

Core Manager

0.0.1

Generated by Doxygen 1.9.1

1 Namespace Index	1
1.1 Namespace List	1
2 Hierarchical Index	3
2.1 Class Hierarchy	3
3 Class Index	5
3.1 Class List	5
4 Namespace Documentation	9
4.1 CorePlugin Namespace Reference	9
4.2 CorePlugin.Core Namespace Reference	9
4.3 CorePlugin.Core.Interface Namespace Reference	9
4.4 CorePlugin.Cross Namespace Reference	9
4.5 CorePlugin.Cross.Events Namespace Reference	9
4.6 CorePlugin.Cross.Events.Interface Namespace Reference	10
4.7 CorePlugin.Cross.SceneData Namespace Reference	10
4.8 CorePlugin.Cross.SceneData.Interface Namespace Reference	10
4.9 CorePlugin.CustomAttributes Namespace Reference	10
4.10 CorePlugin.CustomAttributes.Editor Namespace Reference	10
4.11 CorePlugin.CustomAttributes.Editor.Drawers Namespace Reference	11
4.12 CorePlugin.CustomAttributes.Headers Namespace Reference	11
4.13 CorePlugin.CustomAttributes.Validation Namespace Reference	11
4.14 CorePlugin.CustomAttributes.Validation.Base Namespace Reference	11
4.15 CorePlugin.Editor Namespace Reference	12
4.16 CorePlugin.Extensions Namespace Reference	12
4.17 CorePlugin.FileSystem Namespace Reference	12
4.18 CorePlugin.Interface Namespace Reference	12
4.19 CorePlugin.Logger Namespace Reference	13
4.20 CorePlugin.ReferenceDistribution Namespace Reference	13
4.21 CorePlugin.ReferenceDistribution.Interface Namespace Reference	13
4.22 CorePlugin.Samples Namespace Reference	13
4.23 CorePlugin.Samples.Scripts Namespace Reference	13
4.24 CorePlugin.Samples.Scripts.Demo Namespace Reference	13
4.25 CorePlugin.Samples.Scripts.EventTypes Namespace Reference	13
4.26 CorePlugin.Samples.Scripts.Managers Namespace Reference	13
4.27 CorePlugin.Samples.Scripts.Model Namespace Reference	13
4.28 CorePlugin.Serializable Namespace Reference	13
4.29 CorePlugin.Serializable.Interface Namespace Reference	14
4.30 CorePlugin.Singletons Namespace Reference	14
4.31 CorePlugin.UISystem Namespace Reference	14
4.32 CorePlugin.UISystem.UI Namespace Reference	14
5 Class Documentation	15

5.1 BaseCore Class Reference	15
5.1.1 Detailed Description	15
5.1.2 Member Function Documentation	15
5.1.2.1 InitializeElements()	16
5.2 ClassValidationAttribute Class Reference	16
5.2.1 Detailed Description	16
5.3 CoreManager Class Reference	16
5.3.1 Detailed Description	17
5.4 CustomInspectorCoreManager Class Reference	17
5.4.1 Detailed Description	17
5.5 CustomLogger Class Reference	17
5.5.1 Detailed Description	18
5.6 EventInitializer Class Reference	18
5.6.1 Detailed Description	18
5.6.2 Member Function Documentation	19
5.6.2.1 AddHandler()	19
5.6.2.2 InitializeSubscriptions()	19
5.6.2.3 InvokeBase()	19
5.6.2.4 RemoveHandler()	19
5.6.2.5 Subscribe()	20
5.6.2.6 Unsubscribe()	20
5.7 EventTypes Class Reference	20
5.7.1 Detailed Description	21
5.8 FieldValidationAttribute Class Reference	21
5.8.1 Detailed Description	21
5.9 FileSystem Class Reference	21
5.9.1 Detailed Description	22
5.9.2 Constructor & Destructor Documentation	22
5.9.2.1 FileSystem() [1/3]	22
5.9.2.2 FileSystem() [2/3]	22
5.9.2.3 FileSystem() [3/3]	23
5.9.3 Member Function Documentation	23
5.9.3.1 Load< T >() [1/2]	23
5.9.3.2 Load< T >() [2/2]	24
5.9.3.3 Save< T >() [1/2]	24
5.9.3.4 Save< T >() [2/2]	25
5.10 HasComponentAttribute Class Reference	25
5.10.1 Detailed Description	26
5.11 ICore Interface Reference	26
5.11.1 Detailed Description	26
5.11.2 Member Function Documentation	27
5.11.2.1 InitializeElements()	27

5.12 IDeepClone< out out T > Interface Template Reference	27
5.12.1 Detailed Description	27
5.12.2 Member Function Documentation	27
5.12.2.1 DeepClone() [1/2]	27
5.12.2.2 DeepClone() [2/2]	28
5.13 IDistributingReference Interface Reference	28
5.13.1 Detailed Description	28
5.14 IEventHandler Interface Reference	28
5.14.1 Detailed Description	29
5.14.2 Member Function Documentation	29
5.14.2.1 InvokeEvents()	29
5.14.2.2 Subscribe()	29
5.14.2.3 Unsubscribe()	30
5.15 IEventSubscriber Interface Reference	30
5.15.1 Detailed Description	30
5.15.2 Member Function Documentation	30
5.15.2.1 GetSubscribers()	31
5.16 ISceneData Interface Reference	31
5.16.1 Detailed Description	31
5.17 IUnique Interface Reference	31
5.17.1 Detailed Description	32
5.18 Named< TName, TKey, TValue > Class Template Reference	32
5.18.1 Detailed Description	32
5.19 NotNullAttribute Class Reference	32
5.19.1 Detailed Description	33
5.20 OneAndOnlyAttribute Class Reference	33
5.20.1 Detailed Description	33
5.21 PrefabHeaderAttribute Class Reference	33
5.21.1 Detailed Description	34
5.22 PrefabRequiredAttribute Class Reference	34
5.22.1 Detailed Description	34
5.23 ReferenceDistributor Class Reference	34
5.23.1 Detailed Description	35
5.23.2 Member Function Documentation	35
5.23.2.1 AskReference< T >()	35
5.23.2.2 GetReference< T >()	36
5.23.2.3 GetReferences< T >()	36
5.23.2.4 Initialize()	37
5.24 ReferencesHeaderAttribute Class Reference	37
5.24.1 Detailed Description	37
5.25 SceneDataHandler Class Reference	37
5.25.1 Detailed Description	38

5.25.2 Member Function Documentation	38
5.25.2.1 AddData< T >()	38
5.25.2.2 GetData< T >()	39
5.25.2.3 RemoveData< T >()	39
5.26 SettingsHeaderAttribute Class Reference	39
5.26.1 Detailed Description	40
5.27 Singleton< T > Class Template Reference	40
5.27.1 Detailed Description	40
5.28 StateHeaderAttribute Class Reference	40
5.28.1 Detailed Description	40
5.29 SubPagesController Class Reference	41
5.29.1 Detailed Description	41
5.29.2 Member Function Documentation	41
5.29.2.1 AddPage() [1/2]	41
5.29.2.2 AddPage() [2/2]	42
5.29.2.3 DelayedOpenPage() [1/2]	42
5.29.2.4 DelayedOpenPage() [2/2]	42
5.29.2.5 HideAllTables()	43
5.29.2.6 OpenPage()	43
5.30 SymbolDefiner Class Reference	43
5.30.1 Detailed Description	44
5.30.2 Member Function Documentation	44
5.30.2.1 OnEnable()	44
5.30.2.2 ShowSymbolsButtons()	44
5.31 UIManager Class Reference	45
5.31.1 Detailed Description	45
5.31.2 Member Function Documentation	45
5.31.2.1 InitializeElements()	45
5.32 UIPage Class Reference	46
5.32.1 Detailed Description	46
5.32.2 Member Function Documentation	46
5.32.2.1 Initialize()	46
5.33 UIStateTools Class Reference	46
5.33.1 Detailed Description	47
5.33.2 Member Function Documentation	47
5.33.2.1 ChangeCursorState()	47
5.33.2.2 ChangeGroupState() [1/2]	47
5.33.2.3 ChangeGroupState() [2/2]	48
5.34 Unique Class Reference	48
5.34.1 Detailed Description	48
5.35 UnityEditorExtension Class Reference	49
5.35.1 Detailed Description	49

