RefreshRequest](#)

### RefreshFilesAsync(DriveService, RefreshRequest)

```
IEnumerable<Task<RefreshStatus>> RefreshFilesAsync(DriveService driveService,  
RefreshRequest request)
```

#### Parameters

driveService [DriveService](#)

request [RefreshRequest](#)

#### Returns

[IEnumerable](#)<[Task](#)<[RefreshStatus](#)>>

# Struct RefreshRequest

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public struct RefreshRequest
```

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Fields

### changeHandler

```
public IChangeHandler changeHandler
```

Field Value

[IChangeHandler](#)

### getSetting

```
public Func<File, Res<Optional<SettingItem>, Exception>> getSetting
```

Field Value

[Func<File, Res<Optional<SettingItem>, Exception>>](#)

### googleFiles

```
public IList<File> googleFiles
```

## Field Value

[IList](#) <[File](#)>

# Struct RefreshResult

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public struct RefreshResult
```

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Fields

### newChangeChecksumToken

```
public string newChangeChecksumToken
```

#### Field Value

[string](#)

# Struct RefreshStatus

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public struct RefreshStatus
```

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Constructors

### RefreshStatus(File, IDownloadProgress, float)

```
public RefreshStatus(File file, IDownloadProgress status, float progress)
```

## Parameters

file [File](#)

status [IDownloadProgress](#)

progress [float](#)

## Fields

### file

```
public readonly File file
```

## Field Value

[File](#)

## progress

```
public readonly float progress
```

### Field Value

[float](#)

## status

```
public readonly IDownloadProgress status
```

### Field Value

[IDownloadProgress](#)

# Class SettingItem

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
[Tool]
[GlobalClass]
[ScriptPath("res://Modules/cfGodotEngine/google/drive/SettingItem.cs")]
public class SettingItem : Resource, IDisposable
```

## Inheritance

[object](#) ← GodotObject ← RefCounted ← Resource ← SettingItem

## Implements

[IDisposable](#)

## Inherited Members

Resource.\_GetRid() , Resource.\_ResetState() , [Resource.\\_SetPathCache\(string\)](#) ,  
Resource.\_SetupLocalToScene() , [Resource.TakeOverPath\(string\)](#) ,  
[Resource.SetPathCache\(string\)](#) , Resource.GetRid() , Resource.GetLocalScene() ,  
Resource.SetupLocalToScene() , Resource.ResetState() ,  
[Resource.SetIdForPath\(string, string\)](#) , [Resource.GetIdForPath\(string\)](#) ,  
Resource.IsBuiltIn() , Resource.GenerateSceneUniqueId() , Resource.EmitChanged() ,  
[Resource.Duplicate\(bool\)](#) , Resource.EmitSignalChanged() ,  
Resource.EmitSignalSetupLocalToSceneRequested() ,  
Resource.InvokeGodotClassMethod(in godot\_string\_name, NativeVariantPtrArgs, out godot\_variant) ,  
Resource.HasGodotClassMethod(in godot\_string\_name) ,  
Resource.HasGodotClassSignal(in godot\_string\_name) , Resource.ResourceLocalToScene ,  
Resource.ResourcePath , Resource.ResourceName , Resource.ResourceSceneUniqueId ,  
Resource.Changed , Resource.SetupLocalToSceneRequested , RefCounted.InitRef() ,  
RefCounted.Reference() , RefCounted.Unreference() , RefCounted.GetReferenceCount() ,  
GodotObject.NotificationPostinitialize , GodotObject.NotificationPredelete ,  
GodotObject.NotificationExtensionReloaded , [GodotObject.InstanceFromId\(ulong\)](#) ,  
[GodotObject.IsInstanceIdValid\(ulong\)](#) , GodotObject.IsInstanceIdValid(GodotObject) ,  
GodotObject.WeakRef(GodotObject) , GodotObject.Dispose() , [GodotObject.Dispose\(bool\)](#) ,  
GodotObject.ToString() , GodotObject.ToSignal(GodotObject, StringName) ,  
GodotObject.\_Get(StringName) , GodotObject.\_GetPropertyList() ,  
GodotObject.\_IterGet(Variant) , GodotObject.\_IterInit(Array) , GodotObject.\_IterNext(Array) ,

[GodotObject.Notification\(int\)](#) , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , GodotObject.Set(StringName, Variant) ,  
GodotObject.ValidateProperty(Dictionary) , GodotObject.Free() , GodotObject.GetClass() ,  
[GodotObject.IsClass\(string\)](#) , GodotObject.Set(StringName, Variant) ,  
GodotObject.Get(StringName) , GodotObject.SetIndexed(NodePath, Variant) ,  
GodotObject.GetIndexed(NodePath) , GodotObject.GetPropertyList() ,  
GodotObject.GetMethodList() , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , [GodotObject.Notification\(int, bool\)](#) ,  
GodotObject.GetInstanceId() , GodotObject.SetScript(Variant) , GodotObject.GetScript() ,  
GodotObject.SetMeta(StringName, Variant) , GodotObject.RemoveMeta(StringName) ,  
GodotObject.GetMeta(StringName, Variant) , GodotObject.HasMeta(StringName) ,  
GodotObject.GetMetaList() , [GodotObject.AddUserSignal\(string, Array\)](#) ,  
GodotObject.HasUserSignal(StringName) , GodotObject.RemoveUserSignal(StringName) ,  
GodotObject.EmitSignal(StringName, params Variant[]) ,  
[GodotObject.EmitSignal\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.Call(StringName, params Variant[]) ,  
[GodotObject.Call\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.CallDeferred(StringName, params Variant[]) ,  
[GodotObject.CallDeferred\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.SetDeferred(StringName, Variant) , GodotObject.Callv(StringName, Array) ,  
GodotObject.HasMethod(StringName) ,  
GodotObjectGetMethodArgumentCount(StringName) ,  
GodotObject.HasSignal(StringName) , GodotObject.GetSignalList() ,  
GodotObject.GetSignalConnectionList(StringName) ,  
GodotObject.GetIncomingConnections() ,  
[GodotObject.Connect\(StringName, Callable, uint\)](#) ,  
GodotObject.Disconnect(StringName, Callable) ,  
GodotObject.IsConnected(StringName, Callable) ,  
GodotObject.HasConnections(StringName) , [GodotObject.SetBlockSignals\(bool\)](#) ,  
GodotObject.IsBlockingSignals() , GodotObject.NotifyPropertyListChanged() ,  
[GodotObject.SetMessageTranslation\(bool\)](#) , GodotObject.CanTranslateMessages() ,  
GodotObject.Tr(StringName, StringName) ,  
[GodotObject.TrN\(StringName, StringName, int, StringName\)](#) ,  
GodotObject.GetTranslationDomain() , GodotObject.SetTranslationDomain(StringName) ,  
GodotObject.IsQueuedForDeletion() , GodotObject.CancelFree() ,  
GodotObject.EmitSignalScriptChanged() , GodotObject.EmitSignalPropertyListChanged() ,  
GodotObject.NativeInstance , GodotObject.ScriptChanged ,  
GodotObject.PropertyListChanged , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

# Fields

## assetPath

```
[Export(PropertyHint.None, "")]  
public string assetPath
```

### Field Value

[string](#) ↗

## driveLink

```
[Export(PropertyHint.None, "")]  
public string driveLink
```

### Field Value

[string](#) ↗

# Properties

## fileName

```
[Export(PropertyHint.None, "")]  
public string fileName { get; set; }
```

### Property Value

[string](#) ↗

# Class SettingItem.MethodName

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

Cached StringNames for the methods contained in this class, for fast lookup.

```
public class SettingItem.MethodName : Resource.MethodName
```

## Inheritance

[object](#) ← GodotObject.MethodName ← RefCounted.MethodName ←

Resource.MethodName ← SettingItem.MethodName

## Inherited Members

Resource.MethodName.\_GetRid , Resource.MethodName.\_ResetState ,  
Resource.MethodName.\_SetPathCache , Resource.MethodName.\_SetupLocalToScene ,  
Resource.MethodName.SetPath , Resource.MethodName.TakeOverPath ,  
Resource.MethodName.GetPath , Resource.MethodName.SetPathCache ,  
Resource.MethodNameSetName , Resource.MethodName.GetName ,  
Resource.MethodName.GetRid , Resource.MethodName.SetLocalToScene ,  
Resource.MethodName.IsLocalToScene , Resource.MethodName.GetLocalScene ,  
Resource.MethodName.SetupLocalToScene , Resource.MethodName.ResetState ,  
Resource.MethodName.SetIdForPath , Resource.MethodName.GetIdForPath ,  
Resource.MethodName.IsBuiltIn , Resource.MethodName.GenerateSceneUniqueId ,  
Resource.MethodName.SetSceneUniqueId , Resource.MethodName.GetSceneUniqueId ,  
Resource.MethodName.EmitChanged , Resource.MethodName.Duplicate ,  
RefCounted.MethodName.InitRef , RefCounted.MethodName.Reference ,  
RefCounted.MethodName.Unreference , RefCounted.MethodName.GetReferenceCount ,  
GodotObject.MethodName.\_Get , GodotObject.MethodName.\_GetPropertyList ,  
GodotObject.MethodName.\_IterGet , GodotObject.MethodName.\_IterInit ,  
GodotObject.MethodName.\_IterNext , GodotObject.MethodName.\_Notification ,  
GodotObject.MethodName.\_PropertyCanRevert ,  
GodotObject.MethodName.\_PropertyGetRevert , GodotObject.MethodName.\_Set ,  
GodotObject.MethodName.\_ValidateProperty , GodotObject.MethodName.Free ,  
GodotObject.MethodName.GetClass , GodotObject.MethodName.IsClass ,  
GodotObject.MethodName.Set , GodotObject.MethodName.Get ,  
GodotObject.MethodName.SetIndexed , GodotObject.MethodName.GetIndexed ,  
GodotObject.MethodName.GetPropertyList , GodotObject.MethodName.GetMethodList ,  
GodotObject.MethodName.PropertyCanRevert ,

GodotObject.MethodName.PropertyGetRevert , GodotObject.MethodName.Notification ,  
GodotObject.MethodName.GetInstanceID , GodotObject.MethodName.SetScript ,  
GodotObject.MethodName.GetScript , GodotObject.MethodName.SetMeta ,  
GodotObject.MethodName.RemoveMeta , GodotObject.MethodName.GetMeta ,  
GodotObject.MethodName.HasMeta , GodotObject.MethodName.GetMetaList ,  
GodotObject.MethodName.AddUserSignal , GodotObject.MethodName.HasUserSignal ,  
GodotObject.MethodName.RemoveUserSignal , GodotObject.MethodName.EmitSignal ,  
GodotObject.MethodName.Call , GodotObject.MethodName.CallDeferred ,  
GodotObject.MethodName.SetDeferred , GodotObject.MethodName.Callv ,  
GodotObject.MethodName.HasMethod ,  
GodotObject.MethodName.GetMethodArgumentCount ,  
GodotObject.MethodName.HasSignal , GodotObject.MethodName.GetSignalList ,  
GodotObject.MethodName.GetSignalConnectionList ,  
GodotObject.MethodName.GetIncomingConnections , GodotObject.MethodName.Connect ,  
GodotObject.MethodName.Disconnect , GodotObject.MethodName.IsConnected ,  
GodotObject.MethodName.HasConnections , GodotObject.MethodName.SetBlockSignals ,  
GodotObject.MethodName.IsBlockingSignals ,  
GodotObject.MethodName.NotifyPropertyListChanged ,  
GodotObject.MethodName.SetMessageTranslation ,  
GodotObject.MethodName.CanTranslateMessages , GodotObject.MethodName.Tr ,  
GodotObject.MethodName.TrN , GodotObject.MethodName.GetTranslationDomain ,  
GodotObject.MethodName.SetTranslationDomain ,  
GodotObject.MethodName.IsQueuedForDeletion , GodotObject.MethodName.CancelFree ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class SettingItem.PropertyName

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

Cached StringNames for the properties and fields contained in this class, for fast lookup.

```
public class SettingItem.PropertyName : Resource.PropertyName
```

## Inheritance

[object](#) ← GodotObject(PropertyName) ← RefCounted(PropertyName) ← Resource(PropertyName) ← SettingItem(PropertyName)

## Inherited Members

Resource.PropertyName.ResourceLocalToScene , Resource.PropertyName.ResourcePath ,  
Resource.PropertyName.ResourceName , Resource.PropertyName.ResourceSceneUniqueId ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### \_fileName

Cached name for the '\_fileName' field.

```
public static readonly StringName _fileName
```

### Field Value

StringName

### assetPath

Cached name for the 'assetPath' field.

```
public static readonly StringName assetPath
```

## Field Value

StringName

## driveLink

Cached name for the 'driveLink' field.

```
public static readonly StringName driveLink
```

## Field Value

StringName

## fileName

Cached name for the 'fileName' property.

```
public static readonly StringName fileName
```

## Field Value

StringName

# Class SettingItem.SignalName

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

Cached StringNames for the signals contained in this class, for fast lookup.

```
public class SettingItem.SignalName : Resource.SignalName
```

## Inheritance

[object](#) ← GodotObject.SignalName ← RefCounted.SignalName ← Resource.SignalName ← SettingItem.SignalName

## Inherited Members

Resource.SignalName.Changed , Resource.SignalName.SetupLocalToSceneRequested ,  
GodotObject.SignalName.ScriptChanged , GodotObject.SignalName.PropertyListChanged ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Struct SheetFileHandler

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public struct SheetFileHandler : FileHandler
```

## Implements

[FileHandler](#)

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Methods

### DownloadAsync(FilesResource, DownloadRequest)

```
public Task<IDownloadProgress> DownloadAsync(FilesResource filesResource,  
FileHandler.DownloadRequest downloadRequest)
```

#### Parameters

filesResource [FilesResource](#)

downloadRequest [FileHandler.DownloadRequest](#)

#### Returns

[Task](#)<[IDownloadProgress](#)>

### DownloadWithStatus(FilesResource, in DownloadRequest)

```
public IDownloadProgress DownloadWithStatus(FilesResource filesResource, in  
FileHandler.DownloadRequest downloadRequest)
```

## Parameters

filesResource [FilesResource](#)

downloadRequest [FileHandler.DownloadRequest](#)

## Returns

[IDownloadProgress](#)

# Struct UrlInfo

Namespace: [cfGodotEngine.GoogleDrive](#)

Assembly: CatSweeper.dll

```
public struct UrlInfo
```

## Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [ValueType.ToString\(\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Fields

### fileId

```
public string fileId
```

#### Field Value

[string](#)

### fileType

```
public FileType fileType
```

#### Field Value

[FileType](#)

## Properties

### Empty

```
public static UrlInfo Empty { get; }
```

Property Value

[UrlInfo](#)

# Namespace cfGodotEngine.Info

## Classes

[ResourceInfoLoader<TInfo>](#)

[ResourceInfo<TInfo>](#)

[ResourceInfo<TInfo>.MethodName](#)

Cached StringNames for the methods contained in this class, for fast lookup.

[ResourceInfo<TInfo>.PropertyName](#)

Cached StringNames for the properties and fields contained in this class, for fast lookup.

[ResourceInfo<TInfo>.SignalName](#)

Cached StringNames for the signals contained in this class, for fast lookup.

# Class ResourceInfoLoader<TInfo>

Namespace: [cfGodotEngine.Info](#)

Assembly: CatSweeper.dll

```
public class ResourceInfoLoader<TInfo> : IValueLoader<TInfo>
```

## Type Parameters

**TInfo**

## Inheritance

[object](#) ← ResourceInfoLoader<TInfo>

## Implements

[IValueLoader](#)<TInfo>

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Constructors

### ResourceInfoLoader(string)

```
public ResourceInfoLoader(string resourcePath)
```

## Parameters

**resourcePath** [string](#)

## Methods

### Load(out List<TInfo>)

```
public ObjectPool<List<TInfo>>.Handle Load(out List<TInfo> values)
```

Parameters

values [List](#)<TInfo>

Returns

[ObjectPool](#)<List<TInfo>>.Handle

## LoadAsync(CancellationToken)

```
public Task<List<TInfo>> LoadAsync(CancellationToken cancellationToken)
```

Parameters

cancellationToken  [CancellationToken](#)

Returns

[Task](#)<List<TInfo>>

# Class ResourceInfo<TInfo>

Namespace: [cfGodotEngine.Info](#)

Assembly: CatSweeper.dll

```
[ScriptPath("res://Modules/cfGodotEngine/info/ResourceInfo.cs")]
public abstract class ResourceInfo<TInfo> : Resource, IDisposable
```

## Type Parameters

**TInfo**

## Inheritance

[object](#) ← GodotObject ← RefCounted ← Resource ← ResourceInfo<TInfo>

## Implements

[IDisposable](#)

## Inherited Members

Resource.\_GetRid() , Resource.\_ResetState() , [Resource.\\_SetPathCache\(string\)](#) ,  
Resource.\_SetupLocalToScene() , [Resource.TakeOverPath\(string\)](#) ,  
[Resource.SetPathCache\(string\)](#) , Resource.GetRid() , Resource.GetLocalScene() ,  
Resource.SetupLocalToScene() , Resource.ResetState() ,  
[Resource.SetIdForPath\(string, string\)](#) , [Resource.GetIdForPath\(string\)](#) ,  
Resource.IsBuiltIn() , Resource.GenerateSceneUniqueId() , Resource.EmitChanged() ,  
[Resource.Duplicate\(bool\)](#) , Resource.EmitSignalChanged() ,  
Resource.EmitSignalSetupLocalToSceneRequested() ,  
Resource.InvokeGodotClassMethod(in godot\_string\_name, NativeVariantPtrArgs, out godot\_variant) ,  
Resource.HasGodotClassMethod(in godot\_string\_name) ,  
Resource.HasGodotClassSignal(in godot\_string\_name) , Resource.ResourceLocalToScene ,  
Resource.ResourcePath , Resource.ResourceName , Resource.ResourceSceneUniqueId ,  
Resource.Changed , Resource.SetupLocalToSceneRequested , RefCounted.InitRef() ,  
RefCounted.Reference() , RefCounted.Unreference() , RefCounted.GetReferenceCount() ,  
GodotObject.NotificationPostinitialize , GodotObject.NotificationPredelete ,  
GodotObject.NotificationExtensionReloaded , [GodotObject.InstanceFromId\(ulong\)](#) ,  
[GodotObject.IsInstanceIdValid\(ulong\)](#) , GodotObject.IsInstanceIdValid(GodotObject) ,  
GodotObject.WeakRef(GodotObject) , GodotObject.Dispose() , [GodotObject.Dispose\(bool\)](#) ,  
GodotObject.ToString() , GodotObject.ToSignal(GodotObject, StringName) ,

GodotObject.\_Get(StringName) , GodotObject.\_GetPropertyList() ,  
GodotObject.\_IterGet(Variant) , GodotObject.\_IterInit(Array) , GodotObject.\_IterNext(Array) ,  
[GodotObject.\\_Notification\(int\)](#) , GodotObject.\_PropertyCanRevert(StringName) ,  
GodotObject.\_PropertyGetRevert(StringName) , GodotObject.\_Set(StringName, Variant) ,  
GodotObject.\_ValidateProperty(Dictionary) , GodotObject.Free() , GodotObject.GetClass() ,  
[GodotObject.IsClass\(string\)](#) , GodotObject.Set(StringName, Variant) ,  
GodotObject.Get(StringName) , GodotObject.SetIndexed(NodePath, Variant) ,  
GodotObject.GetIndexed(NodePath) , GodotObject.GetPropertyList() ,  
GodotObject.GetMethodList() , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , [GodotObject.Notification\(int, bool\)](#) ,  
GodotObject.GetInstanceId() , GodotObject.SetScript(Variant) , GodotObject.GetScript() ,  
GodotObject.SetMeta(StringName, Variant) , GodotObject.RemoveMeta(StringName) ,  
GodotObject.GetMeta(StringName, Variant) , GodotObject.HasMeta(StringName) ,  
GodotObject.GetMetaList() , [GodotObject.AddUserSignal\(string, Array\)](#) ,  
GodotObject.HasUserSignal(StringName) , GodotObject.RemoveUserSignal(StringName) ,  
GodotObject.EmitSignal(StringName, params Variant[]) ,  
[GodotObject.EmitSignal\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.Call(StringName, params Variant[]) ,  
[GodotObject.Call\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.CallDeferred(StringName, params Variant[]) ,  
[GodotObject.CallDeferred\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.SetDeferred(StringName, Variant) , GodotObject.Callv(StringName, Array) ,  
GodotObject.HasMethod(StringName) ,  
GodotObjectGetMethodArgumentCount(StringName) ,  
GodotObject.HasSignal(StringName) , GodotObject.GetSignalList() ,  
GodotObject.GetSignalConnectionList(StringName) ,  
GodotObject.GetIncomingConnections() ,  
[GodotObject.Connect\(StringName, Callable, uint\)](#) ,  
GodotObject.Disconnect(StringName, Callable) ,  
GodotObject.IsConnected(StringName, Callable) ,  
GodotObject.HasConnections(StringName) , [GodotObject.SetBlockSignals\(bool\)](#) ,  
GodotObject.IsBlockingSignals() , GodotObject.NotifyPropertyListChanged() ,  
[GodotObject.SetMessageTranslation\(bool\)](#) , GodotObject.CanTranslateMessages() ,  
GodotObject.Tr(StringName, StringName) ,  
[GodotObject.TrN\(StringName, StringName, int, StringName\)](#) ,  
GodotObject.GetTranslationDomain() , GodotObject.SetTranslationDomain(StringName) ,  
GodotObject.IsQueuedForDeletion() , GodotObject.CancelFree() ,  
GodotObject.EmitSignalScriptChanged() , GodotObject.EmitSignalPropertyListChanged() ,  
GodotObject.NativeInstance , GodotObject.ScriptChanged ,  
GodotObject.PropertyListChanged , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,

[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Properties

### GetInfos

```
public abstract IEnumerable<TInfo> GetInfos { get; }
```

Property Value

[IEnumerable](#)<TInfo>

# Class ResourceInfo<TInfo>.MethodName

Namespace: [cfGodotEngine.Info](#)

Assembly: CatSweeper.dll

Cached StringNames for the methods contained in this class, for fast lookup.

