

TriEngine2D
v0.0.6

Generated by Doxygen 1.8.3.1

Mon Feb 11 2013 02:02:39

Contents

1	Main Page	1
2	Namespace Index	3
2.1	Namespace List	3
3	Hierarchical Index	5
3.1	Class Hierarchy	5
4	Class Index	7
4.1	Class List	7
5	File Index	9
5.1	File List	9
6	Namespace Documentation	11
6.1	Package TriDevs	11
6.2	Package TriDevs.TriEngine2D	11
6.3	Package TriDevs.TriEngine2D.Extensions	11
6.4	Package TriDevs.TriEngine2D.Helpers	12
6.5	Package TriDevs.TriEngine2D.Input	12
6.6	Package TriDevs.TriEngine2D.Logging	12
6.7	Package TriDevs.TriEngine2D.Native	12
6.8	Package TriDevs.TriEngine2D.Serializing	12
7	Class Documentation	13
7.1	TriDevs.TriEngine2D.Extensions.EnumerationExtensions Class Reference	13
7.1.1	Detailed Description	13
7.1.2	Member Function Documentation	13
7.1.2.1	Has< T >	13
7.1.2.2	Include< T >	14
7.1.2.3	Missing< T >	14
7.1.2.4	Remove< T >	15
7.2	TriDevs.TriEngine2D.Native.Helpers Class Reference	15
7.2.1	Detailed Description	16

7.2.2	Member Function Documentation	16
7.2.2.1	GetErrorMessage	16
7.2.2.2	GetLastError	16
7.2.2.3	GetLastErrorInfo	16
7.2.2.4	GetLastErrorMessage	17
7.2.2.5	GetLastWin32Exception	17
7.2.2.6	GetWin32Exception	17
7.3	TriDevs.TriEngine2D.Input.IInputManager Interface Reference	17
7.3.1	Detailed Description	19
7.3.2	Member Function Documentation	19
7.3.2.1	ButtonDown	19
7.3.2.2	ButtonPressed	19
7.3.2.3	ButtonReleased	19
7.3.2.4	ButtonUp	20
7.3.2.5	KeyDown	20
7.3.2.6	KeyPressed	20
7.3.2.7	KeyReleased	20
7.3.2.8	KeyUp	21
7.3.2.9	Update	21
7.3.2.10	WheelChange	21
7.3.2.11	WheelChanged	21
7.3.2.12	WheelDown	21
7.3.2.13	WheelUp	22
7.3.3	Property Documentation	22
7.3.3.1	MousePosition	22
7.3.3.2	MouseWheelValue	22
7.3.3.3	MouseX	22
7.3.3.4	MouseY	22
7.3.3.5	this[Key key]	22
7.3.3.6	this[MouseButton button]	22
7.4	TriDevs.TriEngine2D.Input.InputManager Class Reference	23
7.4.1	Detailed Description	24
7.4.2	Constructor & Destructor Documentation	24
7.4.2.1	InputManager	24
7.4.3	Member Function Documentation	24
7.4.3.1	ButtonDown	24
7.4.3.2	ButtonPressed	25
7.4.3.3	ButtonReleased	25
7.4.3.4	ButtonUp	25
7.4.3.5	KeyDown	26

7.4.3.6	KeyPressed	26
7.4.3.7	KeyReleased	26
7.4.3.8	KeyUp	27
7.4.3.9	Update	27
7.4.3.10	WheelChange	27
7.4.3.11	WheelChanged	28
7.4.3.12	WheelDown	28
7.4.3.13	WheelUp	28
7.4.4	Property Documentation	28
7.4.4.1	MousePosition	28
7.4.4.2	MouseWheelValue	28
7.4.4.3	MouseX	28
7.4.4.4	MouseY	28
7.4.4.5	this[Key key]	28
7.4.4.6	this[MouseButton button]	28
7.5	TriDevs.TriEngine2D.Helpers.IO Class Reference	29
7.5.1	Detailed Description	29
7.5.2	Member Function Documentation	29
7.5.2.1	GetAbsolutePath	29
7.6	TriDevs.TriEngine2D.Logging.LogManager Class Reference	29
7.6.1	Detailed Description	30
7.6.2	Member Function Documentation	30
7.6.2.1	ClearOldLogs	30
7.6.2.2	DestroyConsole	30
7.6.2.3	GetLogger	31
7.6.2.4	LoadConfig	31
7.6.2.5	SetupConsole	32
7.7	TriDevs.TriEngine2D.Input.NullInputManager Class Reference	32
7.7.1	Detailed Description	34
7.7.2	Member Function Documentation	34
7.7.2.1	ButtonDown	34
7.7.2.2	ButtonPressed	34
7.7.2.3	ButtonReleased	35
7.7.2.4	ButtonUp	35
7.7.2.5	KeyDown	35
7.7.2.6	KeyPressed	36
7.7.2.7	KeyReleased	36
7.7.2.8	KeyUp	36
7.7.2.9	Update	37
7.7.2.10	WheelChange	37

7.7.2.11	WheelChanged	37
7.7.2.12	WheelDown	37
7.7.2.13	WheelUp	38
7.7.3	Property Documentation	38
7.7.3.1	MousePosition	38
7.7.3.2	MouseWheelValue	38
7.7.3.3	MouseX	38
7.7.3.4	MouseY	38
7.7.3.5	this[Key key]	38
7.7.3.6	this[MouseButton button]	38
7.8	TriDevs.TriEngine2D.Point Struct Reference	38
7.8.1	Detailed Description	38
7.8.2	Constructor & Destructor Documentation	39
7.8.2.1	Point	39
7.8.3	Member Data Documentation	39
7.8.3.1	X	39
7.8.3.2	Y	39
7.9	TriDevs.TriEngine2D.Serializing.Serializer Class Reference	39
7.9.1	Detailed Description	39
7.9.2	Member Function Documentation	39
7.9.2.1	Deserialize< T >	39
7.9.2.2	Serialize< T >	40
7.9.2.3	Serialize< T >	40
7.10	TriDevs.TriEngine2D.Services Class Reference	41
7.10.1	Detailed Description	41
7.10.2	Member Function Documentation	41
7.10.2.1	Provide	41
7.10.3	Property Documentation	42
7.10.3.1	Input	42
7.11	TriDevs.TriEngine2D.Extensions.StringExtensions Class Reference	42
7.11.1	Detailed Description	42
7.11.2	Member Function Documentation	42
7.11.2.1	Replace	42
7.11.2.2	Replace	43
7.11.2.3	ReplaceFirst	43
7.12	TriDevs.TriEngine2D.Helpers.Threading Class Reference	43
7.12.1	Detailed Description	44
7.12.2	Member Function Documentation	44
7.12.2.1	SetCurrentThreadName	44
7.13	TriDevs.TriEngine2D.Version Class Reference	44

7.13.1 Detailed Description	44
7.13.2 Member Data Documentation	45
7.13.2.1 Major	45
7.13.2.2 Minor	45
7.13.2.3 Patch	45
7.13.2.4 Suffix	45
7.13.2.5 VersionStringFormat	45
7.13.2.6 VersionStringFormatWithSuffix	45
7.13.3 Property Documentation	45
7.13.3.1 VersionString	45
7.14 TriDevs.TriEngine2D.Native.WinAPI Class Reference	45
7.14.1 Detailed Description	46
7.14.2 Member Function Documentation	46
7.14.2.1 AllocConsole	46
7.14.2.2 FreeConsole	46
7.14.2.3 GetStdHandle	47
7.14.3 Member Data Documentation	47
7.14.3.1 CODE_PAGE	47
7.14.3.2 STD_OUTPUT_HANDLE	47
8 File Documentation	49
8.1 README.md File Reference	49
8.2 TriDevs.TriEngine2D/Extensions/EnumerationExtensions.cs File Reference	49
8.3 TriDevs.TriEngine2D/Extensions/StringExtensions.cs File Reference	49
8.4 TriDevs.TriEngine2D/Helpers/IO.cs File Reference	49
8.5 TriDevs.TriEngine2D/Helpers/Threading.cs File Reference	50
8.6 TriDevs.TriEngine2D/Input/IInputManager.cs File Reference	50
8.7 TriDevs.TriEngine2D/Input/InputManager.cs File Reference	50
8.8 TriDevs.TriEngine2D/Input/NullInputManager.cs File Reference	50
8.9 TriDevs.TriEngine2D/Logging/LogManager.cs File Reference	51
8.10 TriDevs.TriEngine2D/Native/Helpers.cs File Reference	51
8.11 TriDevs.TriEngine2D/Native/WinAPI.cs File Reference	51
8.12 TriDevs.TriEngine2D/Point.cs File Reference	51
8.13 TriDevs.TriEngine2D/Properties/AssemblyInfo.cs File Reference	52
8.14 TriDevs.TriEngine2D/Serializing/Serializer.cs File Reference	52
8.15 TriDevs.TriEngine2D/Services.cs File Reference	52
8.16 TriDevs.TriEngine2D/Version.cs File Reference	52

Chapter 1

Main Page

2D general-purpose engine in C#/OpenGL

IRC

[TriDevs](#) has an IRC channel, feel free to hop in if you have a question about anything: **Server:** irc.kottnet.net **Port:** 6667, 6697 (SSL) **Channel:** [TriDevs](#)

The channel topic contains further info.

License

Copyright © 2013 by [Adam Hellberg](#), [Sijmen Schoon](#) and [Preston Shumway](#).

TriEngine2D is licensed under the [MIT License](#), more info can be found in the **LICENSE** file.

Contributing

You are free to fork this project and make your own changes, as long as you follow the MIT License.

If you want to make a pull request, please do so to the [main project](#) and not any of the "official" forks.

For your pull request to be accepted, please follow our coding style:

- Indent with 4 spaces, not tabs.
- Curly braces placed on next line.
- All **public** methods, accessors and members must be properly documented.
- Use sensible variable names that describe what they are for.
- Method declarations written as:

```
“c# public void Hello(string world) “
```

- If your method accepts many parameters, it can be useful to put parameters on separate lines, as per this style:

```
“c# public void Hello(string world, bool print) “
```

- Please write tests for your code (not strictly required, but it's a plus)

By looking through the current source code, you should be able to get a good understanding of the formatting we use.

If you're using Visual Studio, you can change the indent behaviour by going to: **Tools -> Options -> Text Editor -> C# -> Tabs** and make sure "Insert spaces" is checked.

If you write tests for your code, please place these tests in their own project: "Namespace >.Tests", create said project if it does not exist (of type Class Library).

We use NUnit as test framework, feel free to use something else if you want to, but make sure you document what framework you are using and that it is freely available for anyone to obtain.

Dependencies

TriEngine2D depends on [log4net](#), which is included in the **libs/log4net** folder.

