

# TriEngine

v0.0.16

Generated by Doxygen 1.8.3.1

Thu Jul 11 2013 19:20:01



# Contents

<b>1</b>	<b>Main Page</b>	<b>1</b>
<b>2</b>	<b>Namespace Index</b>	<b>3</b>
2.1	Namespace List . . . . .	3
<b>3</b>	<b>Hierarchical Index</b>	<b>5</b>
3.1	Class Hierarchy . . . . .	5
<b>4</b>	<b>Class Index</b>	<b>7</b>
4.1	Class List . . . . .	7
<b>5</b>	<b>File Index</b>	<b>11</b>
5.1	File List . . . . .	11
<b>6</b>	<b>Namespace Documentation</b>	<b>13</b>
6.1	Package TriDevs . . . . .	13
6.2	Package TriDevs.TriEngine . . . . .	13
6.3	Package TriDevs.TriEngine.Audio . . . . .	14
6.3.1	Enumeration Type Documentation . . . . .	14
6.3.1.1	AudioFormat . . . . .	14
6.4	Package TriDevs.TriEngine.Extensions . . . . .	15
6.5	Package TriDevs.TriEngine.Graphics . . . . .	15
6.6	Package TriDevs.TriEngine.Helpers . . . . .	15
6.7	Package TriDevs.TriEngine.Input . . . . .	16
6.8	Package TriDevs.TriEngine.Input.Events . . . . .	16
6.8.1	Function Documentation . . . . .	16
6.8.1.1	KeyDownEventHandler . . . . .	16
6.8.1.2	KeyPressEventHandler . . . . .	17
6.8.1.3	KeyUpEventHandler . . . . .	17
6.8.1.4	MouseDownEventHandler . . . . .	17
6.8.1.5	MouseUpEventHandler . . . . .	17
6.8.1.6	MouseWheelChangedEventHandler . . . . .	17
6.8.1.7	MouseWheelDownEventHandler . . . . .	18

6.8.1.8	MouseWheelUpEventHandler	18
6.9	Package TriDevs.TriEngine.Interfaces	18
6.10	Package TriDevs.TriEngine.Logging	18
6.11	Package TriDevs.TriEngine.Native	18
6.12	Package TriDevs.TriEngine.Resources	19
6.13	Package TriDevs.TriEngine.Serializing	19
6.14	Package TriDevs.TriEngine.Shaders	19
6.15	Package TriDevs.TriEngine.StateManagement	19
6.16	Package TriDevs.TriEngine.Text	19
6.16.1	Enumeration Type Documentation	20
6.16.1.1	FontType	20
6.17	Package TriDevs.TriEngine.UI	20
6.18	Package TriDevs.TriEngine.UI.Events	21
6.18.1	Function Documentation	21
6.18.1.1	ControlClickedEventHandler	21
<b>7</b>	<b>Class Documentation</b>	<b>23</b>
7.1	TriDevs.TriEngine.Audio.AudioManager Class Reference	23
7.1.1	Detailed Description	24
7.1.2	Constructor & Destructor Documentation	24
7.1.2.1	AudioManager	24
7.1.3	Member Function Documentation	24
7.1.3.1	Dispose	24
7.1.3.2	StopAll	25
7.1.3.3	StopAllSongs	25
7.1.3.4	StopAllSounds	25
7.2	TriDevs.TriEngine.Extensions.ByteExtensions Class Reference	25
7.2.1	Detailed Description	26
7.2.2	Member Function Documentation	26
7.2.2.1	Clamp	26
7.3	TriDevs.TriEngine.Color Struct Reference	27
7.3.1	Detailed Description	34
7.3.2	Constructor & Destructor Documentation	34
7.3.2.1	Color	34
7.3.2.2	Color	34
7.3.2.3	Color	34
7.3.2.4	Color	34
7.3.2.5	Color	35
7.3.2.6	Color	35
7.3.2.7	Color	35

7.3.2.8	Color	36
7.3.3	Member Function Documentation	36
7.3.3.1	FromAbgr	36
7.3.3.2	FromBgra	36
7.3.3.3	FromRgba	37
7.3.3.4	ToAbgr	37
7.3.3.5	ToArgb	37
7.3.3.6	ToBgra	38
7.3.3.7	ToColor4	38
7.3.3.8	ToRgba	38
7.3.3.9	ToVector3	38
7.3.3.10	ToVector4	39
7.3.4	Member Data Documentation	39
7.3.4.1	A	39
7.3.4.2	AliceBlue	39
7.3.4.3	AntiqueWhite	39
7.3.4.4	Aqua	39
7.3.4.5	Aquamarine	39
7.3.4.6	Azure	39
7.3.4.7	B	39
7.3.4.8	Beige	40
7.3.4.9	Bisque	40
7.3.4.10	Black	40
7.3.4.11	BlanchedAlmond	40
7.3.4.12	Blue	40
7.3.4.13	BlueViolet	40
7.3.4.14	Brown	40
7.3.4.15	BurlyWood	40
7.3.4.16	CadetBlue	40
7.3.4.17	Chartreuse	40
7.3.4.18	Chocolate	40
7.3.4.19	Coral	40
7.3.4.20	CornflowerBlue	41
7.3.4.21	Cornsilk	41
7.3.4.22	Crimson	41
7.3.4.23	Cyan	41
7.3.4.24	DarkBlue	41
7.3.4.25	DarkCyan	41
7.3.4.26	DarkGoldenrod	41
7.3.4.27	DarkGray	41

7.3.4.28	DarkGreen	41
7.3.4.29	DarkKhaki	41
7.3.4.30	DarkMagenta	41
7.3.4.31	DarkOliveGreen	41
7.3.4.32	DarkOrange	42
7.3.4.33	DarkOrchid	42
7.3.4.34	DarkRed	42
7.3.4.35	DarkSalmon	42
7.3.4.36	DarkSeaGreen	42
7.3.4.37	DarkSlateBlue	42
7.3.4.38	DarkSlateGray	42
7.3.4.39	DarkTurquoise	42
7.3.4.40	DarkViolet	42
7.3.4.41	DeepPink	42
7.3.4.42	DeepSkyBlue	42
7.3.4.43	DimGray	42
7.3.4.44	DodgerBlue	43
7.3.4.45	Firebrick	43
7.3.4.46	FloralWhite	43
7.3.4.47	ForestGreen	43
7.3.4.48	Fuchsia	43
7.3.4.49	G	43
7.3.4.50	Gainsboro	43
7.3.4.51	GhostWhite	43
7.3.4.52	Gold	43
7.3.4.53	Goldenrod	43
7.3.4.54	Gray	43
7.3.4.55	Green	43
7.3.4.56	GreenYellow	44
7.3.4.57	HoneyDew	44
7.3.4.58	HotPink	44
7.3.4.59	IndianRed	44
7.3.4.60	Indigo	44
7.3.4.61	Ivory	44
7.3.4.62	Khaki	44
7.3.4.63	Lavender	44
7.3.4.64	LavenderBlush	44
7.3.4.65	LawnGreen	44
7.3.4.66	LemonChiffon	44
7.3.4.67	LightBlue	44

7.3.4.68	LightCoral	45
7.3.4.69	LightCyan	45
7.3.4.70	LightGoldenrodYellow	45
7.3.4.71	LightGray	45
7.3.4.72	LightGreen	45
7.3.4.73	LightPink	45
7.3.4.74	LightSalmon	45
7.3.4.75	LightSeaGreen	45
7.3.4.76	LightSkyBlue	45
7.3.4.77	LightSlateGray	45
7.3.4.78	LightSteelBlue	45
7.3.4.79	LightYellow	45
7.3.4.80	Lime	46
7.3.4.81	LimeGreen	46
7.3.4.82	Linen	46
7.3.4.83	Magenta	46
7.3.4.84	Maroon	46
7.3.4.85	MediumAquamarine	46
7.3.4.86	MediumBlue	46
7.3.4.87	MediumOrchid	46
7.3.4.88	MediumPurple	46
7.3.4.89	MediumSeaGreen	46
7.3.4.90	MediumSlateBlue	46
7.3.4.91	MediumSpringGreen	46
7.3.4.92	MediumTurquoise	47
7.3.4.93	MediumVioletRed	47
7.3.4.94	MidnightBlue	47
7.3.4.95	MintCream	47
7.3.4.96	MistyRose	47
7.3.4.97	Moccasin	47
7.3.4.98	NavajoWhite	47
7.3.4.99	Navy	47
7.3.4.100	OldLace	47
7.3.4.101	Olive	47
7.3.4.102	OliveDrab	47
7.3.4.103	Orange	47
7.3.4.104	OrangeRed	48
7.3.4.105	Orchid	48
7.3.4.106	PaleGoldenrod	48
7.3.4.107	PaleGreen	48

7.3.4.108 PaleTurquoise . . . . .	48
7.3.4.109 PaleVioletRed . . . . .	48
7.3.4.110 PapayaWhip . . . . .	48
7.3.4.111 PeachPuff . . . . .	48
7.3.4.112 Peru . . . . .	48
7.3.4.113 Pink . . . . .	48
7.3.4.114 Plum . . . . .	48
7.3.4.115 PowderBlue . . . . .	48
7.3.4.116 Purple . . . . .	49
7.3.4.117 R . . . . .	49
7.3.4.118 Red . . . . .	49
7.3.4.119 RosyBrown . . . . .	49
7.3.4.120 RoyalBlue . . . . .	49
7.3.4.121 SaddleBrown . . . . .	49
7.3.4.122 Salmon . . . . .	49
7.3.4.123 SandyBrown . . . . .	49
7.3.4.124 SeaGreen . . . . .	49
7.3.4.125 SeaShell . . . . .	49
7.3.4.126 Sienna . . . . .	49
7.3.4.127 Silver . . . . .	49
7.3.4.128 SkyBlue . . . . .	50
7.3.4.129 SlateBlue . . . . .	50
7.3.4.130 SlateGray . . . . .	50
7.3.4.131 Snow . . . . .	50
7.3.4.132 SpringGreen . . . . .	50
7.3.4.133 SteelBlue . . . . .	50
7.3.4.134 Tan . . . . .	50
7.3.4.135 Teal . . . . .	50
7.3.4.136 Thistle . . . . .	50
7.3.4.137 Tomato . . . . .	50
7.3.4.138 TransparentBlack . . . . .	50
7.3.4.139 TransparentWhite . . . . .	50
7.3.4.140 Turquoise . . . . .	51
7.3.4.141 Violet . . . . .	51
7.3.4.142 Wheat . . . . .	51
7.3.4.143 White . . . . .	51
7.3.4.144 WhiteSmoke . . . . .	51
7.3.4.145 Yellow . . . . .	51
7.3.4.146 YellowGreen . . . . .	51
7.4 TriDevs.TriEngine.UI.Control Class Reference . . . . .	51



7.4.1	Detailed Description	53
7.4.2	Member Function Documentation	53
7.4.2.1	Disable	53
7.4.2.2	Draw	54
7.4.2.3	Draw	54
7.4.2.4	Enable	54
7.4.2.5	Hide	54
7.4.2.6	OnClicked	55
7.4.2.7	Show	55
7.4.2.8	Update	55
7.4.3	Property Documentation	55
7.4.3.1	Color	55
7.4.3.2	Enabled	55
7.4.3.3	Position	56
7.4.3.4	Rectangle	56
7.4.3.5	Size	56
7.4.3.6	Text	56
7.4.3.7	Visible	56
7.4.4	Event Documentation	56
7.4.4.1	Clicked	56
7.5	TriDevs.TriEngine.UI.ControlManager Class Reference	56
7.5.1	Detailed Description	58
7.5.2	Constructor & Destructor Documentation	58
7.5.2.1	ControlManager	58
7.5.3	Member Function Documentation	58
7.5.3.1	AddControl	58
7.5.3.2	Disable	59
7.5.3.3	Draw	59
7.5.3.4	Enable	59
7.5.3.5	HasControl	59
7.5.3.6	HasControl	60
7.5.3.7	HasControl	60
7.5.3.8	RemoveAllControls	60
7.5.3.9	RemoveAllControls	61
7.5.3.10	RemoveControl	61
7.5.3.11	Update	62
7.5.4	Property Documentation	62
7.5.4.1	Enabled	62
7.6	TriDevs.TriEngine.Extensions.DecimalExtensions Class Reference	62
7.6.1	Detailed Description	62

7.6.2	Member Function Documentation	62
7.6.2.1	Clamp	62
7.7	TriDevs.TriEngine.Extensions.DoubleExtensions Class Reference	63
7.7.1	Detailed Description	63
7.7.2	Member Function Documentation	64
7.7.2.1	Clamp	64
7.8	TriDevs.TriEngine.EngineException Class Reference	64
7.8.1	Detailed Description	65
7.9	TriDevs.TriEngine.Extensions.EnumerationExtensions Class Reference	65
7.9.1	Detailed Description	66
7.9.2	Member Function Documentation	66
7.9.2.1	Has< T >	66
7.9.2.2	Include< T >	66
7.9.2.3	Missing< T >	67
7.9.2.4	Remove< T >	67
7.10	TriDevs.TriEngine.Helpers.Exceptions Class Reference	68
7.10.1	Detailed Description	68
7.10.2	Member Function Documentation	68
7.10.2.1	LogException	68
7.11	TriDevs.TriEngine.Extensions.FloatExtensions Class Reference	69
7.11.1	Detailed Description	69
7.11.2	Member Function Documentation	69
7.11.2.1	Clamp	69
7.12	TriDevs.TriEngine.Text.Font Class Reference	70
7.12.1	Detailed Description	72
7.12.2	Constructor & Destructor Documentation	72
7.12.2.1	Font	72
7.12.2.2	Font	72
7.12.3	Member Function Documentation	73
7.12.3.1	Dispose	73
7.12.3.2	GetDefaultName	74
7.12.3.3	ToString	74
7.12.4	Property Documentation	74
7.12.4.1	Color	74
7.12.4.2	File	74
7.12.4.3	Name	74
7.12.4.4	QFont	74
7.12.4.5	Size	74
7.12.4.6	Type	74
7.13	TriDevs.TriEngine.Text.FontConstructionConfig Class Reference	75

7.13.1 Detailed Description . . . . .	75
7.13.2 Constructor & Destructor Documentation . . . . .	75
7.13.2.1 FontConstructionConfig . . . . .	75
7.13.2.2 FontConstructionConfig . . . . .	75
7.13.3 Member Data Documentation . . . . .	76
7.13.3.1 BuildConfig . . . . .	76
7.13.3.2 LoadConfig . . . . .	76
7.14 TriDevs.TriEngine.StateManagement.GameState Class Reference . . . . .	76
7.14.1 Detailed Description . . . . .	78
7.14.2 Constructor & Destructor Documentation . . . . .	78
7.14.2.1 GameState . . . . .	78
7.14.3 Member Function Documentation . . . . .	78
7.14.3.1 AddComponent . . . . .	78
7.14.3.2 Draw . . . . .	79
7.14.3.3 GetAllComponents . . . . .	80
7.14.3.4 GetAllComponents . . . . .	80
7.14.3.5 GetAllComponents . . . . .	80
7.14.3.6 GetComponent . . . . .	80
7.14.3.7 GetComponent . . . . .	81
7.14.3.8 HasComponent . . . . .	81
7.14.3.9 HasComponent . . . . .	81
7.14.3.10 HasComponent . . . . .	82
7.14.3.11 Load . . . . .	82
7.14.3.12 Pause . . . . .	82
7.14.3.13 RemoveAllComponents . . . . .	82
7.14.3.14 RemoveAllComponents . . . . .	83
7.14.3.15 RemoveAllComponents . . . . .	83
7.14.3.16 RemoveComponent . . . . .	83
7.14.3.17 Unload . . . . .	84
7.14.3.18 Unpause . . . . .	84
7.14.3.19 Update . . . . .	84
7.14.4 Member Data Documentation . . . . .	84
7.14.4.1 Components . . . . .	84
7.14.5 Property Documentation . . . . .	84
7.14.5.1 Paused . . . . .	84
7.15 TriDevs.TriEngine.StateManagement.GameStateManager Class Reference . . . . .	85
7.15.1 Detailed Description . . . . .	86
7.15.2 Constructor & Destructor Documentation . . . . .	86
7.15.2.1 GameStateManager . . . . .	86
7.15.2.2 GameStateManager . . . . .	86

7.15.3	Member Function Documentation	86
7.15.3.1	Draw	86
7.15.3.2	Peek	87
7.15.3.3	Pop	87
7.15.3.4	Push	87
7.15.3.5	Switch	88
7.15.3.6	Update	88
7.15.4	Property Documentation	89
7.15.4.1	ActiveState	89
7.15.4.2	StateCount	89
7.16	TriDevs.TriEngine.GameWindow2D Class Reference	89
7.16.1	Detailed Description	90
7.16.2	Constructor & Destructor Documentation	90
7.16.2.1	GameWindow2D	90
7.16.3	Member Function Documentation	90
7.16.3.1	Initialize	90
7.16.3.2	OnDraw	90
7.16.3.3	OnLoad	90
7.16.3.4	OnRenderFrame	91
7.16.3.5	OnResize	91
7.16.4	Property Documentation	91
7.16.4.1	ClearColor	91
7.17	TriDevs.TriEngine.Native.Helpers Class Reference	91
7.17.1	Detailed Description	91
7.17.2	Member Function Documentation	92
7.17.2.1	GetErrorMessage	92
7.17.2.2	GetLastError	92
7.17.2.3	GetLastErrorInfo	92
7.17.2.4	GetLastErrorMessage	92
7.18	TriDevs.TriEngine.Audio.IAudioManager Interface Reference	93
7.18.1	Detailed Description	94
7.18.2	Member Function Documentation	94
7.18.2.1	StopAll	94
7.18.2.2	StopAllSongs	94
7.18.2.3	StopAllSounds	94
7.19	TriDevs.TriEngine.UI.IControl Interface Reference	94
7.19.1	Detailed Description	96
7.19.2	Member Function Documentation	96
7.19.2.1	Disable	96
7.19.2.2	Enable	96

7.19.2.3	Hide	97
7.19.2.4	Show	97
7.19.3	Property Documentation	97
7.19.3.1	Color	97
7.19.3.2	Enabled	98
7.19.3.3	Position	98
7.19.3.4	Rectangle	98
7.19.3.5	Size	98
7.19.3.6	Text	98
7.19.3.7	Visible	98
7.19.4	Event Documentation	98
7.19.4.1	Clicked	98
7.20	TriDevs.TriEngine.UI.IControlManager Interface Reference	98
7.20.1	Detailed Description	100
7.20.2	Member Function Documentation	101
7.20.2.1	AddControl	101
7.20.2.2	HasControl	101
7.20.2.3	HasControl	101
7.20.2.4	HasControl	101
7.20.2.5	RemoveAllControls	102
7.20.2.6	RemoveAllControls	102
7.20.2.7	RemoveControl	102
7.21	TriDevs.TriEngine.Interfaces.IDrawable Interface Reference	102
7.21.1	Detailed Description	103
7.21.2	Member Function Documentation	103
7.21.2.1	Draw	103
7.22	TriDevs.TriEngine.Interfaces.IDrawableGameComponent Interface Reference	103
7.22.1	Detailed Description	105
7.23	TriDevs.TriEngine.Interfaces.IGameComponent Interface Reference	105
7.23.1	Detailed Description	106
7.23.2	Member Function Documentation	106
7.23.2.1	Disable	106
7.23.2.2	Enable	107
7.23.3	Property Documentation	107
7.23.3.1	Enabled	107
7.24	TriDevs.TriEngine.StateManagement.IGameState Interface Reference	107
7.24.1	Detailed Description	109
7.24.2	Member Function Documentation	109
7.24.2.1	AddComponent	109
7.24.2.2	GetAllComponents	109

7.24.2.3	<a href="#">GetAllComponents</a>	109
7.24.2.4	<a href="#">GetAllComponents</a>	110
7.24.2.5	<a href="#">GetComponent</a>	110
7.24.2.6	<a href="#">GetComponent</a>	110
7.24.2.7	<a href="#">HasComponent</a>	110
7.24.2.8	<a href="#">HasComponent</a>	111
7.24.2.9	<a href="#">HasComponent</a>	111
7.24.2.10	<a href="#">Load</a>	111
7.24.2.11	<a href="#">Pause</a>	111
7.24.2.12	<a href="#">RemoveAllComponents</a>	112
7.24.2.13	<a href="#">RemoveAllComponents</a>	112
7.24.2.14	<a href="#">RemoveAllComponents</a>	112
7.24.2.15	<a href="#">RemoveComponent</a>	112
7.24.2.16	<a href="#">Unload</a>	112
7.24.2.17	<a href="#">Unpause</a>	113
7.24.3	<a href="#">Property Documentation</a>	113
7.24.3.1	<a href="#">Paused</a>	113
7.25	<a href="#">TriDevs.TriEngine.StateManagement.IGameStateManager Interface Reference</a>	113
7.25.1	<a href="#">Detailed Description</a>	114
7.25.2	<a href="#">Member Function Documentation</a>	114
7.25.2.1	<a href="#">Peek</a>	114
7.25.2.2	<a href="#">Pop</a>	115
7.25.2.3	<a href="#">Push</a>	115
7.25.2.4	<a href="#">Switch</a>	115
7.25.3	<a href="#">Property Documentation</a>	115
7.25.3.1	<a href="#">ActiveState</a>	115
7.25.3.2	<a href="#">StateCount</a>	115
7.26	<a href="#">TriDevs.TriEngine.Input.IInputManager Interface Reference</a>	115
7.26.1	<a href="#">Detailed Description</a>	117
7.26.2	<a href="#">Member Function Documentation</a>	117
7.26.2.1	<a href="#">IsKeyDown</a>	117
7.26.2.2	<a href="#">IsKeyUp</a>	118
7.26.2.3	<a href="#">IsMouseDown</a>	118
7.26.2.4	<a href="#">IsMouseUp</a>	118
7.26.2.5	<a href="#">IsWheelChanged</a>	118
7.26.2.6	<a href="#">IsWheelDown</a>	119
7.26.2.7	<a href="#">IsWheelUp</a>	119
7.26.2.8	<a href="#">KeyPressed</a>	119
7.26.2.9	<a href="#">KeyReleased</a>	119
7.26.2.10	<a href="#">MousePressed</a>	119

7.26.2.11	<a href="#">MouseReleased</a>	120
7.26.2.12	<a href="#">Update</a>	120
7.26.2.13	<a href="#">WheelChange</a>	120
7.26.3	<a href="#">Property Documentation</a>	120
7.26.3.1	<a href="#">MousePosition</a>	120
7.26.3.2	<a href="#">MouseWheelValue</a>	120
7.26.3.3	<a href="#">MouseX</a>	120
7.26.3.4	<a href="#">MouseY</a>	121
7.26.3.5	<a href="#">this[Key key]</a>	121
7.26.3.6	<a href="#">this[MouseButton button]</a>	121
7.26.4	<a href="#">Event Documentation</a>	121
7.26.4.1	<a href="#">KeyDown</a>	121
7.26.4.2	<a href="#">KeyPress</a>	121
7.26.4.3	<a href="#">KeyUp</a>	121
7.26.4.4	<a href="#">MouseDown</a>	121
7.26.4.5	<a href="#">MouseUp</a>	121
7.26.4.6	<a href="#">WheelChanged</a>	122
7.26.4.7	<a href="#">WheelDown</a>	122
7.26.4.8	<a href="#">WheelUp</a>	122
7.27	<a href="#">TriDevs.TriEngine.Input.InputManager Class Reference</a>	122
7.27.1	<a href="#">Detailed Description</a>	124
7.27.2	<a href="#">Constructor &amp; Destructor Documentation</a>	124
7.27.2.1	<a href="#">InputManager</a>	124
7.27.2.2	<a href="#">InputManager</a>	124
7.27.3	<a href="#">Member Function Documentation</a>	124
7.27.3.1	<a href="#">IsKeyDown</a>	124
7.27.3.2	<a href="#">IsKeyUp</a>	125
7.27.3.3	<a href="#">IsMouseDown</a>	125
7.27.3.4	<a href="#">IsMouseUp</a>	125
7.27.3.5	<a href="#">IsWheelChanged</a>	126
7.27.3.6	<a href="#">IsWheelDown</a>	126
7.27.3.7	<a href="#">IsWheelUp</a>	126
7.27.3.8	<a href="#">KeyPressed</a>	126
7.27.3.9	<a href="#">KeyReleased</a>	127
7.27.3.10	<a href="#">MousePressed</a>	127
7.27.3.11	<a href="#">MouseReleased</a>	127
7.27.3.12	<a href="#">Update</a>	128
7.27.3.13	<a href="#">WheelChange</a>	128
7.27.4	<a href="#">Property Documentation</a>	128
7.27.4.1	<a href="#">MousePosition</a>	128

7.27.4.2	<a href="#">MouseWheelValue</a>	128
7.27.4.3	<a href="#">MouseX</a>	128
7.27.4.4	<a href="#">MouseY</a>	128
7.27.4.5	<a href="#">this[Key key]</a>	128
7.27.4.6	<a href="#">this[MouseButton button]</a>	128
7.27.5	<a href="#">Event Documentation</a>	128
7.27.5.1	<a href="#">KeyDown</a>	128
7.27.5.2	<a href="#">KeyPress</a>	129
7.27.5.3	<a href="#">KeyUp</a>	129
7.27.5.4	<a href="#">MouseDown</a>	129
7.27.5.5	<a href="#">MouseUp</a>	129
7.27.5.6	<a href="#">WheelChanged</a>	129
7.27.5.7	<a href="#">WheelDown</a>	129
7.27.5.8	<a href="#">WheelUp</a>	129
7.28	<a href="#">TriDevs.TriEngine.Extensions.IntegerExtensions Class Reference</a>	129
7.28.1	<a href="#">Detailed Description</a>	130
7.28.2	<a href="#">Member Function Documentation</a>	130
7.28.2.1	<a href="#">Clamp</a>	130
7.28.2.2	<a href="#">Clamp</a>	131
7.28.2.3	<a href="#">Clamp</a>	131
7.28.2.4	<a href="#">Clamp</a>	131
7.28.2.5	<a href="#">Clamp</a>	132
7.28.2.6	<a href="#">Clamp</a>	132
7.29	<a href="#">TriDevs.TriEngine.Helpers.IO Class Reference</a>	132
7.29.1	<a href="#">Detailed Description</a>	132
7.29.2	<a href="#">Member Function Documentation</a>	133
7.29.2.1	<a href="#">GetAbsolutePath</a>	133
7.30	<a href="#">TriDevs.TriEngine.Resources.IResource Interface Reference</a>	133
7.30.1	<a href="#">Detailed Description</a>	134
7.30.2	<a href="#">Property Documentation</a>	134
7.30.2.1	<a href="#">Name</a>	134
7.31	<a href="#">TriDevs.TriEngine.Audio.ISong Interface Reference</a>	134
7.31.1	<a href="#">Detailed Description</a>	135
7.31.2	<a href="#">Member Function Documentation</a>	135
7.31.2.1	<a href="#">Pause</a>	135
7.31.2.2	<a href="#">Play</a>	136
7.31.2.3	<a href="#">Resume</a>	136
7.31.2.4	<a href="#">Stop</a>	136
7.31.3	<a href="#">Property Documentation</a>	136
7.31.3.1	<a href="#">File</a>	136