```
public class ResourceInfo<TInfo>.MethodName : Resource.MethodName
```

## Inheritance

```
object ← GodotObject.MethodName ← RefCounted.MethodName ←  
Resource.MethodName ← ResourceInfo<TInfo>.MethodName
```

## Inherited Members

```
Resource.MethodName._GetRid , Resource.MethodName._ResetState ,  
Resource.MethodName._SetPathCache , Resource.MethodName._SetupLocalToScene ,  
Resource.MethodName.SetPath , Resource.MethodName.TakeOverPath ,  
Resource.MethodName.GetPath , Resource.MethodName.SetPathCache ,  
Resource.MethodNameSetName , Resource.MethodName.GetName ,  
Resource.MethodName.GetRid , Resource.MethodName.SetLocalToScene ,  
Resource.MethodName.IsLocalToScene , Resource.MethodName.GetLocalScene ,  
Resource.MethodName.SetupLocalToScene , Resource.MethodName.ResetState ,  
Resource.MethodName.SetIdForPath , Resource.MethodName.GetIdForPath ,  
Resource.MethodName.IsBuiltIn , Resource.MethodName.GenerateSceneUniqueId ,  
Resource.MethodName.SetSceneUniqueId , Resource.MethodName.GetSceneUniqueId ,  
Resource.MethodName.EmitChanged , Resource.MethodName.Duplicate ,  
RefCounted.MethodName.InitRef , RefCounted.MethodName.Reference ,  
RefCounted.MethodName.Unreference , RefCounted.MethodName.GetReferenceCount ,  
GodotObject.MethodName._Get , GodotObject.MethodName._GetPropertyList ,  
GodotObject.MethodName._IterGet , GodotObject.MethodName._IterInit ,  
GodotObject.MethodName._IterNext , GodotObject.MethodName._Notification ,  
GodotObject.MethodName._PropertyCanRevert ,  
GodotObject.MethodName._PropertyGetRevert , GodotObject.MethodName._Set ,  
GodotObject.MethodName._ValidateProperty , GodotObject.MethodName.Free ,  
GodotObject.MethodName.GetClass , GodotObject.MethodName.IsClass ,  
GodotObject.MethodName.Set , GodotObject.MethodName.Get ,  
GodotObject.MethodName.SetIndexed , GodotObject.MethodName.GetIndexed ,  
GodotObject.MethodName.GetPropertyList , GodotObject.MethodName.GetMethodList ,  
GodotObject.MethodName.PropertyCanRevert ,
```

GodotObject.MethodName.PropertyGetRevert , GodotObject.MethodName.Notification ,  
GodotObject.MethodName.GetInstanceID , GodotObject.MethodName.SetScript ,  
GodotObject.MethodName.GetScript , GodotObject.MethodName.SetMeta ,  
GodotObject.MethodName.RemoveMeta , GodotObject.MethodName.GetMeta ,  
GodotObject.MethodName.HasMeta , GodotObject.MethodName.GetMetaList ,  
GodotObject.MethodName.AddUserSignal , GodotObject.MethodName.HasUserSignal ,  
GodotObject.MethodName.RemoveUserSignal , GodotObject.MethodName.EmitSignal ,  
GodotObject.MethodName.Call , GodotObject.MethodName.CallDeferred ,  
GodotObject.MethodName.SetDeferred , GodotObject.MethodName.Callv ,  
GodotObject.MethodName.HasMethod ,  
GodotObject.MethodName.GetMethodArgumentCount ,  
GodotObject.MethodName.HasSignal , GodotObject.MethodName.GetSignalList ,  
GodotObject.MethodName.GetSignalConnectionList ,  
GodotObject.MethodName.GetIncomingConnections , GodotObject.MethodName.Connect ,  
GodotObject.MethodName.Disconnect , GodotObject.MethodName.IsConnected ,  
GodotObject.MethodName.HasConnections , GodotObject.MethodName.SetBlockSignals ,  
GodotObject.MethodName.IsBlockingSignals ,  
GodotObject.MethodName.NotifyPropertyListChanged ,  
GodotObject.MethodName.SetMessageTranslation ,  
GodotObject.MethodName.CanTranslateMessages , GodotObject.MethodName.Tr ,  
GodotObject.MethodName.TrN , GodotObject.MethodName.GetTranslationDomain ,  
GodotObject.MethodName.SetTranslationDomain ,  
GodotObject.MethodName.IsQueuedForDeletion , GodotObject.MethodName.CancelFree ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class ResourceInfo<TInfo>.PropertyName

Namespace: [cfGodotEngine.Info](#)

Assembly: CatSweeper.dll

Cached StringNames for the properties and fields contained in this class, for fast lookup.

```
public class ResourceInfo<TInfo>.PropertyName : Resource.PropertyName
```

## Inheritance

[object](#) ← GodotObject(PropertyName) ← RefCounted(PropertyName) ← Resource(PropertyName) ← ResourceInfo<TInfo>.PropertyName

## Inherited Members

Resource.PropertyName.ResourceLocalToScene , Resource.PropertyName.ResourcePath ,  
Resource.PropertyName.ResourceName , Resource.PropertyName.ResourceSceneUniqueId ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class ResourceInfo<TInfo>.SignalName

Namespace: [cfGodotEngine.Info](#)

Assembly: CatSweeper.dll

Cached StringNames for the signals contained in this class, for fast lookup.

```
public class ResourceInfo<TInfo>.SignalName : Resource.SignalName
```

## Inheritance

[object](#) ← GodotObject.SignalName ← RefCounted.SignalName ← Resource.SignalName ← ResourceInfo<TInfo>.SignalName

## Inherited Members

Resource.SignalName.Changed , Resource.SignalName.SetupLocalToSceneRequested ,  
GodotObject.SignalName.ScriptChanged , GodotObject.SignalName.PropertyListChanged ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Namespace cfGodotEngine.Scene Management

## Classes

[GodotSceneManager](#)

[GodotSceneManager.MethodName](#)

Cached StringNames for the methods contained in this class, for fast lookup.

[GodotSceneManager.PropertyName](#)

Cached StringNames for the properties and fields contained in this class, for fast lookup.

[GodotSceneManager.SignalName](#)

Cached StringNames for the signals contained in this class, for fast lookup.

## Interfaces

[ISceneManager<TScene>](#)

## Enums

[LoadSceneMode](#)

# Class GodotSceneManager

Namespace: [cfGodotEngine.SceneManagement](#)

Assembly: CatSweeper.dll

```
[ScriptPath("res://Modules/cfGodotEngine/scene/GodotSceneManager.cs")]
public class GodotSceneManager : MonoInstance<GodotSceneManager>,
ISceneManager<Node>, IService, IDisposable
```

## Inheritance

[object](#) ← GodotObject ← Node ← [MonoInstance<GodotSceneManager>](#) ← GodotSceneManager

## Implements

[ISceneManager<Node>](#), [IService](#), [IDisposable](#)

## Inherited Members

[MonoInstance<GodotSceneManager>.Instance](#) , Node.NotificationEnterTree , Node.NotificationExitTree , Node.NotificationMovedInParent , Node.NotificationReady , Node.NotificationPaused , Node.NotificationUnpaused , Node.NotificationPhysicsProcess , Node.NotificationProcess , Node.NotificationParented , Node.NotificationUnparented , Node.NotificationSceneInstantiated , Node.NotificationDragBegin , Node.NotificationDragEnd , Node.NotificationPathRenamed , Node.NotificationChildOrderChanged , Node.NotificationInternalProcess , Node.NotificationInternalPhysicsProcess , Node.NotificationPostEnterTree , Node.NotificationDisabled , Node.NotificationEnabled , Node.NotificationResetPhysicsInterpolation , Node.NotificationEditorPreSave , Node.NotificationEditorPostSave , Node.NotificationWMMouseEnter , Node.NotificationWMMouseExit , Node.NotificationWMWindowFocusIn , Node.NotificationWMWindowFocusOut , Node.NotificationWMCloseRequest , Node.NotificationWMGoBackRequest , Node.NotificationWMSizeChanged , Node.NotificationWMDpiChange , Node.NotificationVpMouseEnter , Node.NotificationVpMouseExit , Node.NotificationWMPositionChanged , Node.NotificationOsMemoryWarning , Node.NotificationTranslationChanged , Node.NotificationWMAbout , Node.NotificationCrash , Node.NotificationOslmeUpdate , Node.NotificationApplicationResumed , Node.NotificationApplicationPaused , Node.NotificationApplicationFocusIn , Node.NotificationApplicationFocusOut , Node.NotificationTextServerChanged , Node.GetNode<T>(NodePath) , Node.GetNodeOrNull<T>(NodePath) , [Node.GetChild<T>\(int, bool\)](#) ,

[Node.GetChildOrNull<T>\(int, bool\)](#) , Node.GetOwner<T>() , Node.GetOwnerOrNull<T>() ,  
Node.GetParent<T>() , Node.GetParentOrNull<T>() , Node.\_EnterTree() , Node.\_ExitTree() ,  
Node.\_GetConfigurationWarnings() , Node.\_Input(InputEvent) ,  
[Node.\\_PhysicsProcess\(double\)](#) , Node.\_Ready() , Node.\_ShortcutInput(InputEvent) ,  
Node.\_UnhandledInput(InputEvent) , Node.\_UnhandledKeyInput(InputEvent) ,  
Node.PrintOrphanNodes() , [Node.AddSibling\(Node, bool\)](#) ,  
[Node.AddChild\(Node, bool, Node.InternalMode\)](#) , Node.RemoveChild(Node) ,  
[Node.Reparent\(Node, bool\)](#) , [Node.GetChildCount\(bool\)](#) , [Node.GetChildren\(bool\)](#) ,  
[Node.GetChild\(int, bool\)](#) , Node.HasNode(NodePath) , Node.GetNode(NodePath) ,  
Node.GetNodeOrNull(NodePath) , Node.GetParent() , [Node.FindChild\(string, bool, bool\)](#) ,  
[Node.FindChildren\(string, string, bool, bool\)](#) , [Node.FindParent\(string\)](#) ,  
Node.HasNodeAndResource(NodePath) , Node.GetNodeAndResource(NodePath) ,  
Node.IsInsideTree() , Node.IsPartOfEditedScene() , Node.IsAncestorOf(Node) ,  
Node.IsGreater Than(Node) , Node.GetPath() , [Node.GetPathTo\(Node, bool\)](#) ,  
[Node.AddToGroup\(StringName, bool\)](#) , Node.RemoveFromGroup(StringName) ,  
Node.IsInGroup(StringName) , [Node.MoveChild\(Node, int\)](#) , Node.GetGroups() ,  
[Node.GetIndex\(bool\)](#) , Node.PrintTree() , Node.PrintTreePretty() , Node.GetTreeString() ,  
Node.GetTreeStringPretty() , [Node.PropagateNotification\(int\)](#) ,  
[Node.PropagateCall\(StringName, Array, bool\)](#) , [Node.SetPhysicsProcess\(bool\)](#) ,  
Node.GetPhysicsProcessDeltaTime() , Node.IsPhysicsProcessing() ,  
Node.GetProcessDeltaTime() , [Node.SetProcess\(bool\)](#) , Node.IsProcessing() ,  
[Node.SetProcessInput\(bool\)](#) , Node.IsProcessingInput() ,  
[Node.SetProcessShortcutInput\(bool\)](#) , Node.IsProcessingShortcutInput() ,  
[Node.SetProcessUnhandledInput\(bool\)](#) , Node.IsProcessingUnhandledInput() ,  
[Node.SetProcessUnhandledKeyInput\(bool\)](#) , Node.IsProcessingUnhandledKeyInput() ,  
Node.CanProcess() , [Node.SetDisplayFolded\(bool\)](#) , Node.IsDisplayedFolded() ,  
[Node.SetProcessInternal\(bool\)](#) , Node.IsProcessingInternal() ,  
[Node.SetPhysicsProcessInternal\(bool\)](#) , Node.IsPhysicsProcessingInternal() ,  
Node.IsPhysicsInterpolated() , Node.IsPhysicsInterpolatedAndEnabled() ,  
Node.ResetPhysicsInterpolation() , Node.SetTranslationDomainInherited() ,  
Node.GetWindow() , Node.GetLastExclusiveWindow() , Node.GetTree() ,  
Node.CreateTween() , [Node.Duplicate\(int\)](#) , [Node.ReplaceBy\(Node, bool\)](#) ,  
[Node.setSceneInstanceLoadPlaceholder\(bool\)](#) ,  
Node.GetSceneInstanceLoadPlaceholder() , [Node.setEditableInstance\(Node, bool\)](#) ,  
Node.setEditableInstance(Node) , Node.GetViewport() , Node.QueueFree() ,  
Node.RequestReady() , Node.IsNodeReady() , [Node.SetMultiplayerAuthority\(int, bool\)](#) ,  
Node.GetMultiplayerAuthority() , Node.IsMultiplayerAuthority() ,  
Node.RpcConfig(StringName, Variant) , Node.GetRpcConfig() ,  
[Node.Atr\(string, StringName\)](#) , [Node.AtrN\(string, StringName, int, StringName\)](#) ,  
Node.Rpc(StringName, params Variant[]) ,

[Node.Rpc\(StringName, ReadOnlySpan<Variant>\)](#) ,  
[Node.RpcId\(long, StringName, params Variant\[\]\)](#) ,  
[Node.RpcId\(long, StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.UpdateConfigurationWarnings() ,  
Node.CallDeferredThreadGroup(StringName, params Variant[]) ,  
[Node.CallDeferredThreadGroup\(StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.SetDeferredThreadGroup(StringName, Variant) ,  
[Node.NotifyDeferredThreadGroup\(int\)](#) ,  
Node.CallThreadSafe(StringName, params Variant[]) ,  
[Node.CallThreadSafe\(StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.SetThreadSafe(StringName, Variant) , [Node.NotifyThreadSafe\(int\)](#) ,  
Node.EmitSignalReady() , Node.EmitSignalRenamed() , Node.EmitSignalTreeEntered() ,  
Node.EmitSignalTreeExiting() , Node.EmitSignalTreeExited() ,  
Node.EmitSignalChildEnteredTree(Node) , Node.EmitSignalChildExitingTree(Node) ,  
Node.EmitSignalChildOrderChanged() , Node.EmitSignalReplacingBy(Node) ,  
Node.EmitSignalEditorDescriptionChanged(Node) , Node.EmitSignalEditorStateChanged() ,  
Node.InvokeGodotClassMethod(in godot\_string\_name, NativeVariantPtrArgs, out godot\_variant) ,  
Node.HasGodotClassMethod(in godot\_string\_name) ,  
Node.HasGodotClassSignal(in godot\_string\_name) , Node.Name ,  
Node.UniqueNameInOwner , Node.SceneFilePath , Node.Owner , Node.Multiplayer ,  
Node.ProcessMode , Node.ProcessPriority , Node.ProcessPhysicsPriority ,  
Node.ProcessThreadGroup , Node.ProcessThreadGroupOrder ,  
Node.ProcessThreadMessages , Node.PhysicsInterpolationMode , Node.AutoTranslateMode ,  
Node.EditorDescription , Node.Ready , Node.Renamed , Node.TreeEntered ,  
Node.TreeExiting , Node.TreeExited , Node.ChildEnteredTree , Node.ChildExitingTree ,  
Node.ChildOrderChanged , Node.ReplacingBy , Node.EditorDescriptionChanged ,  
Node.EditorStateChanged , GodotObject.NotificationPostinitialize ,  
GodotObject.NotificationPredelete , GodotObject.NotificationExtensionReloaded ,  
[GodotObject.InstanceFromId\(ulong\)](#) , [GodotObject.IsInstanceIdValid\(ulong\)](#) ,  
GodotObject.IsValid(GodotObject) , GodotObject.WeakRef(GodotObject) ,  
GodotObject.Dispose() , [GodotObject.Dispose\(bool\)](#) , GodotObject.ToString() ,  
GodotObject.ToSignal(GodotObject, StringName) , GodotObject.\_Get(StringName) ,  
GodotObject.\_GetPropertyList() , GodotObject.\_IterGet(Variant) ,  
GodotObject.\_IterInit(Array) , GodotObject.\_IterNext(Array) ,  
[GodotObject.\\_Notification\(int\)](#) , GodotObject.\_PropertyCanRevert(StringName) ,  
GodotObject.\_PropertyGetRevert(StringName) , GodotObject.\_Set(StringName, Variant) ,  
GodotObject.\_ValidateProperty(Dictionary) , GodotObject.Free() , GodotObject.GetClass() ,  
[GodotObject.IsClass\(string\)](#) , GodotObject.Set(StringName, Variant) ,  
GodotObject.Get(StringName) , GodotObject.SetIndexed(NodePath, Variant) ,

GodotObject.GetIndexed(NodePath) , GodotObject.GetPropertyList() ,  
GodotObject.GetMethodList() , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , [GodotObject.Notification\(int, bool\)](#) ,  
GodotObject.GetInstanceId() , GodotObject.SetScript(Variant) , GodotObject.GetScript() ,  
GodotObject.SetMeta(StringName, Variant) , GodotObject.RemoveMeta(StringName) ,  
GodotObject.GetMeta(StringName, Variant) , GodotObject.HasMeta(StringName) ,  
GodotObject.GetMetaList() , [GodotObject.AddUserSignal\(string, Array\)](#) ,  
GodotObject.HasUserSignal(StringName) , GodotObject.RemoveUserSignal(StringName) ,  
GodotObject.EmitSignal(StringName, params Variant[]) ,  
[GodotObject.EmitSignal\(String Name, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.Call(StringName, params Variant[]) ,  
[GodotObject.Call\(String Name, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.CallDeferred(StringName, params Variant[]) ,  
[GodotObject.CallDeferred\(String Name, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.SetDeferred(StringName, Variant) , GodotObject.Callv(StringName, Array) ,  
GodotObject.HasMethod(StringName) ,  
GodotObjectGetMethodArgumentCount(StringName) ,  
GodotObject.HasSignal(StringName) , GodotObject.GetSignalList() ,  
GodotObject.GetSignalConnectionList(StringName) ,  
GodotObject.GetIncomingConnections() ,  
[GodotObject.Connect\(String Name, Callable, uint\)](#) ,  
GodotObject.Disconnect(StringName, Callable) ,  
GodotObject.IsConnected(StringName, Callable) ,  
GodotObject.HasConnections(StringName) , [GodotObject.SetBlockSignals\(bool\)](#) ,  
GodotObject.IsBlockingSignals() , GodotObject.NotifyPropertyListChanged() ,  
[GodotObject.SetMessageTranslation\(bool\)](#) , GodotObject.CanTranslateMessages() ,  
GodotObject.Tr(StringName, StringName) ,  
[GodotObject.TrN\(String Name, String Name, int, String Name\)](#) ,  
GodotObject.GetTranslationDomain() , GodotObject.SetTranslationDomain(StringName) ,  
GodotObject.IsQueuedForDeletion() , GodotObject.CancelFree() ,  
GodotObject.EmitSignalScriptChanged() , GodotObject.EmitSignalPropertyListChanged() ,  
GodotObject.NativeInstance , GodotObject.ScriptChanged ,  
GodotObject.PropertyListChanged , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Extension Methods

[NodeUtil.DontDestroyOnLoad\(Node\)](#)

# Constructors

## GodotSceneManager()