TriEngine2D depends on [OpenTK](#), this is not included and you will have to build/install it yourself. OpenTK depends on OpenGL drivers being installed, they are usually in your normal video card drivers.

TriEngine2D depends on [Json.NET](#), this is not included, but is specified in the NuGet package config. If you [properly configure your NuGet settings](#), NuGet will automatically download Json.NET when building any projects that depend on it.

If you want to run the tests you will need to have [NUnit](#) installed.

Chapter 2

Namespace Index

2.1 Namespace List

Here is a list of all namespaces with brief descriptions:

TriDevs	11
TriDevs.TriEngine2D	11
TriDevs.TriEngine2D.Extensions	11
TriDevs.TriEngine2D.Helpers	12
TriDevs.TriEngine2D.Input	12
TriDevs.TriEngine2D.Logging	12
TriDevs.TriEngine2D.Native	12
TriDevs.TriEngine2D.Serializing	12

Chapter 3

Hierarchical Index

3.1 Class Hierarchy

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

TriDevs.TriEngine2D.Extensions.EnumerationExtensions	13
TriDevs.TriEngine2D.Native.Helpers	15
TriDevs.TriEngine2D.Input.IInputManager	17
TriDevs.TriEngine2D.Input.InputManager	23
TriDevs.TriEngine2D.Input.NullInputManager	32
TriDevs.TriEngine2D.Helpers.IO	29
TriDevs.TriEngine2D.Logging.LogManager	29
TriDevs.TriEngine2D.Point	38
TriDevs.TriEngine2D.Serializing.Serializer	39
TriDevs.TriEngine2D.Services	41
TriDevs.TriEngine2D.Extensions.StringExtensions	42
TriDevs.TriEngine2D.Helpers.Threading	43
TriDevs.TriEngine2D.Version	44
TriDevs.TriEngine2D.Native.WinAPI	45

Chapter 4

Class Index

4.1 Class List

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

TriDevs.TriEngine2D.Extensions.EnumerationExtensions	
Extensions for System.Enum.	13
TriDevs.TriEngine2D.Native.Helpers	
Helper class with various methods to help native coding and debugging.	15
TriDevs.TriEngine2D.Input.IInputManager	
Provides various methods to query input devices like the keyboard.	17
TriDevs.TriEngine2D.Input.InputManager	
Input manager interfacing with input methods provided by a GameWindow.	23
TriDevs.TriEngine2D.Helpers.IO	
Provides various helper functions for doing IO operations.	29
TriDevs.TriEngine2D.Logging.LogManager	
Class to manage logging. ILog interfaces should be obtained from this class' methods, as opposed to calling default log4net methods.	29
TriDevs.TriEngine2D.Input.NullInputManager	
Used as a fallback InputManager object when the service locator fails to find one.	32
TriDevs.TriEngine2D.Point	
A struct representing an X/Y coordinate.	38
TriDevs.TriEngine2D.Serializing.Serializer	
Provides serialization methods.	39
TriDevs.TriEngine2D.Services	
Provides different game-related service interfaces.	41
TriDevs.TriEngine2D.Extensions.StringExtensions	
Extensions for System.String	42
TriDevs.TriEngine2D.Helpers.Threading	
Provides various helper functions for doing threading operations.	43
TriDevs.TriEngine2D.Version	
Version class specifying the version of this project.	44
TriDevs.TriEngine2D.Native.WinAPI	
Holds various WinAPI stuff.	45

Chapter 5

File Index

5.1 File List

Here is a list of all files with brief descriptions:

TriDevs.TriEngine2D/ Point.cs	51
TriDevs.TriEngine2D/ Services.cs	52
TriDevs.TriEngine2D/ Version.cs	52
TriDevs.TriEngine2D/Extensions/ EnumerationExtensions.cs	49
TriDevs.TriEngine2D/Extensions/ StringExtensions.cs	49
TriDevs.TriEngine2D/Helpers/ IO.cs	49
TriDevs.TriEngine2D/Helpers/ Threading.cs	50
TriDevs.TriEngine2D/Input/ IInputManager.cs	50
TriDevs.TriEngine2D/Input/ InputManager.cs	50
TriDevs.TriEngine2D/Input/ NullInputManager.cs	50
TriDevs.TriEngine2D/Logging/ LogManager.cs	51
TriDevs.TriEngine2D/Native/ Helpers.cs	51
TriDevs.TriEngine2D/Native/ WinAPI.cs	51
TriDevs.TriEngine2D/Properties/ AssemblyInfo.cs	52
TriDevs.TriEngine2D/Serializing/ Serializer.cs	52

Chapter 6

Namespace Documentation

6.1 Package TriDevs

Namespaces

- package [TriEngine2D](#)

6.2 Package TriDevs.TriEngine2D

Namespaces

- package [Extensions](#)
- package [Helpers](#)
- package [Input](#)
- package [Logging](#)
- package [Native](#)
- package [Serializing](#)

Classes

- struct [Point](#)
A struct representing an X/Y coordinate.
- class [Services](#)
Provides different game-related service interfaces.
- class [Version](#)
[Version](#) class specifying the version of this project.

6.3 Package TriDevs.TriEngine2D.Extensions

Classes

- class [EnumerationExtensions](#)
[Extensions](#) for System.Enum.
- class [StringExtensions](#)
[Extensions](#) for System.String

6.4 Package TriDevs.TriEngine2D.Helpers

Classes

- class [IO](#)
Provides various helper functions for doing [IO](#) operations.
- class [Threading](#)
Provides various helper functions for doing threading operations.

6.5 Package TriDevs.TriEngine2D.Input

Classes

- interface [IInputManager](#)
Provides various methods to query input devices like the keyboard.
- class [InputManager](#)
[Input](#) manager interfacing with input methods provided by a [GameWindow](#).
- class [NullInputManager](#)
Used as a fallback [InputManager](#) object when the service locator fails to find one.

6.6 Package TriDevs.TriEngine2D.Logging

Classes

- class [LogManager](#)
Class to manage logging. [ILog](#) interfaces should be obtained from this class' methods, as opposed to calling default [log4net](#) methods.

6.7 Package TriDevs.TriEngine2D.Native

Classes

- class [Helpers](#)
Helper class with various methods to help native coding and debugging.
- class [WinAPI](#)
Holds various [WinAPI](#) stuff.

6.8 Package TriDevs.TriEngine2D.Serializing

Classes

- class [Serializer](#)
Provides serialization methods.

Chapter 7

Class Documentation

7.1 TriDevs.TriEngine2D.Extensions.EnumerationExtensions Class Reference

[Extensions](#) for System.Enum.

Static Public Member Functions

- static T [Include< T >](#) (this Enum value, T append)
Includes an enumerated type and returns the new value.
- static T [Remove< T >](#) (this Enum value, T remove)
Removes an enumerated type and returns the new value.
- static bool [Has< T >](#) (this Enum value, T check)
Checks if an enumerated type contains a value.
- static bool [Missing< T >](#) (this Enum obj, T value)
Checks if an enumerated type is missing a value.

7.1.1 Detailed Description

[Extensions](#) for System.Enum.

7.1.2 Member Function Documentation

7.1.2.1 static bool TriDevs.TriEngine2D.Extensions.EnumerationExtensions.Has< T > (this Enum *value*, T *check*)
[static]

Checks if an enumerated type contains a value.

Template Parameters

<i>T</i>	The enum type.
----------	----------------

Parameters

<i>value</i>	The enum to check.
<i>check</i>	Value to check for.

Returns

True if the enum has the value(s), false otherwise.

```

148     {
149         Type type = value.GetType();
150
151         //determine the values
152         var parsed = new _Value(check, type);
153         if (parsed.Signed.HasValue) //if (parsed.Signed is long)
154         {
155             return (Convert.ToInt64(value) & (long)parsed.Signed) == (long)parsed.Signed;
156         }
157         if (parsed.Unsigned.HasValue) //if (parsed.Unsigned is ulong)
158         {
159             return (Convert.ToUInt64(value) & (ulong)parsed.Unsigned) == (ulong)parsed.Unsigned;
160         }
161         return false;
162     }

```

7.1.2.2 `static T TriDevs.TriEngine2D.Extensions.EnumerationExtensions.Include< T > (this Enum value, T append)`
`[static]`

Includes an enumerated type and returns the new value.

Template Parameters

<i>T</i>	The enum type.
----------	----------------

Parameters

<i>value</i>	The enum to append to.
<i>append</i>	Value to append.

Returns

New enum T with the new values.

```

94     {
95         var type = value.GetType();
96
97         //determine the values
98         object result = value;
99         var parsed = new _Value(append, type);
100         if (parsed.Signed.HasValue) //if (parsed.Signed is long)
101         {
102             result = Convert.ToInt64(value) | (long)parsed.Signed;
103         }
104         else if (parsed.Unsigned.HasValue) //else if (parsed.Unsigned is ulong)
105         {
106             result = Convert.ToUInt64(value) | (ulong)parsed.Unsigned;
107         }
108
109         //return the final value
110         return (T)Enum.Parse(type, result.ToString());
111     }

```

7.1.2.3 `static bool TriDevs.TriEngine2D.Extensions.EnumerationExtensions.Missing< T > (this Enum obj, T value)`
`[static]`

Checks if an enumerated type is missing a value.

Template Parameters

<i>T</i>	The enum type.
----------	----------------

Parameters

<i>obj</i>	The enum to check.
<i>value</i>	Value to check for.

Returns

True if the enum is missing the value(s), false otherwise.

```

172     {
173         return !Has(obj, value);
174     }
```

7.1.2.4 `static T TriDevs.TriEngine2D.Extensions.EnumerationExtensions.Remove< T > (this Enum value, T remove)`
[static]

Removes an enumerated type and returns the new value.

Template Parameters

<i>T</i>	The enum type.
----------	----------------

Parameters

<i>value</i>	The enum to remove from.
<i>remove</i>	Value to remove.

Returns

New enum T with the value(s) removed.

```

121     {
122         Type type = value.GetType();
123
124         //determine the values
125         object result = value;
126         var parsed = new _Value(remove, type);
127         if (parsed.Signed.HasValue) //if (parsed.Signed is long)
128         {
129             result = Convert.ToInt64(value) & ~(long)parsed.Signed;
130         }
131         else if (parsed.Unsigned.HasValue) //else if (parsed.Unsigned is ulong)
132         {
133             result = Convert.ToUInt64(value) & ~(ulong)parsed.Unsigned;
134         }
135
136         //return the final value
137         return (T)Enum.Parse(type, result.ToString());
138     }
```

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

- TriDevs.TriEngine2D/Extensions/[EnumerationExtensions.cs](#)

7.2 TriDevs.TriEngine2D.Native.Helpers Class Reference

Helper class with various methods to help native coding and debugging.