5.35.2 Member Function Documentation	49
5.35.2.1 HelpBox() [1/2]	49
5.35.2.2 HelpBox() [2/2]	49
5.35.2.3 IconName()	50
5.36 UnityExtensions Class Reference	50
5.36.1 Detailed Description	51
5.36.2 Member Function Documentation	51
5.36.2.1 Clear< T >()	51
5.36.2.2 IsNotNullAndEqual()	51
5.36.2.3 IsNotNullAndNotEqual()	52
5.36.2.4 TryFindObjectOfType< T >()	52
5.36.2.5 TryFindObjectOfType< T >() [1/2]	53
5.36.2.6 TryFindObjectOfType< T >() [2/2]	53
5.37 ValidationAttribute Class Reference	54
5.37.1 Detailed Description	54
5.38 ValidationAttributeEditor Class Reference	54
5.38.1 Detailed Description	54
Index	55

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

CorePlugin	9
CorePlugin.Core	9
CorePlugin.Core.Interface	9
CorePlugin.Cross	9
CorePlugin.Cross.Events	9
CorePlugin.Cross.Events.Interface	10
CorePlugin.Cross.SceneData	10
CorePlugin.Cross.SceneData.Interface	10
CorePlugin.CustomAttributes	10
CorePlugin.CustomAttributes.Editor	10
CorePlugin.CustomAttributes.Editor.Drawers	11
CorePlugin.CustomAttributes.Headers	11
CorePlugin.CustomAttributes.Validation	11
CorePlugin.CustomAttributes.Validation.Base	11
CorePlugin.Editor	12
CorePlugin.Extensions	12
CorePlugin.FileSystem	12
CorePlugin.Interface	12
CorePlugin.Logger	13
CorePlugin.ReferenceDistribution	13
CorePlugin.ReferenceDistribution.Interface	13
CorePlugin.Samples	13
CorePlugin.Samples.Scripts	13
CorePlugin.Samples.Scripts.Demo	13
CorePlugin.Samples.Scripts.EventTypes	13
CorePlugin.Samples.Scripts.Managers	13
CorePlugin.Samples.Scripts.Model	13
CorePlugin.Serializable	13
CorePlugin.Serializable.Interface	14
CorePlugin.Singletons	14
CorePlugin.UISystem	14
CorePlugin.UISystem.UI	14

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

CoreManager	16
CustomLogger	17
EventInitializer	18
EventTypes	20
FileSystem	21
ICore	26
BaseCore	15
UIManager	45
IDeepClone< out out T >	27
IDistributingReference	28
IEventHandler	28
IEventSubscriber	30
ISceneData	31
IUnique	31
Unique	48
Named< TName, TKey, TValue >	32
Named< string, Object >	32
PrefabHeaderAttribute	33
ReferenceDistributor	34
ReferencesHeaderAttribute	37
SettingsHeaderAttribute	39
Singleton< T >	40
Singleton< SceneDataHandler >	40
SceneDataHandler	37
StateHeaderAttribute	40
SubPagesController	41
SymbolDefiner	43
UIPage	46
UIStateTools	46
UnityEditorExtension	49
UnityExtensions	50
ValidationAttribute	54
ClassValidationAttribute	16
OneAndOnlyAttribute	33

FieldValidationAttribute	21
HasComponentAttribute	25
NotNullAttribute	32
PrefabRequiredAttribute	34
ValidationAttributeEditor	54
CustomInspectorCoreManager	17

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

BaseCore	
Base implementation of IManager	15
ClassValidationAttribute	
Base attribute for class validation	16
16	
17	
CustomLogger	
Custom logger solution for logs	17
18	
EventTypes	
Declare your delegate type to convert method group to delegate	20
FieldValidationAttribute	
Base attribute for field validation	21
21	
HasComponentAttribute	
Attribute validating whether Object in field or all objects in the list have desired component	25

ICore		
	Interface for Manager declaration	26
IDeepClone< out out T >		
	Interface for deep cloning	27
IDistributingReference		
	Flag interface for MonoBehaviour classes what need to referenced from another class	28
IEventHandler		
	Interface for event handler	28
IEventSubscriber		
	Interface for subscribers	30
31		
IUnique		
	Interface for unique objects. For example, you can use it for objects that need to be saved in Json or objects with the same data, but with different identifiers	31
Named< TName, TKey, TValue >		
	Replacement for dictionary in Unity Inspector	32
NotNullAttribute		
	Attribute validating whether field or element in the list equals null	32
OneAndOnlyAttribute		
	Attribute validating whether there is only one copy of this class in the scene	33
PrefabHeaderAttribute		
	Replacement for Header("Prefabs")	33
PrefabRequiredAttribute		
	Attribute validating whether the object or all items in list are prefabs	34
ReferenceDistributor		
	Class responsible for reference distribution inside one scene	34
ReferencesHeaderAttribute		
	Replacement for Header("References")	37
37		
SettingsHeaderAttribute		
	Replacement for Header("Settings")	39
Singleton< T >		
	Base for all singletons. Strongly recommended to use singletons as little as possible	40

StateHeaderAttribute	
Replacement for Header("State")	40
SubPagesController	
Class designated for UI subpages	41
43	
UIManager	
Base implementation of UIManager	45
UIPage	
UI page implementation	46
UIStateTools	
UI state tool for canvas groups	46
Unique	
Base class for classes that need to be saved in Json file	48
UnityEditorExtension	
Extensions for Unity Editor classes	49
UnityExtensions	
Extension class for default Unity classes	50
ValidationAttribute	
Base attribute for validation	54
54	

Chapter 4

Namespace Documentation

4.1 CorePlugin Namespace Reference

4.2 CorePlugin.Core Namespace Reference

Classes

- class [BaseCore](#)
Base implementation of IManager.
- class [CoreManager](#)
Manager for initialization of sub manager in the scene.
See also
[CorePlugin.ReferenceDistribution.ReferenceDistributor](#), [CorePlugin.Core.Interface.ICore](#)
- class [EventInitializer](#)
Class responsible for reference initialization
See also
[CorePlugin.Cross.Events.Interface.IEventHandler](#), [CorePlugin.Cross.Events.Interface.IEventSubscriber](#)

4.3 CorePlugin.Core.Interface Namespace Reference

Classes

- interface [ICore](#)
Interface for Manager declaration.

4.4 CorePlugin.Cross Namespace Reference

4.5 CorePlugin.Cross.Events Namespace Reference

Classes

- class [EventTypes](#)
Declare your delegate type to convert method group to delegate.

4.6 CorePlugin.Cross.Events.Interface Namespace Reference

Classes

- interface [IEventHandler](#)
Interface for event handler.
- interface [IEventSubscriber](#)
Interface for subscribers.

4.7 CorePlugin.Cross.SceneData Namespace Reference

Classes

- class [SceneDataHandler](#)
Singleton for passing data between scenes
See also
[CorePlugin.Cross.SceneData.Interface.ISceneData](#)

4.8 CorePlugin.Cross.SceneData.Interface Namespace Reference

Classes

- interface [ISceneData](#)
Flag interface for classes or structs to pass between scenes
See also
[CorePlugin.Cross.SceneData.SceneDataHandler](#)

4.9 CorePlugin.CustomAttributes Namespace Reference

4.10 CorePlugin.CustomAttributes.Editor Namespace Reference

Classes

- class [ValidationAttributeEditor](#)
Custom editor for Unity.Object class. If you want to create your own custom editor inherit from this class.
See also
[CorePlugin.CustomAttributes.Validation.Base.ValidationAttribute](#), [CorePlugin.CustomAttributes.Validation.Base.FieldValidationAttribute](#),
[CorePlugin.CustomAttributes.Validation.Base.ClassValidationAttribute](#)

4.11 CorePlugin.CustomAttributes.Editor.Drawers Namespace Reference

4.12 CorePlugin.CustomAttributes.Headers Namespace Reference

Classes

- class [PrefabHeaderAttribute](#)
Replacement for Header("Prefabs")
- class [ReferencesHeaderAttribute](#)
Replacement for Header("References")
- class [SettingsHeaderAttribute](#)
Replacement for Header("Settings")
- class [StateHeaderAttribute](#)
Replacement for Header("State")

4.13 CorePlugin.CustomAttributes.Validation Namespace Reference

Classes

- class [HasComponentAttribute](#)
Attribute validating whether Object in field or all objects in the list have desired component.
- class [NotNullAttribute](#)
Attribute validating whether field or element in the list equals null.
- class [OneAndOnlyAttribute](#)
Attribute validating whether there is only one copy of this class in the scene.
- class [PrefabRequiredAttribute](#)
Attribute validating whether the object or all items in list are prefabs.