```
public GodotSceneManager()
```

# Methods

## GetScene(string)

```
public Node GetScene(string sceneName)
```

### Parameters

sceneName [string](#)

### Returns

Node

## LoadScene(string, LoadSceneMode)

```
public Node LoadScene(string sceneKey, LoadSceneMode mode = LoadSceneMode.Single)
```

### Parameters

sceneKey [string](#)

mode [LoadSceneMode](#)

### Returns

Node

# LoadSceneAsync(string, LoadSceneMode, IProgress<float>)

```
public Task<Node> LoadSceneAsync(string sceneKey, LoadSceneMode mode = LoadSceneMode.Single, IProgress<float> progress = null)
```

## Parameters

sceneKey [string](#)

mode [LoadSceneMode](#)

progress [IProgress<float>](#)

## Returns

[Task](#)<Node>

## \_Process(double)

Called during the processing step of the main loop. Processing happens at every frame and as fast as possible, so the `delta` time since the previous frame is not constant. `delta` is in seconds.

It is only called if processing is enabled, which is done automatically if this method is overridden, and can be toggled with [SetProcess\(bool\)](#).

Processing happens in order of Godot.Node.ProcessPriority, lower priority values are called first. Nodes with the same priority are processed in tree order, or top to bottom as seen in the editor (also known as pre-order traversal).

Corresponds to the Godot.Node.NotificationProcess notification in [Notification\(int\)](#).

**Note:** This method is only called if the node is present in the scene tree (i.e. if it's not an orphan).

**Note:** `delta` will be larger than expected if running at a framerate lower than Godot.Engine.PhysicsTicksPerSecond / Godot.Engine.MaxPhysicsStepsPerFrame FPS. This is done to avoid "spiral of death" scenarios where performance would plummet due to an ever-increasing number of physics steps per frame. This behavior affects both [\\_Process\(double\)](#) and [\\_PhysicsProcess\(double\)](#). As a result, avoid using `delta` for time measurements in real-

world seconds. Use the Godot.Time singleton's methods for this purpose instead, such as Godot.Time.GetTicksUsec().

```
public override void _Process(double delta)
```

## Parameters

**delta** double ↗

# Class GodotSceneManager.MethodName

Namespace: [cfGodotEngine.SceneManagement](#)

Assembly: CatSweeper.dll

Cached StringNames for the methods contained in this class, for fast lookup.

```
public class GodotSceneManager.MethodName :  
MonoInstance<GodotSceneManager>.MethodName
```

## Inheritance

[object](#) ← GodotObject.MethodName ← Node.MethodName ←  
[MonoInstance<GodotSceneManager>.MethodName](#) ← GodotSceneManager.MethodName

## Inherited Members

Node.MethodName.\_EnterTree , Node.MethodName.\_ExitTree ,  
Node.MethodName.\_GetConfigurationWarnings , Node.MethodName.\_Input ,  
Node.MethodName.\_PhysicsProcess , Node.MethodName.\_Ready ,  
Node.MethodName.\_ShortcutInput , Node.MethodName.\_UnhandledInput ,  
Node.MethodName.\_UnhandledKeyInput , Node.MethodName.PrintOrphanNodes ,  
Node.MethodName.AddSibling , Node.MethodName.SetName ,  
Node.MethodName.GetName , Node.MethodName.AddChild ,  
Node.MethodName.RemoveChild , Node.MethodName.Reparent ,  
Node.MethodName.GetChildCount , Node.MethodName.GetChildren ,  
Node.MethodName.GetChild , Node.MethodName.HasNode , Node.MethodName.GetNode ,  
Node.MethodName.GetNodeOrNull , Node.MethodName.GetParent ,  
Node.MethodName.FindChild , Node.MethodName.FindChildren ,  
Node.MethodName.FindParent , Node.MethodName.HasNodeAndResource ,  
Node.MethodName.GetNodeAndResource , Node.MethodName.IsInsideTree ,  
Node.MethodName.IsPartOfEditedScene , Node.MethodName.IsAncestorOf ,  
Node.MethodName.IsGreater Than , Node.MethodName.GetPath ,  
Node.MethodName.GetPathTo , Node.MethodName.AddToGroup ,  
Node.MethodName.RemoveFromGroup , Node.MethodName.IsInGroup ,  
Node.MethodName.MoveChild , Node.MethodName.GetGroups ,  
Node.MethodName.SetOwner , Node.MethodName.GetOwner ,  
Node.MethodName.GetIndex , Node.MethodName.PrintTree ,  
Node.MethodName.PrintTreePretty , Node.MethodName.GetTreeString ,  
Node.MethodName.GetTreeStringPretty , Node.MethodName.SetSceneFilePath ,  
Node.MethodName.GetSceneFilePath , Node.MethodName.PropagateNotification ,

Node.MethodName.PropagateCall , Node.MethodName.SetPhysicsProcess ,  
Node.MethodName.GetPhysicsProcessDeltaTime , Node.MethodName.IsPhysicsProcessing ,  
Node.MethodName.GetProcessDeltaTime , Node.MethodName.SetProcess ,  
Node.MethodName.SetProcessPriority , Node.MethodName.GetProcessPriority ,  
Node.MethodName.SetPhysicsProcessPriority ,  
Node.MethodName.GetPhysicsProcessPriority , Node.MethodName.IsProcessing ,  
Node.MethodName.SetProcessInput , Node.MethodName.IsProcessingInput ,  
Node.MethodName.SetProcessShortcutInput ,  
Node.MethodName.IsProcessingShortcutInput ,  
Node.MethodName.SetProcessUnhandledInput ,  
Node.MethodName.IsProcessingUnhandledInput ,  
Node.MethodName.SetProcessUnhandledKeyInput ,  
Node.MethodName.IsProcessingUnhandledKeyInput , Node.MethodName.SetProcessMode ,  
Node.MethodName.GetProcessMode , Node.MethodName.CanProcess ,  
Node.MethodName.SetProcessThreadGroup , Node.MethodName.GetProcessThreadGroup ,  
Node.MethodName.SetProcessThreadMessages ,  
Node.MethodName.GetProcessThreadMessages ,  
Node.MethodName.SetProcessThreadGroupOrder ,  
Node.MethodName.GetProcessThreadGroupOrder , Node.MethodName.SetDisplayFolded ,  
Node.MethodName.IsDisplayedFolded , Node.MethodName.SetProcessInternal ,  
Node.MethodName.IsProcessingInternal , Node.MethodName.SetPhysicsProcessInternal ,  
Node.MethodName.IsPhysicsProcessingInternal ,  
Node.MethodName.SetPhysicsInterpolationMode ,  
Node.MethodName.GetPhysicsInterpolationMode ,  
Node.MethodName.IsPhysicsInterpolated ,  
Node.MethodName.IsPhysicsInterpolatedAndEnabled ,  
Node.MethodName.ResetPhysicsInterpolation , Node.MethodName.SetAutoTranslateMode ,  
Node.MethodName.GetAutoTranslateMode ,  
Node.MethodName.SetTranslationDomainInherited , Node.MethodName.GetWindow ,  
Node.MethodName.GetLastExclusiveWindow , Node.MethodName.GetTree ,  
Node.MethodName.CreateTween , Node.MethodName.Duplicate ,  
Node.MethodName.ReplaceBy , Node.MethodName.SetSceneInstanceLoadPlaceholder ,  
Node.MethodName.GetSceneInstanceLoadPlaceholder ,  
Node.MethodName.SetEditableInstance , Node.MethodName.IsEditableInstance ,  
Node.MethodName.GetViewport , Node.MethodName.QueueFree ,  
Node.MethodName.RequestReady , Node.MethodName.IsNodeReady ,  
Node.MethodName.SetMultiplayerAuthority , Node.MethodName.GetMultiplayerAuthority ,  
Node.MethodName.IsMultiplayerAuthority , Node.MethodName.GetMultiplayer ,  
Node.MethodName.RpcConfig , Node.MethodName.GetRpcConfig ,  
Node.MethodName.SetEditorDescription , Node.MethodName.GetEditorDescription ,

Node.MethodName.\_SetImportPath , Node.MethodName.\_GetImportPath ,  
Node.MethodName.SetUniqueNameInOwner , Node.MethodName.IsUniqueNameInOwner ,  
Node.MethodName.Atr , Node.MethodName.AtrN , Node.MethodName.Rpc ,  
Node.MethodName.RpcId , Node.MethodName.UpdateConfigurationWarnings ,  
Node.MethodName.CallDeferredThreadGroup ,  
Node.MethodName.SetDeferredThreadGroup ,  
Node.MethodName.NotifyDeferredThreadGroup , Node.MethodName.CallThreadSafe ,  
Node.MethodName.SetThreadSafe , Node.MethodName.NotifyThreadSafe ,  
GodotObject.MethodName.\_Get , GodotObject.MethodName.\_GetPropertyList ,  
GodotObject.MethodName.\_IterGet , GodotObject.MethodName.\_IterInit ,  
GodotObject.MethodName.\_IterNext , GodotObject.MethodName.\_Notification ,  
GodotObject.MethodName.\_PropertyCanRevert ,  
GodotObject.MethodName.\_PropertyGetRevert , GodotObject.MethodName.\_Set ,  
GodotObject.MethodName.\_ValidateProperty , GodotObject.MethodName.Free ,  
GodotObject.MethodName.GetClass , GodotObject.MethodName.IsClass ,  
GodotObject.MethodName.Set , GodotObject.MethodName.Get ,  
GodotObject.MethodName.SetIndexed , GodotObject.MethodName.GetIndexed ,  
GodotObject.MethodName.GetPropertyList , GodotObject.MethodName.GetMethodList ,  
GodotObject.MethodName.PropertyCanRevert ,  
GodotObject.MethodName.PropertyGetRevert , GodotObject.MethodName.Notification ,  
GodotObject.MethodName.GetInstanceId , GodotObject.MethodName.SetScript ,  
GodotObject.MethodName.GetScript , GodotObject.MethodName.SetMeta ,  
GodotObject.MethodName.RemoveMeta , GodotObject.MethodName.GetMeta ,  
GodotObject.MethodName.HasMeta , GodotObject.MethodName.GetMetaList ,  
GodotObject.MethodName.AddUserSignal , GodotObject.MethodName.HasUserSignal ,  
GodotObject.MethodName.RemoveUserSignal , GodotObject.MethodName.EmitSignal ,  
GodotObject.MethodName.Call , GodotObject.MethodName.CallDeferred ,  
GodotObject.MethodName.SetDeferred , GodotObject.MethodName.Callv ,  
GodotObject.MethodName.HasMethod ,  
GodotObject.MethodName.GetMethodArgumentCount ,  
GodotObject.MethodName.HasSignal , GodotObject.MethodName.GetSignalList ,  
GodotObject.MethodName.GetSignalConnectionList ,  
GodotObject.MethodName.GetIncomingConnections , GodotObject.MethodName.Connect ,  
GodotObject.MethodName.Disconnect , GodotObject.MethodName.IsConnected ,  
GodotObject.MethodName.HasConnections , GodotObject.MethodName.SetBlockSignals ,  
GodotObject.MethodName.IsBlockingSignals ,  
GodotObject.MethodName.NotifyPropertyListChanged ,  
GodotObject.MethodName.SetMessageTranslation ,  
GodotObject.MethodName.CanTranslateMessages , GodotObject.MethodName.Tr ,  
GodotObject.MethodName.TrN , GodotObject.MethodName.GetTranslationDomain ,

GodotObject.MethodName.SetTranslationDomain ,  
GodotObject.MethodName.IsQueuedForDeletion , GodotObject.MethodName.CancelFree ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### GetScene

Cached name for the 'GetScene' method.

```
public static readonly StringName GetScene
```

### Field Value

StringName

### GetSceneTree

Cached name for the 'GetSceneTree' method.

```
public static readonly StringName GetSceneTree
```

### Field Value

StringName

### LoadScene

Cached name for the 'LoadScene' method.

```
public static readonly StringName LoadScene
```

### Field Value

StringName

## ShowScene

Cached name for the 'ShowScene' method.

```
public static readonly StringName ShowScene
```

Field Value

StringName

## \_Process

Cached name for the '\_Process' method.

```
public static readonly StringName _Process
```

Field Value

StringName

# Class GodotSceneManager.PropertyName

Namespace: [cfGodotEngine.SceneManagement](#)

Assembly: CatSweeper.dll

Cached StringNames for the properties and fields contained in this class, for fast lookup.

```
public class GodotSceneManager.PropertyName :  
MonoInstance<GodotSceneManager>.PropertyName
```

## Inheritance

[object](#) ← GodotObject.PropertyName ← Node.PropertyName ←  
[MonoInstance<GodotSceneManager>.PropertyName](#) ← GodotSceneManager.PropertyName

## Inherited Members

Node.PropertyName.\_ImportPath , Node.PropertyName.Name ,  
Node.PropertyName.UniqueNameInOwner , Node.PropertyName.SceneFilePath ,  
Node.PropertyName.Owner , Node.PropertyName.Multiplayer ,  
Node.PropertyName.ProcessMode , Node.PropertyName.ProcessPriority ,  
Node.PropertyName.ProcessPhysicsPriority , Node.PropertyName.ProcessThreadGroup ,  
Node.PropertyName.ProcessThreadGroupOrder ,  
Node.PropertyName.ProcessThreadMessages ,  
Node.PropertyName.PhysicsInterpolationMode , Node.PropertyName.AutoTranslateMode ,  
Node.PropertyName.EditorDescription , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Fields

### progressArray

Cached name for the 'progressArray' field.

```
public static readonly StringName progressArray
```

## Field Value

StringName



# Class GodotSceneManager.SignalName

Namespace: [cfGodotEngine.SceneManagement](#)

Assembly: CatSweeper.dll

Cached StringNames for the signals contained in this class, for fast lookup.

```
public class GodotSceneManager.SignalName :  
MonoInstance<GodotSceneManager>.SignalName
```

## Inheritance

[object](#) ← GodotObject.SignalName ← Node.SignalName ←  
[MonoInstance<GodotSceneManager>.SignalName](#) ← GodotSceneManager.SignalName

## Inherited Members

Node.SignalName.Ready , Node.SignalName.Renamed , Node.SignalName.TreeEntered ,  
Node.SignalName.TreeExiting , Node.SignalName.TreeExited ,  
Node.SignalName.ChildEnteredTree , Node.SignalName.ChildExitingTree ,  
Node.SignalName.ChildOrderChanged , Node.SignalName.ReplacingBy ,  
Node.SignalName.EditorDescriptionChanged , Node.SignalName.EditorStateChanged ,  
GodotObject.SignalName.ScriptChanged , GodotObject.SignalName.PropertyListChanged ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Interface ISceneManager<TScene>

Namespace: [cfGodotEngine.SceneManagement](#)

Assembly: CatSweeper.dll

```
public interface ISceneManager<TScene> : IService, IDisposable
```

## Type Parameters

TScene

### Inherited Members

[IDisposable.Dispose\(\)](#) ↗

## Methods

### GetScene(string)

```
TScene GetScene(string sceneName)
```

#### Parameters

sceneName [string](#) ↗

#### Returns

TScene

### LoadScene(string, LoadSceneMode)

```
TScene LoadScene(string sceneKey, LoadSceneMode mode = LoadSceneMode.Single)
```

#### Parameters

sceneKey [string](#) ↗

mode [LoadSceneMode](#)

Returns

TScene

**LoadSceneAsync(string, LoadSceneMode, IProgress<float>)**

```
Task<TScene> LoadSceneAsync(string sceneKey, LoadSceneMode mode =  
LoadSceneMode.Single, IProgress<float> progress = null)
```

Parameters

sceneKey [string](#)

mode [LoadSceneMode](#)

progress [IProgress](#)<[float](#)>

Returns

[Task](#)<TScene>

# Enum LoadSceneMode

Namespace: [cfGodotEngine.SceneManagement](#)

Assembly: CatSweeper.dll

```
public enum LoadSceneMode
```

## Extension Methods

[EnumExtension.hasFlag\(Enum, Enum\)](#)

## Fields

Additive = 1

Single = 0

# Namespace cfGodotEngine.Util

## Classes

[Application](#)

[MonoInstance<T>](#)

[MonoInstance<T>.MethodName](#)

Cached StringNames for the methods contained in this class, for fast lookup.

[MonoInstance<T>.PropertyName](#)

Cached StringNames for the properties and fields contained in this class, for fast lookup.

[MonoInstance<T>.SignalName](#)

Cached StringNames for the signals contained in this class, for fast lookup.

[NodeUtil](#)

[SettingPath](#)

[Setting<T>](#)

[Setting<T>.MethodName](#)

Cached StringNames for the methods contained in this class, for fast lookup.

[Setting<T>.PropertyName](#)

Cached StringNames for the properties and fields contained in this class, for fast lookup.

[Setting<T>.SignalName](#)

Cached StringNames for the signals contained in this class, for fast lookup.

[StateExecutionException<TStatId>](#)

[StateMachineNode<TStatId, TState, TStateMachine>](#)

[StateMachineNode<TStatId, TState, TStateMachine>.MethodName](#)

Cached StringNames for the methods contained in this class, for fast lookup.

[StateMachineNode<TStatId, TState, TStateMachine>.PropertyName](#)

Cached StringNames for the properties and fields contained in this class, for fast lookup.

[StateMachineNode<TStatId, TState, TStateMachine>.SignalName](#)

Cached StringNames for the signals contained in this class, for fast lookup.

[StateNode<TStatId, TState, TStateMachine>](#)

### [StateNode<TStatId, TState, TStateMachine>.MethodName](#)

Cached StringNames for the methods contained in this class, for fast lookup.

### [StateNode<TStatId, TState, TStateMachine>.PropertyName](#)

Cached StringNames for the properties and fields contained in this class, for fast lookup.

### [StateNode<TStatId, TState, TStateMachine>.SignalName](#)

Cached StringNames for the signals contained in this class, for fast lookup.

# Class Application

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