Static Public Member Functions

- static int [GetLastError](#) ()

- Gets the last error that was thrown.*
- static int [GetLastErrorInfo](#) (out string message)
Gets information about the last error that was thrown.
- static string [GetErrorMessage](#) (int err)
Gets the error message associated with an error code.
- static string [GetLastErrorMessage](#) ()
Gets the error message associated with the last thrown error.
- static Win32Exception [GetWin32Exception](#) (int err)
Gets the Win32Exception associated with the specified error code.
- static Win32Exception [GetLastWin32Exception](#) ()
Gets the Win32Exception associated with the last thrown error.

7.2.1 Detailed Description

Helper class with various methods to help native coding and debugging.

7.2.2 Member Function Documentation

7.2.2.1 static string TriDevs.TriEngine2D.Native.Helpers.GetErrorMessage (int err) [static]

Gets the error message associated with an error code.

Parameters

<i>err</i>	The error code to get information about.
------------	--

Returns

The error message.

```

61         {
62             return GetWin32Exception(err).Message;
63         }
```

7.2.2.2 static int TriDevs.TriEngine2D.Native.Helpers.GetLastError () [static]

Gets the last error that was thrown.

Returns

The error code associated with the thrown error.

```

39         {
40             return Marshal.GetLastWin32Error();
41         }
```

7.2.2.3 static int TriDevs.TriEngine2D.Native.Helpers.GetLastErrorInfo (out string message) [static]

Gets information about the last error that was thrown.

Parameters

<i>message</i>	Will be set to the error message.
----------------	-----------------------------------

Returns

The error code associated with the thrown error.

```

49      {
50          var err = GetLastError();
51          message = GetWin32Exception(err).Message;
52          return err;
53      }

```

7.2.2.4 static string TriDevs.TriEngine2D.Native.Helpers.GetLastErrorMessage () [static]

Gets the error message associated with the last thrown error.

Returns

The error message.

```

70      {
71          return GetErrorMessage(GetLastError());
72      }

```

7.2.2.5 static Win32Exception TriDevs.TriEngine2D.Native.Helpers.GetLastWin32Exception () [static]

Gets the Win32Exception associated with the last thrown error.

Returns

The Win32Exception.

```

89      {
90          return GetWin32Exception(GetLastError());
91      }

```

7.2.2.6 static Win32Exception TriDevs.TriEngine2D.Native.Helpers.GetWin32Exception (int err) [static]

Gets the Win32Exception associated with the specified error code.

Parameters

<i>err</i>	The error code.
------------	-----------------

Returns

The Win32Exception for the provided error code.

```

80      {
81          return new Win32Exception(err);
82      }

```

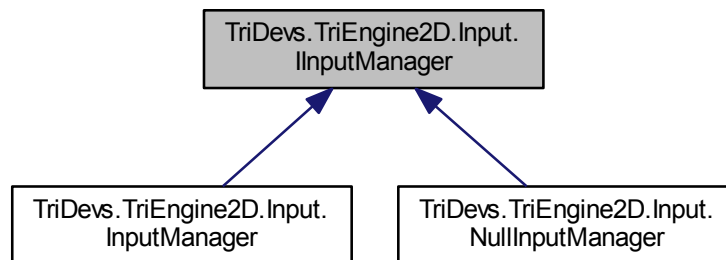
The documentation for this class was generated from the following file:

- TriDevs.TriEngine2D/Native/[Helpers.cs](#)

7.3 TriDevs.TriEngine2D.Input.IInputManager Interface Reference

Provides various methods to query input devices like the keyboard.

Inheritance diagram for TriDevs.TriEngine2D.Input.IInputManager:



Public Member Functions

- void [Update](#) ()
Updates the input manager, refreshing all current and previous states.
- bool [KeyUp](#) (Key key)
Returns whether or not the specified key is currently unpressed.
- bool [KeyDown](#) (Key key)
Returns whether or not the specified key is currently being pressed.
- bool [KeyPressed](#) (Key key)
Returns whether or not the specified key has been pressed.
- bool [KeyReleased](#) (Key key)
Returns whether or not the specified key has been released.
- bool [ButtonUp](#) (MouseButton button)
Returns whether or not the specified mouse button is currently unpressed.
- bool [ButtonDown](#) (MouseButton button)
Returns whether or not the specified mouse button is currently being pressed.
- bool [ButtonPressed](#) (MouseButton button)
Returns whether or not the specified mouse button has been pressed.
- bool [ButtonReleased](#) (MouseButton button)
Returns whether or not the specified mouse button has been released.
- bool [WheelUp](#) ()
Returns whether the mouse wheel was scrolled up.
- bool [WheelDown](#) ()
Returns whether the mouse wheel was scrolled down.
- bool [WheelChanged](#) ()
Returns whether the mouse wheel scrolled at all.
- int [WheelChange](#) ()
Returns the mouse wheel's change in value.

Properties

- int [MouseX](#) [get]
Gets the absolute X position of the pointer, in window pixel coordinates.
- int [MouseY](#) [get]

- Gets the absolute Y position of the pointer, in window pixel coordinates.*
- [Point MousePosition](#) [get]
Gets a [Point](#) representing the position of the mouse pointer, in window pixel coordinates.
- int [MouseWheelValue](#) [get]
Gets the current value of the mouse wheel.
- bool [this\[Key key\]](#) [get]
Gets a boolean value indicating whether the specified [OpenTK.Input.Key](#) is pressed.
- bool [this\[MouseButton button\]](#) [get]
Gets a boolean value indicating whether the specified [OpenTK.Input.MouseButton](#) is pressed.

7.3.1 Detailed Description

Provides various methods to query input devices like the keyboard.

7.3.2 Member Function Documentation

7.3.2.1 bool TriDevs.TriEngine2D.Input.IInputManager.ButtonDown (MouseButton *button*)

Returns whether or not the specified mouse button is currently being pressed.

Parameters

<i>button</i>	The button to query for.
---------------	--------------------------

Returns

True if button is currently being pressed, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

7.3.2.2 bool TriDevs.TriEngine2D.Input.IInputManager.ButtonPressed (MouseButton *button*)

Returns whether or not the specified mouse button has been pressed.

Only returns true if the last state of the mouse button was not pressed.

Parameters

<i>button</i>	Button to query for.
---------------	----------------------

Returns

True if button was pressed, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

7.3.2.3 bool TriDevs.TriEngine2D.Input.IInputManager.ButtonReleased (MouseButton *button*)

Returns whether or not the specified mouse button has been released.

Only returns true if the last state of the button was pressed.

Parameters

<i>button</i>	The button to query for.
---------------	--------------------------

Returns

True if the button was released, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

7.3.2.4 bool TriDevs.TriEngine2D.Input.IInputManager.ButtonUp (MouseButton *button*)

Returns whether or not the specified mouse button is currently unpressed.

Parameters

<i>button</i>	Button to query for.
---------------	----------------------

Returns

True if the button is currently up (not pressed), false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

7.3.2.5 bool TriDevs.TriEngine2D.Input.IInputManager.KeyDown (Key *key*)

Returns whether or not the specified key is currently being pressed.

Parameters

<i>key</i>	Key to query for.
------------	-------------------

Returns

True if key is currently being pressed, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

7.3.2.6 bool TriDevs.TriEngine2D.Input.IInputManager.KeyPressed (Key *key*)

Returns whether or not the specified key has been pressed.

Only returns true if the last state of the key was not pressed.

Parameters

<i>key</i>	Key to query for.
------------	-------------------

Returns

True if key was pressed, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

7.3.2.7 bool TriDevs.TriEngine2D.Input.IInputManager.KeyReleased (Key *key*)

Returns whether or not the specified key has been released.

Only returns true if the last state of the key was pressed.

Parameters

<i>key</i>	Key to query for.
------------	-------------------

Returns

True if key was released, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

7.3.2.8 bool TriDevs.TriEngine2D.Input.IInputManager.KeyUp (Key key)

Returns whether or not the specified key is currently unpressed.

Parameters

<i>key</i>	Key to query for.
------------	-------------------

Returns

True if the key is currently up (not pressed), false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

7.3.2.9 void TriDevs.TriEngine2D.Input.IInputManager.Update ()

Updates the input manager, refreshing all current and previous states.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

7.3.2.10 int TriDevs.TriEngine2D.Input.IInputManager.WheelChange ()

Returns the mouse wheel's change in value.

Returns

Negative value if wheel scrolled down, positive value if scrolled up, zero if not scrolled.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

7.3.2.11 bool TriDevs.TriEngine2D.Input.IInputManager.WheelChanged ()

Returns whether the mouse wheel scrolled at all.

Returns

True if the mouse wheel scrolled, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

7.3.2.12 bool TriDevs.TriEngine2D.Input.IInputManager.WheelDown ()

Returns whether the mouse wheel was scrolled down.

Returns

True if mouse wheel was scrolled down, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

7.3.2.13 `bool TriDevs.TriEngine2D.Input.IInputManager.WheelUp ()`

Returns whether the mouse wheel was scrolled up.

Returns

True if mouse wheel was scrolled up, false otherwise.

Implemented in [TriDevs.TriEngine2D.Input.InputManager](#), and [TriDevs.TriEngine2D.Input.NullInputManager](#).

7.3.3 Property Documentation

7.3.3.1 `Point TriDevs.TriEngine2D.Input.IInputManager.MousePosition` `[get]`

Gets a [Point](#) representing the position of the mouse pointer, in window pixel coordinates.

7.3.3.2 `int TriDevs.TriEngine2D.Input.IInputManager.MouseWheelValue` `[get]`

Gets the current value of the mouse wheel.

7.3.3.3 `int TriDevs.TriEngine2D.Input.IInputManager.MouseX` `[get]`

Gets the absolute X position of the pointer, in window pixel coordinates.

7.3.3.4 `int TriDevs.TriEngine2D.Input.IInputManager.MouseY` `[get]`

Gets the absolute Y position of the pointer, in window pixel coordinates.

7.3.3.5 `bool TriDevs.TriEngine2D.Input.IInputManager.this[Key key]` `[get]`

Gets a boolean value indicating whether the specified `OpenTK.Input.Key` is pressed.

Parameters

<i>key</i>	The key to query.
------------	-------------------

Returns

True if pressed, false otherwise.

7.3.3.6 `bool TriDevs.TriEngine2D.Input.IInputManager.this[MouseButton button]` `[get]`

Gets a boolean value indicating whether the specified `OpenTK.Input.MouseButton` is pressed.

Parameters

<i>button</i>	The button to query.
---------------	----------------------

Returns

True if pressed, false otherwise.