7.31.3.2	IsLooped	136
7.31.3.3	Volume	136
7.32	TriDevs.TriEngine.Audio.ISound Interface Reference	136
7.32.1	Detailed Description	138
7.32.2	Member Function Documentation	138
7.32.2.1	Play	138
7.32.2.2	Stop	138
7.32.3	Property Documentation	138
7.32.3.1	File	138
7.33	TriDevs.TriEngine.Text.ITextObject Interface Reference	138
7.33.1	Detailed Description	139
7.33.2	Member Function Documentation	139
7.33.2.1	Draw	139
7.33.2.2	Draw	139
7.33.2.3	Draw	139
7.33.3	Property Documentation	140
7.33.3.1	Alignment	140
7.33.3.2	Bounds	140
7.33.3.3	Color	140
7.33.3.4	Font	140
7.33.3.5	Position	140
7.33.3.6	Text	140
7.34	TriDevs.TriEngine.Interfaces.IUpdatable Interface Reference	140
7.34.1	Detailed Description	141
7.34.2	Member Function Documentation	141
7.34.2.1	Update	141
7.35	TriDevs.TriEngine.Input.Events.KeyCharEventArgs Class Reference	141
7.35.1	Detailed Description	142
7.35.2	Member Data Documentation	142
7.35.2.1	KeyChar	142
7.36	TriDevs.TriEngine.Input.Events.KeyEventArgs Class Reference	142
7.36.1	Detailed Description	143
7.36.2	Member Data Documentation	143
7.36.2.1	Key	143
7.37	TriDevs.TriEngine.UI.Label Class Reference	144
7.37.1	Detailed Description	145
7.37.2	Member Function Documentation	146
7.37.2.1	Draw	146
7.37.2.2	SetFont	146
7.37.2.3	Update	146

7.37.2.4	UpdateTextObject	146
7.37.3	Property Documentation	147
7.37.3.1	Alignment	147
7.37.3.2	Position	147
7.37.3.3	Text	147
7.38	TriDevs.TriEngine.UI.LinkLabel Class Reference	147
7.38.1	Detailed Description	149
7.38.2	Member Function Documentation	149
7.38.2.1	OnClicked	149
7.38.3	Property Documentation	150
7.38.3.1	Url	150
7.39	TriDevs.TriEngine.Logging.LogManager Class Reference	150
7.39.1	Detailed Description	151
7.39.2	Member Function Documentation	151
7.39.2.1	ClearOldLogs	151
7.39.2.2	DestroyConsole	151
7.39.2.3	GetLogger	151
7.39.2.4	LoadConfig	152
7.39.2.5	SetupConsole	152
7.40	TriDevs.TriEngine.Helpers.Math Class Reference	153
7.40.1	Detailed Description	153
7.40.2	Member Function Documentation	153
7.40.2.1	Clamp	153
7.40.2.2	Clamp	154
7.40.2.3	Clamp	154
7.40.2.4	Clamp	154
7.40.2.5	Clamp	155
7.40.2.6	Clamp	155
7.40.2.7	Clamp	155
7.40.2.8	Clamp	156
7.40.2.9	Clamp	156
7.40.2.10	Clamp	156
7.41	TriDevs.TriEngine.Audio.NullAudioManager Class Reference	157
7.41.1	Detailed Description	158
7.41.2	Member Function Documentation	158
7.41.2.1	Dispose	158
7.41.2.2	StopAll	158
7.41.2.3	StopAllSongs	159
7.41.2.4	StopAllSounds	159
7.42	TriDevs.TriEngine.Input.NullInputManager Class Reference	159

7.42.1	Detailed Description	161
7.42.2	Member Function Documentation	161
7.42.2.1	IsKeyDown	161
7.42.2.2	IsKeyUp	161
7.42.2.3	IsMouseDown	162
7.42.2.4	IsMouseUp	162
7.42.2.5	IsWheelChanged	162
7.42.2.6	IsWheelDown	163
7.42.2.7	IsWheelUp	163
7.42.2.8	KeyPressed	163
7.42.2.9	KeyReleased	163
7.42.2.10	MousePressed	164
7.42.2.11	MouseReleased	164
7.42.2.12	Update	164
7.42.2.13	WheelChange	165
7.42.3	Property Documentation	165
7.42.3.1	MousePosition	165
7.42.3.2	MouseWheelValue	165
7.42.3.3	MouseX	165
7.42.3.4	MouseY	165
7.42.3.5	this[Key key]	165
7.42.3.6	this[MouseButton button]	165
7.42.4	Event Documentation	165
7.42.4.1	KeyDown	165
7.42.4.2	KeyPress	165
7.42.4.3	KeyUp	165
7.42.4.4	MouseDown	165
7.42.4.5	MouseUp	166
7.42.4.6	WheelChanged	166
7.42.4.7	WheelDown	166
7.42.4.8	WheelUp	166
7.43	TriDevs.TriEngine.Audio.NullSong Class Reference	166
7.43.1	Detailed Description	168
7.43.2	Member Function Documentation	168
7.43.2.1	Dispose	168
7.43.2.2	Pause	168
7.43.2.3	Play	168
7.43.2.4	Resume	169
7.43.2.5	Stop	169
7.43.3	Property Documentation	169

7.43.3.1	File	169
7.43.3.2	IsLooped	169
7.43.3.3	Name	169
7.43.3.4	Volume	169
7.44	TriDevs.TriEngine.Audio.NullSound Class Reference	169
7.44.1	Detailed Description	171
7.44.2	Member Function Documentation	171
7.44.2.1	Dispose	171
7.44.2.2	Play	171
7.44.2.3	Stop	171
7.44.3	Property Documentation	171
7.44.3.1	File	171
7.44.3.2	Name	171
7.45	TriDevs.TriEngine.Point< T > Struct Template Reference	172
7.45.1	Detailed Description	172
7.45.2	Constructor & Destructor Documentation	172
7.45.2.1	Point	172
7.45.3	Member Data Documentation	172
7.45.3.1	X	172
7.45.3.2	Y	173
7.45.3.3	Z	173
7.46	TriDevs.TriEngine.Graphics.Primitive Class Reference	173
7.46.1	Detailed Description	174
7.46.2	Constructor & Destructor Documentation	174
7.46.2.1	Primitive	174
7.46.2.2	Primitive	174
7.46.3	Member Function Documentation	175
7.46.3.1	Dispose	175
7.46.3.2	Draw	175
7.46.4	Member Data Documentation	175
7.46.4.1	ColorId	175
7.46.4.2	Colors	175
7.46.4.3	Ids	175
7.46.4.4	Indices	176
7.46.4.5	Vertices	176
7.46.5	Property Documentation	176
7.46.5.1	ColorID	176
7.46.5.2	IndicesID	176
7.46.5.3	VerticesID	176
7.47	TriDevs.TriEngine.Program Class Reference	176

7.47.1 Detailed Description . . . . .	177
7.47.2 Constructor & Destructor Documentation . . . . .	177
7.47.2.1 Program . . . . .	177
7.47.3 Member Function Documentation . . . . .	177
7.47.3.1 Dispose . . . . .	178
7.47.4 Member Data Documentation . . . . .	178
7.47.4.1 ID . . . . .	178
7.48 TriDevs.TriEngine.Rectangle Struct Reference . . . . .	178
7.48.1 Detailed Description . . . . .	179
7.48.2 Constructor & Destructor Documentation . . . . .	179
7.48.2.1 Rectangle . . . . .	179
7.48.2.2 Rectangle . . . . .	179
7.48.3 Member Function Documentation . . . . .	180
7.48.3.1 Equals . . . . .	180
7.48.3.2 Intersects . . . . .	180
7.48.4 Member Data Documentation . . . . .	180
7.48.4.1 Color . . . . .	180
7.48.4.2 Height . . . . .	180
7.48.4.3 Width . . . . .	180
7.48.4.4 X . . . . .	181
7.48.4.5 Y . . . . .	181
7.49 TriDevs.TriEngine.Graphics.Rectangle Class Reference . . . . .	181
7.49.1 Detailed Description . . . . .	182
7.49.2 Constructor & Destructor Documentation . . . . .	182
7.49.2.1 Rectangle . . . . .	182
7.50 TriDevs.TriEngine.Resources.ResourceException Class Reference . . . . .	183
7.50.1 Detailed Description . . . . .	183
7.51 TriDevs.TriEngine.Resources.ResourceManager Class Reference . . . . .	184
7.51.1 Detailed Description . . . . .	184
7.51.2 Member Function Documentation . . . . .	185
7.51.2.1 Add . . . . .	185
7.51.2.2 Get . . . . .	185
7.51.2.3 Get< T > . . . . .	185
7.51.2.4 GetAll< T > . . . . .	186
7.51.2.5 Has . . . . .	186
7.51.2.6 Has< T > . . . . .	187
7.51.2.7 LoadFont . . . . .	187
7.51.2.8 LoadFont . . . . .	188
7.51.2.9 LoadShader . . . . .	188
7.51.2.10 LoadSong . . . . .	188

7.51.2.11	LoadSound	189
7.51.3	Member Data Documentation	189
7.51.3.1	BasePath	189
7.51.3.2	FontPath	189
7.51.3.3	ShaderPath	189
7.51.3.4	SongPath	189
7.51.3.5	SoundPath	190
7.52	TriDevs.TriEngine.Serializing.Serializer Class Reference	190
7.52.1	Detailed Description	190
7.52.2	Member Function Documentation	190
7.52.2.1	Deserialize< T >	190
7.52.2.2	Serialize< T >	191
7.52.2.3	Serialize< T >	191
7.53	TriDevs.TriEngine.Services Class Reference	192
7.53.1	Detailed Description	192
7.53.2	Member Function Documentation	192
7.53.2.1	Provide	192
7.53.2.2	Provide	192
7.53.2.3	Provide	193
7.53.3	Property Documentation	193
7.53.3.1	Audio	193
7.53.3.2	Input	193
7.54	TriDevs.TriEngine.Shaders.Shader Class Reference	193
7.54.1	Detailed Description	194
7.54.2	Constructor & Destructor Documentation	195
7.54.2.1	Shader	195
7.54.3	Member Function Documentation	195
7.54.3.1	Dispose	195
7.54.3.2	GetDefaultName	195
7.54.4	Member Data Documentation	195
7.54.4.1	ID	195
7.54.5	Property Documentation	195
7.54.5.1	File	195
7.54.5.2	Name	196
7.55	TriDevs.TriEngine.Audio.Song Class Reference	196
7.55.1	Detailed Description	197
7.55.2	Member Function Documentation	198
7.55.2.1	Dispose	198
7.55.2.2	Pause	198
7.55.2.3	Play	198

7.55.2.4	Resume	198
7.55.2.5	Stop	198
7.55.3	Property Documentation	199
7.55.3.1	File	199
7.55.3.2	IsLooped	199
7.55.3.3	Name	199
7.55.3.4	Volume	199
7.56	TriDevs.TriEngine.Audio.Sound Class Reference	199
7.56.1	Detailed Description	200
7.56.2	Member Function Documentation	200
7.56.2.1	Dispose	200
7.56.2.2	Play	201
7.56.2.3	Stop	201
7.56.3	Property Documentation	201
7.56.3.1	File	201
7.56.3.2	Name	201
7.57	TriDevs.TriEngine.Extensions.StringExtensions Class Reference	201
7.57.1	Detailed Description	201
7.57.2	Member Function Documentation	202
7.57.2.1	Replace	202
7.57.2.2	Replace	202
7.57.2.3	ReplaceFirst	202
7.58	TriDevs.TriEngine.Text.TextObject Class Reference	203
7.58.1	Detailed Description	204
7.58.2	Constructor & Destructor Documentation	204
7.58.2.1	TextObject	204
7.58.2.2	TextObject	204
7.58.3	Member Function Documentation	205
7.58.3.1	Draw	205
7.58.3.2	Draw	205
7.58.3.3	Draw	205
7.58.4	Property Documentation	205
7.58.4.1	Alignment	205
7.58.4.2	Bounds	205
7.58.4.3	Color	205
7.58.4.4	Font	205
7.58.4.5	Position	205
7.58.4.6	Text	205
7.59	TriDevs.TriEngine.Helpers.Threading Class Reference	206
7.59.1	Detailed Description	206

7.59.2	Member Function Documentation	206
7.59.2.1	SetCurrentThreadName	206
7.60	TriDevs.TriEngine.Graphics.Triangle Class Reference	206
7.60.1	Detailed Description	207
7.60.2	Constructor & Destructor Documentation	208
7.60.2.1	Triangle	208
7.61	TriDevs.TriEngine.Extensions.Vector3Extensions Class Reference	208
7.61.1	Detailed Description	208
7.61.2	Member Function Documentation	208
7.61.2.1	ToFloatArray	208
7.61.2.2	ToFloatArray	208
7.62	TriDevs.TriEngine.Extensions.Vector4Extensions Class Reference	209
7.62.1	Detailed Description	209
7.62.2	Member Function Documentation	209
7.62.2.1	ToFloatArray	209
7.62.2.2	ToFloatArray	210
7.63	TriDevs.TriEngine.Version Class Reference	210
7.63.1	Detailed Description	211
7.63.2	Member Data Documentation	211
7.63.2.1	Major	211
7.63.2.2	Minor	211
7.63.2.3	Patch	211
7.63.2.4	Suffix	211
7.63.2.5	VersionStringFormat	211
7.63.2.6	VersionStringFormatWithSuffix	211
7.63.3	Property Documentation	211
7.63.3.1	VersionString	211
<b>8</b>	<b>File Documentation</b>	<b>213</b>
8.1	README.md File Reference	213
8.2	TriDevs.TriEngine/Audio/AudioFormat.cs File Reference	213
8.3	TriDevs.TriEngine/Audio/AudioManager.cs File Reference	213
8.4	TriDevs.TriEngine/Audio/IAudioManager.cs File Reference	213
8.5	TriDevs.TriEngine/Audio/ISong.cs File Reference	214
8.6	TriDevs.TriEngine/Audio/ISound.cs File Reference	214
8.7	TriDevs.TriEngine/Audio/NullAudioManager.cs File Reference	214
8.8	TriDevs.TriEngine/Audio/NullSong.cs File Reference	214
8.9	TriDevs.TriEngine/Audio/NullSound.cs File Reference	215
8.10	TriDevs.TriEngine/Audio/Song.cs File Reference	215
8.11	TriDevs.TriEngine/Audio/Sound.cs File Reference	215



8.12	<a href="#">TriDevs.TriEngine/Color.cs File Reference</a>	215
8.13	<a href="#">TriDevs.TriEngine/EngineException.cs File Reference</a>	216
8.14	<a href="#">TriDevs.TriEngine/Extensions/ByteExtensions.cs File Reference</a>	216
8.15	<a href="#">TriDevs.TriEngine/Extensions/DecimalExtensions.cs File Reference</a>	216
8.16	<a href="#">TriDevs.TriEngine/Extensions/DoubleExtensions.cs File Reference</a>	216
8.17	<a href="#">TriDevs.TriEngine/Extensions/EnumerationExtensions.cs File Reference</a>	217
8.18	<a href="#">TriDevs.TriEngine/Extensions/FloatExtensions.cs File Reference</a>	217
8.19	<a href="#">TriDevs.TriEngine/Extensions/IntegerExtensions.cs File Reference</a>	217
8.20	<a href="#">TriDevs.TriEngine/Extensions/StringExtensions.cs File Reference</a>	217
8.21	<a href="#">TriDevs.TriEngine/Extensions/Vector3Extensions.cs File Reference</a>	218
8.22	<a href="#">TriDevs.TriEngine/Extensions/Vector4Extensions.cs File Reference</a>	218
8.23	<a href="#">TriDevs.TriEngine/GameWindow2D.cs File Reference</a>	218
8.24	<a href="#">TriDevs.TriEngine/Graphics/Primitive.cs File Reference</a>	218
8.25	<a href="#">TriDevs.TriEngine/Graphics/Rectangle.cs File Reference</a>	219
8.26	<a href="#">TriDevs.TriEngine/Rectangle.cs File Reference</a>	219
8.27	<a href="#">TriDevs.TriEngine/Graphics/Triangle.cs File Reference</a>	219
8.28	<a href="#">TriDevs.TriEngine/Helpers/Exceptions.cs File Reference</a>	219
8.29	<a href="#">TriDevs.TriEngine/Helpers/IO.cs File Reference</a>	220
8.30	<a href="#">TriDevs.TriEngine/Helpers/Math.cs File Reference</a>	220
8.31	<a href="#">TriDevs.TriEngine/Helpers/Threading.cs File Reference</a>	220
8.32	<a href="#">TriDevs.TriEngine/Input/Events/Key.cs File Reference</a>	220
8.33	<a href="#">TriDevs.TriEngine/Input/Events/Mouse.cs File Reference</a>	221
8.34	<a href="#">TriDevs.TriEngine/Input/IInputManager.cs File Reference</a>	221
8.35	<a href="#">TriDevs.TriEngine/Input/InputManager.cs File Reference</a>	221
8.36	<a href="#">TriDevs.TriEngine/Input/NullInputManager.cs File Reference</a>	222
8.37	<a href="#">TriDevs.TriEngine/Interfaces/IDrawable.cs File Reference</a>	222
8.38	<a href="#">TriDevs.TriEngine/Interfaces/IDrawableGameComponent.cs File Reference</a>	222
8.39	<a href="#">TriDevs.TriEngine/Interfaces/IGameComponent.cs File Reference</a>	222
8.40	<a href="#">TriDevs.TriEngine/Interfaces/IUpdatable.cs File Reference</a>	223
8.41	<a href="#">TriDevs.TriEngine/Logging/LogManager.cs File Reference</a>	223
8.42	<a href="#">TriDevs.TriEngine/Native/Helpers.cs File Reference</a>	223
8.43	<a href="#">TriDevs.TriEngine/Native/WinAPI.cs File Reference</a>	223
8.44	<a href="#">TriDevs.TriEngine/Point.cs File Reference</a>	223
8.45	<a href="#">TriDevs.TriEngine/Program.cs File Reference</a>	224
8.46	<a href="#">TriDevs.TriEngine/Properties/AssemblyInfo.cs File Reference</a>	224
8.47	<a href="#">TriDevs.TriEngine/Resources/IResource.cs File Reference</a>	224
8.48	<a href="#">TriDevs.TriEngine/Resources/ResourceException.cs File Reference</a>	224
8.49	<a href="#">TriDevs.TriEngine/Resources/ResourceManager.cs File Reference</a>	224
8.50	<a href="#">TriDevs.TriEngine/Serializing/Serializer.cs File Reference</a>	225
8.51	<a href="#">TriDevs.TriEngine/Services.cs File Reference</a>	225

8.52	<a href="#">TriDevs.TriEngine/Shaders/Shader.cs File Reference</a>	225
8.53	<a href="#">TriDevs.TriEngine/StateManagement/GameState.cs File Reference</a>	225
8.54	<a href="#">TriDevs.TriEngine/StateManagement/GameStateManager.cs File Reference</a>	226
8.55	<a href="#">TriDevs.TriEngine/StateManagement/IGameState.cs File Reference</a>	226
8.56	<a href="#">TriDevs.TriEngine/StateManagement/IGameStateManager.cs File Reference</a>	226
8.57	<a href="#">TriDevs.TriEngine/Text/Font.cs File Reference</a>	226
8.58	<a href="#">TriDevs.TriEngine/Text/FontConstructionConfig.cs File Reference</a>	227
8.59	<a href="#">TriDevs.TriEngine/Text/FontType.cs File Reference</a>	227
8.60	<a href="#">TriDevs.TriEngine/Text/ITextObject.cs File Reference</a>	227
8.61	<a href="#">TriDevs.TriEngine/Text/TextObject.cs File Reference</a>	227
8.62	<a href="#">TriDevs.TriEngine/UI/Control.cs File Reference</a>	228
8.63	<a href="#">TriDevs.TriEngine/UI/ControlManager.cs File Reference</a>	228
8.64	<a href="#">TriDevs.TriEngine/UI/Events/ControlClicked.cs File Reference</a>	228
8.65	<a href="#">TriDevs.TriEngine/UI/IControl.cs File Reference</a>	228
8.66	<a href="#">TriDevs.TriEngine/UI/IControlManager.cs File Reference</a>	229
8.67	<a href="#">TriDevs.TriEngine/UI/Label.cs File Reference</a>	229
8.68	<a href="#">TriDevs.TriEngine/UI/LinkLabel.cs File Reference</a>	229
8.69	<a href="#">TriDevs.TriEngine/Version.cs File Reference</a>	229

## Index

229

# Chapter 1

## Main Page

General-purpose engine in C#/OpenGL

Will eventually have scripting API for Lua (and possible additional languages)

### 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](#).

TriEngine 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: "<strong>Namespace</strong> &gt;.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.

### Platform-specific developing

The solution/project files have several different configurations defined for debugging on different platforms (Linux, Windows and OSX).

Each configuration has a compile constant defined:

- "Debug (Linux)" and "Release (Linux)" have the **LINUX** constant #defined
- "Debug (Windows)" and "Release (Windows)" have the **WINDOWS** constant #defined
- "Debug (OSX)" and "Release (OSX)" have the **OSX** constant #defined

Platform-specific code must be enclosed in `#if {PLATFORM} /* Code goes here */ #endif`

Example:

```
“c# #if WINDOWS DoWindowsStuff(); #elif LINUX DoLinuxStuff(); #elif OSX DoOSXStuff(); #else // Unsupported platform? #endif
```

```
#if WINDOWS DoExclusiveWindowsStuff(); #else AllOtherPlatforms(); #endif “
```

### Dependencies

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

TriEngine 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.

TriEngine 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.

TriEngine depends on [NVorbis](#), which is included in the **libs/NVorbis** folder. [NVorbis is licensed under Ms-PL](#).