```
public static class Application
```

## Inheritance

[object](#) ← Application

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### assetDataPath

```
public static string assetDataPath { get; }
```

Property Value

[string](#)

### exportDataPath

```
public static string exportDataPath { get; }
```

Property Value

[string](#)

### persistentDataPath

```
public static string persistentDataPath { get; }
```

Property Value

[string](#)

## Methods

### GetGlobalizePath(string)

```
public static string GetGlobalizePath(string path)
```

Parameters

path [string](#)

Returns

[string](#)

# Class MonoInstance<T>

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

```
[ScriptPath("res://Modules/cfGodotEngine/util/MonoInstance.cs")]
public abstract class MonoInstance<T> : Node, IDisposable where T :
MonoInstance<T>, new()
```

## Type Parameters

T

### Inheritance

[object](#) ← GodotObject ← Node ← MonoInstance<T>

### Implements

[IDisposable](#)

### Derived

[AsyncResourceLoader](#), [GodotSceneManager](#)

### Inherited Members

Node.NotificationEnterTree , Node.NotificationExitTree , Node.NotificationMovedInParent ,  
Node.NotificationReady , Node.NotificationPaused , Node.NotificationUnpaused ,  
Node.NotificationPhysicsProcess , Node.NotificationProcess , Node.NotificationParented ,  
Node.NotificationUnparented , Node.NotificationSceneInstantiated ,  
Node.NotificationDragBegin , Node.NotificationDragEnd , Node.NotificationPathRenamed ,  
Node.NotificationChildOrderChanged , Node.NotificationInternalProcess ,  
Node.NotificationInternalPhysicsProcess , Node.NotificationPostEnterTree ,  
Node.NotificationDisabled , Node.NotificationEnabled ,  
Node.NotificationResetPhysicsInterpolation , Node.NotificationEditorPreSave ,  
Node.NotificationEditorPostSave , Node.NotificationWMMouseEnter ,  
Node.NotificationWMMouseExit , Node.NotificationWMWindowFocusIn ,  
Node.NotificationWMWindowFocusOut , Node.NotificationWMCloseRequest ,  
Node.NotificationWMGoBackRequest , Node.NotificationWMSizeChanged ,  
Node.NotificationWMDpiChange , Node.NotificationVpMouseEnter ,  
Node.NotificationVpMouseExit , Node.NotificationWMPositionChanged ,  
Node.NotificationOsMemoryWarning , Node.NotificationTranslationChanged ,  
Node.NotificationWMAbout , Node.NotificationCrash , Node.NotificationOslmeUpdate ,

Node.NotificationApplicationResumed , Node.NotificationApplicationPaused ,  
Node.NotificationApplicationFocusIn , Node.NotificationApplicationFocusOut ,  
Node.NotificationTextServerChanged , Node.GetNode<T>(NodePath) ,  
Node.GetNodeOrNull<T>(NodePath) , [Node.GetChild<T>\(int, bool\)](#) ,  
[Node.GetChildOrNull<T>\(int, bool\)](#) , Node.GetOwner<T>() , Node.GetOwnerOrNull<T>() ,  
Node.GetParent<T>() , Node.GetParentOrNull<T>() , Node.\_EnterTree() , Node.\_ExitTree() ,  
Node.\_GetConfigurationWarnings() , Node.\_Input(InputEvent) ,  
[Node.\\_PhysicsProcess\(double\)](#) , [Node.\\_Process\(double\)](#) , Node.\_Ready() ,  
Node.\_ShortcutInput(InputEvent) , Node.\_UnhandledInput(InputEvent) ,  
Node.\_UnhandledKeyInput(InputEvent) , Node.PrintOrphanNodes() ,  
[Node.AddSibling\(Node, bool\)](#) , [Node.AddChild\(Node, bool, Node.InternalMode\)](#) ,  
Node.RemoveChild(Node) , [Node.Reparent\(Node, bool\)](#) , [Node.GetChildCount\(bool\)](#) ,  
[Node.GetChildren\(bool\)](#) , [Node.GetChild\(int, bool\)](#) , Node.HasNode(NodePath) ,  
Node.GetNode(NodePath) , Node.GetNodeOrNull(NodePath) , Node.GetParent() ,  
[Node.FindChild\(string, bool, bool\)](#) , [Node.FindChildren\(string, string, bool, bool\)](#) ,  
[Node.FindParent\(string\)](#) , Node.HasNodeAndResource(NodePath) ,  
Node.GetNodeAndResource(NodePath) , Node.IsInsideTree() , Node.IsPartOfEditedScene() ,  
Node.IsAncestorOf(Node) , Node.IsGreaterThan(Node) , Node.GetPath() ,  
[Node.GetPathTo\(Node, bool\)](#) , [Node.AddToGroup\(StringName, bool\)](#) ,  
Node.RemoveFromGroup(StringName) , Node.IsInGroup(StringName) ,  
[Node.MoveChild\(Node, int\)](#) , Node.GetGroups() , [Node.GetIndex\(bool\)](#) , Node.PrintTree() ,  
Node.PrintTreePretty() , Node.GetTreeString() , Node.GetTreeStringPretty() ,  
[Node.PropagateNotification\(int\)](#) , [Node.PropagateCall\(StringName, Array, bool\)](#) ,  
[Node.SetPhysicsProcess\(bool\)](#) , Node.GetPhysicsProcessDeltaTime() ,  
Node.IsPhysicsProcessing() , Node.GetProcessDeltaTime() , [Node.SetProcess\(bool\)](#) ,  
Node.IsProcessing() , [Node.SetProcessInput\(bool\)](#) , Node.IsProcessingInput() ,  
[Node.SetProcessShortcutInput\(bool\)](#) , Node.IsProcessingShortcutInput() ,  
[Node.SetProcessUnhandledInput\(bool\)](#) , Node.IsProcessingUnhandledInput() ,  
[Node.SetProcessUnhandledKeyInput\(bool\)](#) , Node.IsProcessingUnhandledKeyInput() ,  
Node.CanProcess() , [Node.SetDisplayFolded\(bool\)](#) , Node.IsDisplayedFolded() ,  
[Node.SetProcessInternal\(bool\)](#) , Node.IsProcessingInternal() ,  
[Node.SetPhysicsProcessInternal\(bool\)](#) , Node.IsPhysicsProcessingInternal() ,  
Node.IsPhysicsInterpolated() , Node.IsPhysicsInterpolatedAndEnabled() ,  
Node.ResetPhysicsInterpolation() , Node.SetTranslationDomainInherited() ,  
Node.GetWindow() , Node.GetLastExclusiveWindow() , Node.GetTree() ,  
Node.CreateTween() , [Node.Duplicate\(int\)](#) , [Node.ReplaceBy\(Node, bool\)](#) ,  
[Node.setSceneInstanceLoadPlaceholder\(bool\)](#) ,  
Node.GetSceneInstanceLoadPlaceholder() , [Node.setEditableInstance\(Node, bool\)](#) ,  
Node.setEditableInstance(Node) , Node.GetViewport() , Node.QueueFree() ,  
Node.RequestReady() , Node.isNodeReady() , [Node.SetMultiplayerAuthority\(int, bool\)](#) ,

Node.GetMultiplayerAuthority() , Node.IsMultiplayerAuthority() ,  
Node.RpcConfig(StringName, Variant) , Node.GetRpcConfig() ,  
[Node.Atr\(string, StringName\)](#) , [Node.AtrN\(string, StringName, int, StringName\)](#) ,  
Node.Rpc(StringName, params Variant[]) ,  
[Node.Rpc\(StringName, ReadOnlySpan<Variant>\)](#) ,  
[Node.RpcId\(long, StringName, params Variant\[\]\)](#) ,  
[Node.RpcId\(long, StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.UpdateConfigurationWarnings() ,  
Node.CallDeferredThreadGroup(StringName, params Variant[]) ,  
[Node.CallDeferredThreadGroup\(StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.SetDeferredThreadGroup(StringName, Variant) ,  
[Node.NotifyDeferredThreadGroup\(int\)](#) ,  
Node.CallThreadSafe(StringName, params Variant[]) ,  
[Node.CallThreadSafe\(StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.SetThreadSafe(StringName, Variant) , [Node.NotifyThreadSafe\(int\)](#) ,  
Node.EmitSignalReady() , Node.EmitSignalRenamed() , Node.EmitSignalTreeEntered() ,  
Node.EmitSignalTreeExiting() , Node.EmitSignalTreeExited() ,  
Node.EmitSignalChildEnteredTree(Node) , Node.EmitSignalChildExitingTree(Node) ,  
Node.EmitSignalChildOrderChanged() , Node.EmitSignalReplacingBy(Node) ,  
Node.EmitSignalEditorDescriptionChanged(Node) , Node.EmitSignalEditorStateChanged() ,  
Node.InvokeGodotClassMethod(in godot\_string\_name, NativeVariantPtrArgs, out godot\_variant) ,  
Node.HasGodotClassMethod(in godot\_string\_name) ,  
Node.HasGodotClassSignal(in godot\_string\_name) , Node.Name ,  
Node.UniqueNameInOwner , Node.SceneFilePath , Node.Owner , Node.Multiplayer ,  
Node.ProcessMode , Node.ProcessPriority , Node.ProcessPhysicsPriority ,  
Node.ProcessThreadGroup , Node.ProcessThreadGroupOrder ,  
Node.ProcessThreadMessages , Node.PhysicsInterpolationMode , Node.AutoTranslateMode ,  
Node.EditorDescription , Node.Ready , Node.Renamed , Node.TreeEntered ,  
Node.TreeExiting , Node.TreeExited , Node.ChildEnteredTree , Node.ChildExitingTree ,  
Node.ChildOrderChanged , Node.ReplacingBy , Node.EditorDescriptionChanged ,  
Node.EditorStateChanged , GodotObject.NotificationPostInitialize ,  
GodotObject.NotificationPreDelete , GodotObject.NotificationExtensionReloaded ,  
[GodotObject.InstanceFromId\(ulong\)](#) , [GodotObject.IsInstanceIdValid\(ulong\)](#) ,  
GodotObject.IsInstanceIdValid(GodotObject) , GodotObject.WeakRef(GodotObject) ,  
GodotObject.Dispose() , [GodotObject.Dispose\(bool\)](#) , GodotObject.ToString() ,  
GodotObject.ToSignal(GodotObject, StringName) , GodotObject.\_Get(StringName) ,  
GodotObject.\_GetPropertyList() , GodotObject.\_IterGet(Variant) ,  
GodotObject.\_IterInit(Array) , GodotObject.\_IterNext(Array) ,  
[GodotObject.\\_Notification\(int\)](#) , GodotObject.\_PropertyCanRevert(StringName) ,

GodotObject.\_PropertyGetRevert(StringName) , GodotObject.\_Set(StringName, Variant) ,  
GodotObject.\_ValidateProperty(Dictionary) , GodotObject.Free() , GodotObject.GetClass() ,  
[GodotObject.IsClass\(string\)](#) , GodotObject.Set(StringName, Variant) ,  
GodotObject.Get(StringName) , GodotObject.SetIndexed(NodePath, Variant) ,  
GodotObject.GetIndexed(NodePath) , GodotObject.GetPropertyList() ,  
GodotObject.GetMethodList() , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , [GodotObject.Notification\(int, bool\)](#) ,  
GodotObject.GetInstanceId() , GodotObject.SetScript(Variant) , GodotObject.GetScript() ,  
GodotObject.SetMeta(StringName, Variant) , GodotObject.RemoveMeta(StringName) ,  
GodotObject.GetMeta(StringName, Variant) , GodotObject.HasMeta(StringName) ,  
GodotObject.GetMetaList() , [GodotObject.AddUserSignal\(string, Array\)](#) ,  
GodotObject.HasUserSignal(StringName) , GodotObject.RemoveUserSignal(StringName) ,  
GodotObject.EmitSignal(StringName, params Variant[]) ,  
[GodotObject.EmitSignal\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.Call(StringName, params Variant[]) ,  
[GodotObject.Call\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.CallDeferred(StringName, params Variant[]) ,  
[GodotObject.CallDeferred\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.SetDeferred(StringName, Variant) , GodotObject.Callv(StringName, Array) ,  
GodotObject.HasMethod(StringName) ,  
GodotObjectGetMethodArgumentCount(StringName) ,  
GodotObject.HasSignal(StringName) , GodotObject.GetSignalList() ,  
GodotObject.GetSignalConnectionList(StringName) ,  
GodotObject.GetIncomingConnections() ,  
[GodotObject.Connect\(StringName, Callable, uint\)](#) ,  
GodotObject.Disconnect(StringName, Callable) ,  
GodotObject.IsConnected(StringName, Callable) ,  
GodotObject.HasConnections(StringName) , [GodotObject.SetBlockSignals\(bool\)](#) ,  
GodotObject.IsBlockingSignals() , GodotObject.NotifyPropertyListChanged() ,  
[GodotObject.SetMessageTranslation\(bool\)](#) , GodotObject.CanTranslateMessages() ,  
GodotObject.Tr(StringName, StringName) ,  
[GodotObject.TrN\(StringName, StringName, int, StringName\)](#) ,  
GodotObject.GetTranslationDomain() , GodotObject.SetTranslationDomain(StringName) ,  
GodotObject.IsQueuedForDeletion() , GodotObject.CancelFree() ,  
GodotObject.EmitSignalScriptChanged() , GodotObject.EmitSignalPropertyListChanged() ,  
GodotObject.NativeInstance , GodotObject.ScriptChanged ,  
GodotObject.PropertyListChanged , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Extension Methods

[NodeUtil.DontDestroyOnLoad\(Node\)](#)

## Constructors

### MonoInstance()

```
protected MonoInstance()
```

## Properties

### Instance

```
public static T Instance { get; }
```

Property Value

T

# Class MonoInstance<T>.MethodName

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

Cached StringNames for the methods contained in this class, for fast lookup.

```
public class MonoInstance<T>.MethodName : Node.MethodName
```

## Inheritance

[object](#) ← GodotObject.MethodName ← Node.MethodName ← MonoInstance<T>.MethodName

## Derived

[AsyncResourceLoader.MethodName](#), [GodotSceneManager.MethodName](#)

## Inherited Members

Node.MethodName.\_EnterTree , Node.MethodName.\_ExitTree ,  
Node.MethodName.\_GetConfigurationWarnings , Node.MethodName.\_Input ,  
Node.MethodName.\_PhysicsProcess , Node.MethodName.\_Process ,  
Node.MethodName.\_Ready , Node.MethodName.\_ShortcutInput ,  
Node.MethodName.\_UnhandledInput , Node.MethodName.\_UnhandledKeyInput ,  
Node.MethodName.PrintOrphanNodes , Node.MethodName.AddSibling ,  
Node.MethodName.SetName , Node.MethodName.GetName , Node.MethodName.AddChild ,  
Node.MethodName.RemoveChild , Node.MethodName.Reparent ,  
Node.MethodName.GetChildCount , Node.MethodName.GetChildren ,  
Node.MethodName.GetChild , Node.MethodName.HasNode , Node.MethodName.GetNode ,  
Node.MethodName.GetNodeOrNull , Node.MethodName.GetParent ,  
Node.MethodName.FindChild , Node.MethodName.FindChildren ,  
Node.MethodName.FindParent , Node.MethodName.HasNodeAndResource ,  
Node.MethodName.GetNodeAndResource , Node.MethodName.IsInsideTree ,  
Node.MethodName.IsPartOfEditedScene , Node.MethodName.IsAncestorOf ,  
Node.MethodName.IsGreater Than , Node.MethodName.GetPath ,  
Node.MethodName.GetPathTo , Node.MethodName.AddToGroup ,  
Node.MethodName.RemoveFromGroup , Node.MethodName.IsInGroup ,  
Node.MethodName.MoveChild , Node.MethodName.GetGroups ,  
Node.MethodName.SetOwner , Node.MethodName.GetOwner ,  
Node.MethodName.GetIndex , Node.MethodName.PrintTree ,  
Node.MethodName.PrintTreePretty , Node.MethodName.GetTreeString ,

Node.MethodName.GetTreeStringPretty , Node.MethodName.SetSceneFilePath ,  
Node.MethodName.GetSceneFilePath , Node.MethodName.PropagateNotification ,  
Node.MethodName.PropagateCall , Node.MethodName.SetPhysicsProcess ,  
Node.MethodName.GetPhysicsProcessDeltaTime , Node.MethodName.IsPhysicsProcessing ,  
Node.MethodName.GetProcessDeltaTime , Node.MethodName.SetProcess ,  
Node.MethodName.SetProcessPriority , Node.MethodName.GetProcessPriority ,  
Node.MethodName.SetPhysicsProcessPriority ,  
Node.MethodName.GetPhysicsProcessPriority , Node.MethodName.IsProcessing ,  
Node.MethodName.SetProcessInput , Node.MethodName.IsProcessingInput ,  
Node.MethodName.SetProcessShortcutInput ,  
Node.MethodName.IsProcessingShortcutInput ,  
Node.MethodName.SetProcessUnhandledInput ,  
Node.MethodName.IsProcessingUnhandledInput ,  
Node.MethodName.SetProcessUnhandledKeyInput ,  
Node.MethodName.IsProcessingUnhandledKeyInput , Node.MethodName.SetProcessMode ,  
Node.MethodName.GetProcessMode , Node.MethodName.CanProcess ,  
Node.MethodName.SetProcessThreadGroup , Node.MethodName.GetProcessThreadGroup ,  
Node.MethodName.SetProcessThreadMessages ,  
Node.MethodName.GetProcessThreadMessages ,  
Node.MethodName.SetProcessThreadGroupOrder ,  
Node.MethodName.GetProcessThreadGroupOrder , Node.MethodName.SetDisplayFolded ,  
Node.MethodName.IsDisplayedFolded , Node.MethodName.SetProcessInternal ,  
Node.MethodName.IsProcessingInternal , Node.MethodName.SetPhysicsProcessInternal ,  
Node.MethodName.IsPhysicsProcessingInternal ,  
Node.MethodName.SetPhysicsInterpolationMode ,  
Node.MethodName.GetPhysicsInterpolationMode ,  
Node.MethodName.IsPhysicsInterpolated ,  
Node.MethodName.IsPhysicsInterpolatedAndEnabled ,  
Node.MethodName.ResetPhysicsInterpolation , Node.MethodName.SetAutoTranslateMode ,  
Node.MethodName.GetAutoTranslateMode ,  
Node.MethodName.SetTranslationDomainInherited , Node.MethodName.GetWindow ,  
Node.MethodName.GetLastExclusiveWindow , Node.MethodName.GetTree ,  
Node.MethodName.CreateTween , Node.MethodName.Duplicate ,  
Node.MethodName.ReplaceBy , Node.MethodName.SetSceneInstanceLoadPlaceholder ,  
Node.MethodName.GetSceneInstanceLoadPlaceholder ,  
Node.MethodName.SetEditableInstance , Node.MethodName.IsEditableInstance ,  
Node.MethodName.GetViewport , Node.MethodName.QueueFree ,  
Node.MethodName.RequestReady , Node.MethodName.IsNodeReady ,  
Node.MethodName.SetMultiplayerAuthority , Node.MethodName.GetMultiplayerAuthority ,  
Node.MethodName.IsMultiplayerAuthority , Node.MethodName.GetMultiplayer ,

Node.MethodName.RpcConfig , Node.MethodName.GetRpcConfig ,  
Node.MethodName.SetEditorDescription , Node.MethodName.GetEditorDescription ,  
Node.MethodName.\_SetImportPath , Node.MethodName.\_GetImportPath ,  
Node.MethodName.SetUniqueNameInOwner , Node.MethodName.IsUniqueNameInOwner ,  
Node.MethodName.Atr , Node.MethodName.AtrN , Node.MethodName.Rpc ,  
Node.MethodName.RpcId , Node.MethodName.UpdateConfigurationWarnings ,  
Node.MethodName.CallDeferredThreadGroup ,  
Node.MethodName.SetDeferredThreadGroup ,  
Node.MethodName.NotifyDeferredThreadGroup , Node.MethodName.CallThreadSafe ,  
Node.MethodName.SetThreadSafe , Node.MethodName.NotifyThreadSafe ,  
GodotObject.MethodName.\_Get , GodotObject.MethodName.\_GetPropertyList ,  
GodotObject.MethodName.\_IterGet , GodotObject.MethodName.\_IterInit ,  
GodotObject.MethodName.\_IterNext , GodotObject.MethodName.\_Notification ,  
GodotObject.MethodName.\_PropertyCanRevert ,  
GodotObject.MethodName.\_PropertyGetRevert , GodotObject.MethodName.\_Set ,  
GodotObject.MethodName.\_ValidateProperty , GodotObject.MethodName.Free ,  
GodotObject.MethodName.GetClass , GodotObject.MethodName.IsClass ,  
GodotObject.MethodName.Set , GodotObject.MethodName.Get ,  
GodotObject.MethodName.SetIndexed , GodotObject.MethodName.GetIndexed ,  
GodotObject.MethodName.GetPropertyList , GodotObject.MethodName.GetMethodList ,  
GodotObject.MethodName.PropertyCanRevert ,  
GodotObject.MethodName.PropertyGetRevert , GodotObject.MethodName.Notification ,  
GodotObject.MethodName.GetInstanceId , GodotObject.MethodName.SetScript ,  
GodotObject.MethodName.GetScript , GodotObject.MethodName.SetMeta ,  
GodotObject.MethodName.RemoveMeta , GodotObject.MethodName.GetMeta ,  
GodotObject.MethodName.HasMeta , GodotObject.MethodName.GetMetaList ,  
GodotObject.MethodName.AddUserSignal , GodotObject.MethodName.HasUserSignal ,  
GodotObject.MethodName.RemoveUserSignal , GodotObject.MethodName.EmitSignal ,  
GodotObject.MethodName.Call , GodotObject.MethodName.CallDeferred ,  
GodotObject.MethodName.SetDeferred , GodotObject.MethodName.Callv ,  
GodotObject.MethodName.HasMethod ,  
GodotObject.MethodName.GetMethodArgumentCount ,  
GodotObject.MethodName.HasSignal , GodotObject.MethodName.GetSignalList ,  
GodotObject.MethodName.GetSignalConnectionList ,  
GodotObject.MethodName.GetIncomingConnections , GodotObject.MethodName.Connect ,  
GodotObject.MethodName.Disconnect , GodotObject.MethodName.IsConnected ,  
GodotObject.MethodName.HasConnections , GodotObject.MethodName.SetBlockSignals ,  
GodotObject.MethodName.IsBlockingSignals ,  
GodotObject.MethodName.NotifyPropertyListChanged ,  
GodotObject.MethodName.SetMessageTranslation ,

GodotObject.MethodName.CanTranslateMessages , GodotObject.MethodName.Tr ,  
GodotObject.MethodName.TrN , GodotObject.MethodName.GetTranslationDomain ,  
GodotObject.MethodName.SetTranslationDomain ,  
GodotObject.MethodName.IsQueuedForDeletion , GodotObject.MethodName.CancelFree ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class MonoInstance<T>.PropertyName

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

Cached StringNames for the properties and fields contained in this class, for fast lookup.