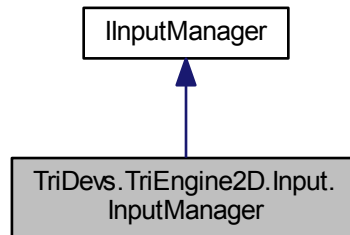
The documentation for this interface was generated from the following file:

- [TriDevs.TriEngine2D/Input/IInputManager.cs](#)

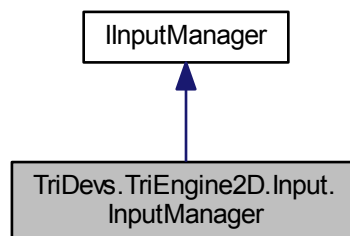
7.4 TriDevs.TriEngine2D.Input.InputManager Class Reference

[Input](#) manager interfacing with input methods provided by a `GameWindow`.

Inheritance diagram for `TriDevs.TriEngine2D.Input.InputManager`:



Collaboration diagram for `TriDevs.TriEngine2D.Input.InputManager`:



Public Member Functions

- [InputManager](#) (`GameWindow window`)
Creates a new [InputManager](#) associated with the specified `GameWindow`.
- void [Update](#) ()
Updates the input manager, refreshing all current and previous states.
- bool [KeyUp](#) (`Key key`)
Returns whether or not the specified key is currently unpressed.
- bool [KeyDown](#) (`Key key`)
Returns whether or not the specified key is currently being pressed.
- bool [KeyPressed](#) (`Key key`)
Returns whether or not the specified key has been pressed.

- bool [KeyReleased](#) (Key key)
Returns whether or not the specified key has been released.
- bool [ButtonUp](#) (MouseButton button)
Returns whether or not the specified mouse button is currently unpressed.
- bool [ButtonDown](#) (MouseButton button)
Returns whether or not the specified mouse button is currently being pressed.
- bool [ButtonPressed](#) (MouseButton button)
Returns whether or not the specified mouse button has been pressed.
- bool [ButtonReleased](#) (MouseButton button)
Returns whether or not the specified mouse button has been released.
- bool [WheelUp](#) ()
Returns whether the mouse wheel was scrolled up.
- bool [WheelDown](#) ()
Returns whether the mouse wheel was scrolled down.
- bool [WheelChanged](#) ()
Returns whether the mouse wheel scrolled at all.
- int [WheelChange](#) ()
Returns the mouse wheel's change in value.

Properties

- int [MouseX](#) [get]
- int [MouseY](#) [get]
- [Point](#) [MousePosition](#) [get]
- int [MouseWheelValue](#) [get]
- bool [this\[Key key\]](#) [get]
- bool [this\[MouseButton button\]](#) [get]

7.4.1 Detailed Description

[Input](#) manager interfacing with input methods provided by a [GameWindow](#).

7.4.2 Constructor & Destructor Documentation

7.4.2.1 [TriDevs.TriEngine2D.Input.InputManager.InputManager](#) ([GameWindow](#) *window*)

Creates a new [InputManager](#) associated with the specified [GameWindow](#).

Parameters

<i>window</i>	The GameWindow this InputManager will interface with.
---------------	---

```

62     {
63         _mouse = window.Mouse;
64     }
```

7.4.3 Member Function Documentation

7.4.3.1 bool [TriDevs.TriEngine2D.Input.InputManager.ButtonDown](#) ([MouseButton](#) *button*)

Returns whether or not the specified mouse button is currently being pressed.

Parameters

<i>button</i>	The button to query for.
---------------	--------------------------

Returns

True if button is currently being pressed, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

101     {
102         return _mouseState[button];
103     }
```

7.4.3.2 bool TriDevs.TriEngine2D.Input.InputManager.ButtonPressed (MouseButton *button*)

Returns whether or not the specified mouse button has been pressed.

Only returns true if the last state of the mouse button was not pressed.

Parameters

<i>button</i>	Button to query for.
---------------	----------------------

Returns

True if button was pressed, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

106     {
107         return _mouseState[button] && !_lastMouseState[button];
108     }
```

7.4.3.3 bool TriDevs.TriEngine2D.Input.InputManager.ButtonReleased (MouseButton *button*)

Returns whether or not the specified mouse button has been released.

Only returns true if the last state of the button was pressed.

Parameters

<i>button</i>	The button to query for.
---------------	--------------------------

Returns

True if the button was released, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

111     {
112         return !_mouseState[button] && _lastMouseState[button];
113     }
```

7.4.3.4 bool TriDevs.TriEngine2D.Input.InputManager.ButtonUp (MouseButton *button*)

Returns whether or not the specified mouse button is currently unpressed.

Parameters

<i>button</i>	Button to query for.
---------------	----------------------

Returns

True if the button is currently up (not pressed), false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

96         {
97             return !_mouseState[button];
98         }

```

7.4.3.5 bool TriDevs.TriEngine2D.Input.InputManager.KeyDown (Key key)

Returns whether or not the specified key is currently being pressed.

Parameters

<i>key</i>	Key to query for.
------------	-------------------

Returns

True if key is currently being pressed, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

81         {
82             return _keyboardState[key];
83         }

```

7.4.3.6 bool TriDevs.TriEngine2D.Input.InputManager.KeyPressed (Key key)

Returns whether or not the specified key has been pressed.

Only returns true if the last state of the key was not pressed.

Parameters

<i>key</i>	Key to query for.
------------	-------------------

Returns

True if key was pressed, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

86         {
87             return _keyboardState[key] && !_lastKeyboardState[key];
88         }

```

7.4.3.7 bool TriDevs.TriEngine2D.Input.InputManager.KeyReleased (Key key)

Returns whether or not the specified key has been released.

Only returns true if the last state of the key was pressed.

Parameters

<i>key</i>	Key to query for.
------------	-------------------

Returns

True if key was released, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

91         {
92             return !_keyboardState[key] && _lastKeyboardState[key];
93         }
```

7.4.3.8 bool TriDevs.TriEngine2D.Input.InputManager.KeyUp (Key key)

Returns whether or not the specified key is currently unpressed.

Parameters

<i>key</i>	Key to query for.
------------	-------------------

Returns

True if the key is currently up (not pressed), false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

76         {
77             return !_keyboardState[key];
78         }
```

7.4.3.9 void TriDevs.TriEngine2D.Input.InputManager.Update ()

Updates the input manager, refreshing all current and previous states.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

67         {
68             _lastKeyboardState = _keyboardState;
69             _keyboardState = Keyboard.GetState();
70
71             _lastMouseState = _mouseState;
72             _mouseState = Mouse.GetState();
73         }
```

7.4.3.10 int TriDevs.TriEngine2D.Input.InputManager.WheelChange ()

Returns the mouse wheel's change in value.

Returns

Negative value if wheel scrolled down, positive value if scrolled up, zero if not scrolled.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

131         {
132             return _mouseState.Wheel - _lastMouseState.Wheel;
133         }
```

7.4.3.11 bool TriDevs.TriEngine2D.Input.InputManager.WheelChanged ()

Returns whether the mouse wheel scrolled at all.

Returns

True if the mouse wheel scrolled, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
126         {
127             return _mouseState.Wheel != _lastMouseState.Wheel;
128         }
```

7.4.3.12 bool TriDevs.TriEngine2D.Input.InputManager.WheelDown ()

Returns whether the mouse wheel was scrolled down.

Returns

True if mouse wheel was scrolled down, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
121         {
122             return _mouseState.Wheel < _lastMouseState.Wheel;
123         }
```

7.4.3.13 bool TriDevs.TriEngine2D.Input.InputManager.WheelUp ()

Returns whether the mouse wheel was scrolled up.

Returns

True if mouse wheel was scrolled up, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
116         {
117             return _mouseState.Wheel > _lastMouseState.Wheel;
118         }
```

7.4.4 Property Documentation

7.4.4.1 Point TriDevs.TriEngine2D.Input.InputManager.MousePosition [get]

7.4.4.2 int TriDevs.TriEngine2D.Input.InputManager.MouseWheelValue [get]

7.4.4.3 int TriDevs.TriEngine2D.Input.InputManager.MouseX [get]

7.4.4.4 int TriDevs.TriEngine2D.Input.InputManager.MouseY [get]

7.4.4.5 bool TriDevs.TriEngine2D.Input.InputManager.this[Key key] [get]

7.4.4.6 bool TriDevs.TriEngine2D.Input.InputManager.this[MouseButton button] [get]

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

- [TriDevs.TriEngine2D/Input/InputManager.cs](#)

7.5 TriDevs.TriEngine2D.Helpers.IO Class Reference

Provides various helper functions for doing [IO](#) operations.

Static Public Member Functions

- static string [GetAbsolutePath](#) (string path)
Resolves the absolute path from a relative path.

7.5.1 Detailed Description

Provides various helper functions for doing [IO](#) operations.

7.5.2 Member Function Documentation

7.5.2.1 static string TriDevs.TriEngine2D.Helpers.IO.GetAbsolutePath (string path) [static]

Resolves the absolute path from a relative path.

Parameters

<i>path</i>	The relative path to resolve.
-------------	-------------------------------

Returns

The absolute path to the item.

```

39         {
40             return Path.Combine(Directory.GetCurrentDirectory(), path);
41         }

```

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

- [TriDevs.TriEngine2D/Helpers/IO.cs](#)

7.6 TriDevs.TriEngine2D.Logging.LogManager Class Reference

Class to manage logging. ILog interfaces should be obtained from this class' methods, as opposed to calling default log4net methods.

Static Public Member Functions

- static void [LoadConfig](#) (string file=null)
Load a config to use with log4net.
- static ILog [GetLogger](#) (object sender)
Gets an ILog object for the specified object.
- static void [SetupConsole](#) ()
Set up a new console for this process. Will not set up a console if a debugger is attached. This method does nothing if DEBUG is not #defined.
- static void [DestroyConsole](#) ()
Destroys the console associated with the process, if loaded. This method does nothing if DEBUG is not #defined.
- static void [ClearOldLogs](#) (int daysOld=7, string logsDir="logs")
Clear logs that are older than the specified amount of days.

7.6.1 Detailed Description

Class to manage logging. ILog interfaces should be obtained from this class' methods, as opposed to calling default log4net methods.

7.6.2 Member Function Documentation

7.6.2.1 static void TriDevs.TriEngine2D.Logging.LogManager.ClearOldLogs (int *daysOld* = 7, string *logsDir* = "logs") [static]

Clear logs that are older than the specified amount of days.

Parameters

<i>daysOld</i>	Logs older than this amount of days will be deleted.
<i>logsDir</i>	The directory to clear.