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:

<a href="#">TriDevs</a>	13
<a href="#">TriDevs.TriEngine</a>	13
<a href="#">TriDevs.TriEngine.Audio</a>	14
<a href="#">TriDevs.TriEngine.Extensions</a>	15
<a href="#">TriDevs.TriEngine.Graphics</a>	15
<a href="#">TriDevs.TriEngine.Helpers</a>	15
<a href="#">TriDevs.TriEngine.Input</a>	16
<a href="#">TriDevs.TriEngine.Input.Events</a>	16
<a href="#">TriDevs.TriEngine.Interfaces</a>	18
<a href="#">TriDevs.TriEngine.Logging</a>	18
<a href="#">TriDevs.TriEngine.Native</a>	18
<a href="#">TriDevs.TriEngine.Resources</a>	19
<a href="#">TriDevs.TriEngine.Serializing</a>	19
<a href="#">TriDevs.TriEngine.Shaders</a>	19
<a href="#">TriDevs.TriEngine.StateManagement</a>	19
<a href="#">TriDevs.TriEngine.Text</a>	19
<a href="#">TriDevs.TriEngine.UI</a>	20
<a href="#">TriDevs.TriEngine.UI.Events</a>	21



## Chapter 3

# Hierarchical Index

### 3.1 Class Hierarchy

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

TriDevs.TriEngine.Extensions.ByteExtensions . . . . .	25
TriDevs.TriEngine.Color . . . . .	27
TriDevs.TriEngine.Extensions.DecimalExtensions . . . . .	62
TriDevs.TriEngine.Extensions.DoubleExtensions . . . . .	63
TriDevs.TriEngine.Extensions.EnumerationExtensions . . . . .	65
EventArgs	
TriDevs.TriEngine.Input.Events.KeyCharEventArgs . . . . .	141
TriDevs.TriEngine.Input.Events.KeyEventArgs . . . . .	142
Exception	
TriDevs.TriEngine.EngineException . . . . .	64
TriDevs.TriEngine.Resources.ResourceException . . . . .	183
TriDevs.TriEngine.Helpers.Exceptions . . . . .	68
TriDevs.TriEngine.Extensions.FloatExtensions . . . . .	69
TriDevs.TriEngine.Text.FontConstructionConfig . . . . .	75
GameWindow	
TriDevs.TriEngine.GameWindow2D . . . . .	89
TriDevs.TriEngine.Native.Helpers . . . . .	91
IDisposable	
TriDevs.TriEngine.Audio.IAudioManager . . . . .	93
TriDevs.TriEngine.Audio.AudioManager . . . . .	23
TriDevs.TriEngine.Audio.NullAudioManager . . . . .	157
TriDevs.TriEngine.Graphics.Primitive . . . . .	173
TriDevs.TriEngine.Graphics.Rectangle . . . . .	181
TriDevs.TriEngine.Graphics.Triangle . . . . .	206
TriDevs.TriEngine.Program . . . . .	176
TriDevs.TriEngine.Resources.IResource . . . . .	133
TriDevs.TriEngine.Audio.ISong . . . . .	134
TriDevs.TriEngine.Audio.NullSong . . . . .	166
TriDevs.TriEngine.Audio.Song . . . . .	196
TriDevs.TriEngine.Audio.ISound . . . . .	136
TriDevs.TriEngine.Audio.NullSound . . . . .	169
TriDevs.TriEngine.Audio.Sound . . . . .	199
TriDevs.TriEngine.Shaders.Shader . . . . .	193
TriDevs.TriEngine.Text.Font . . . . .	70
TriDevs.TriEngine.Interfaces.IDrawable . . . . .	102
TriDevs.TriEngine.Graphics.Primitive . . . . .	173
TriDevs.TriEngine.Interfaces.IDrawableGameComponent . . . . .	103

TriDevs.TriEngine.UI.IControlManager . . . . .	98
TriDevs.TriEngine.UI.ControlManager . . . . .	56
TriDevs.TriEngine.StateManagement.IGameState . . . . .	107
TriDevs.TriEngine.StateManagement.GameState . . . . .	76
TriDevs.TriEngine.StateManagement.IGameStateManager . . . . .	113
TriDevs.TriEngine.StateManagement.GameStateManager . . . . .	85
TriDevs.TriEngine.UI.IControl . . . . .	94
TriDevs.TriEngine.UI.Control . . . . .	51
TriDevs.TriEngine.UI.Label . . . . .	144
TriDevs.TriEngine.UI.LinkLabel . . . . .	147
IEquatable< Rectangle > . . . . .	
TriDevs.TriEngine.Rectangle . . . . .	178
TriDevs.TriEngine.Input.IInputManager . . . . .	115
TriDevs.TriEngine.Input.InputManager . . . . .	122
TriDevs.TriEngine.Input.NullInputManager . . . . .	159
TriDevs.TriEngine.Extensions.IntegerExtensions . . . . .	129
TriDevs.TriEngine.Helpers.IO . . . . .	132
TriDevs.TriEngine.Text.ITextObject . . . . .	138
TriDevs.TriEngine.Text.TextObject . . . . .	203
TriDevs.TriEngine.Interfaces.IUpdatable . . . . .	140
TriDevs.TriEngine.Interfaces.IGameComponent . . . . .	105
TriDevs.TriEngine.Interfaces.IDrawableGameComponent . . . . .	103
TriDevs.TriEngine.StateManagement.IGameState . . . . .	107
TriDevs.TriEngine.StateManagement.IGameStateManager . . . . .	113
TriDevs.TriEngine.UI.IControl . . . . .	94
TriDevs.TriEngine.Logging.LogManager . . . . .	150
TriDevs.TriEngine.Helpers.Math . . . . .	153
TriDevs.TriEngine.Point< T > . . . . .	172
TriDevs.TriEngine.Resources.ResourceManager . . . . .	184
TriDevs.TriEngine.Serializing.Serializer . . . . .	190
TriDevs.TriEngine.Services . . . . .	192
TriDevs.TriEngine.Extensions.StringExtensions . . . . .	201
TriDevs.TriEngine.Helpers.Threading . . . . .	206
TriDevs.TriEngine.Extensions.Vector3Extensions . . . . .	208
TriDevs.TriEngine.Extensions.Vector4Extensions . . . . .	209
TriDevs.TriEngine.Version . . . . .	210



## Chapter 4

# Class Index

### 4.1 Class List

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

<a href="#">TriDevs.TriEngine.Audio.AudioManager</a>	23
Class to manage engine audio. . . . .	
<a href="#">TriDevs.TriEngine.Extensions.ByteExtensions</a>	25
Extensions for System.Byte. . . . .	
<a href="#">TriDevs.TriEngine.Color</a>	27
Represents an RGBA color that can be used with TriEngine2D. . . . .	
<a href="#">TriDevs.TriEngine.UI.Control</a>	51
Base control class that all other controls inherits from. Defines basic UI control behaviour. . . .	
<a href="#">TriDevs.TriEngine.UI.ControlManager</a>	56
Control manager to manage various UI controls for a game. . . . .	
<a href="#">TriDevs.TriEngine.Extensions.DecimalExtensions</a>	62
Extensions for System.Decimal. . . . .	
<a href="#">TriDevs.TriEngine.Extensions.DoubleExtensions</a>	63
Extensions for System.Double. . . . .	
<a href="#">TriDevs.TriEngine.EngineException</a>	64
Base exception class for all engine-related exceptions. The inner exception will contain more info as to what actually happened. . . . .	
<a href="#">TriDevs.TriEngine.Extensions.EnumerationExtensions</a>	65
Extensions for System.Enum. . . . .	
<a href="#">TriDevs.TriEngine.Helpers.Exceptions</a>	68
Provides helper methods for dealing with exceptions. . . . .	
<a href="#">TriDevs.TriEngine.Extensions.FloatExtensions</a>	69
Extensions for System.Single. . . . .	
<a href="#">TriDevs.TriEngine.Text.Font</a>	70
Holds a specific font type. . . . .	
<a href="#">TriDevs.TriEngine.Text.FontConstructionConfig</a>	75
Container class for different QFont configurations for use with the Font constructor. . . . .	
<a href="#">TriDevs.TriEngine.StateManagement.GameState</a>	76
Base GameState class that all other game states derive from, defines basic GameState behaviour. . . . .	
<a href="#">TriDevs.TriEngine.StateManagement.GameStateManager</a>	85
Game state manager that keeps track of the active game states and provides methods to control the states. . . . .	
<a href="#">TriDevs.TriEngine.GameWindow2D</a>	89
Game window class specialized for drawing 2D graphics. . . . .	
<a href="#">TriDevs.TriEngine.Native.Helpers</a>	91
Helper class with various methods to help native coding and debugging. . . . .	

<a href="#">TriDevs.TriEngine.Audio.IAudioManager</a>	Provides various methods to manipulate audio. . . . .	93
<a href="#">TriDevs.TriEngine.UI.IControl</a>	A UI control that can be drawn on screen and interacted with. . . . .	94
<a href="#">TriDevs.TriEngine.UI.IControlManager</a>	Manages various UI controls, automatically updating and drawing them to the screen. . . . .	98
<a href="#">TriDevs.TriEngine.Interfaces.IDrawable</a>	Implements a simple draw method. . . . .	102
<a href="#">TriDevs.TriEngine.Interfaces.IDrawableGameComponent</a>	A game component that can be added to GameState objects. Drawable game components also implement a draw method to draw themselves to screen. . . . .	103
<a href="#">TriDevs.TriEngine.Interfaces.IGameComponent</a>	A game component that can be added to IGameState objects. . . . .	105
<a href="#">TriDevs.TriEngine.StateManagement.IGameState</a>	A game state that can be used with the game state manager. Represent a specific state of the game, like main menu and options screen. . . . .	107
<a href="#">TriDevs.TriEngine.StateManagement.IGameStateManager</a>	Game state manager that keeps track of the active game states and provides methods to control the states. . . . .	113
<a href="#">TriDevs.TriEngine.Input.IInputManager</a>	Provides various methods to query input devices like the keyboard. . . . .	115
<a href="#">TriDevs.TriEngine.Input.InputManager</a>	Input manager interfacing with input methods provided by a GameWindow. . . . .	122
<a href="#">TriDevs.TriEngine.Extensions.IntegerExtensions</a>	Extensions for System.Int16, System.UInt16, System.Int32, System.UInt32, System.Int64 and System.UInt64. . . . .	129
<a href="#">TriDevs.TriEngine.Helpers.IO</a>	Provides various helper functions for doing IO operations. . . . .	132
<a href="#">TriDevs.TriEngine.Resources.IResource</a>	An external resource created from a file. . . . .	133
<a href="#">TriDevs.TriEngine.Audio.ISong</a>	A song that will be streamed in the audio player. . . . .	134
<a href="#">TriDevs.TriEngine.Audio.ISound</a>	A sound file for use with the audio manager. . . . .	136
<a href="#">TriDevs.TriEngine.Text.ITextObject</a>	Implements methods to construct a text object and render it to screen. . . . .	138
<a href="#">TriDevs.TriEngine.Interfaces.IUpdatable</a>	Implements a simple update method. . . . .	140
<a href="#">TriDevs.TriEngine.Input.Events.KeyCharEventArgs</a>	EventArgs class used for keychar-related events. Contains information about the character related with the event. . . . .	141
<a href="#">TriDevs.TriEngine.Input.Events.KeyEventArgs</a>	EventArgs class used for key-related events. Contains information about the key related with the event. . . . .	142
<a href="#">TriDevs.TriEngine.UI.Label</a>	A simple label to display text on the screen. . . . .	144
<a href="#">TriDevs.TriEngine.UI.LinkLabel</a>	A label that, when clicked, will open a URL. . . . .	147
<a href="#">TriDevs.TriEngine.Logging.LogManager</a>	Class to manage logging. ILog interfaces should be obtained from this class' methods, as opposed to calling default log4net methods. . . . .	150
<a href="#">TriDevs.TriEngine.Helpers.Math</a>	Various helper methods for working with math. . . . .	153
<a href="#">TriDevs.TriEngine.Audio.NullAudioManager</a>	Used as a fallback AudioManager object when the service locator fails to find one. . . . .	157
<a href="#">TriDevs.TriEngine.Input.NullInputManager</a>	Used as a fallback InputManager object when the service locator fails to find one. . . . .	159

<a href="#">TriDevs.TriEngine.Audio.NullSong</a>	
Fallback song class used in <a href="#">NullAudioManager</a> .	166
<a href="#">TriDevs.TriEngine.Audio.NullSound</a>	
Fallback sound class used in <a href="#">NullAudioManager</a> .	169
<a href="#">TriDevs.TriEngine.Point&lt; T &gt;</a>	
A struct representing an X/Y/Z coordinate.	172
<a href="#">TriDevs.TriEngine.Graphics.Primitive</a>	
Represents a primitive 2D shape composed of triangles.	173
<a href="#">TriDevs.TriEngine.Program</a>	
An OpenGL program.	176
<a href="#">TriDevs.TriEngine.Rectangle</a>	
A rectangle representing an area in 2D space.	178
<a href="#">TriDevs.TriEngine.Graphics.Rectangle</a>	
A primitive 2D <a href="#">Rectangle</a> shape.	181
<a href="#">TriDevs.TriEngine.Resources.ResourceException</a>	
Exception for resource-related issues. Thrown when there is a problem in <a href="#">ResourceManager</a> .	183
<a href="#">TriDevs.TriEngine.Resources.ResourceManager</a>	
Static class to manage resources.	184
<a href="#">TriDevs.TriEngine.Serializing.Serializer</a>	
Provides serialization methods.	190
<a href="#">TriDevs.TriEngine.Services</a>	
Provides different game-related service interfaces.	192
<a href="#">TriDevs.TriEngine.Shaders.Shader</a>	
GLSL shader object loaded and compiled from a *.glsl shader file.	193
<a href="#">TriDevs.TriEngine.Audio.Song</a>	
Song class that can be used with <a href="#">AudioManager</a> .	196
<a href="#">TriDevs.TriEngine.Audio.Sound</a>	
Sound class that can be used with the <a href="#">AudioManager</a> .	199
<a href="#">TriDevs.TriEngine.Extensions.StringExtensions</a>	
Extensions for System.String	201
<a href="#">TriDevs.TriEngine.Text.TextObject</a>	
Implements the <a href="#">ITextObject</a> interface.	203
<a href="#">TriDevs.TriEngine.Helpers.Threading</a>	
Provides various helper functions for doing threading operations.	206
<a href="#">TriDevs.TriEngine.Graphics.Triangle</a>	
A primitive 2D <a href="#">Triangle</a> shape.	206
<a href="#">TriDevs.TriEngine.Extensions.Vector3Extensions</a>	
Extensions for Vector3.	208
<a href="#">TriDevs.TriEngine.Extensions.Vector4Extensions</a>	
Extensions for Vector4.	209
<a href="#">TriDevs.TriEngine.Version</a>	
Version class specifying the version of this project.	210



## Chapter 5

# File Index

### 5.1 File List

Here is a list of all files with brief descriptions:

TriDevs.TriEngine/Color.cs	215
TriDevs.TriEngine/EngineException.cs	216
TriDevs.TriEngine/GameWindow2D.cs	218
TriDevs.TriEngine/Point.cs	223
TriDevs.TriEngine/Program.cs	224
TriDevs.TriEngine/Rectangle.cs	219
TriDevs.TriEngine/Services.cs	225
TriDevs.TriEngine/Version.cs	229
TriDevs.TriEngine/Audio/AudioFormat.cs	213
TriDevs.TriEngine/Audio/AudioManager.cs	213
TriDevs.TriEngine/Audio/IAudioManager.cs	213
TriDevs.TriEngine/Audio/ISong.cs	214
TriDevs.TriEngine/Audio/ISound.cs	214
TriDevs.TriEngine/Audio/NullAudioManager.cs	214
TriDevs.TriEngine/Audio/NullSong.cs	214
TriDevs.TriEngine/Audio/NullSound.cs	215
TriDevs.TriEngine/Audio/Song.cs	215
TriDevs.TriEngine/Audio/Sound.cs	215
TriDevs.TriEngine/Extensions/ByteExtensions.cs	216
TriDevs.TriEngine/Extensions/DecimalExtensions.cs	216
TriDevs.TriEngine/Extensions/DoubleExtensions.cs	216
TriDevs.TriEngine/Extensions/EnumerationExtensions.cs	217
TriDevs.TriEngine/Extensions/FloatExtensions.cs	217
TriDevs.TriEngine/Extensions/IntegerExtensions.cs	217
TriDevs.TriEngine/Extensions/StringExtensions.cs	217
TriDevs.TriEngine/Extensions/Vector3Extensions.cs	218
TriDevs.TriEngine/Extensions/Vector4Extensions.cs	218
TriDevs.TriEngine/Graphics/Primitive.cs	218
TriDevs.TriEngine/Graphics/Rectangle.cs	219
TriDevs.TriEngine/Graphics/Triangle.cs	219
TriDevs.TriEngine/Helpers/Exceptions.cs	219
TriDevs.TriEngine/Helpers/IO.cs	220
TriDevs.TriEngine/Helpers/Math.cs	220
TriDevs.TriEngine/Helpers/Threading.cs	220
TriDevs.TriEngine/Input/IInputManager.cs	221
TriDevs.TriEngine/Input/InputManager.cs	221
TriDevs.TriEngine/Input/NullInputManager.cs	222
TriDevs.TriEngine/Input/Events/Key.cs	220

TriDevs.TriEngine/Input/Events/ <a href="#">Mouse.cs</a>	221
TriDevs.TriEngine/Interfaces/ <a href="#">IDrawable.cs</a>	222
TriDevs.TriEngine/Interfaces/ <a href="#">IDrawableGameComponent.cs</a>	222
TriDevs.TriEngine/Interfaces/ <a href="#">IGameComponent.cs</a>	222
TriDevs.TriEngine/Interfaces/ <a href="#">IUpdatable.cs</a>	223
TriDevs.TriEngine/Logging/ <a href="#">LogManager.cs</a>	223
TriDevs.TriEngine/Native/ <a href="#">Helpers.cs</a>	223
TriDevs.TriEngine/Native/ <a href="#">WinAPI.cs</a>	223
TriDevs.TriEngine/Properties/ <a href="#">AssemblyInfo.cs</a>	224
TriDevs.TriEngine/Resources/ <a href="#">IResource.cs</a>	224
TriDevs.TriEngine/Resources/ <a href="#">ResourceException.cs</a>	224
TriDevs.TriEngine/Resources/ <a href="#">ResourceManager.cs</a>	224
TriDevs.TriEngine/Serializing/ <a href="#">Serializer.cs</a>	225
TriDevs.TriEngine/Shaders/ <a href="#">Shader.cs</a>	225
TriDevs.TriEngine/StateManagement/ <a href="#">GameState.cs</a>	225
TriDevs.TriEngine/StateManagement/ <a href="#">GameStateManager.cs</a>	226
TriDevs.TriEngine/StateManagement/ <a href="#">IGameState.cs</a>	226
TriDevs.TriEngine/StateManagement/ <a href="#">IGameStateManager.cs</a>	226
TriDevs.TriEngine/Text/ <a href="#">Font.cs</a>	226
TriDevs.TriEngine/Text/ <a href="#">FontConstructionConfig.cs</a>	227
TriDevs.TriEngine/Text/ <a href="#">FontType.cs</a>	227
TriDevs.TriEngine/Text/ <a href="#">ITextObject.cs</a>	227
TriDevs.TriEngine/Text/ <a href="#">TextObject.cs</a>	227
TriDevs.TriEngine/UI/ <a href="#">Control.cs</a>	228
TriDevs.TriEngine/UI/ <a href="#">ControlManager.cs</a>	228
TriDevs.TriEngine/UI/ <a href="#">IControl.cs</a>	228
TriDevs.TriEngine/UI/ <a href="#">IControlManager.cs</a>	229
TriDevs.TriEngine/UI/ <a href="#">Label.cs</a>	229
TriDevs.TriEngine/UI/ <a href="#">LinkLabel.cs</a>	229
TriDevs.TriEngine/UI/Events/ <a href="#">ControlClicked.cs</a>	228

## Chapter 6

# Namespace Documentation

### 6.1 Package TriDevs

#### Namespaces

- package [TriEngine](#)

### 6.2 Package TriDevs.TriEngine

#### Namespaces

- package [Audio](#)
- package [Extensions](#)
- package [Graphics](#)
- package [Helpers](#)
- package [Input](#)
- package [Interfaces](#)
- package [Logging](#)
- package [Native](#)
- package [Resources](#)
- package [Serializing](#)
- package [Shaders](#)
- package [StateManagement](#)
- package [Text](#)
- package [UI](#)

#### Classes

- struct [Color](#)  
*Represents an RGBA color that can be used with TriEngine2D.*
- class [EngineException](#)  
*Base exception class for all engine-related exceptions. The inner exception will contain more info as to what actually happened.*
- class [GameWindow2D](#)  
*Game window class specialized for drawing 2D graphics.*
- struct [Point< T >](#)  
*A struct representing an X/Y/Z coordinate.*

- class [Program](#)  
*An OpenGL program.*
- struct [Rectangle](#)  
*A rectangle representing an area in 2D space.*
- class [Services](#)  
*Provides different game-related service interfaces.*
- class [Version](#)  
*[Version](#) class specifying the version of this project.*

## 6.3 Package TriDevs.TriEngine.Audio

### Classes

- class [AudioManager](#)  
*Class to manage engine audio.*
- interface [IAudioManager](#)  
*Provides various methods to manipulate audio.*
- interface [ISong](#)  
*A song that will be streamed in the audio player.*
- interface [ISound](#)  
*A sound file for use with the audio manager.*
- class [NullAudioManager](#)  
*Used as a fallback [AudioManager](#) object when the service locator fails to find one.*
- class [NullSong](#)  
*Fallback song class used in [NullAudioManager](#).*
- class [NullSound](#)  
*Fallback sound class used in [NullAudioManager](#).*
- class [Song](#)  
*[Song](#) class that can be used with [AudioManager](#).*
- class [Sound](#)  
*[Sound](#) class that can be used with the [AudioManager](#).*

### Enumerations

- enum [AudioFormat](#) { [Ogg](#), [Wav](#) }  
*Different audio formats usable with audio managers.*

#### 6.3.1 Enumeration Type Documentation

##### 6.3.1.1 enum TriDevs.TriEngine.Audio.AudioFormat

Different audio formats usable with audio managers.

#### Enumerator

**Ogg** The Ogg Vorbis format.

**Wav** The WAVE format

```

30     {
34         Ogg,
35
39         Wav
40     }
```



## 6.4 Package TriDevs.TriEngine.Extensions

### Classes

- class [ByteExtensions](#)  
*Extensions for System.Byte.*
- class [DecimalExtensions](#)  
*Extensions for System.Decimal.*
- class [DoubleExtensions](#)  
*Extensions for System.Double.*
- class [EnumerationExtensions](#)  
*Extensions for System.Enum.*
- class [FloatExtensions](#)  
*Extensions for System.Single.*
- class [IntegerExtensions](#)  
*Extensions for System.Int16, System.UInt16, System.Int32, System.UInt32, System.Int64 and System.UInt64.*
- class [StringExtensions](#)  
*Extensions for System.String*
- class [Vector3Extensions](#)  
*Extensions for Vector3.*
- class [Vector4Extensions](#)  
*Extensions for Vector4.*

## 6.5 Package TriDevs.TriEngine.Graphics

### Classes

- class [Primitive](#)  
*Represents a primitive 2D shape composed of triangles.*
- class [Rectangle](#)  
*A primitive 2D [Rectangle](#) shape.*
- class [Triangle](#)  
*A primitive 2D [Triangle](#) shape.*

## 6.6 Package TriDevs.TriEngine.Helpers

### Classes

- class [Exceptions](#)  
*Provides helper methods for dealing with exceptions.*
- class [IO](#)  
*Provides various helper functions for doing [IO](#) operations.*
- class [Math](#)  
*Various helper methods for working with math.*
- class [Threading](#)  
*Provides various helper functions for doing threading operations.*

## 6.7 Package TriDevs.TriEngine.Input

### Namespaces

- package [Events](#)

### 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.*

## 6.8 Package TriDevs.TriEngine.Input.Events

### Classes

- class [KeyEventArgs](#)  
*EventArgs class used for key-related events. Contains information about the key related with the event.*
- class [KeyCharEventArgs](#)  
*EventArgs class used for keychar-related events. Contains information about the character related with the event.*

### Functions

- delegate void [KeyDownEventHandler](#) (object sender, [KeyEventArgs](#) e)  
*Event handler delegate for the KeyDown event.*
- delegate void [KeyUpEventHandler](#) (object sender, [KeyEventArgs](#) e)  
*Event handler delegate for the KeyUp event.*
- delegate void [KeyPressEventHandler](#) (object sender, [KeyCharEventArgs](#) e)  
*Event handler delegate for the KeyPress event.*
- delegate void [MouseDownEventHandler](#) (object sender, [MouseButtonEventArgs](#) e)  
*Event handler delegate for the MouseDown event.*
- delegate void [MouseUpEventHandler](#) (object sender, [MouseButtonEventArgs](#) e)  
*Event handler delegate for the MouseUp event.*
- delegate void [MouseWheelChangedEventHandler](#) (object sender, [MouseWheelEventArgs](#) e)  
*Event handler delegate for the MouseWheelChanged event.*
- delegate void [MouseWheelDownEventHandler](#) (object sender, [MouseWheelEventArgs](#) e)  
*Event handler delegate for the MouseWheelDown event.*
- delegate void [MouseWheelUpEventHandler](#) (object sender, [MouseWheelEventArgs](#) e)  
*Event handler delegate for the MouseWheelUp event.*

### 6.8.1 Function Documentation

#### 6.8.1.1 delegate void TriDevs.TriEngine.Input.Events.KeyDownEventHandler ( object sender, KeyEventArgs e )

Event handler delegate for the KeyDown event.

## Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	<a href="#">KeyEventArgs</a> object with information about the event.

6.8.1.2 `delegate void TriDevs.TriEngine.Input.Events.KeyPressEventHandler ( object sender, KeyCharEventArgs e )`

Event handler delegate for the KeyPress event.

## Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	<a href="#">KeyEventArgs</a> object with information about the event.

6.8.1.3 `delegate void TriDevs.TriEngine.Input.Events.KeyUpEventHandler ( object sender, KeyEventArgs e )`

Event handler delegate for the KeyUp event.

## Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	<a href="#">KeyEventArgs</a> object with information about the event.

6.8.1.4 `delegate void TriDevs.TriEngine.Input.Events.MouseDownEventHandler ( object sender, MouseButtonEventArgs e )`

Event handler delegate for the MouseDown event.

## Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	Mouse button information associated with the event.

6.8.1.5 `delegate void TriDevs.TriEngine.Input.Events.MouseUpEventHandler ( object sender, MouseButtonEventArgs e )`

Event handler delegate for the MouseUp event.

## Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	Mouse button information associated with the event.

6.8.1.6 `delegate void TriDevs.TriEngine.Input.Events.MouseWheelChangedEventHandler ( object sender, MouseWheelEventArgs e )`

Event handler delegate for the MouseWheelChanged event.

## Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	Mouse wheel information associated with the event.

6.8.1.7 `delegate void TriDevs.TriEngine.Input.Events.MouseWheelDownEventHandler ( object sender, MouseEventArgs e )`

Event handler delegate for the MouseWheelDown event.

#### Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	Mouse wheel information associated with the event.

6.8.1.8 `delegate void TriDevs.TriEngine.Input.Events.MouseWheelUpEventHandler ( object sender, MouseEventArgs e )`

Event handler delegate for the MouseWheelUp event.

#### Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	Mouse wheel information associated with the event.

## 6.9 Package TriDevs.TriEngine.Interfaces

### Classes

- interface [IDrawable](#)  
*Implements a simple draw method.*
- interface [IDrawableGameComponent](#)  
*A game component that can be added to GameState objects. Drawable game components also implement a draw method to draw themselves to screen.*
- interface [IGameComponent](#)  
*A game component that can be added to IGameState objects.*
- interface [IUpdatable](#)  
*Implements a simple update method.*

## 6.10 Package TriDevs.TriEngine.Logging

### Classes

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

## 6.11 Package TriDevs.TriEngine.Native

### Classes

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

## 6.12 Package TriDevs.TriEngine.Resources

### Classes

- interface [IResource](#)  
*An external resource created from a file.*
- class [ResourceException](#)  
*Exception for resource-related issues. Thrown when there is a problem in [ResourceManager](#).*
- class [ResourceManager](#)  
*Static class to manage resources.*

## 6.13 Package TriDevs.TriEngine.Serializing

### Classes

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

## 6.14 Package TriDevs.TriEngine.Shaders

### Classes

- class [Shader](#)  
*GLSL shader object loaded and compiled from a \*.glsl shader file.*

## 6.15 Package TriDevs.TriEngine.StateManagement

### Classes

- class [GameState](#)  
*Base [GameState](#) class that all other game states derive from, defines basic [GameState](#) behaviour.*
- class [GameStateManager](#)  
*Game state manager that keeps track of the active game states and provides methods to control the states.*
- interface [IGameState](#)  
*A game state that can be used with the game state manager. Represent a specific state of the game, like main menu and options screen.*
- interface [IGameStateManager](#)  
*Game state manager that keeps track of the active game states and provides methods to control the states.*

## 6.16 Package TriDevs.TriEngine.Text

### Classes

- class [Font](#)  
*Holds a specific font type.*
- class [FontConstructionConfig](#)  
*Container class for different QFont configurations for use with the [Font](#) constructor.*
- interface [ITextObject](#)

*Implements methods to construct a text object and render it to screen.*

- class [TextObject](#)

*Implements the [ITextObject](#) interface.*

## Enumerations

- enum [FontType](#) { [TTF](#), [QFont](#), [Unsupported](#) }

*Different font types supported by TriEngine2D.*

### 6.16.1 Enumeration Type Documentation

#### 6.16.1.1 enum TriDevs.TriEngine.Text.FontType

Different font types supported by TriEngine2D.