4.14 CorePlugin.CustomAttributes.Validation.Base Namespace Reference

Classes

- class [ClassValidationAttribute](#)
Base attribute for class validation.
- class [FieldValidationAttribute](#)
Base attribute for field validation.
- class [ValidationAttribute](#)
Base attribute for validation

4.15 CorePlugin.Editor Namespace Reference

Classes

- class [CustomInspectorCoreManager](#)
Custom [Editor](#) [CoreManager](#) class.
See also
[CorePlugin.Core.CoreManager](#)
- class [SymbolDefiner](#)
Class for "Scripting Define Symbols" defining from [CoreManager](#) Inspector.
See also
[CorePlugin.Core.CoreManager](#)

4.16 CorePlugin.Extensions Namespace Reference

Classes

- class [Named](#)
Replacement for dictionary in Unity Inspector
- class [UIStateTools](#)
UI state tool for canvas groups
- class [UnityEditorExtension](#)
[Extensions](#) for Unity [Editor](#) classes
- class [UnityExtensions](#)
Extension class for default Unity classes

4.17 CorePlugin.FileSystem Namespace Reference

Classes

- class [FileSystem](#)
Class for saving Json file to disk.
See also
[CorePlugin.Serializable.Unique](#)

4.18 CorePlugin.Interface Namespace Reference

Classes

- interface [IDeepClone](#)
[Interface](#) for deep cloning

4.19 CorePlugin.Logger Namespace Reference

Classes

- class [CustomLogger](#)
Custom logger solution for logs.

4.20 CorePlugin.ReferenceDistribution Namespace Reference

Classes

- class [ReferenceDistributor](#)
Class responsible for reference distribution inside one scene.

4.21 CorePlugin.ReferenceDistribution.Interface Namespace Reference

Classes

- interface [IDistributingReference](#)
Flag interface for MonoBehaviour classes what need to be referenced from another class.

4.22 CorePlugin.Samples Namespace Reference

4.23 CorePlugin.Samples.Scripts Namespace Reference

4.24 CorePlugin.Samples.Scripts.Demo Namespace Reference

4.25 CorePlugin.Samples.Scripts.EventTypes Namespace Reference

4.26 CorePlugin.Samples.Scripts.Managers Namespace Reference

4.27 CorePlugin.Samples.Scripts.Model Namespace Reference

4.28 CorePlugin.Serializable Namespace Reference

Classes

- class [Unique](#)
Base class for classes that need to be saved in Json file.

4.29 CorePlugin.Serializable.Interface Namespace Reference

Classes

- interface [IUnique](#)

[Interface](#) for unique objects. For example, you can use it for objects that need to be saved in Json or objects with the same data, but with different identifiers

4.30 CorePlugin.Singletons Namespace Reference

Classes

- class [Singleton](#)

Base for all singletons. Strongly recommended to use singletons as little as possible.

4.31 CorePlugin.UISystem Namespace Reference

Classes

- class [UIManager](#)

Base implementation of [UIManager](#).

4.32 CorePlugin.UISystem.UI Namespace Reference

Classes

- class [SubPagesController](#)

Class designated for [UI](#) subpages.

- class [UIPage](#)

[UI](#) page implementation

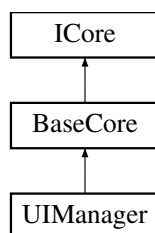
Chapter 5

Class Documentation

5.1 BaseCore Class Reference

Base implementation of IManager.

Inheritance diagram for BaseCore:



Public Member Functions

- virtual void [InitializeElements](#) ()
Initializing all managers elements.

5.1.1 Detailed Description

Base implementation of IManager.

Definition at line 28 of file BaseCore.cs.

5.1.2 Member Function Documentation

5.1.2.1 InitializeElements()

```
virtual void InitializeElements ( ) [inline], [virtual]
```

Initializing all managers elements.

Implements [ICore](#).

Reimplemented in [UIManager](#).

Definition at line 33 of file BaseCore.cs.

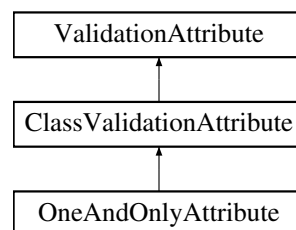
The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/Core/BaseCore.cs

5.2 ClassValidationAttribute Class Reference

[Base](#) attribute for class validation.

Inheritance diagram for ClassValidationAttribute:



5.2.1 Detailed Description

[Base](#) attribute for class validation.

Definition at line 23 of file ClassValidationAttribute.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/CustomAttributes/Validation/Base/ClassValidationAttribute.cs

5.3 CoreManager Class Reference

Manager for initialization of sub manager in the scene.

See also

[CorePlugin.ReferenceDistribution.ReferenceDistributor](#), [CorePlugin.Core.Interface.ICore](#)

Inherits MonoBehaviour.

5.3.1 Detailed Description

Manager for initialization of sub manager in the scene.

See also

[CorePlugin.ReferenceDistribution.ReferenceDistributor](#), [CorePlugin.Core.Interface.ICore](#)

Definition at line 32 of file CoreManager.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/Core/CoreManager.cs

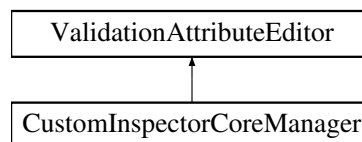
5.4 CustomInspectorCoreManager Class Reference

Custom [Editor](#) CoreManager class.

See also

[CorePlugin.Core.CoreManager](#)

Inheritance diagram for CustomInspectorCoreManager:



5.4.1 Detailed Description

Custom [Editor](#) CoreManager class.

See also

[CorePlugin.Core.CoreManager](#)

Definition at line 32 of file CustomInspectorCoreManager.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/Editor/CustomInspectorCoreManager.cs

5.5 CustomLogger Class Reference

Custom logger solution for logs.

5.5.1 Detailed Description

Custom logger solution for logs.

Logs in this class are dependant on `DEBUG` and `ENABLE_RELEASE_LOGS`. If `ENABLE_RELEASE_LOGS` defined logs will displayed in Release Build. Otherwise only [Editor](#) and Developer Build will display logs. For defining preprocessor open CoreManager or write down in PlayerSettings in field "Scripting Define Symbols".

See also

[CorePlugin.Core.CoreManager](#)

Definition at line 33 of file CustomLogger.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/Logger/CustomLogger.cs

5.6 EventInitializer Class Reference

Class responsible for reference initialization

See also

[CorePlugin.Cross.Events.Interface.IEventHandler](#), [CorePlugin.Cross.Events.Interface.IEventSubscriber](#)

Static Public Member Functions

- static void [InitializeSubscriptions](#) ()
Initialising cross subscriptions for all handlers in the scene.
- static void [Subscribe](#) ([IEventSubscriber](#) subscriber)
Subscribing event subscriber after scene has Awoken to event handlers.
- static void [Unsubscribe](#) ([IEventSubscriber](#) subscriber)
Unsubscribing event subscriber after scene has Awoken from event handlers.
- static void [AddHandler](#) ([IEventHandler](#) handler, bool subscriptionsNeeded=true, bool invokeNeeded=false)
Adding new handler after scene has Awoken to list of event handlers.
- static void [RemoveHandler](#) ([IEventHandler](#) handler)
Removing event handler after scene has Awoken from list
- static void [InvokeBase](#) ()
Invoking event on handlers.

5.6.1 Detailed Description

Class responsible for reference initialization

See also

[CorePlugin.Cross.Events.Interface.IEventHandler](#), [CorePlugin.Cross.Events.Interface.IEventSubscriber](#)

Definition at line 28 of file EventInitializer.cs.

5.6.2 Member Function Documentation

5.6.2.1 AddHandler()

```
static void AddHandler (
    IEventHandler handler,
    bool subscriptionsNeeded = true,
    bool invokeNeeded = false ) [inline], [static]
```

Adding new handler after scene has Awoken to list of event handlers.

Parameters

<i>handler</i>	
<i>subscriptionsNeeded</i>	If false invokeNeeded will not be called
<i>invokeNeeded</i>	

Definition at line 67 of file EventInitializer.cs.

References IEventHandler.InvokeEvents(), and IEventHandler.Subscribe().

5.6.2.2 InitializeSubscriptions()

```
static void InitializeSubscriptions ( ) [inline], [static]
```

Initialising cross subscriptions for all handlers in the scene.