```

136     {
137         var log = GetLogger(typeof(LogManager));
138
139         log.InfoFormat(">> ClearOldLogs({0}, \"{1}\", daysOld, logsDir);
140
141         if (!Directory.Exists(logsDir))
142         {
143             log.InfoFormat("Directory {0} not found, no logs to clear", logsDir);
144             log.Info("<< ClearOldLogs()");
145             return;
146         }
147
148         var now = DateTime.Now;
149         var max = new TimeSpan(daysOld, 0, 0, 0);
150         var count = 0;
151         foreach (var file in from file in Directory.GetFiles(logsDir)
152                             let modTime = File.GetLastAccessTime(file)
153                             let age = now.Subtract(modTime)
154                             where age > max
155                             select file)
156         {
157             try
158             {
159                 File.Delete(file);
160                 log.InfoFormat("Deleted old log file: {0}", file);
161                 count++;
162             }
163             catch (IOException ex)
164             {
165                 log.WarnFormat("Failed to delete log file: {0} ({1})", file, ex.Message);
166             }
167         }
168
169         log.InfoFormat("Done! Cleared {0} log files.", count);
170         log.Info("<< ClearOldLogs()");
171     }

```

7.6.2.2 static void TriDevs.TriEngine2D.Logging.LogManager.DestroyConsole () [static]

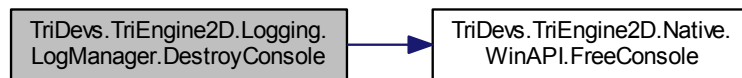
Destroys the console associated with the process, if loaded. This method does nothing if DEBUG is not #defined.

```

123     {
124     #if DEBUG
125         if (_consoleLoaded)
126             WinAPI.FreeConsole();
127     #endif
128     }

```

Here is the call graph for this function:



7.6.2.3 static ILog TriDevs.TriEngine2D.Logging.LogManager.GetLogger (object sender) [static]

Gets an ILog object for the specified object.

To get the logger object for a static class, or from static context, call GetLogger(typeof(YourClass)).

Parameters

<i>sender</i>	The object or Type to get an ILog object for.
---------------	---

Returns

The ILog object.

```

89     {
90         if (!_loaded)
91             LoadConfig();
92
93         return log4net.LogManager.GetLogger(sender.GetType().ToString() == "System.RuntimeType" ? (Type
94             )sender : sender.GetType());
95     }
  
```

7.6.2.4 static void TriDevs.TriEngine2D.Logging.LogManager.LoadConfig (string file = null) [static]

Load a config to use with log4net.

LoadConfig will first try to load the specified file, if not null. If it is unable to find the specified file, it will call itself again with file set to null. If no file is specified, it will attempt to load a config file following the pattern: "(Assembly-Name).config" If it is unable to load the config, it will default to BasicConfigurator.

Parameters

<i>file</i>	The config file to load, null if automatic loading is preferred.
-------------	--

```

57     {
58         if (file == null)
59         {
60             if (File.Exists(AppDomain.CurrentDomain.FriendlyName + ".config"))
61                 XmlConfigurator.Configure();
62             else
63                 BasicConfigurator.Configure();
64         }
65         else
66         {
67             if (File.Exists(file))
68                 XmlConfigurator.Configure(new FileInfo(file));
69             else
70             {
71                 LoadConfig();
72                 return;
73             }
74         }
75     }
  
```

```

76         _loaded = true;
77     }

```

7.6.2.5 static void TriDevs.TriEngine2D.Logging.LogManager.SetupConsole () [static]

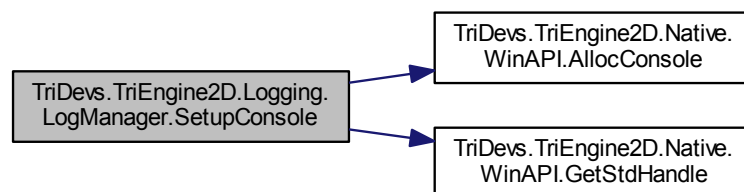
Set up a new console for this process. Will not set up a console if a debugger is attached. This method does nothing if DEBUG is not #defined.

```

102     {
103     #if DEBUG
104         if (System.Diagnostics.Debugger.IsAttached)
105             return;
106
107         WinAPI.AllocConsole();
108         var stdHandle = WinAPI.GetStdHandle(WinAPI.
STD_OUTPUT_HANDLE);
109         var safeFileHandle = new SafeFileHandle(stdHandle, true);
110         var fileStream = new FileStream(safeFileHandle, FileAccess.Write);
111         var encoding = Encoding.GetEncoding(WinAPI.CODE_PAGE);
112         var stdOut = new StreamWriter(fileStream, encoding) { AutoFlush = true };
113         Console.SetOut(stdOut);
114         _consoleLoaded = true;
115     #endif
116     }

```

Here is the call graph for this function:



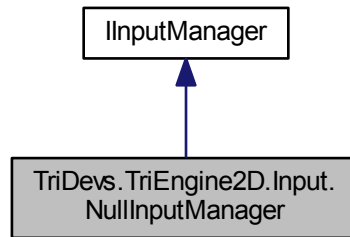
The documentation for this class was generated from the following file:

- [TriDevs.TriEngine2D/Logging/LogManager.cs](#)

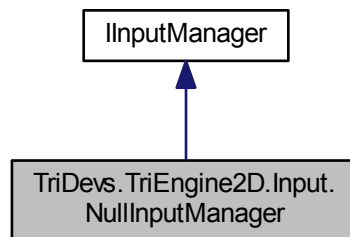
7.7 TriDevs.TriEngine2D.Input.NullInputManager Class Reference

Used as a fallback [InputManager](#) object when the service locator fails to find one.

Inheritance diagram for TriDevs.TriEngine2D.Input.NullInputManager:



Collaboration diagram for TriDevs.TriEngine2D.Input.NullInputManager:



Public Member Functions

- void [Update](#) ()
Updates the input manager, refreshing all current and previous states.
- bool [KeyUp](#) (Key key)
Returns whether or not the specified key is currently unpressed.
- bool [KeyDown](#) (Key key)
Returns whether or not the specified key is currently being pressed.
- bool [KeyPressed](#) (Key key)
Returns whether or not the specified key has been pressed.
- bool [KeyReleased](#) (Key key)
Returns whether or not the specified key has been released.
- bool [ButtonUp](#) (MouseButton button)
Returns whether or not the specified mouse button is currently unpressed.
- bool [ButtonDown](#) (MouseButton button)
Returns whether or not the specified mouse button is currently being pressed.
- bool [ButtonPressed](#) (MouseButton button)
Returns whether or not the specified mouse button has been pressed.

- bool [ButtonReleased](#) (MouseButton button)
Returns whether or not the specified mouse button has been released.
- bool [WheelUp](#) ()
Returns whether the mouse wheel was scrolled up.
- bool [WheelDown](#) ()
Returns whether the mouse wheel was scrolled down.
- bool [WheelChanged](#) ()
Returns whether the mouse wheel scrolled at all.
- int [WheelChange](#) ()
Returns the mouse wheel's change in value.

Properties

- int [MouseX](#) [get]
- int [MouseY](#) [get]
- [Point](#) [MousePosition](#) [get]
- int [MouseWheelValue](#) [get]
- bool [this\[Key key\]](#) [get]
- bool [this\[MouseButton button\]](#) [get]

7.7.1 Detailed Description

Used as a fallback [InputManager](#) object when the service locator fails to find one.

7.7.2 Member Function Documentation

7.7.2.1 bool TriDevs.TriEngine2D.Input.NullInputManager.ButtonDown (MouseButton button)

Returns whether or not the specified mouse button is currently being pressed.

Parameters

<i>button</i>	The button to query for.
---------------	--------------------------

Returns

True if button is currently being pressed, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

79         {
80             return false;
81         }
```

7.7.2.2 bool TriDevs.TriEngine2D.Input.NullInputManager.ButtonPressed (MouseButton button)

Returns whether or not the specified mouse button has been pressed.

Only returns true if the last state of the mouse button was not pressed.

Parameters

<i>button</i>	Button to query for.
---------------	----------------------

Returns

True if button was pressed, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
84         {
85             return false;
86         }
```

7.7.2.3 bool TriDevs.TriEngine2D.Input.NullInputManager.ButtonReleased (MouseButton *button*)

Returns whether or not the specified mouse button has been released.

Only returns true if the last state of the button was pressed.

Parameters

<i>button</i>	The button to query for.
---------------	--------------------------

Returns

True if the button was released, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
89         {
90             return false;
91         }
```

7.7.2.4 bool TriDevs.TriEngine2D.Input.NullInputManager.ButtonUp (MouseButton *button*)

Returns whether or not the specified mouse button is currently unpressed.

Parameters

<i>button</i>	Button to query for.
---------------	----------------------

Returns

True if the button is currently up (not pressed), false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
74         {
75             return true;
76         }
```

7.7.2.5 bool TriDevs.TriEngine2D.Input.NullInputManager.KeyDown (Key *key*)

Returns whether or not the specified key is currently being pressed.

Parameters

<i>key</i>	Key to query for.
------------	-------------------

Returns

True if key is currently being pressed, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

59         {
60             return false;
61         }

```

7.7.2.6 bool TriDevs.TriEngine2D.Input.NullInputManager.KeyPressed (Key key)

Returns whether or not the specified key has been pressed.

Only returns true if the last state of the key was not pressed.

Parameters

<i>key</i>	Key to query for.
------------	-------------------

Returns

True if key was pressed, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

64         {
65             return false;
66         }

```

7.7.2.7 bool TriDevs.TriEngine2D.Input.NullInputManager.KeyReleased (Key key)

Returns whether or not the specified key has been released.

Only returns true if the last state of the key was pressed.

Parameters

<i>key</i>	Key to query for.
------------	-------------------

Returns

True if key was released, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```

69         {
70             return false;
71         }

```

7.7.2.8 bool TriDevs.TriEngine2D.Input.NullInputManager.KeyUp (Key key)

Returns whether or not the specified key is currently unpressed.

Parameters

<i>key</i>	Key to query for.
------------	-------------------

Returns

True if the key is currently up (not pressed), false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
54         {
55             return true;
56         }
```

7.7.2.9 void TriDevs.TriEngine2D.Input.NullInputManager.Update ()

Updates the input manager, refreshing all current and previous states.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
49         {
50             // Do nothing
51         }
```

7.7.2.10 int TriDevs.TriEngine2D.Input.NullInputManager.WheelChange ()

Returns the mouse wheel's change in value.

Returns

Negative value if wheel scrolled down, positive value if scrolled up, zero if not scrolled.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
109         {
110             return 0;
111         }
```

7.7.2.11 bool TriDevs.TriEngine2D.Input.NullInputManager.WheelChanged ()

Returns whether the mouse wheel scrolled at all.

Returns

True if the mouse wheel scrolled, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
104         {
105             return false;
106         }
```

7.7.2.12 bool TriDevs.TriEngine2D.Input.NullInputManager.WheelDown ()

Returns whether the mouse wheel was scrolled down.