```
public class MonoInstance<T>.PropertyName : Node.PropertyName
```

## Inheritance

[object](#) ← GodotObject.PropertyName ← Node.PropertyName ← MonoInstance<T>.PropertyName

## Derived

[AsyncResourceLoader.PropertyName](#), [GodotSceneManager.PropertyName](#)

## Inherited Members

Node.PropertyName.\_ImportPath , Node.PropertyName.Name ,  
Node.PropertyName.UniqueNameInOwner , Node.PropertyName.SceneFilePath ,  
Node.PropertyName.Owner , Node.PropertyName.Multiplayer ,  
Node.PropertyName.ProcessMode , Node.PropertyName.ProcessPriority ,  
Node.PropertyName.ProcessPhysicsPriority , Node.PropertyName.ProcessThreadGroup ,  
Node.PropertyName.ProcessThreadGroupOrder ,  
Node.PropertyName.ProcessThreadMessages ,  
Node.PropertyName.PhysicsInterpolationMode , Node.PropertyName.AutoTranslateMode ,  
Node.PropertyName.EditorDescription , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Class MonoInstance<T>.SignalName

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

Cached StringNames for the signals contained in this class, for fast lookup.

```
public class MonoInstance<T>.SignalName : Node.SignalName
```

## Inheritance

[object](#) ← GodotObject.SignalName ← Node.SignalName ← MonoInstance<T>.SignalName

## Derived

[AsyncResourceLoader.SignalName](#), [GodotSceneManager.SignalName](#)

## Inherited Members

Node.SignalName.Ready , Node.SignalName.Renamed , Node.SignalName.TreeEntered ,  
Node.SignalName.TreeExiting , Node.SignalName.TreeExited ,  
Node.SignalName.ChildEnteredTree , Node.SignalName.ChildExitingTree ,  
Node.SignalName.ChildOrderChanged , Node.SignalName.ReplacingBy ,  
Node.SignalName.EditorDescriptionChanged , Node.SignalName.EditorStateChanged ,  
GodotObject.SignalName.ScriptChanged , GodotObject.SignalName.PropertyListChanged ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class NodeUtil

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

```
public static class NodeUtil
```

## Inheritance

[object](#) ← NodeUtil

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### DontDestroyOnLoad(Node)

```
public static void DontDestroyOnLoad(this Node node)
```

#### Parameters

node Node

### GetSceneTree()

```
public static SceneTree GetSceneTree()
```

#### Returns

SceneTree

# Class SettingPath

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

```
public class SettingPath : Attribute
```

## Inheritance

[object](#) ← [Attribute](#) ← SettingPath

## Inherited Members

[Attribute.Equals\(object\)](#) , [Attribute.GetCustomAttribute\(Assembly, Type\)](#) ,  
[Attribute.GetCustomAttribute\(Assembly, Type, bool\)](#) ,  
[Attribute.GetCustomAttribute\(MemberInfo, Type\)](#) ,  
[Attribute.GetCustomAttribute\(MemberInfo, Type, bool\)](#) ,  
[Attribute.GetCustomAttribute\(Module, Type\)](#) ,  
[Attribute.GetCustomAttribute\(Module, Type, bool\)](#) ,  
[Attribute.GetCustomAttribute\(ParameterInfo, Type\)](#) ,  
[Attribute.GetCustomAttribute\(ParameterInfo, Type, bool\)](#) ,  
[Attribute.GetCustomAttributes\(Assembly\)](#) ,  
[Attribute.GetCustomAttributes\(Assembly, bool\)](#) ,  
[Attribute.GetCustomAttributes\(Assembly, Type\)](#) ,  
[Attribute.GetCustomAttributes\(Assembly, Type, bool\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo, bool\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo, Type\)](#) ,  
[Attribute.GetCustomAttributes\(MemberInfo, Type, bool\)](#) ,  
[Attribute.GetCustomAttributes\(Module\)](#) , [Attribute.GetCustomAttributes\(Module, bool\)](#) ,  
[Attribute.GetCustomAttributes\(Module, Type\)](#) ,  
[Attribute.GetCustomAttributes\(Module, Type, bool\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo, bool\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo, Type\)](#) ,  
[Attribute.GetCustomAttributes\(ParameterInfo, Type, bool\)](#) , [Attribute.GetHashCode\(\)](#) ,  
[Attribute.IsDefaultAttribute\(\)](#) , [Attribute.IsDefined\(Assembly, Type\)](#) ,  
[Attribute.IsDefined\(Assembly, Type, bool\)](#) , [Attribute.IsDefined\(MemberInfo, Type\)](#) ,  
[Attribute.IsDefined\(MemberInfo, Type, bool\)](#) , [Attribute.IsDefined\(Module, Type\)](#) ,  
[Attribute.IsDefined\(Module, Type, bool\)](#) , [Attribute.IsDefined\(ParameterInfo, Type\)](#) ,  
[Attribute.IsDefined\(ParameterInfo, Type, bool\)](#) , [Attribute.Match\(object\)](#) ,

[Attribute.TypeId](#) , [object.Equals\(object, object\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### SettingPath(string)

```
public SettingPath(string path)
```

## Parameters

path [string](#)

## Fields

### path

```
public readonly string path
```

## Field Value

[string](#)

# Class Setting<T>

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

```
[ScriptPath("res://Modules/cfGodotEngine/util/Setting.cs")]
public abstract class Setting<T> : Resource, IDisposable where T : Setting<T>
```

## Type Parameters

T

### Inheritance

[object](#) ← GodotObject ← RefCounted ← Resource ← Setting<T>

### Implements

[IDisposable](#)

### Derived

[DriveMirrorSetting](#)

### Inherited Members

Resource.\_GetRid() , Resource.\_ResetState() , [Resource.\\_SetPathCache\(string\)](#) ,  
Resource.\_SetupLocalToScene() , [Resource.TakeOverPath\(string\)](#) ,  
[Resource.SetPathCache\(string\)](#) , Resource.GetRid() , Resource.GetLocalScene() ,  
Resource.SetupLocalToScene() , Resource.ResetState() ,  
[Resource.SetIdForPath\(string, string\)](#) , [Resource.GetIdForPath\(string\)](#) ,  
Resource.IsBuiltIn() , Resource.GenerateSceneUniqueId() , Resource.EmitChanged() ,  
[Resource.Duplicate\(bool\)](#) , Resource.EmitSignalChanged() ,  
Resource.EmitSignalSetupLocalToSceneRequested() ,  
Resource.InvokeGodotClassMethod(in godot\_string\_name, NativeVariantPtrArgs, out godot\_variant) ,  
Resource.HasGodotClassMethod(in godot\_string\_name) ,  
Resource.HasGodotClassSignal(in godot\_string\_name) , Resource.ResourceLocalToScene ,  
Resource.ResourcePath , Resource.ResourceName , Resource.ResourceSceneUniqueId ,  
Resource.Changed , Resource.SetupLocalToSceneRequested , RefCounted.InitRef() ,  
RefCounted.Reference() , RefCounted.Unreference() , RefCounted.GetReferenceCount() ,  
GodotObject.NotificationPostinitialize , GodotObject.NotificationPredelete ,  
GodotObject.NotificationExtensionReloaded , [GodotObject.InstanceFromId\(ulong\)](#) ,  
[GodotObject.IsInstanceIdValid\(ulong\)](#) , GodotObject.IsInstanceValid(GodotObject) ,

GodotObject.WeakRef(GodotObject) , GodotObject.Dispose() , [GodotObject.Dispose\(bool\)](#) ,  
GodotObject.ToString() , GodotObject.ToSignal(GodotObject, StringName) ,  
GodotObject.\_Get(StringName) , GodotObject.\_GetPropertyList() ,  
GodotObject.\_IterGet(Variant) , GodotObject.\_IterInit(Array) , GodotObject.\_IterNext(Array) ,  
[GodotObject.Notification\(int\)](#) , GodotObject.\_PropertyCanRevert(StringName) ,  
GodotObject.\_PropertyGetRevert(StringName) , GodotObject.\_Set(StringName, Variant) ,  
GodotObject.\_ValidateProperty(Dictionary) , GodotObject.Free() , GodotObject.GetClass() ,  
[GodotObject.IsClass\(string\)](#) , GodotObject.Set(StringName, Variant) ,  
GodotObject.Get(StringName) , GodotObject.SetIndexed(NodePath, Variant) ,  
GodotObject.GetIndexed(NodePath) , GodotObject.GetPropertyList() ,  
GodotObject.GetMethodList() , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , [GodotObject.Notification\(int, bool\)](#) ,  
GodotObject.GetInstanceId() , GodotObject.SetScript(Variant) , GodotObject.GetScript() ,  
GodotObject.SetMeta(StringName, Variant) , GodotObject.RemoveMeta(StringName) ,  
GodotObject.GetMeta(StringName, Variant) , GodotObject.HasMeta(StringName) ,  
GodotObject.GetMetaList() , [GodotObject.AddUserSignal\(string, Array\)](#) ,  
GodotObject.HasUserSignal(StringName) , GodotObject.RemoveUserSignal(StringName) ,  
GodotObject.EmitSignal(StringName, params Variant[]) ,  
[GodotObject.EmitSignal\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.Call(StringName, params Variant[]) ,  
[GodotObject.Call\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.CallDeferred(StringName, params Variant[]) ,  
[GodotObject.CallDeferred\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.SetDeferred(StringName, Variant) , GodotObject.Callv(StringName, Array) ,  
GodotObject.HasMethod(StringName) ,  
GodotObject.GetMethodArgumentCount(StringName) ,  
GodotObject.HasSignal(StringName) , GodotObject.GetSignalList() ,  
GodotObject.GetSignalConnectionList(StringName) ,  
GodotObject.GetIncomingConnections() ,  
[GodotObject.Connect\(StringName, Callable, uint\)](#) ,  
GodotObject.Disconnect(StringName, Callable) ,  
GodotObject.IsConnected(StringName, Callable) ,  
GodotObject.HasConnections(StringName) , [GodotObject.SetBlockSignals\(bool\)](#) ,  
GodotObject.IsBlockingSignals() , GodotObject.NotifyPropertyListChanged() ,  
[GodotObject.SetMessageTranslation\(bool\)](#) , GodotObject.CanTranslateMessages() ,  
GodotObject.Tr(StringName, StringName) ,  
[GodotObject.TrN\(StringName, StringName, int, StringName\)](#) ,  
GodotObject.GetTranslationDomain() , GodotObject.SetTranslationDomain(StringName) ,  
GodotObject.IsQueuedForDeletion() , GodotObject.CancelFree() ,  
GodotObject.EmitSignalScriptChanged() , GodotObject.EmitSignalPropertyListChanged() ,

GodotObject.NativeInstance , GodotObject.ScriptChanged ,  
GodotObject.PropertyListChanged , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Methods

### GetSetting()

```
public static T GetSetting()
```

Returns

T

# Class Setting<T>.MethodName

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

Cached StringNames for the methods contained in this class, for fast lookup.

```
public class Setting<T>.MethodName : Resource.MethodName
```

## Inheritance

[object](#) ← GodotObject.MethodName ← RefCounted.MethodName ← Resource.MethodName ← Setting<T>.MethodName

## Derived

[DriveMirrorSetting.MethodName](#)

## Inherited Members

Resource.MethodName.\_GetRid , Resource.MethodName.\_ResetState ,  
Resource.MethodName.\_SetPathCache , Resource.MethodName.\_SetupLocalToScene ,  
Resource.MethodName.SetPath , Resource.MethodName.TakeOverPath ,  
Resource.MethodName.GetPath , Resource.MethodName.SetPathCache ,  
Resource.MethodNameSetName , Resource.MethodName.GetName ,  
Resource.MethodName.GetRid , Resource.MethodName.SetLocalToScene ,  
Resource.MethodName.IsLocalToScene , Resource.MethodName.GetLocalScene ,  
Resource.MethodName.SetupLocalToScene , Resource.MethodName.ResetState ,  
Resource.MethodName.SetIdForPath , Resource.MethodName.GetIdForPath ,  
Resource.MethodName.IsBuiltIn , Resource.MethodName.GenerateSceneUniqueId ,  
Resource.MethodName.SetSceneUniqueId , Resource.MethodName.GetSceneUniqueId ,  
Resource.MethodName.EmitChanged , Resource.MethodName.Duplicate ,  
RefCounted.MethodName.InitRef , RefCounted.MethodName.Reference ,  
RefCounted.MethodName.Unreference , RefCounted.MethodName.GetReferenceCount ,  
GodotObject.MethodName.\_Get , GodotObject.MethodName.\_GetPropertyList ,  
GodotObject.MethodName.\_IterGet , GodotObject.MethodName.\_IterInit ,  
GodotObject.MethodName.\_IterNext , GodotObject.MethodName.\_Notification ,  
GodotObject.MethodName.\_PropertyCanRevert ,  
GodotObject.MethodName.\_PropertyGetRevert , GodotObject.MethodName.\_Set ,  
GodotObject.MethodName.\_ValidateProperty , GodotObject.MethodName.Free ,  
GodotObject.MethodName.GetClass , GodotObject.MethodName.IsClass ,  
GodotObject.MethodName.Set , GodotObject.MethodName.Get ,

GodotObject.MethodName.SetIndexed , GodotObject.MethodName.GetIndexed ,  
GodotObject.MethodNameGetPropertyList , GodotObject.MethodName.GetMethodList ,  
GodotObject.MethodName.PropertyCanRevert ,  
GodotObject.MethodName.PropertyGetRevert , GodotObject.MethodName.Notification ,  
GodotObject.MethodName.GetInstanceID , GodotObject.MethodName.SetScript ,  
GodotObject.MethodName.GetScript , GodotObject.MethodName.SetMeta ,  
GodotObject.MethodName.RemoveMeta , GodotObject.MethodName.GetMeta ,  
GodotObject.MethodName.HasMeta , GodotObject.MethodName.GetMetaList ,  
GodotObject.MethodName.AddUserSignal , GodotObject.MethodName.HasUserSignal ,  
GodotObject.MethodName.RemoveUserSignal , GodotObject.MethodName.EmitSignal ,  
GodotObject.MethodName.Call , GodotObject.MethodName.CallDeferred ,  
GodotObject.MethodName.SetDeferred , GodotObject.MethodName.Callv ,  
GodotObject.MethodName.HasMethod ,  
GodotObject.MethodName.GetMethodArgumentCount ,  
GodotObject.MethodName.HasSignal , GodotObject.MethodName.GetSignalList ,  
GodotObject.MethodName.GetSignalConnectionList ,  
GodotObject.MethodName.GetIncomingConnections , GodotObject.MethodName.Connect ,  
GodotObject.MethodName.Disconnect , GodotObject.MethodName.IsConnected ,  
GodotObject.MethodName.HasConnections , GodotObject.MethodName.SetBlockSignals ,  
GodotObject.MethodName.IsBlockingSignals ,  
GodotObject.MethodName.NotifyPropertyListChanged ,  
GodotObject.MethodName.SetMessageTranslation ,  
GodotObject.MethodName.CanTranslateMessages , GodotObject.MethodName.Tr ,  
GodotObject.MethodName.TrN , GodotObject.MethodName.GetTranslationDomain ,  
GodotObject.MethodName.SetTranslationDomain ,  
GodotObject.MethodName.IsQueuedForDeletion , GodotObject.MethodName.CancelFree ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class Setting<T>.PropertyName

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

Cached StringNames for the properties and fields contained in this class, for fast lookup.

```
public class Setting<T>.PropertyName : Resource.PropertyName
```

## Inheritance

[object](#) ← GodotObject(PropertyName) ← RefCounted(PropertyName) ← Resource(PropertyName) ← Setting<T>.PropertyName

## Derived

[DriveMirrorSetting.PropertyName](#)

## Inherited Members

Resource.PropertyName.ResourceLocalToScene , Resource.PropertyName.ResourcePath ,  
Resource.PropertyName.ResourceName , Resource.PropertyName.ResourceSceneUniqueID ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class Setting<T>.SignalName

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

Cached StringNames for the signals contained in this class, for fast lookup.