Definition at line 35 of file EventInitializer.cs.

5.6.2.3 InvokeBase()

```
static void InvokeBase ( ) [inline], [static]
```

Invoking event on handlers.

Definition at line 89 of file EventInitializer.cs.

5.6.2.4 RemoveHandler()

```
static void RemoveHandler (
    IEventHandler handler ) [inline], [static]
```

Removing event handler after scene has Awoken from list

Parameters

<i>handler</i>	
----------------	--

Definition at line 81 of file EventInitializer.cs.

5.6.2.5 Subscribe()

```
static void Subscribe (  
    IEventSubscriber subscriber ) [inline], [static]
```

Subscribing event subscriber after scene has Awoken to event handlers.

Parameters

<i>subscriber</i>	
-------------------	--

Definition at line 47 of file EventInitializer.cs.

5.6.2.6 Unsubscribe()

```
static void Unsubscribe (  
    IEventSubscriber subscriber ) [inline], [static]
```

Unsubscribing event subscriber after scene has Awoken from event handlers.

Parameters

<i>subscriber</i>	
-------------------	--

Definition at line 56 of file EventInitializer.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/Core/EventInitializer.cs

5.7 EventTypes Class Reference

Declare your delegate type to convert method group to delegate.

Inherited by CustomEventTypes.

5.7.1 Detailed Description

Declare your delegate type to convert method group to delegate.

```
public delegate void DelegateClass(Type type1);  
public delegate void DelegateClass2(Type type2, Type type3);
```

See also

[CorePlugin.Cross.Events.Interface.IEventHandler](#), [CorePlugin.Cross.Events.Interface.IEventSubscriber](#)

Definition at line 29 of file EventTypes.cs.

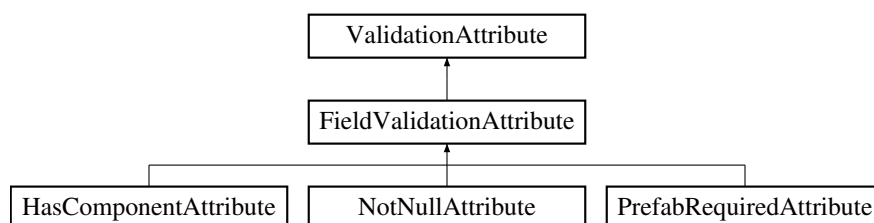
The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/Cross/Events/EventTypes.cs

5.8 FieldValidationAttribute Class Reference

[Base](#) attribute for field validation.

Inheritance diagram for FieldValidationAttribute:



5.8.1 Detailed Description

[Base](#) attribute for field validation.

Definition at line 28 of file FieldValidationAttribute.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/CustomAttributes/Validation/Base/FieldValidationAttribute.cs

5.9 FileSystem Class Reference

Class for saving Json file to disk.

See also

[CorePlugin.Serializable.Unique](#)

Public Member Functions

- [FileSystem](#) ()
Default constructor.
- [FileSystem](#) (string path)
Overloaded constructor which changes default path.
- [FileSystem](#) (string path, string extension)
Overloaded constructor which changes default path and file extension.
- void [Save](#)< T > (T data, Action< Exception > onError, Object context=null)
Saves class to the file with the name of class.
- void [Save](#)< T > (T data, string fileName, Action< Exception > onError, Object context=null)
Saves your class to file.
- void [Load](#)< T > (Action< T > onLoaded, Action< Exception > onError, Object context=null)
Loads file.
- void [Load](#)< T > (string fileName, Action< T > onLoaded, Action< Exception > onError, Object context=null)
Loads file with different file name from passed class.

5.9.1 Detailed Description

Class for saving Json file to disk.

See also

[CorePlugin.Serializable.Unique](#)

Definition at line 29 of file FileSystem.cs.

5.9.2 Constructor & Destructor Documentation

5.9.2.1 FileSystem() [1/3]

```
FileSystem ( ) [inline]
```

Default constructor.

Definition at line 38 of file FileSystem.cs.

5.9.2.2 FileSystem() [2/3]

```
FileSystem (
    string path ) [inline]
```

Overloaded constructor which changes default path.

Parameters

<i>path</i>	
-------------	--

Definition at line 47 of file FileSystem.cs.

5.9.2.3 FileSystem() [3/3]

```
FileSystem (
    string path,
    string extension ) [inline]
```

Overloaded constructor which changes default path and file extension.

Parameters

<i>path</i>	
<i>extension</i>	

Definition at line 57 of file FileSystem.cs.

5.9.3 Member Function Documentation

5.9.3.1 Load< T >() [1/2]

```
void Load< T > (
    Action< T > onLoaded,
    Action< Exception > onError,
    Object context = null ) [inline]
```

Loads file.

Parameters

<i>onLoaded</i>	
<i>onError</i>	
<i>context</i>	Required if errors should be shown on Object

Template Parameters

<i>T</i>	
----------	--

Type Constraints

T : Unique

Definition at line 117 of file FileSystem.cs.

5.9.3.2 Load< T >() [2/2]

```
void Load< T > (
    string fileName,
    Action< T > onLoaded,
    Action< Exception > onError,
    Object context = null ) [inline]
```

Loads file with different file name from passed class.

Parameters

<i>fileName</i>	
<i>onLoaded</i>	
<i>onError</i>	
<i>context</i>	Required if errors should be shown on Object

Template Parameters

<i>T</i>	
----------	--

Type Constraints

T : Unique

Definition at line 130 of file FileSystem.cs.

5.9.3.3 Save< T >() [1/2]

```
void Save< T > (
    T data,
    Action< Exception > onError,
    Object context = null ) [inline]
```

Saves class to the file with the name of class.

Parameters

<i>data</i>	
<i>onError</i>	
<i>context</i>	Required if errors should be shown on Object

Template Parameters

<i>T</i>	
----------	--

Type Constraints

T : Unique

Definition at line 69 of file FileSystem.cs.

5.9.3.4 Save< T >() [2/2]

```
void Save< T > (
    T data,
    string fileName,
    Action< Exception > onError,
    Object context = null ) [inline]
```

Saves your class to file.

Parameters

<i>data</i>	
<i>fileName</i>	
<i>onError</i>	
<i>context</i>	Required if errors should be shown on Object

Template Parameters

<i>T</i>	
----------	--

Type Constraints

T : Unique

Definition at line 82 of file FileSystem.cs.

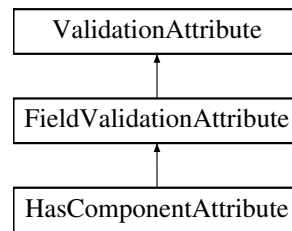
The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/FileSystem/FileSystem.cs

5.10 HasComponentAttribute Class Reference

Attribute validating whether Object in field or all objects in the list have desired component.

Inheritance diagram for HasComponentAttribute:



5.10.1 Detailed Description

Attribute validating whether Object in field or all objects in the list have desired component.

Definition at line 27 of file HasComponentAttribute.cs.

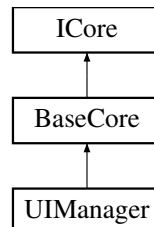
The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/CustomAttributes/Validation/HasComponentAttribute.cs

5.11 ICore Interface Reference

[Interface](#) for Manager declaration.

Inheritance diagram for ICore:



Public Member Functions

- void [InitializeElements](#) ()
Initializing all managers elements.

5.11.1 Detailed Description

[Interface](#) for Manager declaration.

Definition at line 22 of file ICore.cs.

5.11.2 Member Function Documentation

5.11.2.1 InitializeElements()

```
void InitializeElements ( )
```

Initializing all managers elements.

Implemented in [UIManager](#), and [BaseCore](#).

The documentation for this interface was generated from the following file:

- CoreManager/Assets/CorePlugin/Core/Interface/ICore.cs

5.12 IDepClone< out out T > Interface Template Reference

[Interface](#) for deep cloning

Inherits IDepClone.

Public Member Functions

- new T [DeepClone](#) ()
Clones instance of an object
- object [DeepClone](#) ()
Clones instance of an object

5.12.1 Detailed Description

[Interface](#) for deep cloning

Definition at line 21 of file IDepClone.cs.

5.12.2 Member Function Documentation

5.12.2.1 DeepClone() [1/2]

```
new T DeepClone ( )
```

Clones instance of an object

Returns

5.12.2.2 DeepClone() [2/2]

object DeepClone ()

Clones instance of an object

Returns

The documentation for this interface was generated from the following file:

- CoreManager/Assets/CorePlugin/Interface/IDeepClone.cs

5.13 IDistributingReference Interface Reference

Flag interface for MonoBehaviour classes what need to referenced from another class.