Returns

True if mouse wheel was scrolled down, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
99         {
100             return false;
101         }
```

7.7.2.13 bool TriDevs.TriEngine2D.Input.NullInputManager.WheelUp ()

Returns whether the mouse wheel was scrolled up.

Returns

True if mouse wheel was scrolled up, false otherwise.

Implements [TriDevs.TriEngine2D.Input.IInputManager](#).

```
94     {  
95         return false;  
96     }
```

7.7.3 Property Documentation

7.7.3.1 Point TriDevs.TriEngine2D.Input.NullInputManager.MousePosition [get]

7.7.3.2 int TriDevs.TriEngine2D.Input.NullInputManager.MouseWheelValue [get]

7.7.3.3 int TriDevs.TriEngine2D.Input.NullInputManager.MouseX [get]

7.7.3.4 int TriDevs.TriEngine2D.Input.NullInputManager.MouseY [get]

7.7.3.5 bool TriDevs.TriEngine2D.Input.NullInputManager.this[Key key] [get]

7.7.3.6 bool TriDevs.TriEngine2D.Input.NullInputManager.this[MouseButton button] [get]

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

- [TriDevs.TriEngine2D/Input/NullInputManager.cs](#)

7.8 TriDevs.TriEngine2D.Point Struct Reference

A struct representing an X/Y coordinate.

Public Member Functions

- [Point](#) (int x, int y)
Creates a new [Point](#) with the specified X and Y values.

Public Attributes

- int [X](#)
The X value of the coordinate.
- int [Y](#)
The Y value of the coordinate.

7.8.1 Detailed Description

A struct representing an X/Y coordinate.

7.8.2 Constructor & Destructor Documentation

7.8.2.1 TriDevs.TriEngine2D.Point.Point (int x, int y)

Creates a new [Point](#) with the specified X and Y values.

Parameters

<code>x</code>	The X value.
<code>y</code>	The Y value.

```

47     {
48         x = x;
49         y = y;
50     }
```

7.8.3 Member Data Documentation

7.8.3.1 int TriDevs.TriEngine2D.Point.X

The X value of the coordinate.

7.8.3.2 int TriDevs.TriEngine2D.Point.Y

The Y value of the coordinate.

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

- TriDevs.TriEngine2D/[Point.cs](#)

7.9 TriDevs.TriEngine2D.Serializing.Serializer Class Reference

Provides serialization methods.

Static Public Member Functions

- static string [Serialize< T >](#) (T data)
Serialize an object to string.
- static void [Serialize< T >](#) (T data, string file, Formatting formatting=Formatting.Indented)
Serializes an object to file.
- static T [Deserialize< T >](#) (string file)
Deserialize a serialized object from file.

7.9.1 Detailed Description

Provides serialization methods.

7.9.2 Member Function Documentation

7.9.2.1 static T TriDevs.TriEngine2D.Serializing.Serializer.Deserialize< T > (string file) [static]

Deserialize a serialized object from file.

Template Parameters

<i>T</i>	Type of the object being deserialized.
----------	--

Parameters

<i>file</i>	File to read from.
-------------	--------------------

Returns

The deserialized object.

```

84     {
85         T data;
86         // Create the StreamReader
87         using (var reader = new StreamReader(file))
88         {
89             // And the json reader
90             using (var jsonReader = new JsonTextReader(reader))
91             {
92                 // Now deserialize the file to the requested object...
93                 data = JsonSerializer.Value.Deserialize<T>(jsonReader);
94                 // ... and close the json reader.
95                 jsonReader.Close();
96             }
97             // Finally, close the file reader
98             reader.Close();
99         }
100         return data;
101     }

```

7.9.2.2 static string TriDevs.TriEngine2D.Serializing.Serializer.Serialize< T > (T data) [static]

Serialize an object to string.

Template Parameters

<i>T</i>	Type of data.
----------	---------------

Parameters

<i>data</i>	Data to serialize.
-------------	--------------------

Returns

The serialized object in string format.

```

44     {
45         // Create a StringWriter to hold the serialized object
46         var writer = new StringWriter();
47         // Serialize the object into the writer
48         JsonSerializer.Value.Serialize(writer, data);
49         return writer.ToString(); // Return the data as string
50     }

```

7.9.2.3 static void TriDevs.TriEngine2D.Serializing.Serializer.Serialize< T > (T data, string file, Formatting formatting = Formatting.Indented) [static]

Serializes an object to file.

Template Parameters

<i>T</i>	Type of the data.
----------	-------------------

Parameters

<i>data</i>	Data to serialize.
<i>file</i>	File to serialize to.
<i>formatting</i>	The formatting to use for the JSON output.

```

60     {
61         // Create the StreamWriter
62         using (var writer = new StreamWriter(file, false))
63         {
64             // Create the json writer
65             using (var jsonWriter = new JsonTextWriter(writer){Formatting = formatting})
66             {
67                 // Now serialize the object to the file...
68                 JsonSerializer.Value.Serialize(jsonWriter, data);
69                 // ... and close the json writer
70                 jsonWriter.Close();
71             }
72             // Finally, close the file writer
73             writer.Close();
74         }
75     }

```

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

- TriDevs.TriEngine2D/Serializing/[Serializer.cs](#)

7.10 TriDevs.TriEngine2D.Services Class Reference

Provides different game-related service interfaces.

Static Public Member Functions

- static void [Provide](#) ([IInputManager](#) input)
Specifies an input manager service to provide.

Properties

- static [IInputManager Input](#) [get]
The input manager service.

7.10.1 Detailed Description

Provides different game-related service interfaces.

Actual service providers must be supplied from external code. All Service properties are intialized with Null-type services that provide no real functionality.

7.10.2 Member Function Documentation

7.10.2.1 static void TriDevs.TriEngine2D.Services.Provide (IInputManager input) [static]

Specifies an input manager service to provide.

Parameters

<i>input</i>	An object implementing the IInputManager interface.
--------------	---

```

50     {

```

```

51         _input = input;
52     }

```

7.10.3 Property Documentation

7.10.3.1 InputManager TriDevs.TriEngine2D.Services.Input [static], [get]

The input manager service.

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

- TriDevs.TriEngine2D/[Services.cs](#)

7.11 TriDevs.TriEngine2D.Extensions.StringExtensions Class Reference

[Extensions](#) for System.String

Static Public Member Functions

- static string [ReplaceFirst](#) (this string s, string search, string replace, bool caseInsensitive=false)
Returns a string in which the first occurrence of a specified string is replaced with another string.
- static string [Replace](#) (this string s, string search, string replace, int count, bool caseInsensitive=false)
Returns a string in which the N first occurrences of a specified string are replaced with another string.
- static string [Replace](#) (this string s, string search, string replace, bool caseInsensitive=false)
Returns a string in which all occurrences of a specified string are replaced with another string.

7.11.1 Detailed Description

[Extensions](#) for System.String

7.11.2 Member Function Documentation

7.11.2.1 static string TriDevs.TriEngine2D.Extensions.StringExtensions.Replace (this string s, string search, string replace, int count, bool caseInsensitive = false) [static]

Returns a string in which the N first occurrences of a specified string are replaced with another string.

Parameters

<i>s</i>	String to modify.
<i>search</i>	String to search for.
<i>replace</i>	String to replace the match(es) with.
<i>count</i>	Number of occurrences to replace.
<i>caseInsensitive</i>	True for case insensitive search, false for case sensitive.

Returns

The supplied string with the N first occurrences of the specified string replaced with the other.

```

56     {
57         var re = caseInsensitive ? new Regex(search, RegexOptions.IgnoreCase) : new Regex(search);
58         return re.Replace(s, replace, count);
59     }

```

7.11.2.2 `static string TriDevs.TriEngine2D.Extensions.StringExtensions.Replace (this string s, string search, string replace, bool caseInsensitive = false) [static]`

Returns a string in which all occurrences of a specified string are replaced with another string.

This extension method supports case insensitive searches.

Parameters

<i>s</i>	String to modify.
<i>search</i>	String to search for.
<i>replace</i>	String to replace the match(es) with.
<i>caseInsensitive</i>	True for case insensitive search, false for case sensitive.

Returns

The supplied string with all occurrences of the specified string replaced with the other.

```

73     {
74         var re = caseInsensitive ? new Regex(search, RegexOptions.IgnoreCase) : new Regex(search);
75         return re.Replace(s, replace);
76     }

```

7.11.2.3 `static string TriDevs.TriEngine2D.Extensions.StringExtensions.ReplaceFirst (this string s, string search, string replace, bool caseInsensitive = false) [static]`

Returns a string in which the first occurrence of a specified string is replaced with another string.

Parameters

<i>s</i>	String to modify.
<i>search</i>	String to search for.
<i>replace</i>	String to replace the match with.
<i>caseInsensitive</i>	True for case insensitive search, false for case sensitive.

Returns

The supplied string with the first occurrence of the specified string replaced with the other.

```

42     {
43         return Replace(s, search, replace, 1, caseInsensitive);
44     }

```

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

- TriDevs.TriEngine2D/Extensions/[StringExtensions.cs](#)

7.12 TriDevs.TriEngine2D.Helpers.Threading Class Reference

Provides various helper functions for doing threading operations.

Static Public Member Functions

- static void [SetCurrentThreadName](#) (string name)
Sets the name of the current thread, does nothing if the thread already has a name.

7.12.1 Detailed Description

Provides various helper functions for doing threading operations.

7.12.2 Member Function Documentation

7.12.2.1 static void TriDevs.TriEngine2D.Helpers.Threading.SetCurrentThreadName (string *name*) [static]

Sets the name of the current thread, does nothing if the thread already has a name.

Parameters

<i>name</i>	The new name for the current thread
-------------	-------------------------------------