```
public class Setting<T>.SignalName : Resource.SignalName
```

## Inheritance

[object](#) ← GodotObject.SignalName ← RefCounted.SignalName ← Resource.SignalName ← Setting<T>.SignalName

## Derived

[DriveMirrorSetting.SignalName](#)

## Inherited Members

Resource.SignalName.Changed , Resource.SignalName.SetupLocalToSceneRequested ,  
GodotObject.SignalName.ScriptChanged , GodotObject.SignalName.PropertyListChanged ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class StateExecutionException<TStatId>

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

```
public class StateExecutionException<TStateId> : Exception, ISerializable
```

## Type Parameters

TStateId

### Inheritance

[object](#) ↗ ← [Exception](#) ↗ ← StateExecutionException<TStatId>

### Implements

[ISerializable](#) ↗

### Inherited Members

[Exception.GetBaseException\(\)](#) ↗ , [Exception.GetType\(\)](#) ↗ , [Exception.ToString\(\)](#) ↗ ,  
[Exception.Data](#) ↗ , [Exception.HelpLink](#) ↗ , [Exception.HResult](#) ↗ , [Exception.InnerException](#) ↗ ,  
[Exception.Message](#) ↗ , [Exception.Source](#) ↗ , [Exception.StackTrace](#) ↗ , [Exception.TargetSite](#) ↗ ,  
[Exception.SerializeObjectState](#) ↗ , [object.Equals\(object\)](#) ↗ , [object.Equals\(object, object\)](#) ↗ ,  
[object.GetHashCode\(\)](#) ↗ , [object.MemberwiseClone\(\)](#) ↗ ,  
[object.ReferenceEquals\(object, object\)](#) ↗

## Constructors

### StateExecutionException(TStatId, Exception)

```
public StateExecutionException(TStateId stateId, Exception innerException)
```

## Parameters

stateId TStatId

innerException [Exception](#) ↗

# Class StateMachineNode<TStatId, TState, TStateMachine>

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

```
[ScriptPath("res://Modules/cfGodotEngine/util/StateMachineNode/StateMachineNode.cs")]
]
public abstract class StateMachineNode<TStateId, TState, TStateMachine> :
    Node, IDisposable, IStateMachine<TStateId> where TState : StateNode<TStateId,
    TState, TStateMachine> where TStateMachine : StateMachineNode<TStateId,
    TState, TStateMachine>
```

## Type Parameters

TStateId

TState

TStateMachine

## Inheritance

[object](#) ← GodotObject ← Node ← StateMachineNode<TStatId, TState, TStateMachine>

## Implements

[IDisposable](#), [IStateMachine](#)<TStatId>

## Inherited Members

Node.NotificationEnterTree , Node.NotificationExitTree , Node.NotificationMovedInParent ,  
Node.NotificationReady , Node.NotificationPaused , Node.NotificationUnpaused ,  
Node.NotificationPhysicsProcess , Node.NotificationProcess , Node.NotificationParented ,  
Node.NotificationUnparented , Node.NotificationSceneInstantiated ,  
Node.NotificationDragBegin , Node.NotificationDragEnd , Node.NotificationPathRenamed ,  
Node.NotificationChildOrderChanged , Node.NotificationInternalProcess ,  
Node.NotificationInternalPhysicsProcess , Node.NotificationPostEnterTree ,  
Node.NotificationDisabled , Node.NotificationEnabled ,  
Node.NotificationResetPhysicsInterpolation , Node.NotificationEditorPreSave ,  
Node.NotificationEditorPostSave , Node.NotificationWMMouseEnter ,  
Node.NotificationWMMouseExit , Node.NotificationWMWindowFocusIn ,  
Node.NotificationWMWindowFocusOut , Node.NotificationWMCloseRequest ,

Node.NotificationWMGoBackRequest , Node.NotificationWMSizeChanged ,  
Node.NotificationWMDpiChange , Node.NotificationVpMouseEnter ,  
Node.NotificationVpMouseExit , Node.NotificationWMPositionChanged ,  
Node.NotificationOsMemoryWarning , Node.NotificationTranslationChanged ,  
Node.NotificationWMAbout , Node.NotificationCrash , Node.NotificationOsImeUpdate ,  
Node.NotificationApplicationResumed , Node.NotificationApplicationPaused ,  
Node.NotificationApplicationFocusIn , Node.NotificationApplicationFocusOut ,  
Node.NotificationTextServerChanged , Node.GetNode<T>(NodePath) ,  
Node.GetNodeOrNull<T>(NodePath) , [Node.GetChild<T>\(int, bool\)](#) ,  
[Node.GetChildOrNull<T>\(int, bool\)](#) , Node.GetOwner<T>() , Node.GetOwnerOrNull<T>() ,  
Node.GetParent<T>() , Node.GetParentOrNull<T>() , Node.\_EnterTree() , Node.\_ExitTree() ,  
Node.\_GetConfigurationWarnings() , Node.\_Input(InputEvent) ,  
[Node.\\_PhysicsProcess\(double\)](#) , Node.\_ShortcutInput(InputEvent) ,  
Node.\_UnhandledInput(InputEvent) , Node.\_UnhandledKeyInput(InputEvent) ,  
Node.PrintOrphanNodes() , [Node.AddSibling\(Node, bool\)](#) ,  
[Node.AddChild\(Node, bool, Node.InternalMode\)](#) , Node.RemoveChild(Node) ,  
[Node.Reparent\(Node, bool\)](#) , [Node.GetChildCount\(bool\)](#) , [Node.GetChildren\(bool\)](#) ,  
[Node.GetChild\(int, bool\)](#) , Node.HasNode(NodePath) , Node.GetNode(NodePath) ,  
Node.GetNodeOrNull(NodePath) , Node.GetParent() , [Node.FindChild\(string, bool, bool\)](#) ,  
[Node.FindChildren\(string, string, bool, bool\)](#) , [Node.FindParent\(string\)](#) ,  
Node.HasNodeAndResource(NodePath) , Node.GetNodeAndResource(NodePath) ,  
Node.IsInsideTree() , Node.IsPartOfEditedScene() , Node.IsAncestorOf(Node) ,  
Node.IsGreater Than(Node) , Node.GetPath() , [Node.GetPathTo\(Node, bool\)](#) ,  
[Node.AddToGroup\(StringName, bool\)](#) , Node.RemoveFromGroup(StringName) ,  
Node.IsInGroup(StringName) , [Node.MoveChild\(Node, int\)](#) , Node.GetGroups() ,  
[Node.GetIndex\(bool\)](#) , Node.PrintTree() , Node.PrintTreePretty() , Node.GetTreeString() ,  
Node.GetTreeStringPretty() , [Node.PropagateNotification\(int\)](#) ,  
[Node.PropagateCall\(StringName, Array, bool\)](#) , [Node.SetPhysicsProcess\(bool\)](#) ,  
Node.GetPhysicsProcessDeltaTime() , Node.IsPhysicsProcessing() ,  
Node.GetProcessDeltaTime() , [Node.SetProcess\(bool\)](#) , Node.IsProcessing() ,  
[Node.SetProcessInput\(bool\)](#) , Node.IsProcessingInput() ,  
[Node.SetProcessShortcutInput\(bool\)](#) , Node.IsProcessingShortcutInput() ,  
[Node.SetProcessUnhandledInput\(bool\)](#) , Node.IsProcessingUnhandledInput() ,  
[Node.SetProcessUnhandledKeyInput\(bool\)](#) , Node.IsProcessingUnhandledKeyInput() ,  
Node.CanProcess() , [Node.SetDisplayFolded\(bool\)](#) , Node.IsDisplayedFolded() ,  
[Node.SetProcessInternal\(bool\)](#) , Node.IsProcessingInternal() ,  
[Node.SetPhysicsProcessInternal\(bool\)](#) , Node.IsPhysicsProcessingInternal() ,  
Node.IsPhysicsInterpolated() , Node.IsPhysicsInterpolatedAndEnabled() ,  
Node.ResetPhysicsInterpolation() , Node.SetTranslationDomainInherited() ,  
Node.GetWindow() , Node.GetLastExclusiveWindow() , Node.GetTree() ,

Node.CreateTween() , [Node.Duplicate\(int\)](#) , [Node.ReplaceBy\(Node, bool\)](#) ,  
[Node.SetSceneInstanceLoadPlaceholder\(bool\)](#) ,  
Node.GetSceneInstanceLoadPlaceholder() , [Node.setEditableInstance\(Node, bool\)](#) ,  
Node.IsEditableInstance(Node) , Node.GetViewport() , Node.QueueFree() ,  
Node.RequestReady() , Node.IsNodeReady() , [Node.SetMultiplayerAuthority\(int, bool\)](#) ,  
Node.GetMultiplayerAuthority() , Node.IsMultiplayerAuthority() ,  
Node.RpcConfig(StringName, Variant) , Node.GetRpcConfig() ,  
[Node.Atr\(string, StringName\)](#) , [Node.AtrN\(string, StringName, int, StringName\)](#) ,  
Node.Rpc(StringName, params Variant[]) ,  
[Node.Rpc\(StringName, ReadOnlySpan<Variant>\)](#) ,  
[Node.RpcId\(long, StringName, params Variant\[\]\)](#) ,  
[Node.RpcId\(long, StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.UpdateConfigurationWarnings() ,  
Node.CallDeferredThreadGroup(StringName, params Variant[]) ,  
[Node.CallDeferredThreadGroup\(StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.SetDeferredThreadGroup(StringName, Variant) ,  
[Node.NotifyDeferredThreadGroup\(int\)](#) ,  
Node.CallThreadSafe(StringName, params Variant[]) ,  
[Node.CallThreadSafe\(StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.SetThreadSafe(StringName, Variant) , [Node.NotifyThreadSafe\(int\)](#) ,  
Node.EmitSignalReady() , Node.EmitSignalRenamed() , Node.EmitSignalTreeEntered() ,  
Node.EmitSignalTreeExiting() , Node.EmitSignalTreeExited() ,  
Node.EmitSignalChildEnteredTree(Node) , Node.EmitSignalChildExitingTree(Node) ,  
Node.EmitSignalChildOrderChanged() , Node.EmitSignalReplacingBy(Node) ,  
Node.EmitSignalEditorDescriptionChanged(Node) , Node.EmitSignalEditorStateChanged() ,  
Node.InvokeGodotClassMethod(in godot\_string\_name, NativeVariantPtrArgs, out godot\_variant) ,  
Node.HasGodotClassMethod(in godot\_string\_name) ,  
Node.HasGodotClassSignal(in godot\_string\_name) , Node.Name ,  
Node.UniqueNameInOwner , Node.SceneFilePath , Node.Owner , Node.Multiplayer ,  
Node.ProcessMode , Node.ProcessPriority , Node.ProcessPhysicsPriority ,  
Node.ProcessThreadGroup , Node.ProcessThreadGroupOrder ,  
Node.ProcessThreadMessages , Node.PhysicsInterpolationMode , Node.AutoTranslateMode ,  
Node.EditorDescription , Node.Ready , Node.Renamed , Node.TreeEntered ,  
Node.TreeExiting , Node.TreeExited , Node.ChildEnteredTree , Node.ChildExitingTree ,  
Node.ChildOrderChanged , Node.ReplacingBy , Node.EditorDescriptionChanged ,  
Node.EditorStateChanged , GodotObject.NotificationPostinitialize ,  
GodotObject.NotificationPredelete , GodotObject.NotificationExtensionReloaded ,  
[GodotObject.InstanceFromId\(ulong\)](#) , [GodotObject.IsInstanceIdValid\(ulong\)](#) ,  
GodotObject.IsInstanceValid(GodotObject) , GodotObject.WeakRef(GodotObject) ,

GodotObject.Dispose() , [GodotObject.Dispose\(bool\)](#) , GodotObject.ToString() ,  
GodotObject.ToSignal(GodotObject, StringName) , GodotObject.\_Get(StringName) ,  
GodotObject.\_GetPropertyList() , GodotObject.\_IterGet(Variant) ,  
GodotObject.\_IterInit(Array) , GodotObject.\_IterNext(Array) ,  
[GodotObject.Notification\(int\)](#) , GodotObject.\_PropertyCanRevert(StringName) ,  
GodotObject.\_PropertyGetRevert(StringName) , GodotObject.\_Set(StringName, Variant) ,  
GodotObject.\_ValidateProperty(Dictionary) , GodotObject.Free() , GodotObject.GetClass() ,  
[GodotObject.IsClass\(string\)](#) , GodotObject.Set(StringName, Variant) ,  
GodotObject.Get(StringName) , GodotObject.SetIndexed(NodePath, Variant) ,  
GodotObject.GetIndexed(NodePath) , GodotObject.GetPropertyList() ,  
GodotObject.GetMethodList() , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , [GodotObject.Notification\(int, bool\)](#) ,  
GodotObject.GetInstanceId() , GodotObject.SetScript(Variant) , GodotObject.GetScript() ,  
GodotObject.SetMeta(StringName, Variant) , GodotObject.RemoveMeta(StringName) ,  
GodotObject.GetMeta(StringName, Variant) , GodotObject.HasMeta(StringName) ,  
GodotObject.GetMetaList() , [GodotObject.AddUserSignal\(string, Array\)](#) ,  
GodotObject.HasUserSignal(StringName) , GodotObject.RemoveUserSignal(StringName) ,  
GodotObject.EmitSignal(StringName, params Variant[]) ,  
[GodotObject.EmitSignal\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.Call(StringName, params Variant[]) ,  
[GodotObject.Call\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.CallDeferred(StringName, params Variant[]) ,  
[GodotObject.CallDeferred\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.SetDeferred(StringName, Variant) , GodotObject.Callv(StringName, Array) ,  
GodotObject.HasMethod(StringName) ,  
GodotObject.GetMethodArgumentCount(StringName) ,  
GodotObject.HasSignal(StringName) , GodotObject.GetSignalList() ,  
GodotObject.GetSignalConnectionList(StringName) ,  
GodotObject.GetIncomingConnections() ,  
[GodotObject.Connect\(StringName, Callable, uint\)](#) ,  
GodotObject.Disconnect(StringName, Callable) ,  
GodotObject.IsConnected(StringName, Callable) ,  
GodotObject.HasConnections(StringName) , [GodotObject.SetBlockSignals\(bool\)](#) ,  
GodotObject.IsBlockingSignals() , GodotObject.NotifyPropertyListChanged() ,  
[GodotObject.SetMessageTranslation\(bool\)](#) , GodotObject.CanTranslateMessages() ,  
GodotObject.Tr(StringName, StringName) ,  
[GodotObject.TrN\(StringName, StringName, int, StringName\)](#) ,  
GodotObject.GetTranslationDomain() , GodotObject.SetTranslationDomain(StringName) ,  
GodotObject.IsQueuedForDeletion() , GodotObject.CancelFree() ,  
GodotObject.EmitSignalScriptChanged() , GodotObject.EmitSignalPropertyListChanged() ,

```
GodotObject.NativeInstance , GodotObject.ScriptChanged ,  
GodotObject.PropertyListChanged , object.Equals\(object\) , object.Equals\(object, object\) ,  
object.GetHashCode\(\) , object.GetType\(\) , object.MemberwiseClone\(\) ,  
object.ReferenceEquals\(object, object\)
```

## Extension Methods

[NodeUtil.DontDestroyOnLoad\(Node\)](#)

# Properties

## allState

```
protected IEnumerable<TState> allState { get; }
```

### Property Value

[IEnumerable](#)<TState>

## currentState

```
protected TState currentState { get; }
```

### Property Value

TState

## currentStatId

```
public TStateId currentStateId { get; }
```

### Property Value

TStateId

## lastState

```
protected TState lastState { get; }
```

Property Value

TState

## lastStatId

```
public TStateId lastStateId { get; }
```

Property Value

TStatId

## Methods

### CanGoToState(TStatId, StateParam)

```
public bool CanGoToState(TStateId id, StateParam param)
```

Parameters

**id** TStatId

**param** [StateParam](#)

Returns

[bool](#)

### ForceGoToState(TStatId, StateParam)

```
public void ForceGoToState(TStateId nextStateId, StateParam param = null)
```

Parameters

`nextStateId` `TStatId`

`param` [StateParam](#)

## GetStateUnsafe(`TStatId`)

```
public TState GetStateUnsafe(TStateId id)
```

Parameters

`id` `TStatId`

Returns

`TState`

## GetStateUnsafe<`T`>(`TStatId`)

```
public T GetStateUnsafe<T>(TStateId id) where T : TState
```

Parameters

`id` `TStatId`

Returns

`T`

Type Parameters

`T`

## RegisterState(`TState`)

```
public void RegisterState(TState state)
```

Parameters

state TState

## SubscribeAfterStateChange(Action<StateChangeRecord<TStatId>>)

```
public Subscription SubscribeAfterStateChange(Action<StateChangeRecord<TStateId>> listener)
```

Parameters

listener [Action](#)<[StateChangeRecord](#)<TStatId>>

Returns

[Subscription](#)

## SubscribeBeforeStateChange(Action<StateChangeRecord<TStatId>>)

```
public Subscription SubscribeBeforeStateChange(Action<StateChangeRecord<TStateId>> listener)
```

Parameters

listener [Action](#)<[StateChangeRecord](#)<TStatId>>

Returns

[Subscription](#)

## TryGetState(TStatId, out TState)

```
public bool TryGetState(TStateId id, out TState monoState)
```

Parameters

**id** TStatId

**monoState** TState

Returns

[bool](#)

## TryGetState<T>(TStatId, out T)

```
public bool TryGetState<T>(TStateId id, out T state) where T : TState
```

Parameters

**id** TStatId

**state** T

Returns

[bool](#)

Type Parameters

**T**

## TryGoToState(TStatId, StateParam)

```
public bool TryGoToState(TStateId nextStateId, StateParam param = null)
```

Parameters

`nextStateId` `TStatId`

`param` [StateParam](#)

Returns

[bool](#)

## \_Process(double)

Called during the processing step of the main loop. Processing happens at every frame and as fast as possible, so the `delta` time since the previous frame is not constant. `delta` is in seconds.

It is only called if processing is enabled, which is done automatically if this method is overridden, and can be toggled with [SetProcess\(bool\)](#).

Processing happens in order of Godot.Node.ProcessPriority, lower priority values are called first. Nodes with the same priority are processed in tree order, or top to bottom as seen in the editor (also known as pre-order traversal).

Corresponds to the Godot.Node.NotificationProcess notification in [Notification\(int\)](#).

**Note:** This method is only called if the node is present in the scene tree (i.e. if it's not an orphan).

**Note:** `delta` will be larger than expected if running at a framerate lower than Godot.Engine.PhysicsTicksPerSecond / Godot.Engine.MaxPhysicsStepsPerFrame FPS. This is done to avoid "spiral of death" scenarios where performance would plummet due to an ever-increasing number of physics steps per frame. This behavior affects both [\\_Process\(double\)](#) and [\\_PhysicsProcess\(double\)](#). As a result, avoid using `delta` for time measurements in real-world seconds. Use the Godot.Time singleton's methods for this purpose instead, such as Godot.Time.GetTicksUsec().