5.13.1 Detailed Description

Flag interface for MonoBehaviour classes what need to referenced from another class.

Strongly recommended to use [CorePlugin.Cross.Events](#) and [CorePlugin.Cross.Events.Interface](#) instead of direct reference serialization.

Definition at line 22 of file IDistributingReference.cs.

The documentation for this interface was generated from the following file:

- CoreManager/Assets/CorePlugin/ReferenceDistribution/Interface/IDistributingReference.cs

5.14 IEventHandler Interface Reference

[Interface](#) for event handler.

Inherited by CrossSceneDataSenderDemo.

Public Member Functions

- void [InvokeEvents](#) ()
Invoking events that need to be invoked on scene initialization.
- void [Subscribe](#) (IEnumerable< Delegate > subscribers)
Subscribing delegates to event
- void [Unsubscribe](#) (IEnumerable< Delegate > unsubscribers)
Unsubscribing delegates to event

5.14.1 Detailed Description

[Interface](#) for event handler.

```
public void Subscribe(params Delegate[] subscriber)
{
    foreach (var action in subscriber.OfType<EventTypes.DelegateClass>())
    {
        MyEvent += action;
    }
}
```

See also

[CorePlugin.Cross.Events.EventTypes](#)

Definition at line 34 of file IEventHandler.cs.

5.14.2 Member Function Documentation

5.14.2.1 InvokeEvents()

```
void InvokeEvents ( )
```

Invoking events that need to be invoked on scene initialization.

Referenced by EventInitializer.AddHandler().

5.14.2.2 Subscribe()

```
void Subscribe (
    IEnumerable< Delegate > subscribers )
```

Subscribing delegates to event

Parameters

<i>subscribers</i>	
--------------------	--

Referenced by EventInitializer.AddHandler().

5.14.2.3 Unsubscribe()

```
void Unsubscribe (
    IEnumerable< Delegate > unsubscribers )
```

Unsubscribing delegates to event

Parameters

<i>unsubscribers</i>	
----------------------	--

The documentation for this interface was generated from the following file:

- CoreManager/Assets/CorePlugin/Cross/Events/Interface/IEventHandler.cs

5.15 IEventSubscriber Interface Reference

[Interface](#) for subscribers.

Inherited by [SceneLoader](#).

Public Member Functions

- [IEnumerable< Delegate > GetSubscribers \(\)](#)
Returns IEnumerable with all methods which need to be subscribed.

5.15.1 Detailed Description

[Interface](#) for subscribers.

```
public IEnumerable<Delegate> GetSubscribers \(\)
{
    var list = new Delegate[] { (CrossEventTypes.DelegateClass) MyMethod, (Cross↵
EventTypes.DelegateClass2) MyMethod2};
    return list;
}
```

See also

[CorePlugin.Cross.Events.EventTypes](#)

Definition at line 33 of file [IEventSubscriber.cs](#).

5.15.2 Member Function Documentation

5.15.2.1 GetSubscribers()

```
IEnumerable<Delegate> GetSubscribers ( )
```

Returns IEnumerable with all methods which need to be subscribed.

Returns

The documentation for this interface was generated from the following file:

- CoreManager/Assets/CorePlugin/Cross/Events/Interface/IEventSubscriber.cs

5.16 ISceneData Interface Reference

Flag interface for classes or structs to pass between scenes

See also

[CorePlugin.Cross.SceneData.SceneDataHandler](#)

Inherited by DataTransfer.

5.16.1 Detailed Description

Flag interface for classes or structs to pass between scenes

See also

[CorePlugin.Cross.SceneData.SceneDataHandler](#)

Definition at line 22 of file ISceneData.cs.

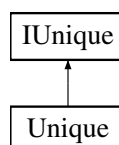
The documentation for this interface was generated from the following file:

- CoreManager/Assets/CorePlugin/Cross/SceneData/Interface/ISceneData.cs

5.17 IUnique Interface Reference

[Interface](#) for unique objects. For example, you can use it for objects that need to be saved in Json or objects with the same data, but with different identifiers

Inheritance diagram for IUnique:



5.17.1 Detailed Description

[Interface](#) for unique objects. For example, you can use it for objects that need to be saved in Json or objects with the same data, but with different identifiers

Definition at line 22 of file IUnique.cs.

The documentation for this interface was generated from the following file:

- CoreManager/Assets/CorePlugin/Serializable/Interface/IUnique.cs

5.18 Named< TName, TKey, TValue > Class Template Reference

Replacement for dictionary in Unity Inspector

5.18.1 Detailed Description

Replacement for dictionary in Unity Inspector

More complex list for dictionary

Template Parameters

<i>TKey</i>	
<i>TValue</i>	

Template Parameters

<i>TName</i>	
<i>TKey</i>	
<i>TValue</i>	

Definition at line 57 of file Named.cs.

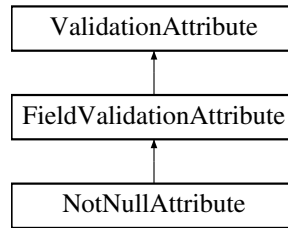
The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/Extensions/Named.cs

5.19 NotNullAttribute Class Reference

Attribute validating whether field or element in the list equals null.

Inheritance diagram for NotNullAttribute:



5.19.1 Detailed Description

Attribute validating whether field or element in the list equals null.

Definition at line 25 of file NotNullAttribute.cs.

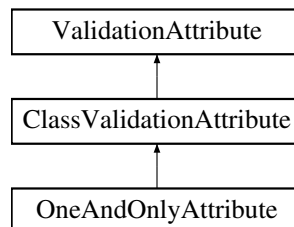
The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/CustomAttributes/Validation/NotNullAttribute.cs

5.20 OneAndOnlyAttribute Class Reference

Attribute validating whether there is only one copy of this class in the scene.

Inheritance diagram for OneAndOnlyAttribute:



5.20.1 Detailed Description

Attribute validating whether there is only one copy of this class in the scene.

Definition at line 28 of file OneAndOnlyAttribute.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/CustomAttributes/Validation/OneAndOnlyAttribute.cs

5.21 PrefabHeaderAttribute Class Reference

Replacement for Header("Prefabs")

Inherits HeaderAttribute.

5.21.1 Detailed Description

Replacement for Header("Prefabs")

Definition at line 25 of file PrefabHeaderAttribute.cs.

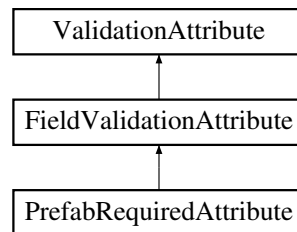
The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/CustomAttributes/Headers/PrefabHeaderAttribute.cs

5.22 PrefabRequiredAttribute Class Reference

Attribute validating whether the object or all items in list are prefabs.

Inheritance diagram for PrefabRequiredAttribute:



5.22.1 Detailed Description

Attribute validating whether the object or all items in list are prefabs.

Definition at line 27 of file PrefabRequiredAttribute.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/CustomAttributes/Validation/PrefabRequiredAttribute.cs

5.23 ReferenceDistributor Class Reference

Class responsible for reference distribution inside one scene.

Inherits MonoBehaviour.

Public Member Functions

- void [Initialize](#) ()
Initializing distribution references

Static Public Member Functions

- static T [GetReference< T > \(\)](#)
Getting reference by type from list
- static bool [AskReference< T > \(ref T reference\)](#)
Finding reference if passed parameter is null. Use this if you need reference not in Start() and/or reference should be received in some event
- static IEnumerable< T > [GetReferences< T > \(\)](#)
Getting references by type from list

5.23.1 Detailed Description

Class responsible for reference distribution inside one scene.

Strongly recommended to use [CorePlugin.Cross.Events](#) and [CorePlugin.Cross.Events.Interface](#) instead of direct reference serialization.

Definition at line 31 of file ReferenceDistributor.cs.

5.23.2 Member Function Documentation

5.23.2.1 AskReference< T >()

```
static bool AskReference< T > (
    ref T reference ) [inline], [static]
```

Finding reference if passed parameter is null. Use this if you need reference not in Start() and/or reference should be received in some event

Parameters

<i>reference</i>	
------------------	--

Template Parameters

<i>T</i>	
----------	--

Returns

Type Constraints

T : MonoBehaviour
T : IDistributingReference

Definition at line 68 of file ReferenceDistributor.cs.