#### Enumerator

***TTF*** TrueType font file

***QFont*** QFont \*.qfont files.

***Unsupported*** Specifies that the font type is not supported by the library.

```

30     {
34         TTF,
35
39         QFont,
40
44         Unsupported
45     }
```

## 6.17 Package TriDevs.TriEngine.UI

### Namespaces

- package [Events](#)

### Classes

- class [Control](#)

*Base control class that all other controls inherits from. Defines basic [UI](#) control behaviour.*

- class [ControlManager](#)

*[Control](#) manager to manage various [UI](#) controls for a game.*

- interface [IControl](#)

*A [UI](#) control that can be drawn on screen and interacted with.*

- interface [IControlManager](#)

*Manages various [UI](#) controls, automatically updating and drawing them to the screen.*

- class [Label](#)

*A simple label to display text on the screen.*

- class [LinkLabel](#)

*A label that, when clicked, will open a URL.*

## 6.18 Package TriDevs.TriEngine.UI.Events

### Functions

- delegate void [ControlClickedEventHandler](#) (object sender, EventArgs e)  
*Delegate handler for the control clicked event.*

### 6.18.1 Function Documentation

6.18.1.1 delegate void TriDevs.TriEngine.UI.Events.ControlClickedEventHandler ( object *sender*, EventArgs *e* )

Delegate handler for the control clicked event.

#### Parameters

<i>sender</i>	The object that raised the event.
<i>e</i>	





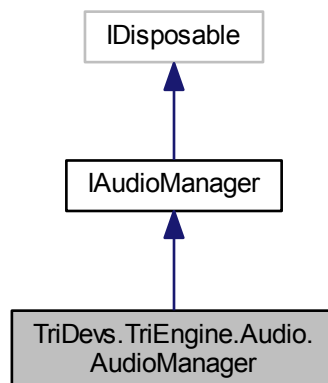
## Chapter 7

# Class Documentation

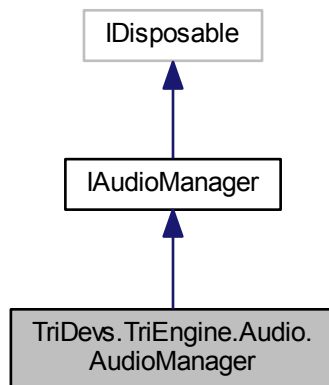
### 7.1 TriDevs.TriEngine.Audio.AudioManager Class Reference

Class to manage engine audio.

Inheritance diagram for TriDevs.TriEngine.Audio.AudioManager:



Collaboration diagram for TriDevs.TriEngine.Audio.AudioManager:



## Public Member Functions

- [AudioManager](#) ()  
*Creates a new instance of [AudioManager](#).*
- void [Dispose](#) ()
- void [StopAll](#) ()  
*Immediately stops all playback of sounds and songs.*
- void [StopAllSounds](#) ()  
*Immediately stops playback of all sounds.*
- void [StopAllSongs](#) ()  
*Immediately stops playback of all songs.*

### 7.1.1 Detailed Description

Class to manage engine audio.

### 7.1.2 Constructor & Destructor Documentation

#### 7.1.2.1 TriDevs.TriEngine.Audio.AudioManager.AudioManager ( )

Creates a new instance of [AudioManager](#).

```

42     {
43         _context = new AudioContext();
44         _oggStreamer = new OggStreamer();
45     }
  
```

### 7.1.3 Member Function Documentation

#### 7.1.3.1 void TriDevs.TriEngine.Audio.AudioManager.Dispose ( )

```

48     {
  
```

```

49         if (_oggStreamer != null)
50         {
51             _oggStreamer.Dispose();
52             _oggStreamer = null;
53         }
54
55         if (_context != null)
56         {
57             _context.Dispose();
58             _context = null;
59         }
60     }

```

### 7.1.3.2 void TriDevs.TriEngine.Audio.AudioManager.StopAll ( )

Immediately stops all playback of sounds and songs.

Implements [TriDevs.TriEngine.Audio.IAudioManager](#).

```

63     {
64         StopAllSounds();
65         StopAllSongs();
66     }

```

### 7.1.3.3 void TriDevs.TriEngine.Audio.AudioManager.StopAllSongs ( )

Immediately stops playback of all songs.

Implements [TriDevs.TriEngine.Audio.IAudioManager](#).

```

75     {
76         foreach (var song in ResourceManager.GetAll<ISong>())
77             song.Stop();
78     }

```

### 7.1.3.4 void TriDevs.TriEngine.Audio.AudioManager.StopAllSounds ( )

Immediately stops playback of all sounds.

Implements [TriDevs.TriEngine.Audio.IAudioManager](#).

```

69     {
70         foreach (var sound in ResourceManager.GetAll<ISound>())
71             sound.Stop();
72     }

```

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

- TriDevs.TriEngine/Audio/[AudioManager.cs](#)

## 7.2 TriDevs.TriEngine.Extensions.ByteExtensions Class Reference

[Extensions](#) for System.Byte.

### Static Public Member Functions

- static byte [Clamp](#) (this byte value, byte min, byte max)  
*Clamps the specified byte between a minimum and maximum value.*

### 7.2.1 Detailed Description

[Extensions](#) for System.Byte.

### 7.2.2 Member Function Documentation

**7.2.2.1** static byte TriDevs.TriEngine.Extensions.ByteExtensions.Clamp ( this byte *value*, byte *min*, byte *max* ) [static]

Clamps the specified byte between a minimum and maximum value.

#### Parameters

<i>value</i>	Value to clamp.
<i>min</i>	If the specified byte is below this value, then this will be returned.
<i>max</i>	If the specified byte is above this value, then this will be returned.

#### Returns

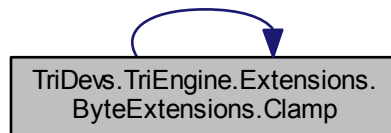
The clamped value of the byte.

```

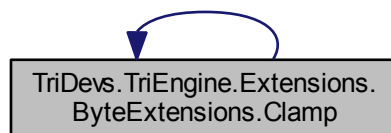
39         {
40             return Helpers.Math.Clamp(value, min, max);
41         }

```

Here is the call graph for this function:



Here is the caller graph for this function:



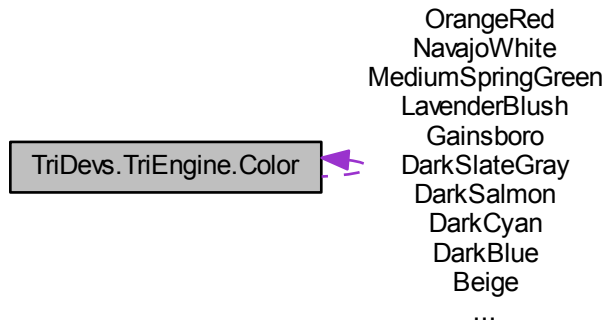
The documentation for this class was generated from the following file:

- TriDevs.TriEngine/Extensions/[ByteExtensions.cs](#)

## 7.3 TriDevs.TriEngine.Color Struct Reference

Represents an RGBA color that can be used with TriEngine2D.

Collaboration diagram for TriDevs.TriEngine.Color:



### Public Member Functions

- [Color](#) (Color4 color)  
*Creates a new color from a Color4 color.*
- [Color](#) (Vector3 vector)  
*Creates a new color from a Vector3 vector.*
- [Color](#) (Vector4 vector)  
*Creates a new color from a Vector4 vector.*
- [Color](#) (uint value)  
*Creates a new color from an ARGB value.*
- [Color](#) ([Color](#) @base, byte a)  
*Creates a new color from a base color with new alpha value.*
- [Color](#) ([Color](#) @base, float a)  
*Creates a new color from a base color with new alpha value.*
- [Color](#) (byte r, byte g, byte b, byte a=255)  
*Creates a new color with the specified red, green, blue and alpha values.*
- [Color](#) (float r, float g, float b, float a=1.0f)  
*Creates a new color with the specified red, green, blue and alpha values.*
- uint [ToArgb](#) ()  
*Converts this color to an ARGB value.*
- uint [ToRgba](#) ()  
*Converts this color to an RGBA value.*
- uint [ToBgra](#) ()  
*Converts this color to an BGRA value.*
- uint [ToAbgr](#) ()  
*Converts this color to an ABGR value.*
- Vector3 [ToVector3](#) ()  
*Returns a Vector3 representation of this color (ommits alpha value). This can be used with most OpenTK methods.*
- Vector4 [ToVector4](#) ()

*Returns a `Vector4` representation of this color. This can be used with most OpenTK methods.*

- `Color4 ToColor4 ()`

*Returns a `Color4` representation of this color. This can be used with most OpenTK methods.*

## Static Public Member Functions

- static `Color FromRgba (uint value)`  
*Creates a new `Color` from an `RGBA` value.*
- static `Color FromBgra (uint value)`  
*Creates a new `Color` from a `BGRA` value.*
- static `Color FromAbgr (uint value)`  
*Creates a new `Color` from an `ABGR` value.*

## Public Attributes

- readonly float `R`  
*The red component of the color.*
- readonly float `G`  
*The green component of the color.*
- readonly float `B`  
*The blue component of the color.*
- readonly float `A`  
*The color's alpha value.*

## Static Public Attributes

- static readonly `Color AliceBlue` = new `Color`(240, 248, 255)  
*The color AliceBlue.*
- static readonly `Color AntiqueWhite` = new `Color`(250, 235, 215)  
*The color AntiqueWhite.*
- static readonly `Color Aqua` = new `Color`(0, 255, 255)  
*The color Aqua.*
- static readonly `Color Aquamarine` = new `Color`(127, 255, 212)  
*The color Aquamarine.*
- static readonly `Color Azure` = new `Color`(240, 255, 255)  
*The color Azure.*
- static readonly `Color Beige` = new `Color`(245, 245, 220)  
*The color Beige.*
- static readonly `Color Bisque` = new `Color`(255, 228, 196)  
*The color Bisque.*
- static readonly `Color Black` = new `Color`(0, 0, 0)  
*The color Black.*
- static readonly `Color BlanchedAlmond` = new `Color`(255, 235, 205)  
*The color BlanchedAlmond.*
- static readonly `Color Blue` = new `Color`(0, 0, 255)  
*The color Blue.*
- static readonly `Color BlueViolet` = new `Color`(138, 43, 226)  
*The color BlueViolet.*
- static readonly `Color Brown` = new `Color`(165, 42, 42)  
*The color Brown.*

- static readonly [Color BurlyWood](#) = new [Color](#)(222, 184, 135)  
*The color BurlyWood.*
- static readonly [Color CadetBlue](#) = new [Color](#)(95, 158, 160)  
*The color CadetBlue.*
- static readonly [Color Chartreuse](#) = new [Color](#)(127, 255, 0)  
*The color Chartreuse.*
- static readonly [Color Chocolate](#) = new [Color](#)(210, 105, 30)  
*The color Chocolate.*
- static readonly [Color Coral](#) = new [Color](#)(255, 127, 80)  
*The color Coral.*
- static readonly [Color CornflowerBlue](#) = new [Color](#)(100, 149, 237)  
*The color CornflowerBlue.*
- static readonly [Color Cornsilk](#) = new [Color](#)(255, 248, 220)  
*The color Cornsilk.*
- static readonly [Color Crimson](#) = new [Color](#)(220, 20, 60)  
*The color Crimson.*
- static readonly [Color Cyan](#) = new [Color](#)(0, 255, 255)  
*The color Cyan.*
- static readonly [Color DarkBlue](#) = new [Color](#)(0, 0, 139)  
*The color DarkBlue.*
- static readonly [Color DarkCyan](#) = new [Color](#)(0, 139, 139)  
*The color DarkCyan.*
- static readonly [Color DarkGoldenrod](#) = new [Color](#)(184, 134, 11)  
*The color DarkGoldenrod.*
- static readonly [Color DarkGray](#) = new [Color](#)(169, 169, 169)  
*The color DarkGray.*
- static readonly [Color DarkGreen](#) = new [Color](#)(0, 100, 0)  
*The color DarkGreen.*
- static readonly [Color DarkKhaki](#) = new [Color](#)(189, 183, 107)  
*The color DarkKhaki.*
- static readonly [Color DarkMagenta](#) = new [Color](#)(139, 0, 139)  
*The color DarkMagenta.*
- static readonly [Color DarkOliveGreen](#) = new [Color](#)(85, 107, 47)  
*The color DarkOliveGreen.*
- static readonly [Color DarkOrange](#) = new [Color](#)(255, 140, 0)  
*The color DarkOrange.*
- static readonly [Color DarkOrchid](#) = new [Color](#)(153, 50, 204)  
*The color DarkOrchid.*
- static readonly [Color DarkRed](#) = new [Color](#)(139, 0, 0)  
*The color DarkRed.*
- static readonly [Color DarkSalmon](#) = new [Color](#)(233, 150, 122)  
*The color DarkSalmon.*
- static readonly [Color DarkSeaGreen](#) = new [Color](#)(143, 188, 139)  
*The color DarkSeaGreen.*
- static readonly [Color DarkSlateBlue](#) = new [Color](#)(72, 61, 139)  
*The color DarkSlateBlue.*
- static readonly [Color DarkSlateGray](#) = new [Color](#)(47, 79, 79)  
*The color DarkSlateGray.*
- static readonly [Color DarkTurquoise](#) = new [Color](#)(0, 206, 209)  
*The color DarkTurquoise.*
- static readonly [Color DarkViolet](#) = new [Color](#)(148, 0, 211)

- The color DarkViolet.*
- static readonly `Color DeepPink` = new `Color`(255, 20, 147)  
*The color DeepPink.*
- static readonly `Color DeepSkyBlue` = new `Color`(0, 191, 255)  
*The color DeepSkyBlue.*
- static readonly `Color DimGray` = new `Color`(105, 105, 105)  
*The color DimGray.*
- static readonly `Color DodgerBlue` = new `Color`(30, 144, 255)  
*The color DodgerBlue.*
- static readonly `Color Firebrick` = new `Color`(178, 34, 34)  
*The color Firebrick.*
- static readonly `Color FloralWhite` = new `Color`(255, 250, 240)  
*The color FloralWhite.*
- static readonly `Color ForestGreen` = new `Color`(34, 139, 34)  
*The color ForestGreen.*
- static readonly `Color Fuchsia` = new `Color`(255, 0, 255)  
*The color Fuchsia.*
- static readonly `Color Gainsboro` = new `Color`(220, 220, 220)  
*The color Gainsboro.*
- static readonly `Color GhostWhite` = new `Color`(248, 248, 255)  
*The color GhostWhite.*
- static readonly `Color Gold` = new `Color`(255, 215, 0)  
*The color Gold.*
- static readonly `Color Goldenrod` = new `Color`(218, 165, 32)  
*The color Goldenrod.*
- static readonly `Color Gray` = new `Color`(128, 128, 128)  
*The color Gray.*
- static readonly `Color Green` = new `Color`(0, 128, 0)  
*The color Green.*
- static readonly `Color GreenYellow` = new `Color`(173, 255, 47)  
*The color GreenYellow.*
- static readonly `Color HoneyDew` = new `Color`(240, 255, 240)  
*The color HoneyDew.*
- static readonly `Color HotPink` = new `Color`(255, 105, 180)  
*The color HotPink.*
- static readonly `Color IndianRed` = new `Color`(205, 92, 92)  
*The color IndianRed.*
- static readonly `Color Indigo` = new `Color`(75, 0, 130)  
*The color Indigo.*
- static readonly `Color Ivory` = new `Color`(255, 255, 240)  
*The color Ivory.*
- static readonly `Color Khaki` = new `Color`(240, 230, 140)  
*The color Khaki.*
- static readonly `Color Lavender` = new `Color`(230, 230, 250)  
*The color Lavender.*
- static readonly `Color LavenderBlush` = new `Color`(255, 240, 245)  
*The color LavenderBlush.*
- static readonly `Color LawnGreen` = new `Color`(124, 252, 0)  
*The color LawnGreen.*
- static readonly `Color LemonChiffon` = new `Color`(255, 250, 205)  
*The color LemonChiffon.*



- static readonly [Color LightBlue](#) = new [Color](#)(173, 216, 230)  
*The color LightBlue.*
- static readonly [Color LightCoral](#) = new [Color](#)(240, 128, 128)  
*The color LightCoral.*
- static readonly [Color LightCyan](#) = new [Color](#)(224, 255, 255)  
*The color LightCyan.*
- static readonly [Color LightGoldenrodYellow](#) = new [Color](#)(250, 250, 210)  
*The color LightGoldenrodYellow.*
- static readonly [Color LightGray](#) = new [Color](#)(211, 211, 211)  
*The color LightGray.*
- static readonly [Color LightGreen](#) = new [Color](#)(144, 238, 144)  
*The color LightGreen.*
- static readonly [Color LightPink](#) = new [Color](#)(255, 182, 193)  
*The color LightPink.*
- static readonly [Color LightSalmon](#) = new [Color](#)(255, 160, 122)  
*The color LightSalmon.*
- static readonly [Color LightSeaGreen](#) = new [Color](#)(32, 178, 170)  
*The color LightSeaGreen.*
- static readonly [Color LightSkyBlue](#) = new [Color](#)(135, 206, 250)  
*The color LightSkyBlue.*
- static readonly [Color LightSlateGray](#) = new [Color](#)(119, 136, 153)  
*The color LightSlateGray.*
- static readonly [Color LightSteelBlue](#) = new [Color](#)(176, 196, 222)  
*The color LightSteelBlue.*
- static readonly [Color LightYellow](#) = new [Color](#)(255, 255, 224)  
*The color LightYellow.*
- static readonly [Color Lime](#) = new [Color](#)(0, 255, 0)  
*The color Lime.*
- static readonly [Color LimeGreen](#) = new [Color](#)(50, 205, 50)  
*The color LimeGreen.*
- static readonly [Color Linen](#) = new [Color](#)(250, 240, 230)  
*The color Linen.*
- static readonly [Color Magenta](#) = new [Color](#)(255, 0, 255)  
*The color Magenta.*
- static readonly [Color Maroon](#) = new [Color](#)(128, 0, 0)  
*The color Maroon.*
- static readonly [Color MediumAquamarine](#) = new [Color](#)(102, 205, 170)  
*The color MediumAquamarine.*
- static readonly [Color MediumBlue](#) = new [Color](#)(0, 0, 205)  
*The color MediumBlue.*
- static readonly [Color MediumOrchid](#) = new [Color](#)(186, 85, 211)  
*The color MediumOrchid.*
- static readonly [Color MediumPurple](#) = new [Color](#)(147, 112, 219)  
*The color MediumPurple.*
- static readonly [Color MediumSeaGreen](#) = new [Color](#)(60, 179, 113)  
*The color MediumSeaGreen.*
- static readonly [Color MediumSlateBlue](#) = new [Color](#)(123, 104, 238)  
*The color MediumSlateBlue.*
- static readonly [Color MediumSpringGreen](#) = new [Color](#)(0, 250, 154)  
*The color MediumSpringGreen.*
- static readonly [Color MediumTurquoise](#) = new [Color](#)(72, 209, 204)

- The color MediumTurquoise.*

  - static readonly `Color MediumVioletRed` = new `Color`(199, 21, 133)

*The color MediumVioletRed.*
- static readonly `Color MidnightBlue` = new `Color`(25, 25, 112)

*The color MidnightBlue.*
- static readonly `Color MintCream` = new `Color`(245, 255, 250)

*The color MintCream.*
- static readonly `Color MistyRose` = new `Color`(255, 228, 225)

*The color MistyRose.*
- static readonly `Color Moccasin` = new `Color`(255, 228, 181)

*The color Moccasin.*
- static readonly `Color NavajoWhite` = new `Color`(255, 222, 173)

*The color NavajoWhite.*
- static readonly `Color Navy` = new `Color`(0, 0, 128)

*The color Navy.*
- static readonly `Color OldLace` = new `Color`(253, 245, 230)

*The color OldLace.*
- static readonly `Color Olive` = new `Color`(128, 128, 0)

*The color Olive.*
- static readonly `Color OliveDrab` = new `Color`(107, 142, 35)

*The color OliveDrab.*
- static readonly `Color Orange` = new `Color`(255, 165, 0)

*The color Orange.*
- static readonly `Color OrangeRed` = new `Color`(255, 69, 0)

*The color OrangeRed.*
- static readonly `Color Orchid` = new `Color`(218, 112, 214)

*The color Orchid.*
- static readonly `Color PaleGoldenrod` = new `Color`(238, 232, 170)

*The color PaleGoldenrod.*
- static readonly `Color PaleGreen` = new `Color`(152, 251, 152)

*The color PaleGreen.*
- static readonly `Color PaleTurquoise` = new `Color`(175, 238, 238)

*The color PaleTurquoise.*
- static readonly `Color PaleVioletRed` = new `Color`(219, 112, 147)

*The color PaleVioletRed.*
- static readonly `Color PapayaWhip` = new `Color`(225, 239, 213)

*The color PapayaWhip.*
- static readonly `Color PeachPuff` = new `Color`(255, 218, 185)

*The color PeachPuff.*
- static readonly `Color Peru` = new `Color`(205, 133, 63)

*The color Peru.*
- static readonly `Color Pink` = new `Color`(255, 192, 203)

*The color Pink.*
- static readonly `Color Plum` = new `Color`(221, 160, 221)

*The color Plum.*
- static readonly `Color PowderBlue` = new `Color`(176, 224, 230)

*The color PowderBlue.*
- static readonly `Color Purple` = new `Color`(128, 0, 128)

*The color Purple.*
- static readonly `Color Red` = new `Color`(255, 0, 0)

*The color Red.*

- static readonly [Color RosyBrown](#) = new [Color](#)(188, 143, 143)  
*The color RosyBrown.*
- static readonly [Color RoyalBlue](#) = new [Color](#)(65, 105, 225)  
*The color RoyalBlue.*
- static readonly [Color SaddleBrown](#) = new [Color](#)(139, 69, 19)  
*The color SaddleBrown.*
- static readonly [Color Salmon](#) = new [Color](#)(250, 128, 114)  
*The color Salmon.*
- static readonly [Color SandyBrown](#) = new [Color](#)(244, 164, 96)  
*The color SandyBrown.*
- static readonly [Color SeaGreen](#) = new [Color](#)(46, 139, 87)  
*The color SeaGreen.*
- static readonly [Color SeaShell](#) = new [Color](#)(255, 245, 238)  
*The color SeaShell.*
- static readonly [Color Sienna](#) = new [Color](#)(160, 82, 45)  
*The color Sienna.*
- static readonly [Color Silver](#) = new [Color](#)(192, 192, 192)  
*The color Silver.*
- static readonly [Color SkyBlue](#) = new [Color](#)(135, 206, 235)  
*The color SkyBlue.*
- static readonly [Color SlateBlue](#) = new [Color](#)(106, 90, 205)  
*The color SlateBlue.*
- static readonly [Color SlateGray](#) = new [Color](#)(112, 128, 144)  
*The color SlateGray.*
- static readonly [Color Snow](#) = new [Color](#)(255, 250, 250)  
*The color Snow.*
- static readonly [Color SpringGreen](#) = new [Color](#)(0, 255, 127)  
*The color SpringGreen.*
- static readonly [Color SteelBlue](#) = new [Color](#)(70, 130, 180)  
*The color SteelBlue.*
- static readonly [Color Tan](#) = new [Color](#)(210, 180, 140)  
*The color Tan.*
- static readonly [Color Teal](#) = new [Color](#)(0, 128, 128)  
*The color Teal.*
- static readonly [Color Thistle](#) = new [Color](#)(216, 191, 216)  
*The color Thistle.*
- static readonly [Color Tomato](#) = new [Color](#)(255, 99, 71)  
*The color Tomato.*
- static readonly [Color TransparentBlack](#) = new [Color](#)(0, 0, 0, 0)  
*Transparent black color.*
- static readonly [Color TransparentWhite](#) = new [Color](#)(255, 255, 255, 0)  
*Transparent white color.*
- static readonly [Color Turquoise](#) = new [Color](#)(64, 224, 208)  
*The color Turquoise.*
- static readonly [Color Violet](#) = new [Color](#)(238, 130, 238)  
*The color Violet.*
- static readonly [Color Wheat](#) = new [Color](#)(245, 222, 179)  
*The color Wheat.*
- static readonly [Color White](#) = new [Color](#)(255, 255, 255)  
*The color White.*
- static readonly [Color WhiteSmoke](#) = new [Color](#)(245, 245, 245)

*The color WhiteSmoke.*

- static readonly `Color Yellow` = new `Color`(255, 255, 0)

*The color Yellow.*

- static readonly `Color YellowGreen` = new `Color`(154, 205, 50)

*The color YellowGreen.*

### 7.3.1 Detailed Description

Represents an RGBA color that can be used with TriEngine2D.

### 7.3.2 Constructor & Destructor Documentation

#### 7.3.2.1 TriDevs.TriEngine.Color.Color ( Color4 color )

Creates a new color from a Color4 color.

##### Parameters

<i>color</i>	The base Color4 to use, RGBA will be copied from this color.
--------------	--

```

776                                     : this(color.R, color.G, color.B, color.A)
777     {
778
779     }
```

#### 7.3.2.2 TriDevs.TriEngine.Color.Color ( Vector3 vector )

Creates a new color from a Vector3 vector.

##### Parameters

<i>vector</i>	The base Vector3 to use, RGB will be copied from its XYZ values.
---------------	--

```

785                                     : this(vector.X, vector.Y, vector.Z)
786     {
787
788     }
```

#### 7.3.2.3 TriDevs.TriEngine.Color.Color ( Vector4 vector )

Creates a new color from a Vector4 vector.