```
public override void _Process(double delta)
```

Parameters

`delta` [double](#)

## \_Ready()

Called when the node is "ready", i.e. when both the node and its children have entered the scene tree. If the node has children, their Godot.Node.\_Ready() callbacks get triggered first, and the parent node will receive the ready notification afterwards.

Corresponds to the Godot.Node.NotificationReady notification in [\\_Notification\(int\)](#). See also the `@onready` annotation for variables.

Usually used for initialization. For even earlier initialization, Godot.GodotObject.GodotObject() may be used. See also Godot.Node.\_EnterTree().

**Note:** This method may be called only once for each node. After removing a node from the scene tree and adding it again, Godot.Node.\_Ready() will **not** be called a second time. This can be bypassed by requesting another call with Godot.Node.RequestReady(), which may be called anywhere before adding the node again.

```
public override void _Ready()
```

## \_Process(double)

```
protected virtual void __Process(double delta)
```

Parameters

`delta` [double](#)

## \_Ready()

```
protected virtual void __Ready()
```

# Class StateMachineNode<TStatId, TState, TStateMachine>.MethodName

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

Cached StringNames for the methods contained in this class, for fast lookup.

```
public class StateMachineNode<TStateId, TState, TStateMachine>.MethodName  
: Node.MethodName
```

## Inheritance

```
object ← GodotObject.MethodName ← Node.MethodName ←  
StateMachineNode<TStatId, TState, TStateMachine>.MethodName
```

## Inherited Members

```
Node.MethodName._EnterTree , Node.MethodName._ExitTree ,  
Node.MethodName._GetConfigurationWarnings , Node.MethodName._Input ,  
Node.MethodName._PhysicsProcess , Node.MethodName._ShortcutInput ,  
Node.MethodName._UnhandledInput , Node.MethodName._UnhandledKeyInput ,  
Node.MethodName.PrintOrphanNodes , Node.MethodName.AddSibling ,  
Node.MethodName.SetName , Node.MethodName.GetName , Node.MethodName.AddChild ,  
Node.MethodName.RemoveChild , Node.MethodName.Reparent ,  
Node.MethodName.GetChildCount , Node.MethodName.GetChildren ,  
Node.MethodName.GetChild , Node.MethodName.HasNode , Node.MethodName.GetNode ,  
Node.MethodName.GetNodeOrNull , Node.MethodName.GetParent ,  
Node.MethodName.FindChild , Node.MethodName.FindChildren ,  
Node.MethodName.FindParent , Node.MethodName.HasNodeAndResource ,  
Node.MethodName.GetNodeAndResource , Node.MethodName.IsInsideTree ,  
Node.MethodName.IsPartOfEditedScene , Node.MethodName.IsAncestorOf ,  
Node.MethodName.IsGreater Than , Node.MethodName.GetPath ,  
Node.MethodName.GetPathTo , Node.MethodName.AddToGroup ,  
Node.MethodName.RemoveFromGroup , Node.MethodName.IsInGroup ,  
Node.MethodName.MoveChild , Node.MethodName.GetGroups ,  
Node.MethodName.SetOwner , Node.MethodName.GetOwner ,  
Node.MethodName.GetIndex , Node.MethodName.PrintTree ,  
Node.MethodName.PrintTreePretty , Node.MethodName.GetTreeString ,  
Node.MethodName.GetTreeStringPretty , Node.MethodName.SetSceneFilePath ,  
Node.MethodName.GetSceneFilePath , Node.MethodName.PropagateNotification ,
```

Node.MethodName.PropagateCall , Node.MethodName.SetPhysicsProcess ,  
Node.MethodName.GetPhysicsProcessDeltaTime , Node.MethodName.IsPhysicsProcessing ,  
Node.MethodName.GetProcessDeltaTime , Node.MethodName.SetProcess ,  
Node.MethodName.SetProcessPriority , Node.MethodName.GetProcessPriority ,  
Node.MethodName.SetPhysicsProcessPriority ,  
Node.MethodName.GetPhysicsProcessPriority , Node.MethodName.IsProcessing ,  
Node.MethodName.SetProcessInput , Node.MethodName.IsProcessingInput ,  
Node.MethodName.SetProcessShortcutInput ,  
Node.MethodName.IsProcessingShortcutInput ,  
Node.MethodName.SetProcessUnhandledInput ,  
Node.MethodName.IsProcessingUnhandledInput ,  
Node.MethodName.SetProcessUnhandledKeyInput ,  
Node.MethodName.IsProcessingUnhandledKeyInput , Node.MethodName.SetProcessMode ,  
Node.MethodName.GetProcessMode , Node.MethodName.CanProcess ,  
Node.MethodName.SetProcessThreadGroup , Node.MethodName.GetProcessThreadGroup ,  
Node.MethodName.SetProcessThreadMessages ,  
Node.MethodName.GetProcessThreadMessages ,  
Node.MethodName.SetProcessThreadGroupOrder ,  
Node.MethodName.GetProcessThreadGroupOrder , Node.MethodName.SetDisplayFolded ,  
Node.MethodName.IsDisplayedFolded , Node.MethodName.SetProcessInternal ,  
Node.MethodName.IsProcessingInternal , Node.MethodName.SetPhysicsProcessInternal ,  
Node.MethodName.IsPhysicsProcessingInternal ,  
Node.MethodName.SetPhysicsInterpolationMode ,  
Node.MethodName.GetPhysicsInterpolationMode ,  
Node.MethodName.IsPhysicsInterpolated ,  
Node.MethodName.IsPhysicsInterpolatedAndEnabled ,  
Node.MethodName.ResetPhysicsInterpolation , Node.MethodName.SetAutoTranslateMode ,  
Node.MethodName.GetAutoTranslateMode ,  
Node.MethodName.SetTranslationDomainInherited , Node.MethodName.GetWindow ,  
Node.MethodName.GetLastExclusiveWindow , Node.MethodName.GetTree ,  
Node.MethodName.CreateTween , Node.MethodName.Duplicate ,  
Node.MethodName.ReplaceBy , Node.MethodName.SetSceneInstanceLoadPlaceholder ,  
Node.MethodName.GetSceneInstanceLoadPlaceholder ,  
Node.MethodName.SetEditableInstance , Node.MethodName.IsEditableInstance ,  
Node.MethodName.GetViewport , Node.MethodName.QueueFree ,  
Node.MethodName.RequestReady , Node.MethodName.IsNodeReady ,  
Node.MethodName.SetMultiplayerAuthority , Node.MethodName.GetMultiplayerAuthority ,  
Node.MethodName.IsMultiplayerAuthority , Node.MethodName.GetMultiplayer ,  
Node.MethodName.RpcConfig , Node.MethodName.GetRpcConfig ,  
Node.MethodName.SetEditorDescription , Node.MethodName.GetEditorDescription ,

Node.MethodName.\_SetImportPath , Node.MethodName.\_GetImportPath ,  
Node.MethodName.SetUniqueNameInOwner , Node.MethodName.IsUniqueNameInOwner ,  
Node.MethodName.Atr , Node.MethodName.AtrN , Node.MethodName.Rpc ,  
Node.MethodName.RpcId , Node.MethodName.UpdateConfigurationWarnings ,  
Node.MethodName.CallDeferredThreadGroup ,  
Node.MethodName.SetDeferredThreadGroup ,  
Node.MethodName.NotifyDeferredThreadGroup , Node.MethodName.CallThreadSafe ,  
Node.MethodName.SetThreadSafe , Node.MethodName.NotifyThreadSafe ,  
GodotObject.MethodName.\_Get , GodotObject.MethodName.\_GetPropertyList ,  
GodotObject.MethodName.\_IterGet , GodotObject.MethodName.\_IterInit ,  
GodotObject.MethodName.\_IterNext , GodotObject.MethodName.\_Notification ,  
GodotObject.MethodName.\_PropertyCanRevert ,  
GodotObject.MethodName.\_PropertyGetRevert , GodotObject.MethodName.\_Set ,  
GodotObject.MethodName.\_ValidateProperty , GodotObject.MethodName.Free ,  
GodotObject.MethodName.GetClass , GodotObject.MethodName.IsClass ,  
GodotObject.MethodName.Set , GodotObject.MethodName.Get ,  
GodotObject.MethodName.SetIndexed , GodotObject.MethodName.GetIndexed ,  
GodotObject.MethodName.GetPropertyList , GodotObject.MethodName.GetMethodList ,  
GodotObject.MethodName.PropertyCanRevert ,  
GodotObject.MethodName.PropertyGetRevert , GodotObject.MethodName.Notification ,  
GodotObject.MethodName.GetInstanceId , GodotObject.MethodName.SetScript ,  
GodotObject.MethodName.GetScript , GodotObject.MethodName.SetMeta ,  
GodotObject.MethodName.RemoveMeta , GodotObject.MethodName.GetMeta ,  
GodotObject.MethodName.HasMeta , GodotObject.MethodName.GetMetaList ,  
GodotObject.MethodName.AddUserSignal , GodotObject.MethodName.HasUserSignal ,  
GodotObject.MethodName.RemoveUserSignal , GodotObject.MethodName.EmitSignal ,  
GodotObject.MethodName.Call , GodotObject.MethodName.CallDeferred ,  
GodotObject.MethodName.SetDeferred , GodotObject.MethodName.Callv ,  
GodotObject.MethodName.HasMethod ,  
GodotObject.MethodName.GetMethodArgumentCount ,  
GodotObject.MethodName.HasSignal , GodotObject.MethodName.GetSignalList ,  
GodotObject.MethodName.GetSignalConnectionList ,  
GodotObject.MethodName.GetIncomingConnections , GodotObject.MethodName.Connect ,  
GodotObject.MethodName.Disconnect , GodotObject.MethodName.IsConnected ,  
GodotObject.MethodName.HasConnections , GodotObject.MethodName.SetBlockSignals ,  
GodotObject.MethodName.IsBlockingSignals ,  
GodotObject.MethodName.NotifyPropertyChanged ,  
GodotObject.MethodName.SetMessageTranslation ,  
GodotObject.MethodName.CanTranslateMessages , GodotObject.MethodName.Tr ,  
GodotObject.MethodName.TrN , GodotObject.MethodName.GetTranslationDomain ,

GodotObject.MethodName.SetTranslationDomain ,  
GodotObject.MethodName.IsQueuedForDeletion , GodotObject.MethodName.CancelFree ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### \_Process

Cached name for the '\_Process' method.

```
public static readonly StringName _Process
```

### Field Value

StringName

### \_Ready

Cached name for the '\_Ready' method.

```
public static readonly StringName _Ready
```

### Field Value

StringName

### \_\_Process

Cached name for the '\_\_Process' method.

```
public static readonly StringName __Process
```

### Field Value

StringName

## \_\_Ready

Cached name for the '\_\_Ready' method.

```
public static readonly StringName __Ready
```

Field Value

StringName

# Class StateMachineNode<TStatId, TState, TStateMachine>.PropertyName

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

Cached StringNames for the properties and fields contained in this class, for fast lookup.

```
public class StateMachineNode<TStateId, TState, TStateMachine>.PropertyName  
: Node.PropertyName
```

## Inheritance

[object](#) ← GodotObject(PropertyName) ← Node(PropertyName) ← StateMachineNode<TStatId, TState, TStateMachine>.PropertyName

## Inherited Members

Node(PropertyName.\_ImportPath , Node(PropertyName.Name ,  
Node(PropertyName.UniqueNameInOwner , Node(PropertyName.SceneFilePath ,  
Node(PropertyName.Owner , Node(PropertyName.Multiplayer ,  
Node(PropertyName.ProcessMode , Node(PropertyName.ProcessPriority ,  
Node(PropertyName.ProcessPhysicsPriority , Node(PropertyName.ProcessThreadGroup ,  
Node(PropertyName.ProcessThreadGroupOrder ,  
Node(PropertyName.ProcessThreadMessages ,  
Node(PropertyName.PhysicsInterpolationMode , Node(PropertyName.AutoTranslateMode ,  
Node(PropertyName.EditorDescription , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Class StateMachineNode<TStatId, TState, TStateMachine>.SignalName

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

Cached StringNames for the signals contained in this class, for fast lookup.

```
public class StateMachineNode<TStateId, TState, TStateMachine>.SignalName  
: Node.SignalName
```

## Inheritance

[object](#) ← GodotObject.SignalName ← Node.SignalName ← StateMachineNode<TStatId, TState, TStateMachine>.SignalName

## Inherited Members

Node.SignalName.Ready , Node.SignalName.Renamed , Node.SignalName.TreeEntered ,  
Node.SignalName.TreeExiting , Node.SignalName.TreeExited ,  
Node.SignalName.ChildEnteredTree , Node.SignalName.ChildExitingTree ,  
Node.SignalName.ChildOrderChanged , Node.SignalName.ReplacingBy ,  
Node.SignalName.EditorDescriptionChanged , Node.SignalName.EditorStateChanged ,  
GodotObject.SignalName.ScriptChanged , GodotObject.SignalName.PropertyListChanged ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

# Class StateNode<TStatId, TState, TStateMachine>

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

```
[ScriptPath("res://Modules/cfGodotEngine/util/StateMachineNode/StateNode.cs")]
public abstract class StateNode<TStateId, TState, TStateMachine> : Node, IDisposable
where TState : StateNode<TStateId, TState, TStateMachine> where TStateMachine :
StateMachineNode<TStateId, TState, TStateMachine>
```

## Type Parameters

TStateId

TState

TStateMachine

## Inheritance

[object](#) ← GodotObject ← Node ← StateNode<TStatId, TState, TStateMachine>

## Implements

[IDisposable](#)

## Inherited Members

Node.NotificationEnterTree , Node.NotificationExitTree , Node.NotificationMovedInParent ,  
Node.NotificationReady , Node.NotificationPaused , Node.NotificationUnpaused ,  
Node.NotificationPhysicsProcess , Node.NotificationProcess , Node.NotificationParented ,  
Node.NotificationUnparented , Node.NotificationSceneInstantiated ,  
Node.NotificationDragBegin , Node.NotificationDragEnd , Node.NotificationPathRenamed ,  
Node.NotificationChildOrderChanged , Node.NotificationInternalProcess ,  
Node.NotificationInternalPhysicsProcess , Node.NotificationPostEnterTree ,  
Node.NotificationDisabled , Node.NotificationEnabled ,  
Node.NotificationResetPhysicsInterpolation , Node.NotificationEditorPreSave ,  
Node.NotificationEditorPostSave , Node.NotificationWMMouseEnter ,  
Node.NotificationWMMouseExit , Node.NotificationWMWindowFocusIn ,  
Node.NotificationWMWindowFocusOut , Node.NotificationWMCloseRequest ,  
Node.NotificationWMGoBackRequest , Node.NotificationWMSizeChanged ,

Node.NotificationWMDpiChange , Node.NotificationVpMouseEnter ,  
Node.NotificationVpMouseExit , Node.NotificationWMPositionChanged ,  
Node.NotificationOsMemoryWarning , Node.NotificationTranslationChanged ,  
Node.NotificationWMAbout , Node.NotificationCrash , Node.NotificationOslmeUpdate ,  
Node.NotificationApplicationResumed , Node.NotificationApplicationPaused ,  
Node.NotificationApplicationFocusIn , Node.NotificationApplicationFocusOut ,  
Node.NotificationTextServerChanged , Node.GetNode<T>(NodePath) ,  
Node.GetNodeOrNull<T>(NodePath) , [Node.GetChild<T>\(int, bool\)](#) ,  
[Node.GetChildOrNull<T>\(int, bool\)](#) , Node.GetOwner<T>() , Node.GetOwnerOrNull<T>() ,  
Node.GetParent<T>() , Node.GetParentOrNull<T>() , Node.\_EnterTree() , Node.\_ExitTree() ,  
Node.\_GetConfigurationWarnings() , Node.\_Input(InputEvent) ,  
[Node.\\_PhysicsProcess\(double\)](#) , Node.\_ShortcutInput(InputEvent) ,  
Node.\_UnhandledInput(InputEvent) , Node.\_UnhandledKeyInput(InputEvent) ,  
Node.PrintOrphanNodes() , [Node.AddSibling\(Node, bool\)](#) ,  
[Node.AddChild\(Node, bool, Node.InternalMode\)](#) , Node.RemoveChild(Node) ,  
[Node.Reparent\(Node, bool\)](#) , [Node.GetChildCount\(bool\)](#) , [Node.GetChildren\(bool\)](#) ,  
[Node.GetChild\(int, bool\)](#) , Node.HasNode(NodePath) , Node.GetNode(NodePath) ,  
Node.GetNodeOrNull(NodePath) , Node.GetParent() , [Node.FindChild\(string, bool, bool\)](#) ,  
[Node.FindChildren\(string, string, bool, bool\)](#) , [Node.FindParent\(string\)](#) ,  
Node.HasNodeAndResource(NodePath) , Node.GetNodeAndResource(NodePath) ,  
Node.IsInsideTree() , Node.IsPartOfEditedScene() , Node.IsAncestorOf(Node) ,  
Node.IsGreaterThan(Node) , Node.GetPath() , [Node.GetPathTo\(Node, bool\)](#) ,  
[Node.AddToGroup\(StringName, bool\)](#) , Node.RemoveFromGroup(StringName) ,  
Node.IsInGroup(StringName) , [Node.MoveChild\(Node, int\)](#) , Node.GetGroups() ,  
[Node.GetIndex\(bool\)](#) , Node.PrintTree() , Node.PrintTreePretty() , Node.GetTreeString() ,  
Node.GetTreeStringPretty() , [Node.PropagateNotification\(int\)](#) ,  
[Node.PropagateCall\(StringName, Array, bool\)](#) , [Node.SetPhysicsProcess\(bool\)](#) ,  
Node.GetPhysicsProcessDeltaTime() , Node.IsPhysicsProcessing() ,  
Node.GetProcessDeltaTime() , [Node.SetProcess\(bool\)](#) , Node.IsProcessing() ,  
[Node.SetProcessInput\(bool\)](#) , Node.IsProcessingInput() ,  
[Node.SetProcessShortcutInput\(bool\)](#) , Node.IsProcessingShortcutInput() ,  
[Node.SetProcessUnhandledInput\(bool\)](#) , Node.IsProcessingUnhandledInput() ,  
[Node.SetProcessUnhandledKeyInput\(bool\)](#) , Node.IsProcessingUnhandledKeyInput() ,  
Node.CanProcess() , [Node.SetDisplayFolded\(bool\)](#) , Node.IsDisplayedFolded() ,  
[Node.SetProcessInternal\(bool\)](#) , Node.IsProcessingInternal() ,  
[Node.SetPhysicsProcessInternal\(bool\)](#) , Node.IsPhysicsProcessingInternal() ,  
Node.IsPhysicsInterpolated() , Node.IsPhysicsInterpolatedAndEnabled() ,  
Node.ResetPhysicsInterpolation() , Node.SetTranslationDomainInherited() ,  
Node.GetWindow() , Node.GetLastExclusiveWindow() , Node.GetTree() ,  
Node.CreateTween() , [Node.Duplicate\(int\)](#) , [Node.ReplaceBy\(Node, bool\)](#) ,

[Node.SetSceneInstanceLoadPlaceholder\(bool\)](#) ,  
Node.GetSceneInstanceLoadPlaceholder() , [Node.SetEditableInstance\(Node, bool\)](#) ,  
Node.IsEditableInstance(Node) , Node.GetViewport() , Node.QueueFree() ,  
Node.RequestReady() , Node.IsNodeReady() , [Node.SetMultiplayerAuthority\(int, bool\)](#) ,  
Node.GetMultiplayerAuthority() , Node.IsMultiplayerAuthority() ,  
Node.RpcConfig(StringName, Variant) , Node.GetRpcConfig() ,  
[Node.Atr\(string, StringName\)](#) , [Node.AtrN\(string, StringName, int, StringName\)](#) ,  
Node.Rpc(StringName, params Variant[]) ,  
[Node.Rpc\(StringName, ReadOnlySpan<Variant>\)](#) ,  
[Node.RpcId\(long, StringName, params Variant\[\]\)](#) ,  
[Node.RpcId\(long, StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.UpdateConfigurationWarnings() ,  
Node.CallDeferredThreadGroup(StringName, params Variant[]) ,  
[Node.CallDeferredThreadGroup\(StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.SetDeferredThreadGroup(StringName, Variant) ,  
[Node.NotifyDeferredThreadGroup\(int\)](#) ,  
Node.CallThreadSafe(StringName, params Variant[]) ,  
[Node.CallThreadSafe\(StringName, ReadOnlySpan<Variant>\)](#) ,  
Node.SetThreadSafe(StringName, Variant) , [Node.NotifyThreadSafe\(int\)](#) ,  
Node.EmitSignalReady() , Node.EmitSignalRenamed() , Node.EmitSignalTreeEntered() ,  
Node.EmitSignalTreeExiting() , Node.EmitSignalTreeExited() ,  
Node.EmitSignalChildEnteredTree(Node) , Node.EmitSignalChildExitingTree(Node) ,  
Node.EmitSignalChildOrderChanged() , Node.EmitSignalReplacingBy(Node) ,  
Node.EmitSignalEditorDescriptionChanged(Node) , Node.EmitSignalEditorStateChanged() ,  
Node.InvokeGodotClassMethod(in godot\_string\_name, NativeVariantPtrArgs, out godot\_variant) ,  
Node.HasGodotClassMethod(in godot\_string\_name) ,  
Node.HasGodotClassSignal(in godot\_string\_name) , Node.Name ,  
Node.UniqueNameInOwner , Node.SceneFilePath , Node.Owner , Node.Multiplayer ,  
Node.ProcessMode , Node.ProcessPriority , Node.ProcessPhysicsPriority ,  
Node.ProcessThreadGroup , Node.ProcessThreadGroupOrder ,  
Node.ProcessThreadMessages , Node.PhysicsInterpolationMode , Node.AutoTranslateMode ,  
Node.EditorDescription , Node.Ready , Node.Renamed , Node.TreeEntered ,  
Node.TreeExiting , Node.TreeExited , Node.ChildEnteredTree , Node.ChildExitingTree ,  
Node.ChildOrderChanged , Node.ReplacingBy , Node.EditorDescriptionChanged ,  
Node.EditorStateChanged , GodotObject.NotificationPostinitialize ,  
GodotObject.NotificationPredelete , GodotObject.NotificationExtensionReloaded ,  
[GodotObject.InstanceFromId\(ulong\)](#) , [GodotObject.IsInstanceIdValid\(ulong\)](#) ,  
GodotObject.IsInstanceIdValid(GodotObject) , GodotObject.WeakRef(GodotObject) ,  
GodotObject.Dispose() , [GodotObject.Dispose\(bool\)](#) , GodotObject.ToString()

GodotObject.ToSignal(GodotObject, StringName) , GodotObject.\_Get(StringName) ,  
GodotObject.\_GetPropertyList() , GodotObject.\_IterGet(Variant) ,  
GodotObject.\_IterInit(Array) , GodotObject.\_IterNext(Array) ,  
[GodotObject.Notification\(int\)](#) , GodotObject.\_PropertyCanRevert(StringName) ,  
GodotObject.\_PropertyGetRevert(StringName) , GodotObject.\_Set(StringName, Variant) ,  
GodotObject.\_ValidateProperty(Dictionary) , GodotObject.Free() , GodotObject.GetClass() ,  
[GodotObject.IsClass\(string\)](#) , GodotObject.Set(StringName, Variant) ,  
GodotObject.Get(StringName) , GodotObject.SetIndexed(NodePath, Variant) ,  
GodotObject.GetIndexed(NodePath) , GodotObject.GetPropertyList() ,  
GodotObject.GetMethodList() , GodotObject.PropertyCanRevert(StringName) ,  
GodotObject.PropertyGetRevert(StringName) , [GodotObject.Notification\(int, bool\)](#) ,  
GodotObject.GetInstanceId() , GodotObject.SetScript(Variant) , GodotObject.GetScript() ,  
GodotObject.SetMeta(StringName, Variant) , GodotObject.RemoveMeta(StringName) ,  
GodotObject.GetMeta(StringName, Variant) , GodotObject.HasMeta(StringName) ,  
GodotObject.GetMetaList() , [GodotObject.AddUserSignal\(string, Array\)](#) ,  
GodotObject.HasUserSignal(StringName) , GodotObject.RemoveUserSignal(StringName) ,  
GodotObject.EmitSignal(StringName, params Variant[]) ,  
[GodotObject.EmitSignal\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.Call(StringName, params Variant[]) ,  
[GodotObject.Call\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.CallDeferred(StringName, params Variant[]) ,  
[GodotObject.CallDeferred\(StringName, ReadOnlySpan<Variant>\)](#) ,  
GodotObject.SetDeferred(StringName, Variant) , GodotObject.Callv(StringName, Array) ,  
GodotObject.HasMethod(StringName) ,  
GodotObject.GetMethodArgumentCount(StringName) ,  
GodotObject.HasSignal(StringName) , GodotObject.GetSignalList() ,  
GodotObject.GetSignalConnectionList(StringName) ,  
GodotObject.GetIncomingConnections() ,  
[GodotObject.Connect\(StringName, Callable, uint\)](#) ,  
GodotObject.Disconnect(StringName, Callable) ,  
GodotObject.IsConnected(StringName, Callable) ,  
GodotObject.HasConnections(StringName) , [GodotObject.SetBlockSignals\(bool\)](#) ,  
GodotObject.IsBlockingSignals() , GodotObject.NotifyPropertyChanged() ,  
[GodotObject.SetMessageTranslation\(bool\)](#) , GodotObject.CanTranslateMessages() ,  
GodotObject.Tr(StringName, StringName) ,  
[GodotObject.TrN\(StringName, StringName, int, StringName\)](#) ,  
GodotObject.GetTranslationDomain() , GodotObject.SetTranslationDomain(StringName) ,  
GodotObject.IsQueuedForDeletion() , GodotObject.CancelFree() ,  
GodotObject.EmitSignalScriptChanged() , GodotObject.EmitSignalPropertyChanged() ,  
GodotObject.NativeInstance , GodotObject.ScriptChanged ,

GodotObject.PropertyListChanged , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Extension Methods

[NodeUtil.DontDestroyOnLoad\(Node\)](#)

# Properties

## Id