References ReferenceDistributor.GetReference< T >().

5.23.2.2 GetReference< T >()

```
static T GetReference< T > ( ) [inline], [static]
```

Getting reference by type from list

Template Parameters

<i>T</i>	
----------	--

Returns

Type Constraints

T : MonoBehaviour

T : IDistributingReference

Definition at line 56 of file ReferenceDistributor.cs.

Referenced by ReferenceDistributor.AskReference< T >().

5.23.2.3 GetReferences< T >()

```
static IEnumerable<T> GetReferences< T > ( ) [inline], [static]
```

Getting references by type from list

Template Parameters

<i>T</i>	
----------	--

Returns

Type Constraints

T : MonoBehaviour

T : IDistributingReference

Definition at line 79 of file ReferenceDistributor.cs.

5.23.2.4 Initialize()

```
void Initialize ( ) [inline]
```

Initializing distribution references

Definition at line 45 of file ReferenceDistributor.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/ReferenceDistribution/ReferenceDistributor.cs

5.24 ReferencesHeaderAttribute Class Reference

Replacement for Header("References")

Inherits HeaderAttribute.

5.24.1 Detailed Description

Replacement for Header("References")

Definition at line 25 of file ReferencesHeaderAttribute.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/CustomAttributes/Headers/ReferencesHeaderAttribute.cs

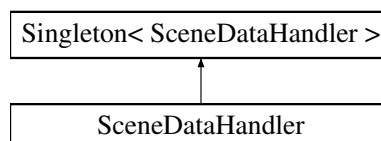
5.25 SceneDataHandler Class Reference

Singleton for passing data between scenes

See also

[CorePlugin.Cross.SceneData.Interface.ISceneData](#)

Inheritance diagram for SceneDataHandler:



Public Member Functions

- void [AddData< T >](#) (T data)
Adding data to dictionary by passed Type
- bool [GetData< T >](#) (out T data)
Getting data from dictionary by passed Type
- void [RemoveData< T >](#) ()
Removing data from dictionary by passed Type

5.25.1 Detailed Description

Singleton for passing data between scenes

See also

[CorePlugin.Cross.SceneData.Interface.ISceneData](#)

Definition at line 29 of file SceneDataHandler.cs.

5.25.2 Member Function Documentation

5.25.2.1 AddData< T >()

```
void AddData< T > (
    T data ) [inline]
```

Adding data to dictionary by passed Type

Parameters

<i>data</i>	
-------------	--

Template Parameters

<i>T</i>	
----------	--

Type Constraints

T* : *ISceneData

T* : *new()

Definition at line 50 of file SceneDataHandler.cs.

5.25.2.2 GetData< T >()

```
bool GetData< T > (
    out T data ) [inline]
```

Getting data from dictionary by passed Type

Parameters

<i>data</i>	
-------------	--

Template Parameters

<i>T</i>	
----------	--

Type Constraints

T : ISceneData

T : new()

Definition at line 65 of file SceneDataHandler.cs.

5.25.2.3 RemoveData< T >()

```
void RemoveData< T > ( ) [inline]
```

Removing data from dictionary by passed Type

Template Parameters

<i>T</i>	
----------	--

Type Constraints

T : ISceneData

T : new()

Definition at line 76 of file SceneDataHandler.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/Cross/SceneData/SceneDataHandler.cs

5.26 SettingsHeaderAttribute Class Reference

Replacement for Header("Settings")

Inherits HeaderAttribute.

5.26.1 Detailed Description

Replacement for Header("Settings")

Definition at line 25 of file SettingsHeaderAttribute.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/CustomAttributes/Headers/SettingsHeaderAttribute.cs

5.27 Singleton< T > Class Template Reference

Base for all singletons. Strongly recommended to use singletons as little as possible.

Inherits MonoBehaviour.

5.27.1 Detailed Description

Base for all singletons. Strongly recommended to use singletons as little as possible.

Template Parameters

<i>T</i>	
----------	--

Type Constraints

T : MonoBehaviour

Definition at line 26 of file Singleton.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/Singletons/Singleton.cs

5.28 StateHeaderAttribute Class Reference

Replacement for Header("State")

Inherits HeaderAttribute.

5.28.1 Detailed Description

Replacement for Header("State")

Definition at line 25 of file StateHeaderAttribute.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/CustomAttributes/Headers/StateHeaderAttribute.cs

5.29 SubPagesController Class Reference

Class designated for [UI](#) subpages.

Inherits [MonoBehaviour](#).

Public Member Functions

- void [OpenPage](#) ([UIPage](#) page)
Showing canvas group sent thought parameter and disabling all others.
- void [AddPage](#) ([UIPage](#) page, out Action openPage)
Adding new [UIPage](#) to this subpages controller
- void [AddPage](#) ([UIPage](#) page, out UnityAction openPage)
Adding new [UIPage](#) to this subpages controller
- void [DelayedOpenPage](#) ([UIPage](#) page)
Enables page interaction after delay
- void [DelayedOpenPage](#) ([UIPage](#) page, float delay)
Enables page interaction after delay
- void [HideAllTables](#) ()
Hides all pages in this subpage controller

5.29.1 Detailed Description

Class designated for [UI](#) subpages.

Definition at line 28 of file [SubPagesController.cs](#).

5.29.2 Member Function Documentation

5.29.2.1 [AddPage\(\)](#) [1/2]

```
void AddPage (
    UIPage page,
    out Action openPage ) [inline]
```

Adding new [UIPage](#) to this subpages controller

Parameters

<i>page</i>	
<i>openPage</i>	

Definition at line 63 of file [SubPagesController.cs](#).

References [SubPagesController.OpenPage\(\)](#).

Referenced by `UIManager.InitializeElements()`.

5.29.2.2 `AddPage()` [2/2]

```
void AddPage (
    UIPage page,
    out UnityAction openPage ) [inline]
```

Adding new [UIPage](#) to this subpages controller

Parameters

<i>page</i>	
<i>openPage</i>	

Definition at line 74 of file `SubPagesController.cs`.

References `SubPagesController.OpenPage()`.

5.29.2.3 `DelayedOpenPage()` [1/2]

```
void DelayedOpenPage (
    UIPage page ) [inline]
```

Enables page interaction after delay

Parameters

<i>page</i>	
-------------	--

Definition at line 99 of file `SubPagesController.cs`.

5.29.2.4 `DelayedOpenPage()` [2/2]

```
void DelayedOpenPage (
    UIPage page,
    float delay ) [inline]
```

Enables page interaction after delay

Parameters

<i>page</i>	
<i>delay</i>	

Definition at line 109 of file SubPagesController.cs.

References `UIStateTools.ChangeGroupState()`, and `SubPagesController.HideAllTables()`.

5.29.2.5 HideAllTables()

```
void HideAllTables ( ) [inline]
```

Hides all pages in this subpage controller

Definition at line 120 of file SubPagesController.cs.

References `UIStateTools.ChangeGroupState()`.

Referenced by `SubPagesController.DelayedOpenPage()`, and `SubPagesController.OpenPage()`.

5.29.2.6 OpenPage()

```
void OpenPage (
    UIPage page ) [inline]
```

Showing canvas group sent thought parameter and disabling all others.

Parameters

<i>page</i>	
-------------	--

Definition at line 50 of file SubPagesController.cs.

References `UIStateTools.ChangeGroupState()`, and `SubPagesController.HideAllTables()`.

Referenced by `SubPagesController.AddPage()`.

The documentation for this class was generated from the following file:

- `CoreManager/Assets/CorePlugin/UISystem/UI/SubPagesController.cs`

5.30 SymbolDefiner Class Reference

Class for "Scripting Define Symbols" defining from CoreManager Inspector.

See also

[CorePlugin.Core.CoreManager](#)

Public Member Functions

- void [ShowSymbolsButtons](#) ()
Shows buttons in Inspector.
- void [OnEnable](#) ()
Check for whether the are already defined symbols OnEnable

5.30.1 Detailed Description

Class for "Scripting Define Symbols" defining from CoreManager Inspector.

See also

[CorePlugin.Core.CoreManager](#)

Definition at line 28 of file SymbolDefiner.cs.

5.30.2 Member Function Documentation

5.30.2.1 OnEnable()

```
void OnEnable ( ) [inline]
```

Check for whether the are already defined symbols OnEnable

Definition at line 87 of file SymbolDefiner.cs.