```

39     {
40         // We can't set the name on a thread if it's already set, it would throw an exception
41         // So we have to check if the current name is null before trying to set a new one
42         if (string.IsNullOrEmpty(Thread.CurrentThread.Name))
43             Thread.CurrentThread.Name = name;
44     }

```

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

- TriDevs.TriEngine2D/Helpers/[Threading.cs](#)

7.13 TriDevs.TriEngine2D.Version Class Reference

[Version](#) class specifying the version of this project.

Public Attributes

- const int [Major](#) = 0
Major version of the project.
- const int [Minor](#) = 0
Minor version of the project.
- const int [Patch](#) = 6
Patch version of the project.
- const string [Suffix](#) = ""
Optional suffix, empty if no suffix for this version.
- const string [VersionStringFormat](#) = "{0}.{1}.{2}"
The format string used when formatting major, minor and patch version to their string representation.
- const string [VersionStringFormatWithSuffix](#) = [VersionStringFormat](#) + "-{3}"
The format string used when formatting major, minor and patch version to their string representation (with suffix).

Properties

- static string [VersionString](#) [get]
String representation of the current project version.

7.13.1 Detailed Description

[Version](#) class specifying the version of this project.

7.13.2 Member Data Documentation

7.13.2.1 `const int TriDevs.TriEngine2D.Version.Major = 0`

Major version of the project.

7.13.2.2 `const int TriDevs.TriEngine2D.Version.Minor = 0`

Minor version of the project.

7.13.2.3 `const int TriDevs.TriEngine2D.Version.Patch = 6`

Patch version of the project.

7.13.2.4 `const string TriDevs.TriEngine2D.Version.Suffix = ""`

Optional suffix, empty if no suffix for this version.

Example values could be "beta" and "alpha".

7.13.2.5 `const string TriDevs.TriEngine2D.Version.VersionStringFormat = "{0}.{1}.{2}"`

The format string used when formatting major, minor and patch version to their string representation.

7.13.2.6 `const string TriDevs.TriEngine2D.Version.VersionStringFormatWithSuffix = VersionStringFormat + "-{3}"`

The format string used when formatting major, minor and patch version to their string representation (with suffix).

7.13.3 Property Documentation

7.13.3.1 `string TriDevs.TriEngine2D.Version.VersionString` `[static], [get]`

String representation of the current project version.

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

- TriDevs.TriEngine2D/[Version.cs](#)

7.14 TriDevs.TriEngine2D.Native.WinAPI Class Reference

Holds various [WinAPI](#) stuff.

Public Member Functions

- static IntPtr [GetStdHandle](#) (int nStdHandle)
Retrieves a handle to the specified standard device (standard input, standard output, or standard error).
- static bool [AllocConsole](#) ()
Allocates a new console for the calling process.
- static int [FreeConsole](#) ()
Detaches the calling process from its console.

Public Attributes

- const int `STD_OUTPUT_HANDLE` = -11
The standard output device. Initially, this is the active console screen buffer, CONOUT\$.
- const int `CODE_PAGE` = 437
The code page to use for the console.

7.14.1 Detailed Description

Holds various [WinAPI](#) stuff.

7.14.2 Member Function Documentation

7.14.2.1 static bool TriDevs.TriEngine2D.Native.WinAPI.AllocConsole ()

Allocates a new console for the calling process.

Returns

If the function succeeds, the return value is nonzero. If the function fails, the return value is zero. To get extended error information, call GetLastError.

Here is the caller graph for this function:



7.14.2.2 static int TriDevs.TriEngine2D.Native.WinAPI.FreeConsole ()

Detaches the calling process from its console.

Returns

If the function succeeds, the return value is nonzero. If the function fails, the return value is zero. To get extended error information, call GetLastError.

Here is the caller graph for this function:



7.14.2.3 static IntPtr TriDevs.TriEngine2D.Native.WinAPI.GetStdHandle (int *nStdHandle*)

Retrieves a handle to the specified standard device (standard input, standard output, or standard error).

Parameters

<i>nStdHandle</i>	The standard device.
-------------------	----------------------

Returns

If the function succeeds, the return value is a handle to the specified device, or a redirected handle set by a previous call to SetStdHandle. The handle has GENERIC_READ and GENERIC_WRITE access rights, unless the application has used SetStdHandle to set a standard handle with lesser access. If the function fails, the return value is INVALID_HANDLE_VALUE. To get extended error information, call GetLastError. If an application does not have associated standard handles, such as a service running on an interactive desktop, and has not redirected them, the return value is NULL.

Here is the caller graph for this function:



7.14.3 Member Data Documentation

7.14.3.1 const int TriDevs.TriEngine2D.Native.WinAPI.CODE_PAGE = 437

The code page to use for the console.

7.14.3.2 const int TriDevs.TriEngine2D.Native.WinAPI.STD_OUTPUT_HANDLE = -11

The standard output device. Initially, this is the active console screen buffer, CONOUT\$.

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

- TriDevs.TriEngine2D/Native/[WinAPI.cs](#)

Chapter 8

File Documentation

8.1 README.md File Reference

8.2 TriDevs.TriEngine2D/Extensions/EnumerationExtensions.cs File Reference

Classes

- class [TriDevs.TriEngine2D.Extensions.EnumerationExtensions](#)
Extensions for System.Enum.

Namespaces

- package [TriDevs.TriEngine2D.Extensions](#)

8.3 TriDevs.TriEngine2D/Extensions/StringExtensions.cs File Reference

Classes

- class [TriDevs.TriEngine2D.Extensions.StringExtensions](#)
Extensions for System.String

Namespaces

- package [TriDevs.TriEngine2D.Extensions](#)

8.4 TriDevs.TriEngine2D/Helpers/IO.cs File Reference

Classes

- class [TriDevs.TriEngine2D.Helpers.IO](#)
Provides various helper functions for doing [IO](#) operations.

Namespaces

- package [TriDevs.TriEngine2D.Helpers](#)

8.5 TriDevs.TriEngine2D/Helpers/Threading.cs File Reference

Classes

- class [TriDevs.TriEngine2D.Helpers.Threading](#)
Provides various helper functions for doing threading operations.

Namespaces

- package [TriDevs.TriEngine2D.Helpers](#)

8.6 TriDevs.TriEngine2D/Input/IInputManager.cs File Reference

Classes

- interface [TriDevs.TriEngine2D.Input.IInputManager](#)
Provides various methods to query input devices like the keyboard.

Namespaces

- package [TriDevs.TriEngine2D.Input](#)

8.7 TriDevs.TriEngine2D/Input/InputManager.cs File Reference

Classes

- class [TriDevs.TriEngine2D.Input.InputManager](#)
Input manager interfacing with input methods provided by a GameWindow.

Namespaces

- package [TriDevs.TriEngine2D.Input](#)

8.8 TriDevs.TriEngine2D/Input/NullInputManager.cs File Reference

Classes

- class [TriDevs.TriEngine2D.Input.NullInputManager](#)
Used as a fallback [IInputManager](#) object when the service locator fails to find one.

Namespaces

- package [TriDevs.TriEngine2D.Input](#)

8.9 TriDevs.TriEngine2D/Logging/LogManager.cs File Reference

Classes

- class [TriDevs.TriEngine2D.Logging.LogManager](#)

Class to manage logging. ILog interfaces should be obtained from this class' methods, as opposed to calling default log4net methods.

Namespaces

- package [TriDevs.TriEngine2D.Logging](#)

8.10 TriDevs.TriEngine2D/Native/Helpers.cs File Reference

Classes

- class [TriDevs.TriEngine2D.Native.Helpers](#)

Helper class with various methods to help native coding and debugging.

Namespaces

- package [TriDevs.TriEngine2D.Native](#)

8.11 TriDevs.TriEngine2D/Native/WinAPI.cs File Reference

Classes

- class [TriDevs.TriEngine2D.Native.WinAPI](#)

Holds various [WinAPI](#) stuff.

Namespaces

- package [TriDevs.TriEngine2D.Native](#)

8.12 TriDevs.TriEngine2D/Point.cs File Reference

Classes

- struct [TriDevs.TriEngine2D.Point](#)

A struct representing an X/Y coordinate.

Namespaces

- package [TriDevs.TriEngine2D](#)

8.13 TriDevs.TriEngine2D/Properties/AssemblyInfo.cs File Reference

8.14 TriDevs.TriEngine2D/Serializing/Serializer.cs File Reference

Classes

- class [TriDevs.TriEngine2D.Serializing.Serializer](#)
Provides serialization methods.

Namespaces

- package [TriDevs.TriEngine2D.Serializing](#)

8.15 TriDevs.TriEngine2D/Services.cs File Reference

Classes

- class [TriDevs.TriEngine2D.Services](#)
Provides different game-related service interfaces.

Namespaces

- package [TriDevs.TriEngine2D](#)

8.16 TriDevs.TriEngine2D/Version.cs File Reference

Classes

- class [TriDevs.TriEngine2D.Version](#)
[Version](#) class specifying the version of this project.

Namespaces

- package [TriDevs.TriEngine2D](#)

Index

- AllocConsole
 - TriDevs::TriEngine2D::Native::WinAPI, [46](#)
- ButtonDown
 - TriDevs::TriEngine2D::Input::IInputManager, [19](#)
 - TriDevs::TriEngine2D::Input::InputManager, [24](#)
 - TriDevs::TriEngine2D::Input::NullInputManager, [34](#)
- ButtonPressed
 - TriDevs::TriEngine2D::Input::IInputManager, [19](#)
 - TriDevs::TriEngine2D::Input::InputManager, [25](#)