##### Parameters

<i>vector</i>	The base Vector4 to use, RGBA will be copied from its XYZW values.
---------------	--

```

794                                     : this(vector.X, vector.Y, vector.Z, vector.W)
795     {
796
797     }
```

#### 7.3.2.4 TriDevs.TriEngine.Color.Color ( uint value )

Creates a new color from an ARGB value.

## Parameters

<i>value</i>	The ARGB value to create the color from.
--------------	--

The format of the ARGB value is expected to be 0xAARRGGBB.

```

805         : this((byte)((value >> 16) & 0xFF),
806             (byte)((value >> 8) & 0xFF),
807             (byte)((value >> 0) & 0xFF),
808             (byte)((value >> 24) & 0xFF))
809     {
810
811     }
```

## 7.3.2.5 TriDevs.TriEngine.Color.Color ( Color @ base, byte a )

Creates a new color from a base color with new alpha value.

## Parameters

<i>base</i>	The base color to use, RGB will be copied from this color.
<i>a</i>	The new alpha value to assign (0-255).

```

818                                     : this(@base, a / 255.0f)
819     {
820
821     }
```

## 7.3.2.6 TriDevs.TriEngine.Color.Color ( Color @ base, float a )

Creates a new color from a base color with new alpha value.

## Parameters

<i>base</i>	The base color to use, RGB will be copied from this color.
<i>a</i>	The new alpha value to assign (0.0-1.0).

```

828                                     : this(@base.R, @base.G, @base.B, a)
829     {
830
831     }
```

## 7.3.2.7 TriDevs.TriEngine.Color.Color ( byte r, byte g, byte b, byte a = 255 )

Creates a new color with the specified red, green, blue and alpha values.

## Parameters

<i>r</i>	Value of the red component (0-255).
<i>g</i>	Value of the green component (0-255).
<i>b</i>	Value of the blue component (0-255).
<i>a</i>	Alpha value (0-255) where 0 is transparent and 255 is opaque.

```

840                                     : this(r / 255.0f, g / 255.0f, b / 255.0f, a /
841         255.0f)
842     {
843     }
```

### 7.3.2.8 TriDevs.TriEngine.Color.Color ( float *r*, float *g*, float *b*, float *a* = 1.0f )

Creates a new color with the specified red, green, blue and alpha values.

#### Parameters

<i>r</i>	Value of the red component (0.0-1.0).
<i>g</i>	Value of the green component (0.0-1.0).
<i>b</i>	Value of the blue component (0.0-1.0).
<i>a</i>	Alpha value (0.0-1.0) where 0.0 is transparent and 1.0 is opaque.

```

853     {
854         R = Helpers.Math.Clamp(r, 0.0f, 1.0f);
855         G = Helpers.Math.Clamp(g, 0.0f, 1.0f);
856         B = Helpers.Math.Clamp(b, 0.0f, 1.0f);
857         A = Helpers.Math.Clamp(a, 0.0f, 1.0f);
858     }

```

## 7.3.3 Member Function Documentation

### 7.3.3.1 static Color TriDevs.TriEngine.Color.FromAbgr ( uint *value* ) [static]

Creates a new [Color](#) from an ABGR value.

#### Parameters

<i>value</i>	The ABGR value to create the color from.
--------------	--

#### Returns

A new [Color](#) created from the supplied ABGR value.

The ABGR value is expected to be in the format 0xAABBGRR.

```

894     {
895         // 0xAABBGRR
896         uint a = value & 0xFF000000; // Alpha
897         uint r = (value & 0x000000FF) << 16; // Red
898         uint g = value & 0x0000FF00; // Green
899         uint b = (value & 0x00FF0000) >> 16; // Blue
900         return new Color(a + r + g + b);
901     }

```

### 7.3.3.2 static Color TriDevs.TriEngine.Color.FromBgra ( uint *value* ) [static]

Creates a new [Color](#) from a BGRA value.

#### Parameters

<i>value</i>	The BGRA value to create the color from.
--------------	--

#### Returns

A new [Color](#) created from the supplied BGRA value.

The BGRA value is expected to be in the format 0xBBGGRRAA.

```

878     {
879         // 0xBBGGRRAA
880         uint a = (value & 0x000000FF) << 24; // Alpha
881         uint r = (value & 0x0000FF00) << 8; // Red
882         uint g = (value & 0x00FF0000) >> 8; // Green

```

```

883         uint b = (value & 0xFF000000) >> 24; // Blue
884         return new Color(a + r + g + b);
885     }

```

### 7.3.3.3 static Color TriDevs.TriEngine.Color.FromRgba ( uint value ) [static]

Creates a new [Color](#) from an RGBA value.

#### Parameters

<i>value</i>	The RGBA value to create the color from.
--------------	--

#### Returns

A new [Color](#) created from the supplied RGBA value.

The RGBA value is expected to be in the format 0xRRGGBBAA.

```

867     {
868         return new Color(((value & 0xFF) << 24) + (value >> 8));
869     }

```

### 7.3.3.4 uint TriDevs.TriEngine.Color.ToAbgr ( )

Converts this color to an ABGR value.

#### Returns

The ABGR value of this color.

The format of the ABGR value is 0xAABBGGRR.

```

955     {
956         // 0xAABBGGRR
957         return (uint) (((byte) (A * 255)) << 24) + // Alpha
958                     (((byte) (B * 255)) << 16) + // Blue
959                     (((byte) (G * 255)) << 8) + // Green
960                     (((byte) (R * 255)) << 0); // Red
961     }

```

### 7.3.3.5 uint TriDevs.TriEngine.Color.ToArgb ( )

Converts this color to an ARGB value.

#### Returns

The ARGB value of this color.

The format of the ARGB value is 0xAARRGGBB.

```

910     {
911         // 0xAARRGGBB
912         return (uint) (((byte) (A * 255)) << 24) + // Alpha
913                     (((byte) (R * 255)) << 16) + // Red
914                     (((byte) (G * 255)) << 8) + // Green
915                     (((byte) (B * 255)) << 0); // Blue
916     }

```

### 7.3.3.6 uint TriDevs.TriEngine.Color.ToBgra ( )

Converts this color to an BGRA value.

#### Returns

The BGRA value of this color.

The format of the BGRA value is 0xBBGGRRAA.

```

940     {
941         // 0xBBGGRRAA
942         return (uint) (((byte) (B * 255)) << 24) + // Red
943                     (((byte) (G * 255)) << 16) + // Green
944                     (((byte) (R * 255)) << 8) + // Red
945                     ((byte) (A * 255)) << 0); // Alpha
946     }
```

### 7.3.3.7 Color4 TriDevs.TriEngine.Color.ToColor4 ( )

Returns a Color4 representation of this color. This can be used with most OpenTK methods.

#### Returns

Color4 representation of this color.

```

992     {
993         return new Color4(R, G, B, A);
994     }
```

### 7.3.3.8 uint TriDevs.TriEngine.Color.ToRgba ( )

Converts this color to an RGBA value.

#### Returns

The RGBA value of this color.

The format of the RGBA value is 0xRRGGBBAA.

```

925     {
926         // 0xRRGGBBAA
927         return (uint) (((byte) (R * 255)) << 24) + // Red
928                     (((byte) (G * 255)) << 16) + // Green
929                     (((byte) (B * 255)) << 8) + // Blue
930                     ((byte) (A * 255)) << 0); // Alpha
931     }
```

### 7.3.3.9 Vector3 TriDevs.TriEngine.Color.ToVector3 ( )

Returns a Vector3 representation of this color (ommits alpha value). This can be used with most OpenTK methods.

#### Returns

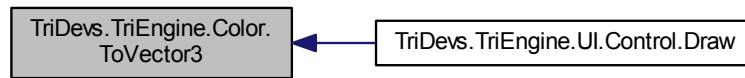
Vector3 representation of this color.

```

970     {
971         return new Vector3(R, G, B);
972     }
```



Here is the caller graph for this function:



#### 7.3.3.10 Vector4 TriDevs.TriEngine.Color.ToVector4 ( )

Returns a Vector4 representation of this color. This can be used with most OpenTK methods.

##### Returns

Vector4 representation of this color.

```

981         {
982             return new Vector4(R, G, B, A);
983         }
  
```

### 7.3.4 Member Data Documentation

#### 7.3.4.1 readonly float TriDevs.TriEngine.Color.A

The color's alpha value.

#### 7.3.4.2 readonly Color TriDevs.TriEngine.Color.AliceBlue = new Color(240, 248, 255) [static]

The color AliceBlue.

#### 7.3.4.3 readonly Color TriDevs.TriEngine.Color.AntiqueWhite = new Color(250, 235, 215) [static]

The color AntiqueWhite.

#### 7.3.4.4 readonly Color TriDevs.TriEngine.Color.Aqua = new Color(0, 255, 255) [static]

The color Aqua.

#### 7.3.4.5 readonly Color TriDevs.TriEngine.Color.Aquamarine = new Color(127, 255, 212) [static]

The color Aquamarine.

#### 7.3.4.6 readonly Color TriDevs.TriEngine.Color.Azure = new Color(240, 255, 255) [static]

The color Azure.

#### 7.3.4.7 readonly float TriDevs.TriEngine.Color.B

The blue component of the color.

**7.3.4.8**    **readonly Color** TriDevs.TriEngine.Color.Beige = new **Color**(245, 245, 220)    [static]

The color Beige.

**7.3.4.9**    **readonly Color** TriDevs.TriEngine.Color.Bisque = new **Color**(255, 228, 196)    [static]

The color Bisque.

**7.3.4.10**   **readonly Color** TriDevs.TriEngine.Color.Black = new **Color**(0, 0, 0)    [static]

The color Black.

**7.3.4.11**   **readonly Color** TriDevs.TriEngine.Color.BlanchedAlmond = new **Color**(255, 235, 205)    [static]

The color BlanchedAlmond.

**7.3.4.12**   **readonly Color** TriDevs.TriEngine.Color.Blue = new **Color**(0, 0, 255)    [static]

The color Blue.

**7.3.4.13**   **readonly Color** TriDevs.TriEngine.Color.BlueViolet = new **Color**(138, 43, 226)    [static]

The color BlueViolet.

**7.3.4.14**   **readonly Color** TriDevs.TriEngine.Color.Brown = new **Color**(165, 42, 42)    [static]

The color Brown.

**7.3.4.15**   **readonly Color** TriDevs.TriEngine.Color.BurlyWood = new **Color**(222, 184, 135)    [static]

The color BurlyWood.

**7.3.4.16**   **readonly Color** TriDevs.TriEngine.Color.CadetBlue = new **Color**(95, 158, 160)    [static]

The color CadetBlue.

**7.3.4.17**   **readonly Color** TriDevs.TriEngine.Color.Chartreuse = new **Color**(127, 255, 0)    [static]

The color Chartreuse.

**7.3.4.18**   **readonly Color** TriDevs.TriEngine.Color.Chocolate = new **Color**(210, 105, 30)    [static]

The color Chocolate.

**7.3.4.19**   **readonly Color** TriDevs.TriEngine.Color.Coral = new **Color**(255, 127, 80)    [static]

The color Coral.

**7.3.4.20**    **readonly Color** TriDevs.TriEngine.Color.CornflowerBlue = new **Color**(100, 149, 237)    [static]

The color CornflowerBlue.

**7.3.4.21**    **readonly Color** TriDevs.TriEngine.Color.Cornsilk = new **Color**(255, 248, 220)    [static]

The color Cornsilk.

**7.3.4.22**    **readonly Color** TriDevs.TriEngine.Color.Crimson = new **Color**(220, 20, 60)    [static]

The color Crimson.

**7.3.4.23**    **readonly Color** TriDevs.TriEngine.Color.Cyan = new **Color**(0, 255, 255)    [static]

The color Cyan.

**7.3.4.24**    **readonly Color** TriDevs.TriEngine.Color.DarkBlue = new **Color**(0, 0, 139)    [static]

The color DarkBlue.

**7.3.4.25**    **readonly Color** TriDevs.TriEngine.Color.DarkCyan = new **Color**(0, 139, 139)    [static]

The color DarkCyan.

**7.3.4.26**    **readonly Color** TriDevs.TriEngine.Color.DarkGoldenrod = new **Color**(184, 134, 11)    [static]

The color DarkGoldenrod.

**7.3.4.27**    **readonly Color** TriDevs.TriEngine.Color.DarkGray = new **Color**(169, 169, 169)    [static]

The color DarkGray.

**7.3.4.28**    **readonly Color** TriDevs.TriEngine.Color.DarkGreen = new **Color**(0, 100, 0)    [static]

The color DarkGreen.

**7.3.4.29**    **readonly Color** TriDevs.TriEngine.Color.DarkKhaki = new **Color**(189, 183, 107)    [static]

The color DarkKhaki.

**7.3.4.30**    **readonly Color** TriDevs.TriEngine.Color.DarkMagenta = new **Color**(139, 0, 139)    [static]

The color DarkMagenta.

**7.3.4.31**    **readonly Color** TriDevs.TriEngine.Color.DarkOliveGreen = new **Color**(85, 107, 47)    [static]

The color DarkOliveGreen.

**7.3.4.32**   `readonly Color TriDevs.TriEngine.Color.DarkOrange = new Color(255, 140, 0)`   `[static]`

The color DarkOrange.

**7.3.4.33**   `readonly Color TriDevs.TriEngine.Color.DarkOrchid = new Color(153, 50, 204)`   `[static]`

The color DarkOrchid.

**7.3.4.34**   `readonly Color TriDevs.TriEngine.Color.DarkRed = new Color(139, 0, 0)`   `[static]`

The color DarkRed.

**7.3.4.35**   `readonly Color TriDevs.TriEngine.Color.DarkSalmon = new Color(233, 150, 122)`   `[static]`

The color DarkSalmon.

**7.3.4.36**   `readonly Color TriDevs.TriEngine.Color.DarkSeaGreen = new Color(143, 188, 139)`   `[static]`

The color DarkSeaGreen.

**7.3.4.37**   `readonly Color TriDevs.TriEngine.Color.DarkSlateBlue = new Color(72, 61, 139)`   `[static]`

The color DarkSlateBlue.

**7.3.4.38**   `readonly Color TriDevs.TriEngine.Color.DarkSlateGray = new Color(47, 79, 79)`   `[static]`

The color DarkSlateGray.

**7.3.4.39**   `readonly Color TriDevs.TriEngine.Color.DarkTurquoise = new Color(0, 206, 209)`   `[static]`

The color DarkTurquoise.

**7.3.4.40**   `readonly Color TriDevs.TriEngine.Color.DarkViolet = new Color(148, 0, 211)`   `[static]`

The color DarkViolet.

**7.3.4.41**   `readonly Color TriDevs.TriEngine.Color.DeepPink = new Color(255, 20, 147)`   `[static]`

The color DeepPink.

**7.3.4.42**   `readonly Color TriDevs.TriEngine.Color.DeepSkyBlue = new Color(0, 191, 255)`   `[static]`

The color DeepSkyBlue.

**7.3.4.43**   `readonly Color TriDevs.TriEngine.Color.DimGray = new Color(105, 105, 105)`   `[static]`

The color DimGray.

**7.3.4.44**    `readonly Color TriDevs.TriEngine.Color.DodgerBlue = new Color(30, 144, 255)    [static]`

The color DodgerBlue.

**7.3.4.45**    `readonly Color TriDevs.TriEngine.Color.Firebrick = new Color(178, 34, 34)    [static]`

The color Firebrick.

**7.3.4.46**    `readonly Color TriDevs.TriEngine.Color.FloralWhite = new Color(255, 250, 240)    [static]`

The color FloralWhite.

**7.3.4.47**    `readonly Color TriDevs.TriEngine.Color.ForestGreen = new Color(34, 139, 34)    [static]`

The color ForestGreen.

**7.3.4.48**    `readonly Color TriDevs.TriEngine.Color.Fuchsia = new Color(255, 0, 255)    [static]`

The color Fuchsia.

**7.3.4.49**    `readonly float TriDevs.TriEngine.Color.G`

The green component of the color.

**7.3.4.50**    `readonly Color TriDevs.TriEngine.Color.Gainsboro = new Color(220, 220, 220)    [static]`

The color Gainsboro.

**7.3.4.51**    `readonly Color TriDevs.TriEngine.Color.GhostWhite = new Color(248, 248, 255)    [static]`

The color GhostWhite.

**7.3.4.52**    `readonly Color TriDevs.TriEngine.Color.Gold = new Color(255, 215, 0)    [static]`

The color Gold.

**7.3.4.53**    `readonly Color TriDevs.TriEngine.Color.Goldenrod = new Color(218, 165, 32)    [static]`

The color Goldenrod.

**7.3.4.54**    `readonly Color TriDevs.TriEngine.Color.Gray = new Color(128, 128, 128)    [static]`

The color Gray.

**7.3.4.55**    `readonly Color TriDevs.TriEngine.Color.Green = new Color(0, 128, 0)    [static]`

The color Green.

**7.3.4.56**   `readonly Color TriDevs.TriEngine.Color.GreenYellow = new Color(173, 255, 47)   [static]`

The color GreenYellow.

**7.3.4.57**   `readonly Color TriDevs.TriEngine.Color.HoneyDew = new Color(240, 255, 240)   [static]`

The color HoneyDew.

**7.3.4.58**   `readonly Color TriDevs.TriEngine.Color.HotPink = new Color(255, 105, 180)   [static]`

The color HotPink.

**7.3.4.59**   `readonly Color TriDevs.TriEngine.Color.IndianRed = new Color(205, 92, 92)   [static]`

The color IndianRed.

**7.3.4.60**   `readonly Color TriDevs.TriEngine.Color.Indigo = new Color(75, 0, 130)   [static]`

The color Indigo.

**7.3.4.61**   `readonly Color TriDevs.TriEngine.Color.Ivory = new Color(255, 255, 240)   [static]`

The color Ivory.

**7.3.4.62**   `readonly Color TriDevs.TriEngine.Color.Khaki = new Color(240, 230, 140)   [static]`

The color Khaki.

**7.3.4.63**   `readonly Color TriDevs.TriEngine.Color.Lavender = new Color(230, 230, 250)   [static]`

The color Lavender.

**7.3.4.64**   `readonly Color TriDevs.TriEngine.Color.LavenderBlush = new Color(255, 240, 245)   [static]`

The color LavenderBlush.

**7.3.4.65**   `readonly Color TriDevs.TriEngine.Color.LawnGreen = new Color(124, 252, 0)   [static]`

The color LawnGreen.

**7.3.4.66**   `readonly Color TriDevs.TriEngine.Color.LemonChiffon = new Color(255, 250, 205)   [static]`

The color LemonChiffon.

**7.3.4.67**   `readonly Color TriDevs.TriEngine.Color.LightBlue = new Color(173, 216, 230)   [static]`

The color LightBlue.

**7.3.4.68**    **readonly Color** TriDevs.TriEngine.Color.LightCoral = new **Color**(240, 128, 128)    [static]

The color LightCoral.

**7.3.4.69**    **readonly Color** TriDevs.TriEngine.Color.LightCyan = new **Color**(224, 255, 255)    [static]

The color LightCyan.

**7.3.4.70**    **readonly Color** TriDevs.TriEngine.Color.LightGoldenrodYellow = new **Color**(250, 250, 210)    [static]

The color LightGoldenrodYellow.

**7.3.4.71**    **readonly Color** TriDevs.TriEngine.Color.LightGray = new **Color**(211, 211, 211)    [static]

The color LightGray.

**7.3.4.72**    **readonly Color** TriDevs.TriEngine.Color.LightGreen = new **Color**(144, 238, 144)    [static]

The color LightGreen.

**7.3.4.73**    **readonly Color** TriDevs.TriEngine.Color.LightPink = new **Color**(255, 182, 193)    [static]

The color LightPink.

**7.3.4.74**    **readonly Color** TriDevs.TriEngine.Color.LightSalmon = new **Color**(255, 160, 122)    [static]

The color LightSalmon.

**7.3.4.75**    **readonly Color** TriDevs.TriEngine.Color.LightSeaGreen = new **Color**(32, 178, 170)    [static]

The color LightSeaGreen.

**7.3.4.76**    **readonly Color** TriDevs.TriEngine.Color.LightSkyBlue = new **Color**(135, 206, 250)    [static]

The color LightSkyBlue.

**7.3.4.77**    **readonly Color** TriDevs.TriEngine.Color.LightSlateGray = new **Color**(119, 136, 153)    [static]

The color LightSlateGray.

**7.3.4.78**    **readonly Color** TriDevs.TriEngine.Color.LightSteelBlue = new **Color**(176, 196, 222)    [static]

The color LightSteelBlue.

**7.3.4.79**    **readonly Color** TriDevs.TriEngine.Color.LightYellow = new **Color**(255, 255, 224)    [static]

The color LightYellow.

**7.3.4.80**   `readonly Color TriDevs.TriEngine.Color.Lime = new Color(0, 255, 0)`   `[static]`

The color Lime.

**7.3.4.81**   `readonly Color TriDevs.TriEngine.Color.LimeGreen = new Color(50, 205, 50)`   `[static]`

The color LimeGreen.

**7.3.4.82**   `readonly Color TriDevs.TriEngine.Color.Linen = new Color(250, 240, 230)`   `[static]`

The color Linen.

**7.3.4.83**   `readonly Color TriDevs.TriEngine.Color.Magenta = new Color(255, 0, 255)`   `[static]`

The color Magenta.

**7.3.4.84**   `readonly Color TriDevs.TriEngine.Color.Maroon = new Color(128, 0, 0)`   `[static]`

The color Maroon.

**7.3.4.85**   `readonly Color TriDevs.TriEngine.Color.MediumAquamarine = new Color(102, 205, 170)`   `[static]`

The color MediumAquamarine.

**7.3.4.86**   `readonly Color TriDevs.TriEngine.Color.MediumBlue = new Color(0, 0, 205)`   `[static]`

The color MediumBlue.

**7.3.4.87**   `readonly Color TriDevs.TriEngine.Color.MediumOrchid = new Color(186, 85, 211)`   `[static]`

The color MediumOrchid.

**7.3.4.88**   `readonly Color TriDevs.TriEngine.Color.MediumPurple = new Color(147, 112, 219)`   `[static]`

The color MediumPurple.

**7.3.4.89**   `readonly Color TriDevs.TriEngine.Color.MediumSeaGreen = new Color(60, 179, 113)`   `[static]`

The color MediumSeaGreen.

**7.3.4.90**   `readonly Color TriDevs.TriEngine.Color.MediumSlateBlue = new Color(123, 104, 238)`   `[static]`

The color MediumSlateBlue.

**7.3.4.91**   `readonly Color TriDevs.TriEngine.Color.MediumSpringGreen = new Color(0, 250, 154)`   `[static]`

The color MediumSpringGreen.



7.3.4.92 `readonly Color TriDevs.TriEngine.Color.MediumTurquoise = new Color(72, 209, 204) [static]`

The color MediumTurquoise.

7.3.4.93 `readonly Color TriDevs.TriEngine.Color.MediumVioletRed = new Color(199, 21, 133) [static]`

The color MediumVioletRed.

7.3.4.94 `readonly Color TriDevs.TriEngine.Color.MidnightBlue = new Color(25, 25, 112) [static]`

The color MidnightBlue.

7.3.4.95 `readonly Color TriDevs.TriEngine.Color.MintCream = new Color(245, 255, 250) [static]`

The color MintCream.

7.3.4.96 `readonly Color TriDevs.TriEngine.Color.MistyRose = new Color(255, 228, 225) [static]`

The color MistyRose.

7.3.4.97 `readonly Color TriDevs.TriEngine.Color.Moccasin = new Color(255, 228, 181) [static]`

The color Moccasin.

7.3.4.98 `readonly Color TriDevs.TriEngine.Color.NavajoWhite = new Color(255, 222, 173) [static]`

The color NavajoWhite.

7.3.4.99 `readonly Color TriDevs.TriEngine.Color.Navy = new Color(0, 0, 128) [static]`

The color Navy.

7.3.4.100 `readonly Color TriDevs.TriEngine.Color.OldLace = new Color(253, 245, 230) [static]`

The color OldLace.

7.3.4.101 `readonly Color TriDevs.TriEngine.Color.Olive = new Color(128, 128, 0) [static]`

The color Olive.

7.3.4.102 `readonly Color TriDevs.TriEngine.Color.OliveDrab = new Color(107, 142, 35) [static]`

The color OliveDrab.

7.3.4.103 `readonly Color TriDevs.TriEngine.Color.Orange = new Color(255, 165, 0) [static]`

The color Orange.

7.3.4.104    readonly **Color** TriDevs.TriEngine.Color.OrangeRed = new **Color**(255, 69, 0)    [static]

The color OrangeRed.

7.3.4.105    readonly **Color** TriDevs.TriEngine.Color.Orchid = new **Color**(218, 112, 214)    [static]

The color Orchid.

7.3.4.106    readonly **Color** TriDevs.TriEngine.Color.PaleGoldenrod = new **Color**(238, 232, 170)    [static]

The color PaleGoldenrod.

7.3.4.107    readonly **Color** TriDevs.TriEngine.Color.PaleGreen = new **Color**(152, 251, 152)    [static]

The color PaleGreen.

7.3.4.108    readonly **Color** TriDevs.TriEngine.Color.PaleTurquoise = new **Color**(175, 238, 238)    [static]

The color PaleTurquoise.

7.3.4.109    readonly **Color** TriDevs.TriEngine.Color.PaleVioletRed = new **Color**(219, 112, 147)    [static]

The color PaleVioletRed.

7.3.4.110    readonly **Color** TriDevs.TriEngine.Color.PapayaWhip = new **Color**(225, 239, 213)    [static]

The color PapayaWhip.

7.3.4.111    readonly **Color** TriDevs.TriEngine.Color.PeachPuff = new **Color**(255, 218, 185)    [static]

The color PeachPuff.

7.3.4.112    readonly **Color** TriDevs.TriEngine.Color.Peru = new **Color**(205, 133, 63)    [static]

The color Peru.

7.3.4.113    readonly **Color** TriDevs.TriEngine.Color.Pink = new **Color**(255, 192, 203)    [static]

The color Pink.