5.30.2.2 ShowSymbolsButtons()

```
void ShowSymbolsButtons ( ) [inline]
```

Shows buttons in Inspector.

Definition at line 36 of file SymbolDefiner.cs.

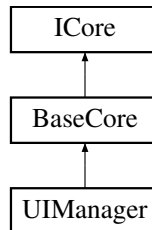
The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/Editor/SymbolDefiner.cs

5.31 UIManager Class Reference

Base implementation of [UIManager](#).

Inheritance diagram for UIManager:



Public Member Functions

- override void [InitializeElements](#) ()
Initializing all managers elements.

5.31.1 Detailed Description

Base implementation of [UIManager](#).

Definition at line 30 of file UIManager.cs.

5.31.2 Member Function Documentation

5.31.2.1 InitializeElements()

```
override void InitializeElements ( ) [inline], [virtual]
```

Initializing all managers elements.

Reimplemented from [BaseCore](#).

Definition at line 38 of file UIManager.cs.

References [SubPagesController.AddPage\(\)](#).

The documentation for this class was generated from the following file:

- [CoreManager/Assets/CorePlugin/UISystem/UIManager.cs](#)

5.32 UIPage Class Reference

UI page implementation

Inherits MonoBehaviour.

Public Member Functions

- [UIPage Initialize \(\)](#)
Initializes elements for this [UIPage](#)

5.32.1 Detailed Description

UI page implementation

Definition at line 28 of file UIPage.cs.

5.32.2 Member Function Documentation

5.32.2.1 Initialize()

```
UIPage Initialize ( ) [inline]
```

Initializes elements for this [UIPage](#)

Returns

Definition at line 49 of file UIPage.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/UISystem/UI/UIPage.cs

5.33 UIStateTools Class Reference

UI state tool for canvas groups

Static Public Member Functions

- static void [ChangeGroupState](#) (CanvasGroup group, bool isVisible)
Changing canvas visibility and interactivity
- static void [ChangeCursorState](#) (bool state)
Changing state of mouse cursor
- static IEnumerator [ChangeGroupState](#) (CanvasGroup group, bool isVisible, float delay)
Changing canvas visibility and interactivity after delay

5.33.1 Detailed Description

UI state tool for canvas groups

Definition at line 24 of file UIStateTools.cs.

5.33.2 Member Function Documentation

5.33.2.1 ChangeCursorState()

```
static void ChangeCursorState (
    bool state ) [inline], [static]
```

Changing state of mouse cursor

Parameters

<i>state</i>	
--------------	--

Definition at line 42 of file UIStateTools.cs.

5.33.2.2 ChangeGroupState() [1/2]

```
static void ChangeGroupState (
    CanvasGroup group,
    bool isVisible ) [inline], [static]
```

Changing canvas visibility and interactivity

Parameters

<i>group</i>	
<i>isVisible</i>	

Definition at line 31 of file UIStateTools.cs.

Referenced by SubPagesController.DelayedOpenPage(), SubPagesController.HideAllTables(), and SubPagesController.OpenPage().

5.33.2.3 ChangeGroupState() [2/2]

```
static IEnumerator ChangeGroupState (
    CanvasGroup group,
    bool isVisible,
    float delay ) [inline], [static]
```

Changing canvas visibility and interactivity after delay

Parameters

<i>group</i>	
<i>isVisible</i>	
<i>delay</i>	

Definition at line 54 of file UIStateTools.cs.

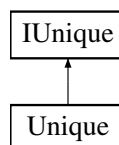
The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/Extensions/UIStateTools.cs

5.34 Unique Class Reference

Base class for classes that need to be saved in Json file.

Inheritance diagram for Unique:



5.34.1 Detailed Description

Base class for classes that need to be saved in Json file.

Definition at line 26 of file Unique.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/Serializable/Unique.cs

5.35 UnityEditorExtension Class Reference

[Extensions](#) for Unity [Editor](#) classes

Static Public Member Functions

- static void [HelpBox](#) (string message, MessageType type)
Override for default Inspector HelpBox with RTF text
- static void [HelpBox](#) (string message, MessageType type, GUIStyle style)
Override for default Inspector HelpBox with style
- static string [IconName](#) (MessageType type)
Getting Icon Name from Unity Inspector

5.35.1 Detailed Description

[Extensions](#) for Unity [Editor](#) classes

Definition at line 24 of file UnityEditorExtension.cs.

5.35.2 Member Function Documentation

5.35.2.1 HelpBox() [1/2]

```
static void HelpBox (
    string message,
    MessageType type ) [inline], [static]
```

Override for default Inspector HelpBox with RTF text

Parameters

<i>message</i>	
<i>type</i>	

Definition at line 31 of file UnityEditorExtension.cs.

5.35.2.2 HelpBox() [2/2]

```
static void HelpBox (
    string message,
    MessageType type,
    GUIStyle style ) [inline], [static]
```

Override for default Inspector HelpBox with style

Parameters

<i>message</i>	
<i>type</i>	
<i>style</i>	

Definition at line 43 of file UnityEditorExtension.cs.

References UnityEditorExtension.IconName().

5.35.2.3 IconName()

```
static string IconName (
    MessageType type ) [inline], [static]
```

Getting Icon Name from Unity Inspector

Parameters

<i>type</i>	
-------------	--

Returns

Definition at line 55 of file UnityEditorExtension.cs.

Referenced by UnityEditorExtension.HelpBox().

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/Extensions/UnityEditorExtension.cs

5.36 UnityExtensions Class Reference

Extension class for default Unity classes

Static Public Member Functions

- static bool [IsNotNullAndNotEqual](#) (this [IUnique](#) lci, [IUnique](#) rci)
Checks whether the left item is null and doesn't equal right item
- static bool [IsNotNullAndEqual](#) (this [IUnique](#) lci, [IUnique](#) rci)
Checks whether the left item is null and equals right item
- static void [Clear](#)< T > (ref List< T > list)
Clearing list and destroying its items
- static bool [TryFindObjectOfType](#)< T > (out T result)
Trying to find object on scene that was inherited from T
- static bool [TryFindObjectTypesOfType](#)< T > (out IEnumerable< T > result)
Trying to find objects on scene that were inherited from T
- static bool [TryFindObjectTypesOfType](#)< T > (out IList< T > result)
Trying to find objects on scene that were inherited from T and return a list of said objects

5.36.1 Detailed Description

Extension class for default Unity classes

Definition at line 29 of file UnityExtensions.cs.

5.36.2 Member Function Documentation

5.36.2.1 Clear< T >()

```
static void Clear< T > (  
    ref List< T > list ) [inline], [static]
```

Clearing list and destroying its items

Parameters

<i>list</i>	
-------------	--

Template Parameters

<i>T</i>	
----------	--

Type Constraints

T* : *MonoBehaviour

Definition at line 58 of file UnityExtensions.cs.

5.36.2.2 IsNotNullAndEqual()

```
static bool IsNotNullAndEqual (  
    this IUnique lci,  
    IUnique rci ) [inline], [static]
```

Checks whether the left item is null and equals right item

Parameters

<i>lci</i>	Left compare item
<i>rci</i>	Right compare item

Returns

Definition at line 48 of file UnityExtensions.cs.

5.36.2.3 IsNotNullAndNotEqual()

```
static bool IsNotNullAndNotEqual (
    this IUnique lci,
    IUnique rci ) [inline], [static]
```

Checks whether the left item is null and doesn't equal right item

Parameters

<i>lci</i>	Left compare item
<i>rci</i>	Right compare item

Returns

Definition at line 37 of file UnityExtensions.cs.

5.36.2.4 TryFindObjectOfType< T >()

```
static bool TryFindObjectOfType< T > (
    out T result ) [inline], [static]
```

Trying to find object on scene that was inherited from T

Parameters

<i>result</i>	
---------------	--

Template Parameters

<i>T</i>	
----------	--

Returns

Definition at line 99 of file UnityExtensions.cs.

5.36.2.5 TryFindObjectOfType< T >() [1/2]

```
static bool TryFindObjectOfType< T > (
    out IEnumerable< T > result ) [inline], [static]
```

Trying to find objects on scene that were inherited from T

Parameters

<i>result</i>	
---------------	--

Template Parameters

<i>T</i>	
----------	--

Returns

Definition at line 112 of file UnityExtensions.cs.

5.36.2.6 TryFindObjectOfType< T >() [2/2]

```
static bool TryFindObjectOfType< T > (
    out IList< T > result ) [inline], [static]
```

Trying to find objects on scene that were inherited from T and return a list of said objects

Parameters

<i>result</i>	
---------------	--

Template Parameters

<i>T</i>	
----------	--

Returns

Definition at line 124 of file UnityExtensions.cs.