```
public abstract TStateId Id { get; }
```

## Property Value

TStateId

## stateMachine

```
protected TStateMachine stateMachine { get; }
```

## Property Value

TStateMachine

# Methods

## CanUpdate()

```
public virtual bool CanUpdate()
```

## Returns

[bool](#)

## IsReady(StateParam)

```
public virtual bool IsReady(StateParam param)
```

Parameters

param [StateParam](#)

Returns

[bool](#)

## SetStateMachine(TStateMachine)

```
public void SetStateMachine(TStateMachine stateMachine)
```

Parameters

stateMachine [TStateMachine](#)

## \_OnEndContext()

```
protected virtual void _OnEndContext()
```

## \_Process(double)

```
public virtual void _Process(double delta)
```

Parameters

delta [double](#)

## \_Ready()

```
public virtual void _Ready()  
  
_StartContext(StateParam)  
  
protected abstract void _StartContext(StateParam param)
```

## Parameters

param [StateParam](#)

# Class StateNode<TStatId, TState, TStateMachine>.MethodName

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

Cached StringNames for the methods contained in this class, for fast lookup.

```
public class StateNode<TStateId, TState, TStateMachine>.MethodName : Node.MethodName
```

## Inheritance

```
object ← GodotObject.MethodName ← Node.MethodName ←  
StateNode<TStatId, TState, TStateMachine>.MethodName
```

## Inherited Members

```
Node.MethodName._EnterTree , Node.MethodName._ExitTree ,  
Node.MethodName._GetConfigurationWarnings , Node.MethodName._Input ,  
Node.MethodName._PhysicsProcess , Node.MethodName._ShortcutInput ,  
Node.MethodName._UnhandledInput , Node.MethodName._UnhandledKeyInput ,  
Node.MethodName.PrintOrphanNodes , Node.MethodName.AddSibling ,  
Node.MethodName.SetName , Node.MethodName.GetName , Node.MethodName.AddChild ,  
Node.MethodName.RemoveChild , Node.MethodName.Reparent ,  
Node.MethodName.GetChildCount , Node.MethodName.GetChildren ,  
Node.MethodName.GetChild , Node.MethodName.HasNode , Node.MethodName.GetNode ,  
Node.MethodName.GetNodeOrNull , Node.MethodName.GetParent ,  
Node.MethodName.FindChild , Node.MethodName.FindChildren ,  
Node.MethodName.FindParent , Node.MethodName.HasNodeAndResource ,  
Node.MethodName.GetNodeAndResource , Node.MethodName.IsInsideTree ,  
Node.MethodName.IsPartOfEditedScene , Node.MethodName.IsAncestorOf ,  
Node.MethodName.IsGreater Than , Node.MethodName.GetPath ,  
Node.MethodName.GetPathTo , Node.MethodName.AddToGroup ,  
Node.MethodName.RemoveFromGroup , Node.MethodName.IsInGroup ,  
Node.MethodName.MoveChild , Node.MethodName.GetGroups ,  
Node.MethodName.SetOwner , Node.MethodName.GetOwner ,  
Node.MethodName.GetIndex , Node.MethodName.PrintTree ,  
Node.MethodName.PrintTreePretty , Node.MethodName.GetTreeString ,  
Node.MethodName.GetTreeStringPretty , Node.MethodName.SetSceneFilePath ,  
Node.MethodName.GetSceneFilePath , Node.MethodName.PropagateNotification ,
```

Node.MethodName.PropagateCall , Node.MethodName.SetPhysicsProcess ,  
Node.MethodName.GetPhysicsProcessDeltaTime , Node.MethodName.IsPhysicsProcessing ,  
Node.MethodName.GetProcessDeltaTime , Node.MethodName.SetProcess ,  
Node.MethodName.SetProcessPriority , Node.MethodName.GetProcessPriority ,  
Node.MethodName.SetPhysicsProcessPriority ,  
Node.MethodName.GetPhysicsProcessPriority , Node.MethodName.IsProcessing ,  
Node.MethodName.SetProcessInput , Node.MethodName.IsProcessingInput ,  
Node.MethodName.SetProcessShortcutInput ,  
Node.MethodName.IsProcessingShortcutInput ,  
Node.MethodName.SetProcessUnhandledInput ,  
Node.MethodName.IsProcessingUnhandledInput ,  
Node.MethodName.SetProcessUnhandledKeyInput ,  
Node.MethodName.IsProcessingUnhandledKeyInput , Node.MethodName.SetProcessMode ,  
Node.MethodName.GetProcessMode , Node.MethodName.CanProcess ,  
Node.MethodName.SetProcessThreadGroup , Node.MethodName.GetProcessThreadGroup ,  
Node.MethodName.SetProcessThreadMessages ,  
Node.MethodName.GetProcessThreadMessages ,  
Node.MethodName.SetProcessThreadGroupOrder ,  
Node.MethodName.GetProcessThreadGroupOrder , Node.MethodName.SetDisplayFolded ,  
Node.MethodName.IsDisplayedFolded , Node.MethodName.SetProcessInternal ,  
Node.MethodName.IsProcessingInternal , Node.MethodName.SetPhysicsProcessInternal ,  
Node.MethodName.IsPhysicsProcessingInternal ,  
Node.MethodName.SetPhysicsInterpolationMode ,  
Node.MethodName.GetPhysicsInterpolationMode ,  
Node.MethodName.IsPhysicsInterpolated ,  
Node.MethodName.IsPhysicsInterpolatedAndEnabled ,  
Node.MethodName.ResetPhysicsInterpolation , Node.MethodName.SetAutoTranslateMode ,  
Node.MethodName.GetAutoTranslateMode ,  
Node.MethodName.SetTranslationDomainInherited , Node.MethodName.GetWindow ,  
Node.MethodName.GetLastExclusiveWindow , Node.MethodName.GetTree ,  
Node.MethodName.CreateTween , Node.MethodName.Duplicate ,  
Node.MethodName.ReplaceBy , Node.MethodName.SetSceneInstanceLoadPlaceholder ,  
Node.MethodName.GetSceneInstanceLoadPlaceholder ,  
Node.MethodName.SetEditableInstance , Node.MethodName.IsEditableInstance ,  
Node.MethodName.GetViewport , Node.MethodName.QueueFree ,  
Node.MethodName.RequestReady , Node.MethodName.IsNodeReady ,  
Node.MethodName.SetMultiplayerAuthority , Node.MethodName.GetMultiplayerAuthority ,  
Node.MethodName.IsMultiplayerAuthority , Node.MethodName.GetMultiplayer ,  
Node.MethodName.RpcConfig , Node.MethodName.GetRpcConfig ,  
Node.MethodName.SetEditorDescription , Node.MethodName.GetEditorDescription ,

Node.MethodName.\_SetImportPath , Node.MethodName.\_GetImportPath ,  
Node.MethodName.SetUniqueNameInOwner , Node.MethodName.IsUniqueNameInOwner ,  
Node.MethodName.Atr , Node.MethodName.AtrN , Node.MethodName.Rpc ,  
Node.MethodName.RpcId , Node.MethodName.UpdateConfigurationWarnings ,  
Node.MethodName.CallDeferredThreadGroup ,  
Node.MethodName.SetDeferredThreadGroup ,  
Node.MethodName.NotifyDeferredThreadGroup , Node.MethodName.CallThreadSafe ,  
Node.MethodName.SetThreadSafe , Node.MethodName.NotifyThreadSafe ,  
GodotObject.MethodName.\_Get , GodotObject.MethodName.\_GetPropertyList ,  
GodotObject.MethodName.\_IterGet , GodotObject.MethodName.\_IterInit ,  
GodotObject.MethodName.\_IterNext , GodotObject.MethodName.\_Notification ,  
GodotObject.MethodName.\_PropertyCanRevert ,  
GodotObject.MethodName.\_PropertyGetRevert , GodotObject.MethodName.\_Set ,  
GodotObject.MethodName.\_ValidateProperty , GodotObject.MethodName.Free ,  
GodotObject.MethodName.GetClass , GodotObject.MethodName.IsClass ,  
GodotObject.MethodName.Set , GodotObject.MethodName.Get ,  
GodotObject.MethodName.SetIndexed , GodotObject.MethodName.GetIndexed ,  
GodotObject.MethodName.GetPropertyList , GodotObject.MethodName.GetMethodList ,  
GodotObject.MethodName.PropertyCanRevert ,  
GodotObject.MethodName.PropertyGetRevert , GodotObject.MethodName.Notification ,  
GodotObject.MethodName.GetInstanceId , GodotObject.MethodName.SetScript ,  
GodotObject.MethodName.GetScript , GodotObject.MethodName.SetMeta ,  
GodotObject.MethodName.RemoveMeta , GodotObject.MethodName.GetMeta ,  
GodotObject.MethodName.HasMeta , GodotObject.MethodName.GetMetaList ,  
GodotObject.MethodName.AddUserSignal , GodotObject.MethodName.HasUserSignal ,  
GodotObject.MethodName.RemoveUserSignal , GodotObject.MethodName.EmitSignal ,  
GodotObject.MethodName.Call , GodotObject.MethodName.CallDeferred ,  
GodotObject.MethodName.SetDeferred , GodotObject.MethodName.Callv ,  
GodotObject.MethodName.HasMethod ,  
GodotObject.MethodName.GetMethodArgumentCount ,  
GodotObject.MethodName.HasSignal , GodotObject.MethodName.GetSignalList ,  
GodotObject.MethodName.GetSignalConnectionList ,  
GodotObject.MethodName.GetIncomingConnections , GodotObject.MethodName.Connect ,  
GodotObject.MethodName.Disconnect , GodotObject.MethodName.IsConnected ,  
GodotObject.MethodName.HasConnections , GodotObject.MethodName.SetBlockSignals ,  
GodotObject.MethodName.IsBlockingSignals ,  
GodotObject.MethodName.NotifyPropertyListChanged ,  
GodotObject.MethodName.SetMessageTranslation ,  
GodotObject.MethodName.CanTranslateMessages , GodotObject.MethodName.Tr ,  
GodotObject.MethodName.TrN , GodotObject.MethodName.GetTranslationDomain ,

GodotObject.MethodName.SetTranslationDomain ,  
GodotObject.MethodName.IsQueuedForDeletion , GodotObject.MethodName.CancelFree ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Fields

### CanUpdate

Cached name for the 'CanUpdate' method.

```
public static readonly StringName CanUpdate
```

### Field Value

StringName

### OnEndContext

Cached name for the 'OnEndContext' method.

```
public static readonly StringName OnEndContext
```

### Field Value

StringName

### \_OnEndContext

Cached name for the '\_OnEndContext' method.

```
public static readonly StringName _OnEndContext
```

### Field Value

StringName

## \_Process

Cached name for the '\_Process' method.

```
public static readonly StringName _Process
```

Field Value

StringName

## \_Ready

Cached name for the '\_Ready' method.

```
public static readonly StringName _Ready
```

Field Value

StringName

# Class StateNode<TStatId, TState, TStateMachine>.PropertyName

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

Cached StringNames for the properties and fields contained in this class, for fast lookup.

```
public class StateNode<TStateId, TState, TStateMachine>.PropertyName  
: Node.PropertyName
```

## Inheritance

[object](#) ← GodotObject(PropertyName) ← Node(PropertyName) ← StateNode<TStatId, TState, TStateMachine>.PropertyName

## Inherited Members

Node(PropertyName.\_ImportPath , Node(PropertyName.Name ,  
Node(PropertyName.UniqueNameInOwner , Node(PropertyName.SceneFilePath ,  
Node(PropertyName.Owner , Node(PropertyName.Multiplayer ,  
Node(PropertyName.ProcessMode , Node(PropertyName.ProcessPriority ,  
Node(PropertyName.ProcessPhysicsPriority , Node(PropertyName.ProcessThreadGroup ,  
Node(PropertyName.ProcessThreadGroupOrder ,  
Node(PropertyName.ProcessThreadMessages ,  
Node(PropertyName.PhysicsInterpolationMode , Node(PropertyName.AutoTranslateMode ,  
Node(PropertyName.EditorDescription , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Fields

### stateNodes

Cached name for the 'stateNodes' field.

```
public static readonly StringName stateNodes
```

## Field Value

StringName

# Class StateNode<TStatId, TState, TStateMachine>.SignalName

Namespace: [cfGodotEngine.Util](#)

Assembly: CatSweeper.dll

Cached StringNames for the signals contained in this class, for fast lookup.

```
public class StateNode<TStateId, TState, TStateMachine>.SignalName : Node.SignalName
```

## Inheritance

```
object ← GodotObject.SignalName ← Node.SignalName ←  
StateNode<TStatId, TState, TStateMachine>.SignalName
```

## Inherited Members

```
Node.SignalName.Ready , Node.SignalName.Renamed , Node.SignalName.TreeEntered ,  
Node.SignalName.TreeExiting , Node.SignalName.TreeExited ,  
Node.SignalName.ChildEnteredTree , Node.SignalName.ChildExitingTree ,  
Node.SignalName.ChildOrderChanged , Node.SignalName.ReplacingBy ,  
Node.SignalName.EditorDescriptionChanged , Node.SignalName.EditorStateChanged ,  
GodotObject.SignalName.ScriptChanged , GodotObject.SignalName.PropertyListChanged ,  
object.Equals\(object\) , object.Equals\(object, object\) , object.GetHashCode\(\) ,  
object.GetType\(\) , object.MemberwiseClone\(\) , object.ReferenceEquals\(object, object\) ,  
object.ToString\(\)
```