 - TriDevs::TriEngine2D::Input::NullInputManager, [34](#)
- ButtonReleased
 - TriDevs::TriEngine2D::Input::IInputManager, [19](#)
 - TriDevs::TriEngine2D::Input::InputManager, [25](#)
 - TriDevs::TriEngine2D::Input::NullInputManager, [35](#)
- ButtonUp
 - TriDevs::TriEngine2D::Input::IInputManager, [20](#)
 - TriDevs::TriEngine2D::Input::InputManager, [25](#)
 - TriDevs::TriEngine2D::Input::NullInputManager, [35](#)
- CODE_PAGE
 - TriDevs::TriEngine2D::Native::WinAPI, [47](#)
- ClearOldLogs
 - TriDevs::TriEngine2D::Logging::LogManager, [30](#)
- Deserialize< T >
 - TriDevs::TriEngine2D::Serializing::Serializer, [39](#)
- DestroyConsole
 - TriDevs::TriEngine2D::Logging::LogManager, [30](#)
- FreeConsole
 - TriDevs::TriEngine2D::Native::WinAPI, [46](#)
- GetAbsolutePath
 - TriDevs::TriEngine2D::Helpers::IO, [29](#)
- GetErrorMessage
 - TriDevs::TriEngine2D::Native::Helpers, [16](#)
- GetLastError
 - TriDevs::TriEngine2D::Native::Helpers, [16](#)
- GetLastErrorInfo
 - TriDevs::TriEngine2D::Native::Helpers, [16](#)
- GetLastErrorMessage
 - TriDevs::TriEngine2D::Native::Helpers, [17](#)
- GetLastWin32Exception
 - TriDevs::TriEngine2D::Native::Helpers, [17](#)
- GetLogger
 - TriDevs::TriEngine2D::Logging::LogManager, [31](#)
- GetStdHandle
 - TriDevs::TriEngine2D::Native::WinAPI, [46](#)
- GetWin32Exception
 - TriDevs::TriEngine2D::Native::Helpers, [17](#)
- Has< T >
 - TriDevs::TriEngine2D::Extensions::Enumeration-Extensions, [13](#)
- Include< T >
 - TriDevs::TriEngine2D::Extensions::Enumeration-Extensions, [14](#)
- Input
 - TriDevs::TriEngine2D::Services, [42](#)
- InputManager
 - TriDevs::TriEngine2D::Input::InputManager, [24](#)
- KeyDown
 - TriDevs::TriEngine2D::Input::IInputManager, [20](#)
 - TriDevs::TriEngine2D::Input::InputManager, [26](#)
 - TriDevs::TriEngine2D::Input::NullInputManager, [35](#)
- KeyPressed
 - TriDevs::TriEngine2D::Input::IInputManager, [20](#)
 - TriDevs::TriEngine2D::Input::InputManager, [26](#)
 - TriDevs::TriEngine2D::Input::NullInputManager, [36](#)
- KeyReleased
 - TriDevs::TriEngine2D::Input::IInputManager, [20](#)
 - TriDevs::TriEngine2D::Input::InputManager, [26](#)
 - TriDevs::TriEngine2D::Input::NullInputManager, [36](#)
- KeyUp
 - TriDevs::TriEngine2D::Input::IInputManager, [21](#)
 - TriDevs::TriEngine2D::Input::InputManager, [27](#)
 - TriDevs::TriEngine2D::Input::NullInputManager, [36](#)
- LoadConfig
 - TriDevs::TriEngine2D::Logging::LogManager, [31](#)
- Major
 - TriDevs::TriEngine2D::Version, [45](#)
- Minor
 - TriDevs::TriEngine2D::Version, [45](#)
- Missing< T >
 - TriDevs::TriEngine2D::Extensions::Enumeration-Extensions, [14](#)
- MousePosition
 - TriDevs::TriEngine2D::Input::IInputManager, [22](#)
 - TriDevs::TriEngine2D::Input::InputManager, [28](#)
 - TriDevs::TriEngine2D::Input::NullInputManager, [38](#)
- MouseWheelValue
 - TriDevs::TriEngine2D::Input::IInputManager, [22](#)
 - TriDevs::TriEngine2D::Input::InputManager, [28](#)
 - TriDevs::TriEngine2D::Input::NullInputManager, [38](#)
- MouseX
 - TriDevs::TriEngine2D::Input::IInputManager, [22](#)
 - TriDevs::TriEngine2D::Input::InputManager, [28](#)

- TriDevs::TriEngine2D::Input::NullInputManager, 38
- MouseY
 - TriDevs::TriEngine2D::Input::InputManager, 22
 - TriDevs::TriEngine2D::Input::InputManager, 28
 - TriDevs::TriEngine2D::Input::NullInputManager, 38
- Patch
 - TriDevs::TriEngine2D::Version, 45
- Point
 - TriDevs::TriEngine2D::Point, 39
- Provide
 - TriDevs::TriEngine2D::Services, 41
- README.md, 49
- Remove< T >
 - TriDevs::TriEngine2D::Extensions::EnumerationExtensions, 15
- Replace
 - TriDevs::TriEngine2D::Extensions::StringExtensions, 42
- ReplaceFirst
 - TriDevs::TriEngine2D::Extensions::StringExtensions, 43
- Serialize< T >
 - TriDevs::TriEngine2D::Serializing::Serializer, 40
- SetCurrentThreadName
 - TriDevs::TriEngine2D::Helpers::Threading, 44
- SetupConsole
 - TriDevs::TriEngine2D::Logging::LogManager, 32
- Suffix
 - TriDevs::TriEngine2D::Version, 45
- TriDevs, 11
- TriDevs.TriEngine2D, 11
- TriDevs.TriEngine2D.Extensions, 11
- TriDevs.TriEngine2D.Extensions.EnumerationExtensions, 13
- TriDevs.TriEngine2D.Extensions.StringExtensions, 42
- TriDevs.TriEngine2D.Helpers, 12
- TriDevs.TriEngine2D.Helpers.IO, 29
- TriDevs.TriEngine2D.Helpers.Threading, 43
- TriDevs.TriEngine2D.Input, 12
- TriDevs.TriEngine2D.Input.InputManager, 17
- TriDevs.TriEngine2D.Input.InputManager, 23
- TriDevs.TriEngine2D.Input.NullInputManager, 32
- TriDevs.TriEngine2D.Logging, 12
- TriDevs.TriEngine2D.Logging.LogManager, 29
- TriDevs.TriEngine2D.Native, 12
- TriDevs.TriEngine2D.Native.Helpers, 15
- TriDevs.TriEngine2D.Native.WinAPI, 45
- TriDevs.TriEngine2D.Point, 38
- TriDevs.TriEngine2D.Serializing, 12
- TriDevs.TriEngine2D.Serializing.Serializer, 39
- TriDevs.TriEngine2D.Services, 41
- TriDevs.TriEngine2D.Version, 44
- TriDevs.TriEngine2D.Extensions/EnumerationExtensions.cs, 49
- TriDevs.TriEngine2D.Extensions/StringExtensions.cs, 49
- TriDevs.TriEngine2D/Helpers/IO.cs, 49
- TriDevs.TriEngine2D/Helpers/Threading.cs, 50
- TriDevs.TriEngine2D/Input/NullInputManager.cs, 50
- TriDevs.TriEngine2D/Input/InputManager.cs, 50
- TriDevs.TriEngine2D/Input/NullInputManager.cs, 50
- TriDevs.TriEngine2D/Logging/LogManager.cs, 51
- TriDevs.TriEngine2D/Native/Helpers.cs, 51
- TriDevs.TriEngine2D/Native/WinAPI.cs, 51
- TriDevs.TriEngine2D/Point.cs, 51
- TriDevs.TriEngine2D/Properties/AssemblyInfo.cs, 52
- TriDevs.TriEngine2D/Serializing/Serializer.cs, 52
- TriDevs.TriEngine2D/Services.cs, 52
- TriDevs.TriEngine2D/Version.cs, 52
- TriDevs::TriEngine2D::Extensions::EnumerationExtensions
 - Has< T >, 13
 - Include< T >, 14
 - Missing< T >, 14
 - Remove< T >, 15
- TriDevs::TriEngine2D::Extensions::StringExtensions
 - Replace, 42
 - ReplaceFirst, 43
- TriDevs::TriEngine2D::Helpers::IO
 - GetAbsolutePath, 29
- TriDevs::TriEngine2D::Helpers::Threading
 - SetCurrentThreadName, 44
- TriDevs::TriEngine2D::Input::InputManager
 - ButtonDown, 19
 - ButtonPressed, 19
 - ButtonReleased, 19
 - ButtonUp, 20
 - KeyDown, 20
 - KeyPressed, 20
 - KeyReleased, 20
 - KeyUp, 21
 - MousePosition, 22
 - MouseWheelValue, 22
 - MouseX, 22
 - MouseY, 22
 - Update, 21
 - WheelChange, 21
 - WheelChanged, 21
 - WheelDown, 21
 - WheelUp, 21
- TriDevs::TriEngine2D::Input::InputManager
 - ButtonDown, 24
 - ButtonPressed, 25
 - ButtonReleased, 25
 - ButtonUp, 25
 - InputManager, 24
 - KeyDown, 26
 - KeyPressed, 26
 - KeyReleased, 26
 - KeyUp, 27
 - MousePosition, 28
 - MouseWheelValue, 28
 - MouseX, 28

- MouseY, [28](#)
 - Update, [27](#)
 - WheelChange, [27](#)
 - WheelChanged, [27](#)
 - WheelDown, [28](#)
 - WheelUp, [28](#)
- TriDevs::TriEngine2D::Input::NullInputManager
 - ButtonDown, [34](#)
 - ButtonPressed, [34](#)
 - ButtonReleased, [35](#)
 - ButtonUp, [35](#)
 - KeyDown, [35](#)
 - KeyPressed, [36](#)
 - KeyReleased, [36](#)
 - KeyUp, [36](#)
 - MousePosition, [38](#)
 - MouseWheelValue, [38](#)
 - MouseX, [38](#)
 - MouseY, [38](#)
 - Update, [37](#)
 - WheelChange, [37](#)
 - WheelChanged, [37](#)
 - WheelDown, [37](#)
 - WheelUp, [37](#)
- TriDevs::TriEngine2D::Logging::LogManager
 - ClearOldLogs, [30](#)
 - DestroyConsole, [30](#)
 - GetLogger, [31](#)
 - LoadConfig, [31](#)
 - SetupConsole, [32](#)
- TriDevs::TriEngine2D::Native::Helpers
 - GetErrorMessage, [16](#)
 - GetLastError, [16](#)
 - GetLastErrorInfo, [16](#)
 - GetLastErrorMessages, [17](#)
 - GetLastWin32Exception, [17](#)
 - GetWin32Exception, [17](#)
- TriDevs::TriEngine2D::Native::WinAPI
 - AllocConsole, [46](#)
 - FreeConsole, [46](#)
 - GetStdHandle, [46](#)
- TriDevs::TriEngine2D::Point
 - Point, [39](#)
 - X, [39](#)
 - Y, [39](#)
- TriDevs::TriEngine2D::Serializing::Serializer
 - Deserialize< T >, [39](#)
 - Serialize< T >, [40](#)
- TriDevs::TriEngine2D::Services
 - Input, [42](#)
 - Provide, [41](#)
- TriDevs::TriEngine2D::Version
 - Major, [45](#)
 - Minor, [45](#)
 - Patch, [45](#)
 - Suffix, [45](#)
 - VersionString, [45](#)
 - VersionStringFormat, [45](#)
 - VersionStringFormatWithSuffix, [45](#)
- Update
 - TriDevs::TriEngine2D::Input::InputManager, [21](#)
 - TriDevs::TriEngine2D::Input::InputManager, [27](#)
 - TriDevs::TriEngine2D::Input::NullInputManager, [37](#)
- VersionString
 - TriDevs::TriEngine2D::Version, [45](#)
- VersionStringFormat
 - TriDevs::TriEngine2D::Version, [45](#)
- VersionStringFormatWithSuffix
 - TriDevs::TriEngine2D::Version, [45](#)
- WheelChange
 - TriDevs::TriEngine2D::Input::InputManager, [21](#)
 - TriDevs::TriEngine2D::Input::InputManager, [27](#)
 - TriDevs::TriEngine2D::Input::NullInputManager, [37](#)
- WheelChanged
 - TriDevs::TriEngine2D::Input::InputManager, [21](#)
 - TriDevs::TriEngine2D::Input::InputManager, [27](#)
 - TriDevs::TriEngine2D::Input::NullInputManager, [37](#)
- WheelDown
 - TriDevs::TriEngine2D::Input::InputManager, [21](#)
 - TriDevs::TriEngine2D::Input::InputManager, [28](#)
 - TriDevs::TriEngine2D::Input::NullInputManager, [37](#)
- WheelUp
 - TriDevs::TriEngine2D::Input::InputManager, [21](#)
 - TriDevs::TriEngine2D::Input::InputManager, [28](#)
 - TriDevs::TriEngine2D::Input::NullInputManager, [37](#)
- X
 - TriDevs::TriEngine2D::Point, [39](#)
- Y
 - TriDevs::TriEngine2D::Point, [39](#)