7.3.4.114    readonly **Color** TriDevs.TriEngine.Color.Plum = new **Color**(221, 160, 221)    [static]

The color Plum.

7.3.4.115    readonly **Color** TriDevs.TriEngine.Color.PowderBlue = new **Color**(176, 224, 230)    [static]

The color PowderBlue.

7.3.4.116   readonly **Color** TriDevs.TriEngine.Color.Purple = new **Color**(128, 0, 128)   [static]

The color Purple.

7.3.4.117   readonly float TriDevs.TriEngine.Color.R

The red component of the color.

7.3.4.118   readonly **Color** TriDevs.TriEngine.Color.Red = new **Color**(255, 0, 0)   [static]

The color Red.

7.3.4.119   readonly **Color** TriDevs.TriEngine.Color.RosyBrown = new **Color**(188, 143, 143)   [static]

The color RosyBrown.

7.3.4.120   readonly **Color** TriDevs.TriEngine.Color.RoyalBlue = new **Color**(65, 105, 225)   [static]

The color RoyalBlue.

7.3.4.121   readonly **Color** TriDevs.TriEngine.Color.SaddleBrown = new **Color**(139, 69, 19)   [static]

The color SaddleBrown.

7.3.4.122   readonly **Color** TriDevs.TriEngine.Color.Salmon = new **Color**(250, 128, 114)   [static]

The color Salmon.

7.3.4.123   readonly **Color** TriDevs.TriEngine.Color.SandyBrown = new **Color**(244, 164, 96)   [static]

The color SandyBrown.

7.3.4.124   readonly **Color** TriDevs.TriEngine.Color.SeaGreen = new **Color**(46, 139, 87)   [static]

The color SeaGreen.

7.3.4.125   readonly **Color** TriDevs.TriEngine.Color.SeaShell = new **Color**(255, 245, 238)   [static]

The color SeaShell.

7.3.4.126   readonly **Color** TriDevs.TriEngine.Color.Sienna = new **Color**(160, 82, 45)   [static]

The color Sienna.

7.3.4.127   readonly **Color** TriDevs.TriEngine.Color.Silver = new **Color**(192, 192, 192)   [static]

The color Silver.

7.3.4.128   readonly **Color** TriDevs.TriEngine.Color.SkyBlue = new **Color**(135, 206, 235)   [static]

The color SkyBlue.

7.3.4.129   readonly **Color** TriDevs.TriEngine.Color.SlateBlue = new **Color**(106, 90, 205)   [static]

The color SlateBlue.

7.3.4.130   readonly **Color** TriDevs.TriEngine.Color.SlateGray = new **Color**(112, 128, 144)   [static]

The color SlateGray.

7.3.4.131   readonly **Color** TriDevs.TriEngine.Color.Snow = new **Color**(255, 250, 250)   [static]

The color Snow.

7.3.4.132   readonly **Color** TriDevs.TriEngine.Color.SpringGreen = new **Color**(0, 255, 127)   [static]

The color SpringGreen.

7.3.4.133   readonly **Color** TriDevs.TriEngine.Color.SteelBlue = new **Color**(70, 130, 180)   [static]

The color SteelBlue.

7.3.4.134   readonly **Color** TriDevs.TriEngine.Color.Tan = new **Color**(210, 180, 140)   [static]

The color Tan.

7.3.4.135   readonly **Color** TriDevs.TriEngine.Color.Teal = new **Color**(0, 128, 128)   [static]

The color Teal.

7.3.4.136   readonly **Color** TriDevs.TriEngine.Color.Thistle = new **Color**(216, 191, 216)   [static]

The color Thistle.

7.3.4.137   readonly **Color** TriDevs.TriEngine.Color.Tomato = new **Color**(255, 99, 71)   [static]

The color Tomato.

7.3.4.138   readonly **Color** TriDevs.TriEngine.Color.TransparentBlack = new **Color**(0, 0, 0, 0)   [static]

Transparent black color.

7.3.4.139   readonly **Color** TriDevs.TriEngine.Color.TransparentWhite = new **Color**(255, 255, 255, 0)   [static]

Transparent white color.

7.3.4.140    readonly **Color** TriDevs.TriEngine.Color.Turquoise = new **Color**(64, 224, 208)    [static]

The color Turquoise.

7.3.4.141    readonly **Color** TriDevs.TriEngine.Color.Violet = new **Color**(238, 130, 238)    [static]

The color Violet.

7.3.4.142    readonly **Color** TriDevs.TriEngine.Color.Wheat = new **Color**(245, 222, 179)    [static]

The color Wheat.

7.3.4.143    readonly **Color** TriDevs.TriEngine.Color.White = new **Color**(255, 255, 255)    [static]

The color White.

7.3.4.144    readonly **Color** TriDevs.TriEngine.Color.WhiteSmoke = new **Color**(245, 245, 245)    [static]

The color WhiteSmoke.

7.3.4.145    readonly **Color** TriDevs.TriEngine.Color.Yellow = new **Color**(255, 255, 0)    [static]

The color Yellow.

7.3.4.146    readonly **Color** TriDevs.TriEngine.Color.YellowGreen = new **Color**(154, 205, 50)    [static]

The color YellowGreen.

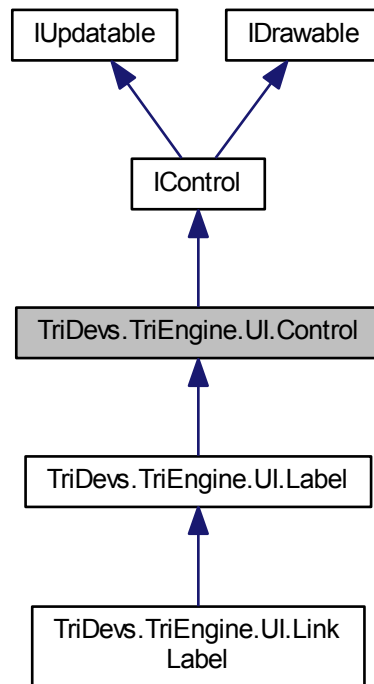
The documentation for this struct was generated from the following file:

- TriDevs.TriEngine/[Color.cs](#)

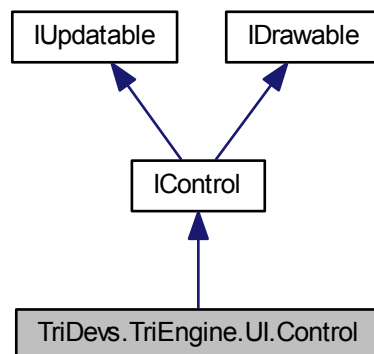
## 7.4 TriDevs.TriEngine.UI.Control Class Reference

Base control class that all other controls inherits from. Defines basic [UI](#) control behaviour.

Inheritance diagram for TriDevs.TriEngine.UI.Control:



Collaboration diagram for TriDevs.TriEngine.UI.Control:



## Public Member Functions

- virtual void [Enable](#) ()

*Enables the control.*

- virtual void [Disable](#) ()

*Disables the control.*

- virtual void [Show](#) ()

*Shows the control.*

- virtual void [Hide](#) ()

*Hides the control.*

- virtual void [Update](#) ()

*Updates the object.*

- virtual void [Draw](#) ()

*Draw the object to screen.*

## Protected Member Functions

- virtual void [OnClicked](#) ()
- virtual void [Draw](#) (Point< int > position)

## Properties

- virtual bool [Enabled](#) [get, set]
- virtual bool [Visible](#) [get, set]
- virtual [Color](#) [Color](#) [get, set]
- virtual Point< int > [Position](#) [get, set]
- virtual Point< int > [Size](#) [get, set]
- virtual [Rectangle](#) [Rectangle](#) [get, set]
- virtual string [Text](#) [get, set]

## Events

- ControlClickedEventHandler [Clicked](#)  
*Raised when this control is clicked on by the user.*

### 7.4.1 Detailed Description

Base control class that all other controls inherits from. Defines basic [UI](#) control behaviour.

### 7.4.2 Member Function Documentation

#### 7.4.2.1 virtual void TriDevs.TriEngine.UI.Control.Disable ( ) [virtual]

Disables the control.

Implements [TriDevs.TriEngine.UI.IControl](#).

```
86         {
87             Enabled = false;
88         }
```

#### 7.4.2.2 virtual void TriDevs.TriEngine.UI.Control.Draw ( ) [virtual]

Draw the object to screen.

Implements [TriDevs.TriEngine.Interfaces.IDrawable](#).

Reimplemented in [TriDevs.TriEngine.UI.Label](#).

```

116     {
117         Draw(Position);
118     }

```

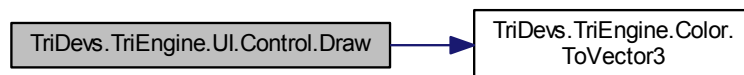
#### 7.4.2.3 virtual void TriDevs.TriEngine.UI.Control.Draw ( Point< int > position ) [protected],[virtual]

```

121     {
122         // Placeholder drawing,
123         // we should replace this with proper control drawing
124         GL.Disable(EnableCap.Texture2D);
125         var color = Color.ToVector3();
126         GL.Color3(color);
127         GL.Begin(BeginMode.Quads);
128         GL.Vertex2(position.X, position.Y);
129         GL.Vertex2(position.X + Size.X, position.Y);
130         GL.Vertex2(position.X + Size.X, position.Y + Size.Y);
131         GL.Vertex2(position.X, position.Y + Size.Y);
132         GL.End();
133     }

```

Here is the call graph for this function:



#### 7.4.2.4 virtual void TriDevs.TriEngine.UI.Control.Enable ( ) [virtual]

Enables the control.

Implements [TriDevs.TriEngine.UI.IControl](#).

```

81     {
82         Enabled = true;
83     }

```

#### 7.4.2.5 virtual void TriDevs.TriEngine.UI.Control.Hide ( ) [virtual]

Hides the control.

Implements [TriDevs.TriEngine.UI.IControl](#).

```

96     {
97         Visible = false;
98     }

```



**7.4.2.6** virtual void TriDevs.TriEngine.UI.Control.OnClicked ( ) [protected],[virtual]

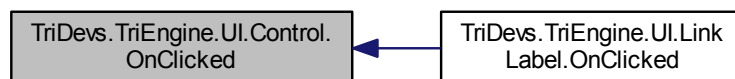
Reimplemented in [TriDevs.TriEngine.UI.LinkLabel](#).

```

64     {
65         var func = Clicked;
66         if (func != null)
67             func(this, null);
68     }

```

Here is the caller graph for this function:

**7.4.2.7** virtual void TriDevs.TriEngine.UI.Control.Show ( ) [virtual]

Shows the control.

Implements [TriDevs.TriEngine.UI.IControl](#).

```

91     {
92         Visible = true;
93     }

```

**7.4.2.8** virtual void TriDevs.TriEngine.UI.Control.Update ( ) [virtual]

Updates the object.

Implements [TriDevs.TriEngine.Interfaces.IUpdatable](#).

Reimplemented in [TriDevs.TriEngine.UI.Label](#).

```

101    {
102        // Return immediately if there is no mouse click
103        // We only run the click handlers if the user has is releasing
104        // the mouse button while on a control, to mimic how most UIs
105        // handle click events.
106        if (!Services.Input.MouseReleased(MouseButton.Left))
107            return;
108
109        var mousePos = Services.Input.MousePosition;
110        if ((mousePos.X >= Rectangle.X && mousePos.X <= (Rectangle.
X + Rectangle.Width))
111            && (mousePos.Y >= Rectangle.Y && mousePos.Y <= (
Rectangle.Y + Rectangle.Height)))
112            OnClicked();
113    }

```

**7.4.3** Property Documentation**7.4.3.1** virtual Color TriDevs.TriEngine.UI.Control.Color [get],[set]**7.4.3.2** virtual bool TriDevs.TriEngine.UI.Control.Enabled [get],[set]

7.4.3.3 virtual `Point<int>` `TriDevs.TriEngine.UI.Control.Position` `[get]`, `[set]`

7.4.3.4 virtual `Rectangle` `TriDevs.TriEngine.UI.Control.Rectangle` `[get]`, `[set]`

7.4.3.5 virtual `Point<int>` `TriDevs.TriEngine.UI.Control.Size` `[get]`, `[set]`

7.4.3.6 virtual `string` `TriDevs.TriEngine.UI.Control.Text` `[get]`, `[set]`

7.4.3.7 virtual `bool` `TriDevs.TriEngine.UI.Control.Visible` `[get]`, `[set]`

## 7.4.4 Event Documentation

7.4.4.1 `ControlClickedEventHandler` `TriDevs.TriEngine.UI.Control.Clicked`

Raised when this control is clicked on by the user.

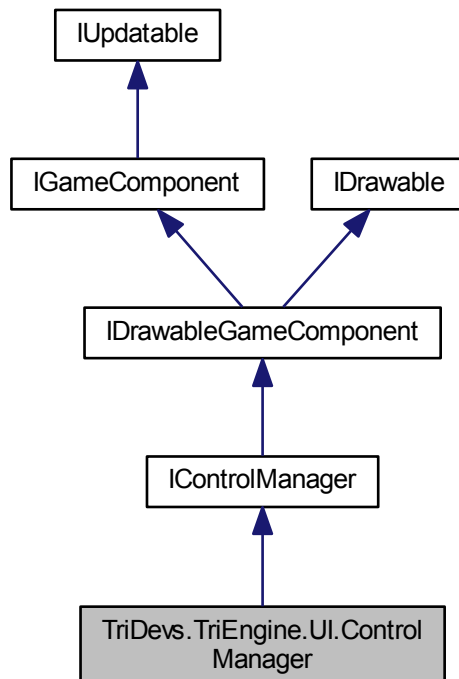
The documentation for this class was generated from the following file:

- `TriDevs.TriEngine/UI/Control.cs`

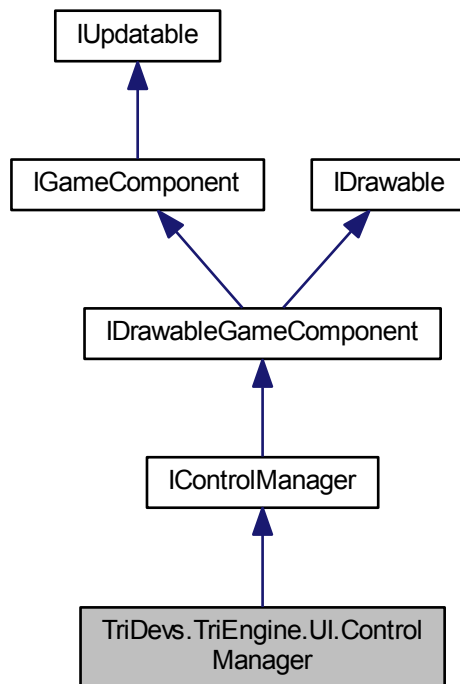
## 7.5 TriDevs.TriEngine.UI.ControlManager Class Reference

`Control` manager to manage various `UI` controls for a game.

Inheritance diagram for `TriDevs.TriEngine.UI.ControlManager`:



Collaboration diagram for TriDevs.TriEngine.UI.ControlManager:



## Public Member Functions

- `ControlManager ()`  
*Initializes a new instance of this control manager.*
- `void Enable ()`  
*Enables this game component.*
- `void Disable ()`  
*Disables this game component.*
- `void Update ()`  
*Updates the object.*
- `void Draw ()`  
*Draw the object to screen.*
- `IControl AddControl (IControl control)`  
*Adds a new control to this control manager.*
- `void RemoveControl (IControl control)`  
*Removes a control from this control manager.*
- `void RemoveAllControls (Type type)`  
*Removes all controls of a specific type from this control manager.*
- `void RemoveAllControls (Func< IControl, bool > func)`  
*Removes all controls matching the supplied predicate function.*
- `bool HasControl (IControl control)`  
*Returns whether this control manager contains the specified control.*

- bool [HasControl](#) (Type type)  
*Returns whether this control manager contains any control of the specified type.*
- bool [HasControl](#) (Func< [IControl](#), bool > func)  
*Returns whether this control manager contains any control matching the supplied predicate function.*

## Properties

- bool [Enabled](#) [get, set]

### 7.5.1 Detailed Description

[Control](#) manager to manage various [UI](#) controls for a game.

### 7.5.2 Constructor & Destructor Documentation

#### 7.5.2.1 [TriDevs.TriEngine.UI.ControlManager.ControlManager](#) ( )

Initializes a new instance of this control manager.

```

53     {
54         _controls = new List<IControl>();
55     }
```

### 7.5.3 Member Function Documentation

#### 7.5.3.1 [IControl](#) [TriDevs.TriEngine.UI.ControlManager.AddControl](#) ( [IControl](#) control )

Adds a new control to this control manager.

##### Parameters

<i>control</i>	The control to add.
----------------	---------------------

##### Returns

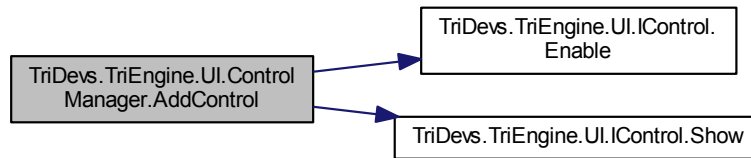
The control that was added.

Implements [TriDevs.TriEngine.UI.IControlManager](#).

```

78     {
79         if (HasControl(control))
80             throw new InvalidOperationException("Cannot add a control more than once.");
81         control.Enable();
82         _controls.Add(control);
83         control.Show();
84         return control;
85     }
```

Here is the call graph for this function:



### 7.5.3.2 void TriDevs.TriEngine.UI.ControlManager.Disable ( )

Disables this game component.

Implements [TriDevs.TriEngine.Interfaces.IGameComponent](#).

```

63     {
64         Enabled = false;
65     }
  
```

### 7.5.3.3 void TriDevs.TriEngine.UI.ControlManager.Draw ( )

Draw the object to screen.

Implements [TriDevs.TriEngine.Interfaces.IDrawable](#).

```

73     {
74         _controls.ForEach(c => c.Draw());
75     }
  
```

### 7.5.3.4 void TriDevs.TriEngine.UI.ControlManager.Enable ( )

Enables this game component.

Implements [TriDevs.TriEngine.Interfaces.IGameComponent](#).

```

58     {
59         Enabled = true;
60     }
  
```

### 7.5.3.5 bool TriDevs.TriEngine.UI.ControlManager.HasControl ( IControl control )

Returns whether this control manager contains the specified control.

#### Parameters

<i>control</i>	The control to check.
----------------	-----------------------

**Returns**

True if the specified control exists in this control manager, false otherwise.

Implements [TriDevs.TriEngine.UI.IControlManager](#).

```
119         {
120             return HasControl(c => c == control);
121         }
```

**7.5.3.6 bool TriDevs.TriEngine.UI.ControlManager.HasControl ( Type type )**

Returns whether this control manager contains any control of the specified type.

**Parameters**

<i>type</i>	The type to check.
-------------	--------------------

**Returns**

True if this control manager contains a control of the specified type, false otherwise.

Implements [TriDevs.TriEngine.UI.IControlManager](#).

```
124         {
125             return HasControl(c => c.GetType() == type);
126         }
```

**7.5.3.7 bool TriDevs.TriEngine.UI.ControlManager.HasControl ( Func< IControl, bool > func )**

Returns whether this control manager contains any control matching the supplied predicate function.

**Parameters**

<i>func</i>	The predicate function.
-------------	-------------------------

**Returns**

True if this control manager contains a control matching the specified predicate, false otherwise.

Implements [TriDevs.TriEngine.UI.IControlManager](#).

```
129         {
130             return _controls.Any(func);
131         }
```

**7.5.3.8 void TriDevs.TriEngine.UI.ControlManager.RemoveAllControls ( Type type )**

Removes all controls of a specific type from this control manager.

**Parameters**

<i>type</i>	The type of control to remove.
-------------	--------------------------------

Implements [TriDevs.TriEngine.UI.IControlManager](#).

```

100     {
101         RemoveAllControls (c => c.GetType() == type);
102     }

```

### 7.5.3.9 void TriDevs.TriEngine.UI.ControlManager.RemoveAllControls ( Func< IControl, bool > func )

Removes all controls matching the supplied predicate function.

#### Parameters

<i>func</i>	The predicate function to use.
-------------	--------------------------------

Implements [TriDevs.TriEngine.UI.IControlManager](#).

```

105     {
106         var toRemove = _controls.Where(func);
107         var controls = toRemove as IList<IControl> ?? toRemove.ToList();
108         if (controls.Count < 0)
109             return;
110         controls.ToList().ForEach(c =>
111         {
112             c.Hide();
113             c.Disable();
114         });
115         _controls.RemoveAll(c => func(c));
116     }

```

### 7.5.3.10 void TriDevs.TriEngine.UI.ControlManager.RemoveControl ( IControl control )

Removes a control from this control manager.

#### Parameters

<i>control</i>	The control to remove.
----------------	------------------------

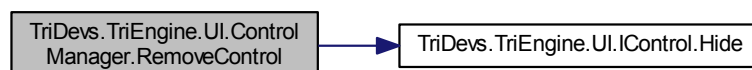
Implements [TriDevs.TriEngine.UI.IControlManager](#).

```

88     {
89         if (!HasControl(control))
90             return;
91         var match = _controls.FirstOrDefault(c => c == control);
92         if (match == null)
93             return;
94         match.Hide();
95         match.Disable();
96         _controls.Remove(match);
97     }

```

Here is the call graph for this function:



### 7.5.3.11 void TriDevs.TriEngine.UI.ControlManager.Update ( )

Updates the object.

Implements [TriDevs.TriEngine.Interfaces.IUpdatable](#).

```
68         {
69             _controls.ForEach(c => c.Update());
70         }
```

## 7.5.4 Property Documentation

### 7.5.4.1 bool TriDevs.TriEngine.UI.ControlManager.Enabled [get], [set]

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

- [TriDevs.TriEngine/UI/ControlManager.cs](#)

## 7.6 TriDevs.TriEngine.Extensions.DecimalExtensions Class Reference

[Extensions](#) for System.Decimal.

### Static Public Member Functions

- static decimal [Clamp](#) (this decimal value, decimal min, decimal max)  
*Clamps the specified decimal between a minimum and maximum value.*

### 7.6.1 Detailed Description

[Extensions](#) for System.Decimal.

### 7.6.2 Member Function Documentation

#### 7.6.2.1 static decimal TriDevs.TriEngine.Extensions.DecimalExtensions.Clamp ( this decimal value, decimal min, decimal max ) [static]

Clamps the specified decimal between a minimum and maximum value.

#### Parameters

<i>value</i>	Value to clamp.
<i>min</i>	If the specified decimal is below this value, then this will be returned.
<i>max</i>	If the specified decimal is above this value, then this will be returned.

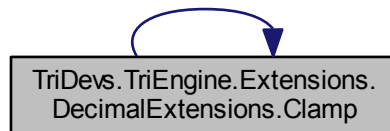


### Returns

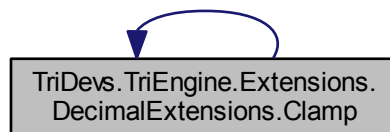
The clamped value of the decimal.

```
39         {  
40             return Helpers.Math.Clamp(value, min, max);  
41         }
```

Here is the call graph for this function:



Here is the caller graph for this function:



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

- TriDevs.TriEngine/Extensions/[DecimalExtensions.cs](#)

## 7.7 TriDevs.TriEngine.Extensions.DoubleExtensions Class Reference

[Extensions](#) for System.Double.

### Static Public Member Functions

- static double [Clamp](#) (this double value, double min, double max)  
*Clamps the specified double between a minimum and maximum value.*

#### 7.7.1 Detailed Description

[Extensions](#) for System.Double.

## 7.7.2 Member Function Documentation

**7.7.2.1** `static double TriDevs.TriEngine.Extensions.DoubleExtensions.Clamp ( this double value, double min, double max )`  
`[static]`

Clamps the specified double between a minimum and maximum value.

### Parameters

<i>value</i>	Value to clamp.
<i>min</i>	If the specified double is below this value, then this will be returned.
<i>max</i>	If the specified double is above this value, then this will be returned.

### Returns

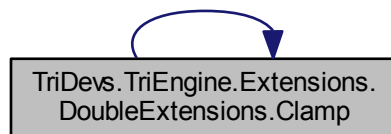
The clamped value of the double.

```

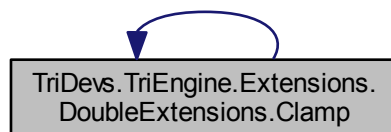
39         {
40             return Helpers.Math.Clamp(value, min, max);
41         }

```

Here is the call graph for this function:



Here is the caller graph for this function:



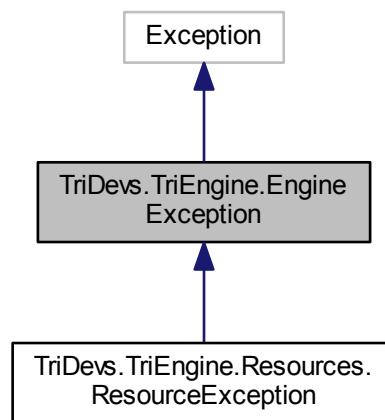
The documentation for this class was generated from the following file:

- TriDevs.TriEngine/Extensions/[DoubleExtensions.cs](#)

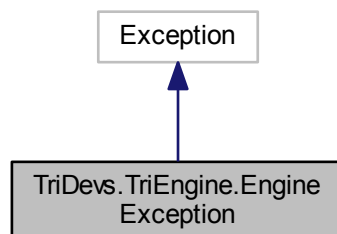
## 7.8 TriDevs.TriEngine.EngineException Class Reference

Base exception class for all engine-related exceptions. The inner exception will contain more info as to what actually happened.

Inheritance diagram for TriDevs.TriEngine.EngineException:



Collaboration diagram for TriDevs.TriEngine.EngineException:



### 7.8.1 Detailed Description

Base exception class for all engine-related exceptions. The inner exception will contain more info as to what actually happened.

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

- TriDevs.TriEngine/[EngineException.cs](#)

## 7.9 TriDevs.TriEngine.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.*

### 7.9.1 Detailed Description

[Extensions](#) for System.Enum.