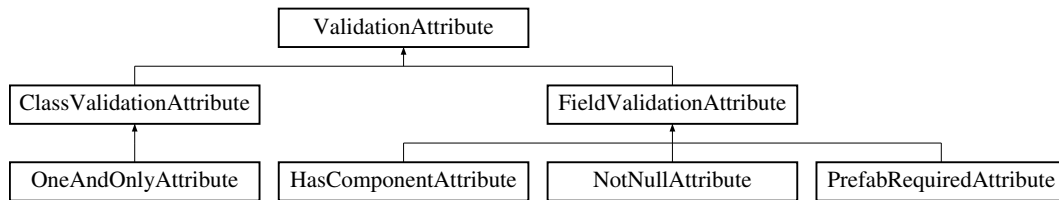
The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/Extensions/UnityExtensions.cs

5.37 ValidationAttribute Class Reference

Base attribute for validation

Inheritance diagram for ValidationAttribute:



5.37.1 Detailed Description

Base attribute for validation

Definition at line 23 of file ValidationAttribute.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/CustomAttributes/Validation/Base/ValidationAttribute.cs

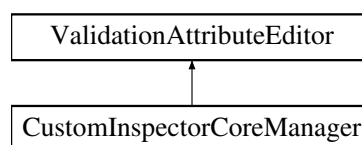
5.38 ValidationAttributeEditor Class Reference

Custom editor for Unity.Object class. If you want to create your own custom editor inherit from this class.

See also

[CorePlugin.CustomAttributes.Validation.Base.ValidationAttribute](#), [CorePlugin.CustomAttributes.Validation.Base.FieldValidationAttribute](#), [CorePlugin.CustomAttributes.Validation.Base.ClassValidationAttribute](#)

Inheritance diagram for ValidationAttributeEditor:



5.38.1 Detailed Description

Custom editor for Unity.Object class. If you want to create your own custom editor inherit from this class.

See also

[CorePlugin.CustomAttributes.Validation.Base.ValidationAttribute](#), [CorePlugin.CustomAttributes.Validation.Base.FieldValidationAttribute](#), [CorePlugin.CustomAttributes.Validation.Base.ClassValidationAttribute](#)

Definition at line 36 of file ValidationAttributeEditor.cs.

The documentation for this class was generated from the following file:

- CoreManager/Assets/CorePlugin/CustomAttributes/Editor/ValidationAttributeEditor.cs

Index

AddData< T >
 SceneDataHandler, 38
AddHandler
 EventInitializer, 19
AddPage
 SubPagesController, 41, 42
AskReference< T >
 ReferenceDistributor, 35

BaseCore, 15
 InitializeElements, 15

ChangeCursorState
 UIStateTools, 47
ChangeGroupState
 UIStateTools, 47, 48
ClassValidationAttribute, 16
Clear< T >
 UnityExtensions, 51
CoreManager, 16
CorePlugin, 9
CorePlugin.Core, 9
CorePlugin.Core.Interface, 9
CorePlugin.Cross, 9
CorePlugin.Cross.Events, 9
CorePlugin.Cross.Events.Interface, 10
CorePlugin.Cross.SceneData, 10
CorePlugin.Cross.SceneData.Interface, 10
CorePlugin.CustomAttributes, 10
CorePlugin.CustomAttributes.Editor, 10
CorePlugin.CustomAttributes.Editor.Drawers, 11
CorePlugin.CustomAttributes.Headers, 11
CorePlugin.CustomAttributes.Validation, 11
CorePlugin.CustomAttributes.Validation.Base, 11
CorePlugin.Editor, 12
CorePlugin.Extensions, 12
CorePlugin.FileSystem, 12
CorePlugin.Interface, 12
CorePlugin.Logger, 13
CorePlugin.ReferenceDistribution, 13
CorePlugin.ReferenceDistribution.Interface, 13
CorePlugin.Samples, 13
CorePlugin.Samples.Scripts, 13
CorePlugin.Samples.Scripts.Demo, 13
CorePlugin.Samples.Scripts.EventTypes, 13
CorePlugin.Samples.Scripts.Managers, 13
CorePlugin.Samples.Scripts.Model, 13
CorePlugin.Serializable, 13
CorePlugin.Serializable.Interface, 14
CorePlugin.Singletons, 14

CorePlugin.UISystem, 14
CorePlugin.UISystem.UI, 14
CustomInspectorCoreManager, 17
CustomLogger, 17

DeepClone
 IDeepClone< out out T >, 27
DelayedOpenPage
 SubPagesController, 42

EventInitializer, 18
 AddHandler, 19
 InitializeSubscriptions, 19
 InvokeBase, 19
 RemoveHandler, 19
 Subscribe, 20
 Unsubscribe, 20
EventTypes, 20

FieldValidationAttribute, 21
FileSystem, 21
 FileSystem, 22, 23
 Load< T >, 23, 24
 Save< T >, 24, 25

GetData< T >
 SceneDataHandler, 38
GetReference< T >
 ReferenceDistributor, 36
GetReferences< T >
 ReferenceDistributor, 36
GetSubscribers
 IEventSubscriber, 30

HasComponentAttribute, 25
HelpBox
 UnityEditorExtension, 49
HideAllTables
 SubPagesController, 43

IconName
 UnityEditorExtension, 50
ICore, 26
 InitializeElements, 27
IDeepClone< out out T >, 27
 DeepClone, 27
IDistributingReference, 28
IEventHandler, 28
 InvokeEvents, 29
 Subscribe, 29
 Unsubscribe, 29

- IEventSubscriber, 30
 - GetSubscribers, 30
- Initialize
 - ReferenceDistributor, 36
 - UIPage, 46
- InitializeElements
 - BaseCore, 15
 - ICore, 27
 - UIManager, 45
- InitializeSubscriptions
 - EventInitializer, 19
- InvokeBase
 - EventInitializer, 19
- InvokeEvents
 - EventHandler, 29
- ISceneData, 31
- IsNotNullAndEqual
 - UnityExtensions, 51
- IsNotNullAndNotEqual
 - UnityExtensions, 52
- IUnique, 31
- Load< T >
 - FileSystem, 23, 24
- Named< TName, TKey, TValue >, 32
- NotNullAttribute, 32
- OneAndOnlyAttribute, 33
- OnEnable
 - SymbolDefiner, 44
- OpenPage
 - SubPagesController, 43
- PrefabHeaderAttribute, 33
- PrefabRequiredAttribute, 34
- ReferenceDistributor, 34
 - AskReference< T >, 35
 - GetReference< T >, 36
 - GetReferences< T >, 36
 - Initialize, 36
- ReferencesHeaderAttribute, 37
- RemoveData< T >
 - SceneDataHandler, 39
- RemoveHandler
 - EventInitializer, 19
- Save< T >
 - FileSystem, 24, 25
- SceneDataHandler, 37
 - AddData< T >, 38
 - GetData< T >, 38
 - RemoveData< T >, 39
- SettingsHeaderAttribute, 39
- ShowSymbolsButtons
 - SymbolDefiner, 44
- Singleton< T >, 40
- StateHeaderAttribute, 40
- SubPagesController, 41
 - AddPage, 41, 42
 - DelayedOpenPage, 42
 - HideAllTables, 43
 - OpenPage, 43
- Subscribe
 - EventInitializer, 20
 - EventHandler, 29
- SymbolDefiner, 43
 - OnEnable, 44
 - ShowSymbolsButtons, 44
- TryFindObjectOfType< T >
 - UnityExtensions, 52
- TryFindObjectsOfType< T >
 - UnityExtensions, 52, 53
- UIManager, 45
 - InitializeElements, 45
- UIPage, 46
 - Initialize, 46
- UIStateTools, 46
 - ChangeCursorState, 47
 - ChangeGroupState, 47, 48
- Unique, 48
- UnityEditorExtension, 49
 - HelpBox, 49
 - IconName, 50
- UnityExtensions, 50
 - Clear< T >, 51
 - IsNotNullAndEqual, 51
 - IsNotNullAndNotEqual, 52
 - TryFindObjectOfType< T >, 52
 - TryFindObjectsOfType< T >, 52, 53
- Unsubscribe
 - EventInitializer, 20
 - EventHandler, 29
- ValidationAttribute, 54
- ValidationAttributeEditor, 54