### 7.9.2 Member Function Documentation

**7.9.2.1** static bool TriDevs.TriEngine.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     }
```

**7.9.2.2** static T TriDevs.TriEngine.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     }
```

**7.9.2.3** static bool TriDevs.TriEngine.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     }
```

**7.9.2.4** static T TriDevs.TriEngine.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.TriEngine/Extensions/[EnumerationExtensions.cs](#)

## 7.10 TriDevs.TriEngine.Helpers.Exceptions Class Reference

Provides helper methods for dealing with exceptions.

### Static Public Member Functions

- static void [LogException](#) (Exception exception)  
*Outputs exception details to default logger.*

#### 7.10.1 Detailed Description

Provides helper methods for dealing with exceptions.

#### 7.10.2 Member Function Documentation

##### 7.10.2.1 static void TriDevs.TriEngine.Helpers.Exceptions.LogException ( Exception *exception* ) [static]

Outputs exception details to default logger.

## Parameters

<i>exception</i>	The exception to log.
------------------	-----------------------

```

56     {
57         var type = "Generic";
58         if (exception is EngineException)
59             type = "Engine";
60
61         _log.ErrorFormat("{0} exception {1} occurred with message: {2}", type, exception.GetType(),
exception.Message);

```

```

62         if (exception.InnerException != null)
63             _log.ErrorFormat("Inner exception {0} with message: {1}", exception.InnerException.GetType(
64             ),
65                             exception.InnerException.Message);
66         _log.Error("Exception details:", exception);
67         if (exception.InnerException != null)
68             _log.Error("Exception details for inner:", exception.InnerException);
69     }

```

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

- TriDevs.TriEngine/Helpers/[Exceptions.cs](#)

## 7.11 TriDevs.TriEngine.Extensions.FloatExtensions Class Reference

[Extensions](#) for System.Single.

### Static Public Member Functions

- static float [Clamp](#) (this float value, float min, float max)  
*Clamps the specified float between a minimum and maximum value.*

#### 7.11.1 Detailed Description

[Extensions](#) for System.Single.

#### 7.11.2 Member Function Documentation

**7.11.2.1** static float TriDevs.TriEngine.Extensions.FloatExtensions.Clamp ( this float value, float min, float max )  
[static]

Clamps the specified float between a minimum and maximum value.

##### Parameters

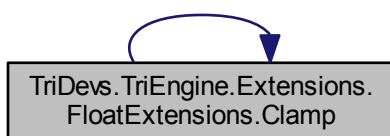
<i>value</i>	Value to clamp.
<i>min</i>	If the specified float is below this value, then this will be returned.
<i>max</i>	If the specified float is above this value, then this will be returned.

### Returns

The clamped value of the float.

```
39         {  
40             return Helpers.Math.Clamp(value, min, max);  
41         }
```

Here is the call graph for this function:



Here is the caller graph for this function:



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

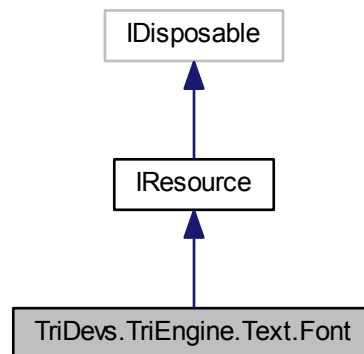
- TriDevs.TriEngine/Extensions/[FloatExtensions.cs](#)

## 7.12 TriDevs.TriEngine.Text.Font Class Reference

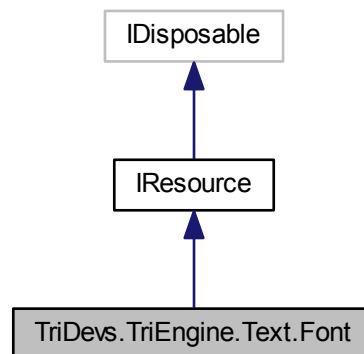
Holds a specific font type.



Inheritance diagram for TriDevs.TriEngine.Text.Font:



Collaboration diagram for TriDevs.TriEngine.Text.Font:



### Public Member Functions

- [Font](#) (string name, string file, int size, bool dropShadow=false, [FontType](#) type=FontType.TTF)  
*Initializes a new [Font](#) instance.*
- [Font](#) (string name, string file, int size, [FontType](#) type, [FontConstructionConfig](#) fontConstructionConfig)  
*Initializes a new [Font](#) instance using the specified builder configuration.*
- override string [ToString](#) ()
- void [Dispose](#) ()  
*Disposes of this font instance, releasing all unmanaged resources.*

### Static Public Member Functions

- static string [GetDefaultName](#) (string file, int size)

Returns an auto-generated font name based on the file name and font size.

## Properties

- string **Name** [get]  
Gets the name of this font instance.
- string **File** [get]  
Gets the file used to create this font instance.
- int **Size** [get]  
Gets the size of this font in points.
- **FontType** **Type** [get]  
Gets the font type.
- QFont **QFont** [get]  
Gets the QFont instance associated with this font.
- **Color** **Color** [get, set]  
Gets or sets the color of this font.

### 7.12.1 Detailed Description

Holds a specific font type.

### 7.12.2 Constructor & Destructor Documentation

**7.12.2.1** `TriDevs.TriEngine.Text.Font.Font ( string name, string file, int size, bool dropShadow = false, FontType type = FontType.TTF )`

Initializes a new **Font** instance.

#### Parameters

<i>name</i>	Name to use for identifying this font, must be unique. Can be set to null to allow the constructor to auto-generate a name for the font.
<i>file</i>	Path to the font file (TTF or qfont).
<i>size</i>	Size (in points) to use for this font.
<i>dropShadow</i>	Whether or not this font should have shadows.
<i>type</i>	The type of font. This will be detected by the file extension, but can be manually specified to control the fallback type used if one was not detected from the file name

```

94         : this(
95             name, file, size, type,
96             new FontConstructionConfig(new QFontBuilderConfiguration(dropShadow),
97                                     new QFontLoaderConfiguration(dropShadow)))
98     {
99
100    }
```

**7.12.2.2** `TriDevs.TriEngine.Text.Font.Font ( string name, string file, int size, FontType type, FontConstructionConfig fontConstructionConfig )`

Initializes a new **Font** instance using the specified builder configuration.

#### Parameters

<i>name</i>	Name to use for identifying this font, must be unique. Can be set to null to allow the constructor to auto-generate a name for the font.
-------------	--

<i>file</i>	Path to the font file (TTF or qfont).
<i>size</i>	Size (in points) to use for this font.
<i>type</i>	The type of font. This will be detected by the file extension, but can be manually specified to control the fallback type used if one was not detected from the file name
<i>font-Construction-Config</i>	The <a href="#">FontConstructionConfig</a> containing relevant font build/load configurations.

```

118     {
119         _file = file;
120         _size = size;
121
122         var ext = Path.GetExtension(_file);
123
124         if (string.IsNullOrEmpty(ext))
125             throw new EngineException("Failed to get file extension of font file!",
126                                     new IOException("System.IO.Path.GetExtension returned null or
empty for specified file."));
127
128         // Disable resharper warning, we are checking for null, resharper doesn't like IsNullOrEmpty
129         // ReSharper disable PossibleNullReferenceException
130         ext = ext.TrimStart('.').ToLower();
131         // ReSharper restore PossibleNullReferenceException
132
133         switch (ext)
134         {
135             case "ttf":
136                 type = FontType.TTF;
137                 break;
138             case "qfont":
139                 type = FontType.QFont;
140                 break;
141         }
142
143         _type = type;
144
145         QFont font;
146
147         switch (_type)
148         {
149             case FontType.TTF:
150                 if (fontConstructionConfig.BuildConfig == null)
151                     throw new EngineException("Builder configuration was null but requested font type
requires a builder config!",
152                                             new ArgumentException("BuildConfig was null.", "
fontConstructionConfig"));
153                 font = new QFont(_file, _size, fontConstructionConfig.BuildConfig);
154                 break;
155             case FontType.QFont:
156                 if (fontConstructionConfig.LoadConfig == null)
157                     throw new EngineException("Loader configuration was null but requested font type
requires a loader config!",
158                                             new ArgumentException("LoadConfig was null.", "
fontConstructionConfig"));
159                 font = QFont.FromQFontFile(_file, fontConstructionConfig.LoadConfig);
160                 break;
161             default:
162                 throw new EngineException("Unsupported font type: " + _type,
163                                         new ArgumentException("Font type unsupported.", "type"));
164         }
165
166         if (font == null)
167             throw new EngineException("Font failed to initialize!",
168                                     new ArgumentException("Font failed to initialize.", "file"));
169
170         _qfont = font;
171
172         _name = name ?? GetDefaultName(file, size);
173     }

```

## 7.12.3 Member Function Documentation

### 7.12.3.1 void TriDevs.TriEngine.Text.Font.Dispose ( )

Disposes of this font instance, releasing all unmanaged resources.

```

196     {

```

```

197         QFont.Dispose();
198     }

```

#### 7.12.3.2 static string TriDevs.TriEngine.Text.Font.GetDefaultName ( string *file*, int *size* ) [static]

Returns an auto-generated font name based on the file name and font size.

##### Parameters

<i>file</i>	The file name.
<i>size</i>	The size (in points) of the font.

##### Returns

The auto-generated font name.

```

182     {
183         return string.Format(NameFormat, Path.GetFileNameWithoutExtension(file), size);
184     }

```

#### 7.12.3.3 override string TriDevs.TriEngine.Text.Font.ToString ( )

```

187     {
188         return Name;
189     }

```

### 7.12.4 Property Documentation

#### 7.12.4.1 Color TriDevs.TriEngine.Text.Font.Color [get], [set]

Gets or sets the color of this font.

#### 7.12.4.2 string TriDevs.TriEngine.Text.Font.File [get]

Gets the file used to create this font instance.

#### 7.12.4.3 string TriDevs.TriEngine.Text.Font.Name [get]

Gets the name of this font instance.

#### 7.12.4.4 QFont TriDevs.TriEngine.Text.Font.QFont [get]

Gets the QFont instance associated with this font.

#### 7.12.4.5 int TriDevs.TriEngine.Text.Font.Size [get]

Gets the size of this font in points.

#### 7.12.4.6 FontType TriDevs.TriEngine.Text.Font.Type [get]

Gets the font type.

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

- TriDevs.TriEngine/Text/[Font.cs](#)

## 7.13 TriDevs.TriEngine.Text.FontConstructionConfig Class Reference

Container class for different QFont configurations for use with the [Font](#) constructor.

### Public Member Functions

- [FontConstructionConfig](#) (QFontLoaderConfiguration loadConfig)  
*Initialize a new [FontConstructionConfig](#) with a load configuration.*
- [FontConstructionConfig](#) (QFontBuilderConfiguration buildConfig, QFontLoaderConfiguration loadConfig=null)  
*Initializes a new [FontConstructionConfig](#) with a builder configuration and optional loader configuration.*

### Public Attributes

- QFontBuilderConfiguration [BuildConfig](#)  
*The builder configuration used when constructing fonts from non-qfont files.*
- QFontLoaderConfiguration [LoadConfig](#)  
*The loader configuration used when contruscting fonts from a qfont file.*

#### 7.13.1 Detailed Description

Container class for different QFont configurations for use with the [Font](#) constructor.

#### 7.13.2 Constructor & Destructor Documentation

##### 7.13.2.1 TriDevs.TriEngine.Text.FontConstructionConfig.FontConstructionConfig ( QFontLoaderConfiguration loadConfig )

Initialize a new [FontConstructionConfig](#) with a load configuration.

##### Parameters

<i>loadConfig</i>	The QFontLoaderConfiguration to use.
-------------------	--------------------------------------

```

50         : this(null, loadConfig)
51     {
52
53     }
```

##### 7.13.2.2 TriDevs.TriEngine.Text.FontConstructionConfig.FontConstructionConfig ( QFontBuilderConfiguration buildConfig, QFontLoaderConfiguration loadConfig = null )

Initializes a new [FontConstructionConfig](#) with a builder configuration and optional loader configuration.

##### Parameters

<i>buildConfig</i>	The builder configuration to use.
<i>loadConfig</i>	Optional loader configuration to set.

```

62     {
63         BuildConfig = buildConfig;
64         LoadConfig = loadConfig;
65     }
```

### 7.13.3 Member Data Documentation

#### 7.13.3.1 QFontBuilderConfiguration TriDevs.TriEngine.Text.FontConstructionConfig.BuildConfig

The builder configuration used when constructing fonts from non-qfont files.

#### 7.13.3.2 QFontLoaderConfiguration TriDevs.TriEngine.Text.FontConstructionConfig.LoadConfig

The loader configuration used when contruscting fonts from a qfont file.

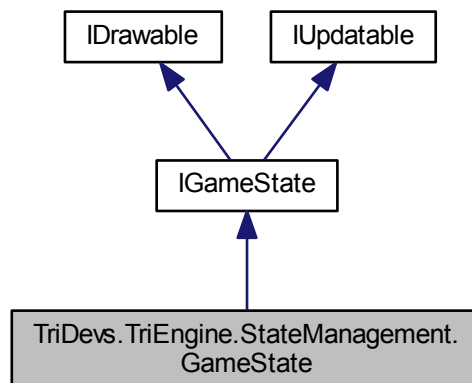
The documentation for this class was generated from the following file:

- TriDevs.TriEngine/Text/[FontConstructionConfig.cs](#)

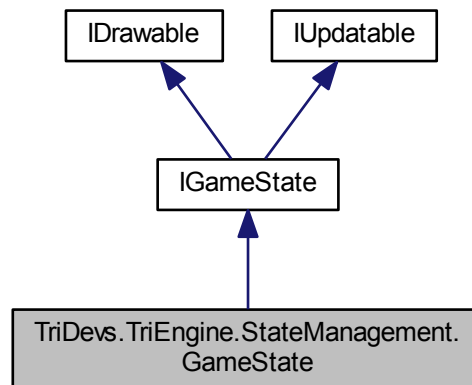
## 7.14 TriDevs.TriEngine.StateManagement.GameState Class Reference

Base [GameState](#) class that all other game states derive from, defines basic [GameState](#) behaviour.

Inheritance diagram for TriDevs.TriEngine.StateManagement.GameState:



Collaboration diagram for TriDevs.TriEngine.StateManagement.GameState:



## Public Member Functions

- virtual void [Update](#) ()  
*Updates the object.*
- virtual void [Draw](#) ()  
*Draw the object to screen.*
- virtual void [Load](#) ()  
*Loads resources associated with this game component.*
- virtual void [Unload](#) ()  
*Unloads resources that were loaded in the [Load](#) method.*
- virtual void [Pause](#) ()  
*Pauses the game state, preventing update calls from running.*
- virtual void [Unpause](#) ()  
*Unpauses the game state, enabling update calls again.*
- [IGameComponent AddComponent](#) ([IGameComponent](#) component)  
*Adds a game component to this game state.*
- void [RemoveComponent](#) ([IGameComponent](#) component)  
*Removes the specified component from this game state.*
- void [RemoveAllComponents](#) ()  
*Removes all components from the game state.*
- void [RemoveAllComponents](#) (Type type)  
*Removes all components of the specified type from the game state.*
- void [RemoveAllComponents](#) (Predicate< [IGameComponent](#) > predicate)  
*Removes all components that match the supplied predicate function.*
- bool [HasComponent](#) ([IGameComponent](#) component)  
*Returns whether this game state contains the specified IGameComponent.*
- bool [HasComponent](#) (Type type)  
*Returns whether this game state contains a specific type of component.*
- bool [HasComponent](#) (Func< [IGameComponent](#), bool > func)  
*Returns whether this game state contains a component that matches the supplied predicate.*

- `IEnumerable< IGameComponent > GetAllComponents ()`  
*Returns a read-only collection of all components in this game state.*
- `IGameComponent GetComponent (Type type)`  
*Returns the specified component type if it exists in this game state.*
- `IEnumerable< IGameComponent > GetAllComponents (Type type)`  
*Returns all components of the specified type.*
- `IGameComponent GetComponent (Func< IGameComponent, bool > func)`  
*Returns the first component that matches the supplied predicate function.*
- `IEnumerable< IGameComponent > GetAllComponents (Func< IGameComponent, bool > func)`  
*Returns all components that matches the supplied predicate function.*

## Protected Member Functions

- `GameState ()`

## Protected Attributes

- readonly `List< IGameComponent > Components`  
*List of components currently added to this [GameState](#).*

## Properties

- bool `Paused` [get, set]

### 7.14.1 Detailed Description

Base [GameState](#) class that all other game states derive from, defines basic [GameState](#) behaviour.

### 7.14.2 Constructor & Destructor Documentation

#### 7.14.2.1 `TriDevs.TriEngine.StateManagement.GameState.GameState ( )` [protected]

```

45         {
46             Components = new List<IGameComponent> ();
47         }

```

### 7.14.3 Member Function Documentation

#### 7.14.3.1 `IGameComponent TriDevs.TriEngine.StateManagement.GameState.AddComponent ( IGameComponent component )`

Adds a game component to this game state.

#### Parameters

<code>component</code>	The component to add.
------------------------	-----------------------

#### Returns

The component that was added.

Implements [TriDevs.TriEngine.StateManagement.IGameState](#).

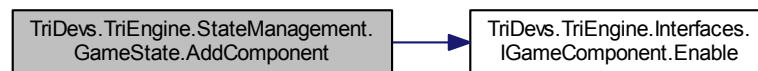


```

81     {
82         if (HasComponent(component))
83             throw new InvalidOperationException("Cannot add the same component more than once.");
84
85         Components.Add(component);
86         component.Enable();
87         return component; // var comp = someState.AddComponent(new SomeComponent());
88     }

```

Here is the call graph for this function:



#### 7.14.3.2 virtual void TriDevs.TriEngine.StateManagement.GameState.Draw ( ) [virtual]

Draw the object to screen.

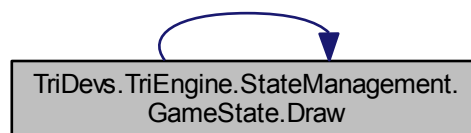
Implements [TriDevs.TriEngine.Interfaces.IDrawable](#).

```

55     {
56         foreach (var component in Components.OfType<
57             IDrawableGameComponent>())
58             component.Draw();
59     }

```

Here is the call graph for this function:



Here is the caller graph for this function:



#### 7.14.3.3 IEnumerable<IGameComponent> TriDevs.TriEngine.StateManagement.GameState.GetAllComponents ( )

Returns a read-only collection of all components in this game state.

##### Returns

Read-only collection of components.

Implements [TriDevs.TriEngine.StateManagement.IGameState](#).

```
137     {
138         return Components.AsReadOnly();
139     }
```

#### 7.14.3.4 IEnumerable<IGameComponent> TriDevs.TriEngine.StateManagement.GameState.GetAllComponents ( Type type )

Returns all components of the specified type.

##### Parameters

<i>type</i>	The type of game component requested.
-------------	---------------------------------------

##### Returns

A collection of all components of matching type.

Implements [TriDevs.TriEngine.StateManagement.IGameState](#).

```
147     {
148         return Components.FindAll(c => c.GetType() == type);
149     }
```

#### 7.14.3.5 IEnumerable<IGameComponent> TriDevs.TriEngine.StateManagement.GameState.GetAllComponents ( Func<IGameComponent, bool> func )

Returns all components that matches the supplied predicate function.

##### Parameters

<i>func</i>	The predicate function.
-------------	-------------------------

##### Returns

Collection of all matching components, empty collection if no matches were found.

Implements [TriDevs.TriEngine.StateManagement.IGameState](#).

```
157     {
158         return Components.Where(func);
159     }
```

#### 7.14.3.6 IGameComponent TriDevs.TriEngine.StateManagement.GameState.GetComponent ( Type type )

Returns the specified component type if it exists in this game state.

## Parameters

<i>type</i>	The component type to get.
-------------	----------------------------

## Returns

The component object, or null if it's not added to this game state.

Implements [TriDevs.TriEngine.StateManagement.IGameState](#).

```

142     {
143         return Components.FirstOrDefault(c => c.GetType() == type);
144     }
```

### 7.14.3.7 IGameComponent TriDevs.TriEngine.StateManagement.GameState.GetComponent ( Func< IGameComponent, bool > func )

Returns the first component that matches the supplied predicate function.

## Parameters

<i>func</i>	The predicate function.
-------------	-------------------------

## Returns

Component that matches the predicate, null if no matches were found.

Implements [TriDevs.TriEngine.StateManagement.IGameState](#).

```

152     {
153         return Components.FirstOrDefault(func);
154     }
```

### 7.14.3.8 bool TriDevs.TriEngine.StateManagement.GameState.HasComponent ( IGameComponent component )

Returns whether this game state contains the specified IGameComponent.

## Parameters

<i>component</i>	The component to check for.
------------------	-----------------------------

## Returns

True if the component has been added to this game state, false otherwise.

Implements [TriDevs.TriEngine.StateManagement.IGameState](#).

```

122     {
123         return Components.Contains(component);
124     }
```

### 7.14.3.9 bool TriDevs.TriEngine.StateManagement.GameState.HasComponent ( Type type )

Returns whether this game state contains a specific type of component.

**Parameters**

<i>type</i>	The type to check for.
-------------	------------------------

**Returns**

True if the type of component has been added to this game state, false otherwise.

Implements [TriDevs.TriEngine.StateManagement.IGameState](#).

```

127     {
128         return Components.Any(c => c.GetType() == type);
129     }
```

#### 7.14.3.10 bool TriDevs.TriEngine.StateManagement.GameState.HasComponent ( Func< IGameComponent, bool > func )

Returns whether this game state contains a component that matches the supplied predicate.

**Parameters**

<i>func</i>	Predicate function to use for search.
-------------	---------------------------------------

**Returns**

True if the game state contains a matching component, false otherwise.

Implements [TriDevs.TriEngine.StateManagement.IGameState](#).

```

132     {
133         return Components.Any(func);
134     }
```

#### 7.14.3.11 virtual void TriDevs.TriEngine.StateManagement.GameState.Load ( ) [virtual]

Loads resources associated with this game component.

Implements [TriDevs.TriEngine.StateManagement.IGameState](#).

```

61     {
62
63     }
```

#### 7.14.3.12 virtual void TriDevs.TriEngine.StateManagement.GameState.Pause ( ) [virtual]

Pauses the game state, preventing update calls from running.

Implements [TriDevs.TriEngine.StateManagement.IGameState](#).

```

71     {
72         Paused = true;
73     }
```

#### 7.14.3.13 void TriDevs.TriEngine.StateManagement.GameState.RemoveAllComponents ( )

Removes all components from the game state.

Implements [TriDevs.TriEngine.StateManagement.IGameState](#).

```

101     {
102         Components.ForEach(c => c.Disable());
103         Components.Clear();
104     }

```

#### 7.14.3.14 void TriDevs.TriEngine.StateManagement.GameState.RemoveAllComponents ( Type *type* )

Removes all components of the specified type from the game state.

##### Parameters

<i>type</i>	The type of component to remove.
-------------	----------------------------------

Implements [TriDevs.TriEngine.StateManagement.IGameState](#).

```

107     {
108         RemoveAllComponents(c => c.GetType() == type);
109     }

```

#### 7.14.3.15 void TriDevs.TriEngine.StateManagement.GameState.RemoveAllComponents ( Predicate< IGameComponent > *predicate* )

Removes all components that match the supplied predicate function.

##### Parameters

<i>predicate</i>	The predicate function.
------------------	-------------------------

Implements [TriDevs.TriEngine.StateManagement.IGameState](#).

```

112     {
113         var removed = Components.FindAll(predicate);
114         if (removed.Count < 1)
115             return;
116
117         Components.RemoveAll(predicate);
118         removed.ForEach(c => c.Disable());
119     }

```

#### 7.14.3.16 void TriDevs.TriEngine.StateManagement.GameState.RemoveComponent ( IGameComponent *component* )

Removes the specified component from this game state.

##### Parameters

<i>component</i>	
------------------	--

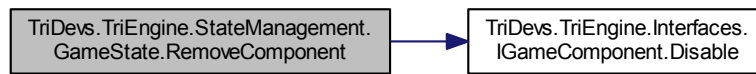
Implements [TriDevs.TriEngine.StateManagement.IGameState](#).

```

91     {
92         var match = Components.FirstOrDefault(c => c == component);
93         if (match == null)
94             return;
95
96         Components.Remove(match);
97         match.Disable();
98     }

```

Here is the call graph for this function:



#### 7.14.3.17 virtual void TriDevs.TriEngine.StateManagement.GameState.Unload ( ) [virtual]

Unloads resources that were loaded in the [Load](#) method.

Implements [TriDevs.TriEngine.StateManagement.IGameState](#).

```

66      {
67
68      }
```

#### 7.14.3.18 virtual void TriDevs.TriEngine.StateManagement.GameState.Unpause ( ) [virtual]

Unpauses the game state, enabling update calls again.

Implements [TriDevs.TriEngine.StateManagement.IGameState](#).

```

76      {
77          Paused = false;
78      }
```

#### 7.14.3.19 virtual void TriDevs.TriEngine.StateManagement.GameState.Update ( ) [virtual]

Updates the object.

Implements [TriDevs.TriEngine.Interfaces.IUpdatable](#).

```

50      {
51          Components.ForEach(c => c.Update());
52      }
```

### 7.14.4 Member Data Documentation

#### 7.14.4.1 readonly List<IGameComponent> TriDevs.TriEngine.StateManagement.GameState.Components [protected]

List of components currently added to this [GameState](#).

### 7.14.5 Property Documentation

#### 7.14.5.1 bool TriDevs.TriEngine.StateManagement.GameState.Paused [get], [set]

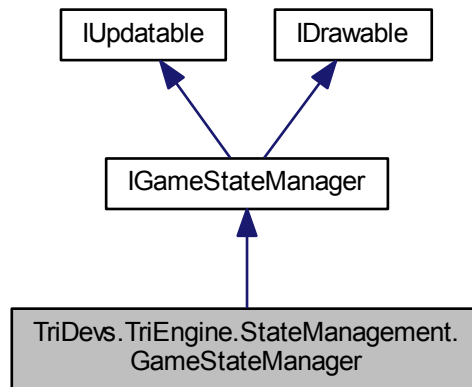
The documentation for this class was generated from the following file:

- [TriDevs.TriEngine/StateManagement/GameState.cs](#)

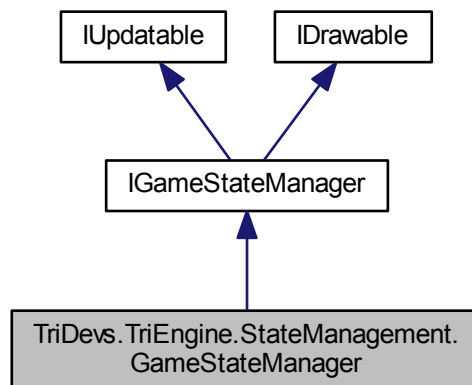
## 7.15 TriDevs.TriEngine.StateManagement.GameStateManager Class Reference

Game state manager that keeps track of the active game states and provides methods to control the states.

Inheritance diagram for TriDevs.TriEngine.StateManagement.GameStateManager:



Collaboration diagram for TriDevs.TriEngine.StateManagement.GameStateManager:



### Public Member Functions

- [GameStateManager](#) ()  
*Creates a new [GameStateManager](#) with an empty state stack.*
- [GameStateManager](#) (IGameState state)  
*Creates a new [GameStateManager](#) and pushes an initial state onto the stack.*
- void [Update](#) ()

- Updates the object.*
- void [Draw](#) ()  
*Draw the object to screen.*
- [IGameState Push](#) ([IGameState](#) state)  
*Pushes a new game state onto the stack, pausing the current one.*
- [IGameState Pop](#) ()  
*Pops the currently active state from the stack, unpausing the previous one.*
- [IGameState Peek](#) ()  
*Returns the game state at the top of the stack, without popping it.*
- [IGameState Switch](#) ([IGameState](#) state)  
*Switches to a new game state, discarding all previous ones in the stack.*

## Properties

- int [StateCount](#) [get]
- [IGameState ActiveState](#) [get, set]

## 7.15.1 Detailed Description

Game state manager that keeps track of the active game states and provides methods to control the states.

## 7.15.2 Constructor & Destructor Documentation

### 7.15.2.1 [TriDevs.TriEngine.StateManagement.GameStateManager.GameStateManager](#) ( )

Creates a new [GameStateManager](#) with an empty state stack.

```

52         {
53             _states = new Stack<IGameState>();
54         }
```

### 7.15.2.2 [TriDevs.TriEngine.StateManagement.GameStateManager.GameStateManager](#) ( [IGameState](#) state )

Creates a new [GameStateManager](#) and pushes an initial state onto the stack.

#### Parameters

<i>state</i>	The state to initialize with.
--------------	-------------------------------

```

61                                     : this()
62         {
63             Push(state);
64         }
```

## 7.15.3 Member Function Documentation

### 7.15.3.1 void [TriDevs.TriEngine.StateManagement.GameStateManager.Draw](#) ( )

Draw the object to screen.

Implements [TriDevs.TriEngine.Interfaces.IDrawable](#).

```

73         {
74             if (ActiveState != null)
75                 ActiveState.Draw();
```



```
76      }
```

### 7.15.3.2 IGameState TriDevs.TriEngine.StateManagement.GameStateManager.Peek ( )

Returns the game state at the top of the stack, without popping it.

#### Returns

The state at the top of the stack.

Implements [TriDevs.TriEngine.StateManagement.IGameStateManager](#).

```
102      {
103          return _states.Peek();
104      }
```

### 7.15.3.3 IGameState TriDevs.TriEngine.StateManagement.GameStateManager.Pop ( )

Pops the currently active state from the stack, unpausing the previous one.

#### Returns

The state that was popped.

Implements [TriDevs.TriEngine.StateManagement.IGameStateManager](#).

```
90      {
91          if (_states.Count == 1)
92              throw new InvalidOperationException("Cannot pop the last remaining game state from stack.");
93      };
94      var state = _states.Pop();
95      state.Unload();
96      ActiveState = _states.Peek();
97      ActiveState.Unpause();
98      return state;
99      }
```

### 7.15.3.4 IGameState TriDevs.TriEngine.StateManagement.GameStateManager.Push ( IGameState state )

Pushes a new game state onto the stack, pausing the current one.

#### Parameters

<i>state</i>	The new game state to push onto the stack.
--------------	--

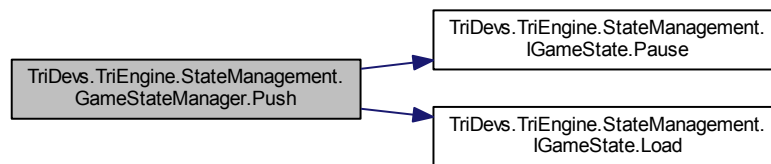
#### Returns

The game state that was pushed.

Implements [TriDevs.TriEngine.StateManagement.IGameStateManager](#).

```
79      {
80          if (ActiveState != null)
81              ActiveState.Pause();
82          state.Load();
83          _states.Push(state);
84          ActiveState = state;
85          return state;
86      }
```

Here is the call graph for this function:



#### 7.15.3.5 IGameState TriDevs.TriEngine.StateManagement.GameStateManager.Switch ( IGameState state )

Switches to a new game state, discarding all previous ones in the stack.

##### Parameters

<i>state</i>	The new state to switch to.
--------------	-----------------------------

##### Returns

The state that was switched to.

Implements [TriDevs.TriEngine.StateManagement.IGameStateManager](#).

```

107     {
108         while (_states.Count > 0)
109             _states.Pop().Unload();
110
111         _states = new Stack<IGameState>();
112         return Push(state);
113     }
  
```

Here is the call graph for this function:



#### 7.15.3.6 void TriDevs.TriEngine.StateManagement.GameStateManager.Update ( )

Updates the object.

Implements [TriDevs.TriEngine.Interfaces.IUpdatable](#).

```

67     {
68         if (ActiveState != null)
69             ActiveState.Update();
70     }
  
```

### 7.15.4 Property Documentation

7.15.4.1 **IGameState** `TriDevs.TriEngine.StateManagement.GameStateManager.ActiveState` `[get]`, `[set]`

7.15.4.2 **int** `TriDevs.TriEngine.StateManagement.GameStateManager.StateCount` `[get]`

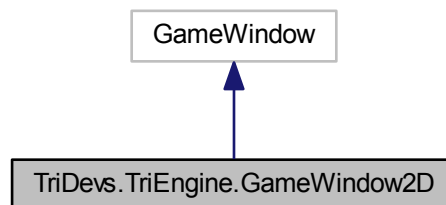
The documentation for this class was generated from the following file:

- `TriDevs.TriEngine/StateManagement/GameStateManager.cs`

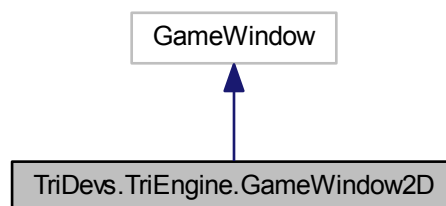
## 7.16 TriDevs.TriEngine.GameWindow2D Class Reference

Game window class specialized for drawing 2D graphics.

Inheritance diagram for `TriDevs.TriEngine.GameWindow2D`:



Collaboration diagram for `TriDevs.TriEngine.GameWindow2D`:



### Protected Member Functions

- `GameWindow2D` (int width, int height, string title, bool vsync=true)
- override void `OnResize` (EventArgs e)
- override void `OnLoad` (EventArgs e)
- sealed override void `OnRenderFrame` (FrameEventArgs e)
- abstract void `OnDraw` (FrameEventArgs e)
- void `Initialize` ()

## Properties

- [Color ClearColor](#) [get, set]  
*Gets or sets the clear color for this window.*

### 7.16.1 Detailed Description

Game window class specialized for drawing 2D graphics.

### 7.16.2 Constructor & Destructor Documentation

#### 7.16.2.1 `TriDevs.TriEngine.GameWindow2D.GameWindow2D ( int width, int height, string title, bool vsync = true )` [protected]

```

52         : base(width, height, GraphicsMode.Default, title)
53     {
54         VSync = vsync ? VSyncMode.On : VSyncMode.Off;
55
56         ClearColor = Color.CornflowerBlue;
57     }

```

### 7.16.3 Member Function Documentation

#### 7.16.3.1 `void TriDevs.TriEngine.GameWindow2D.Initialize ( )` [protected]

```

95     {
96         try
97         {
98             GL.Viewport(0, 0, ClientRectangle.Width, ClientRectangle.Height);
99
100             GL.MatrixMode(MatrixMode.Projection);
101
102             GL.LoadIdentity();
103
104             GL.Ortho(0, ClientRectangle.Width - 1, ClientRectangle.Height - 1, 0, -1, 1);
105
106             GL.MatrixMode(MatrixMode.Modelview);
107
108             // Disable the Z-buffer, this is a 2D game window
109             //GL.Disable(EnableCap.DepthTest);
110             //GL.Disable(EnableCap.CullFace);
111             GL.Enable(EnableCap.Blend);
112             GL.BlendEquation(BlendEquationMode.FuncAdd);
113             GL.BlendFunc(BlendingFactorSrc.SrcAlpha, BlendingFactorDest.OneMinusSrcAlpha);
114             //GL.PixelStore(PixelStoreParameter.UnpackAlignment, 1);
115             //GL.RenderMode(RenderingMode.Render);
116
117             GL.ClearColor(ClearColor.ToColor4());
118         }
119         catch (Exception ex)
120         {
121             Helpers.Exceptions.Throw(ex, "Initialization of 2D game window failed during OpenGL setup!");
122         }
123     }

```

#### 7.16.3.2 `abstract void TriDevs.TriEngine.GameWindow2D.OnDraw ( FrameEventArgs e )` [protected], [pure virtual]

#### 7.16.3.3 `override void TriDevs.TriEngine.GameWindow2D.OnLoad ( EventArgs e )` [protected]

```

75     {
76         base.OnLoad(e);
77
78         Initialize();
79     }

```

**7.16.3.4 sealed override void TriDevs.TriEngine.GameWindow2D.OnRenderFrame ( FrameEventArgs e ) [protected]**

```

82      {
83          base.OnRenderFrame(e);
84
85          GL.Clear(ClearBufferMask.ColorBufferBit);
86
87          OnDraw(e);
88
89          SwapBuffers();
90      }

```

**7.16.3.5 override void TriDevs.TriEngine.GameWindow2D.OnResize ( EventArgs e ) [protected]**

```

60      {
61          base.OnResize(e);
62
63          GL.Viewport(0, 0, ClientRectangle.Width, ClientRectangle.Height);
64
65          GL.MatrixMode(MatrixMode.Projection);
66
67          GL.LoadIdentity();
68
69          GL.Ortho(0, ClientRectangle.Width - 1, ClientRectangle.Height - 1, 0, -1, 1);
70
71          GL.MatrixMode(MatrixMode.Modelview);
72      }

```

**7.16.4 Property Documentation****7.16.4.1 Color TriDevs.TriEngine.GameWindow2D.ClearColor [get],[set],[protected]**

Gets or sets the clear color for this window.

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

- TriDevs.TriEngine/[GameWindow2D.cs](#)

**7.17 TriDevs.TriEngine.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.*

**7.17.1 Detailed Description**

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

## 7.17.2 Member Function Documentation

### 7.17.2.1 static string TriDevs.TriEngine.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.

```

73     {
74 #if WINDOWS
75     return GetWin32Exception(err).Message;
76 #else
77     throw new System.NotImplementedException();
78 #endif
79     }
```

### 7.17.2.2 static int TriDevs.TriEngine.Native.Helpers.GetLastError ( ) [static]

Gets the last error that was thrown.

#### Returns

The error code associated with the thrown error.

```

43     {
44 #if WINDOWS
45     return Marshal.GetLastWin32Error();
46 #else
47     throw new System.NotImplementedException();
48 #endif
49     }
```

### 7.17.2.3 static int TriDevs.TriEngine.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.

```

57     {
58         var err = GetLastError();
59 #if WINDOWS
60         message = GetWin32Exception(err).Message;
61 #else
62         throw new System.NotImplementedException();
63 #endif
64         return err;
65     }
```

### 7.17.2.4 static string TriDevs.TriEngine.Native.Helpers.GetLastErrorMessage ( ) [static]

Gets the error message associated with the last thrown error.

**Returns**

The error message.

```
86         {  
87             return GetErrorMessage(GetLastError());  
88         }
```

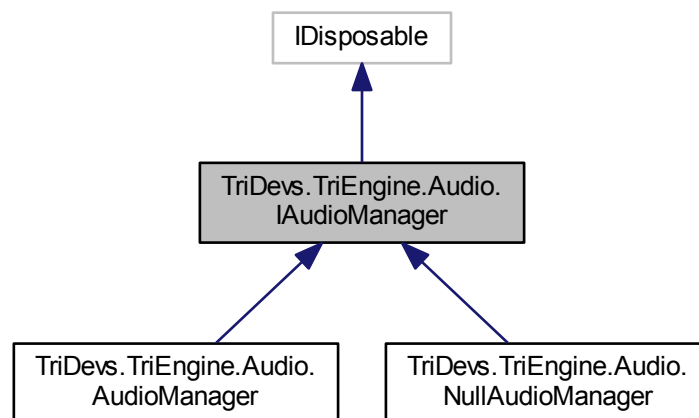
The documentation for this class was generated from the following file:

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

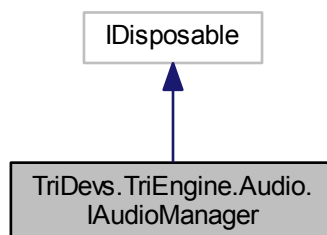
## 7.18 TriDevs.TriEngine.Audio.IAudioManager Interface Reference

Provides various methods to manipulate audio.

Inheritance diagram for TriDevs.TriEngine.Audio.IAudioManager:



Collaboration diagram for TriDevs.TriEngine.Audio.IAudioManager:



## Public Member Functions

- void [StopAll](#) ()

*Immediately stops all playback of sounds and songs.*

- void [StopAllSounds](#) ()

*Immediately stops playback of all sounds.*

- void [StopAllSongs](#) ()

*Immediately stops playback of all songs.*

### 7.18.1 Detailed Description

Provides various methods to manipulate audio.

### 7.18.2 Member Function Documentation

#### 7.18.2.1 void [TriDevs.TriEngine.Audio.IAudioManager.StopAll](#) ( )

Immediately stops all playback of sounds and songs.

Implemented in [TriDevs.TriEngine.Audio.AudioManager](#), and [TriDevs.TriEngine.Audio.NullAudioManager](#).

#### 7.18.2.2 void [TriDevs.TriEngine.Audio.IAudioManager.StopAllSongs](#) ( )

Immediately stops playback of all songs.

Implemented in [TriDevs.TriEngine.Audio.AudioManager](#), and [TriDevs.TriEngine.Audio.NullAudioManager](#).

#### 7.18.2.3 void [TriDevs.TriEngine.Audio.IAudioManager.StopAllSounds](#) ( )

Immediately stops playback of all sounds.

Implemented in [TriDevs.TriEngine.Audio.AudioManager](#), and [TriDevs.TriEngine.Audio.NullAudioManager](#).

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

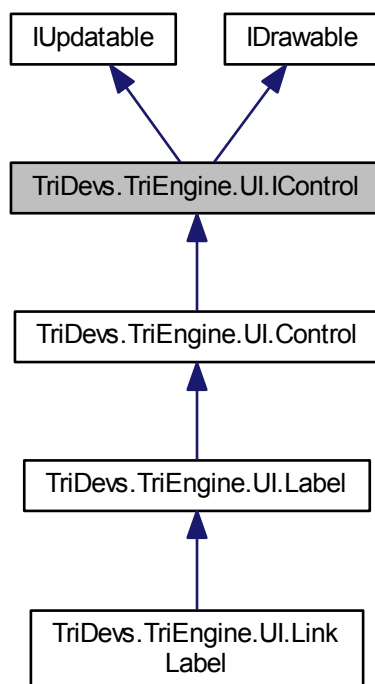
- [TriDevs.TriEngine/Audio/IAudioManager.cs](#)

## 7.19 TriDevs.TriEngine.UI.IControl Interface Reference

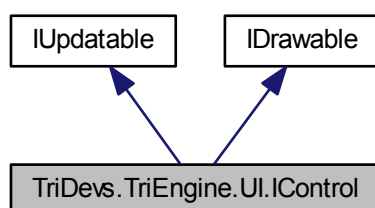
A [UI](#) control that can be drawn on screen and interacted with.



Inheritance diagram for TriDevs.TriEngine.UI.IControl:



Collaboration diagram for TriDevs.TriEngine.UI.IControl:



## Public Member Functions

- void `Enable` ()  
*Enables the control.*
- void `Disable` ()  
*Disables the control.*
- void `Show` ()

*Shows the control.*

- void [Hide](#) ()

*Hides the control.*

## Properties

- bool [Enabled](#) [get, set]

*Gets or sets a value indicating whether this control can be interacted with.*

- bool [Visible](#) [get, set]

*Gets or sets a value indicating whether this control should be drawn to the screen.*

- [Color](#) [Color](#) [get, set]

*The color of this control.*

- Point< int > [Position](#) [get, set]

*The position of this control, in pixel coordinates.*

- Point< int > [Size](#) [get, set]

*Gets or sets the size of this control, in pixels.*

- [Rectangle](#) [Rectangle](#) [get, set]

*Gets the rectangular area on the screen that this control is occupying.*

- string [Text](#) [get, set]

*Gets or sets the text associated with this control.*

## Events

- ControlClickedEventHandler [Clicked](#)

*Raised when this control is clicked on by the user.*

## 7.19.1 Detailed Description

A [UI](#) control that can be drawn on screen and interacted with.

## 7.19.2 Member Function Documentation

### 7.19.2.1 void TriDevs.TriEngine.UI.IControl.Disable ( )

Disables the control.

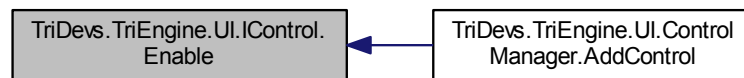
Implemented in [TriDevs.TriEngine.UI.Control](#).

### 7.19.2.2 void TriDevs.TriEngine.UI.IControl.Enable ( )

Enables the control.

Implemented in [TriDevs.TriEngine.UI.Control](#).

Here is the caller graph for this function:

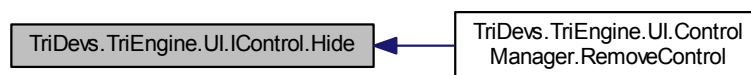


#### 7.19.2.3 void TriDevs.TriEngine.UI.IControl.Hide ( )

Hides the control.

Implemented in [TriDevs.TriEngine.UI.Control](#).

Here is the caller graph for this function:

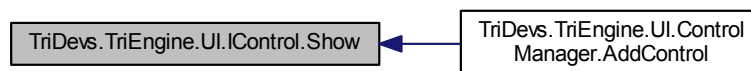


#### 7.19.2.4 void TriDevs.TriEngine.UI.IControl.Show ( )

Shows the control.

Implemented in [TriDevs.TriEngine.UI.Control](#).

Here is the caller graph for this function:



### 7.19.3 Property Documentation

#### 7.19.3.1 Color TriDevs.TriEngine.UI.IControl.Color [get], [set]

The color of this control.

#### 7.19.3.2 `bool TriDevs.TriEngine.UI.IControl.Enabled` `[get], [set]`

Gets or sets a value indicating whether this control can be interacted with.

#### 7.19.3.3 `Point<int> TriDevs.TriEngine.UI.IControl.Position` `[get], [set]`

The position of this control, in pixel coordinates.

#### 7.19.3.4 `Rectangle TriDevs.TriEngine.UI.IControl.Rectangle` `[get], [set]`

Gets the rectangular area on the screen that this control is occupying.

#### 7.19.3.5 `Point<int> TriDevs.TriEngine.UI.IControl.Size` `[get], [set]`

Gets or sets the size of this control, in pixels.

#### 7.19.3.6 `string TriDevs.TriEngine.UI.IControl.Text` `[get], [set]`

Gets or sets the text associated with this control.

#### 7.19.3.7 `bool TriDevs.TriEngine.UI.IControl.Visible` `[get], [set]`

Gets or sets a value indicating whether this control should be drawn to the screen.

### 7.19.4 Event Documentation

#### 7.19.4.1 `ControlClickedEventHandler TriDevs.TriEngine.UI.IControl.Clicked`

Raised when this control is clicked on by the user.

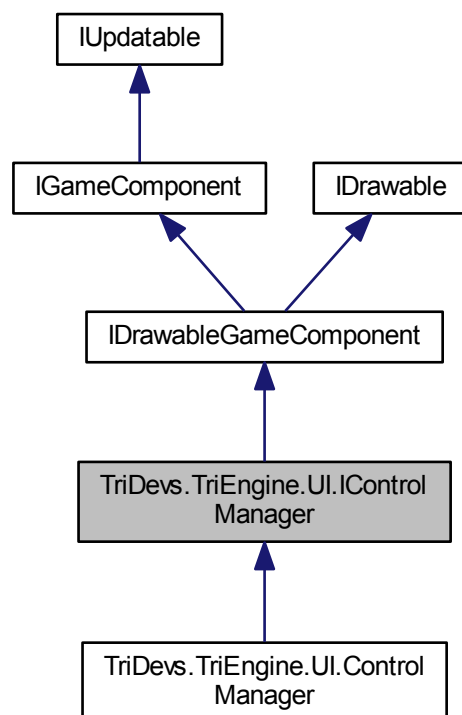
The documentation for this interface was generated from the following file:

- [TriDevs.TriEngine/UI/IControl.cs](#)

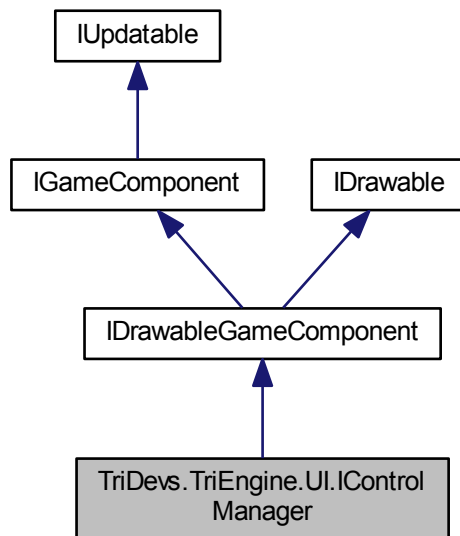
## 7.20 TriDevs.TriEngine.UI.IControlManager Interface Reference

Manages various [UI](#) controls, automatically updating and drawing them to the screen.

Inheritance diagram for TriDevs.TriEngine.UI.IControlManager:



Collaboration diagram for TriDevs.TriEngine.UI.IControlManager:



## Public Member Functions

- `IControl AddControl (IControl control)`  
*Adds a new control to this control manager.*
- `void RemoveControl (IControl control)`  
*Removes a control from this control manager.*
- `void RemoveAllControls (Type type)`  
*Removes all controls of a specific type from this control manager.*
- `void RemoveAllControls (Func< IControl, bool > func)`  
*Removes all controls matching the supplied predicate function.*
- `bool HasControl (IControl control)`  
*Returns whether this control manager contains the specified control.*
- `bool HasControl (Type type)`  
*Returns whether this control manager contains any control of the specified type.*
- `bool HasControl (Func< IControl, bool > func)`  
*Returns whether this control manager contains any control matching the supplied predicate function.*

## Additional Inherited Members

### 7.20.1 Detailed Description

Manages various [UI](#) controls, automatically updating and drawing them to the screen.

## 7.20.2 Member Function Documentation

### 7.20.2.1 IControl TriDevs.TriEngine.UI.IControlManager.AddControl ( IControl *control* )

Adds a new control to this control manager.

#### Parameters

<i>control</i>	The control to add.
----------------	---------------------

#### Returns

The control that was added.

Implemented in [TriDevs.TriEngine.UI.ControlManager](#).

### 7.20.2.2 bool TriDevs.TriEngine.UI.IControlManager.HasControl ( IControl *control* )

Returns whether this control manager contains the specified control.

#### Parameters

<i>control</i>	The control to check.
----------------	-----------------------

#### Returns

True if the specified control exists in this control manager, false otherwise.

Implemented in [TriDevs.TriEngine.UI.ControlManager](#).

### 7.20.2.3 bool TriDevs.TriEngine.UI.IControlManager.HasControl ( Type *type* )

Returns whether this control manager contains any control of the specified type.

#### Parameters

<i>type</i>	The type to check.
-------------	--------------------

#### Returns

True if this control manager contains a control of the specified type, false otherwise.

Implemented in [TriDevs.TriEngine.UI.ControlManager](#).

### 7.20.2.4 bool TriDevs.TriEngine.UI.IControlManager.HasControl ( Func< IControl, bool > *func* )

Returns whether this control manager contains any control matching the supplied predicate function.

#### Parameters

<i>func</i>	The predicate function.
-------------	-------------------------

**Returns**

True if this control manager contains a control matching the specified predicate, false otherwise.

Implemented in [TriDevs.TriEngine.UI.ControlManager](#).

**7.20.2.5 void TriDevs.TriEngine.UI.IControlManager.RemoveAllControls ( Type *type* )**

Removes all controls of a specific type from this control manager.

**Parameters**

<i>type</i>	The type of control to remove.
-------------	--------------------------------

Implemented in [TriDevs.TriEngine.UI.ControlManager](#).

**7.20.2.6 void TriDevs.TriEngine.UI.IControlManager.RemoveAllControls ( Func< IControl, bool > *func* )**

Removes all controls matching the supplied predicate function.

**Parameters**

<i>func</i>	The predicate function to use.
-------------	--------------------------------

Implemented in [TriDevs.TriEngine.UI.ControlManager](#).

**7.20.2.7 void TriDevs.TriEngine.UI.IControlManager.RemoveControl ( IControl *control* )**

Removes a control from this control manager.

**Parameters**

<i>control</i>	The control to remove.
----------------	------------------------

Implemented in [TriDevs.TriEngine.UI.ControlManager](#).

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

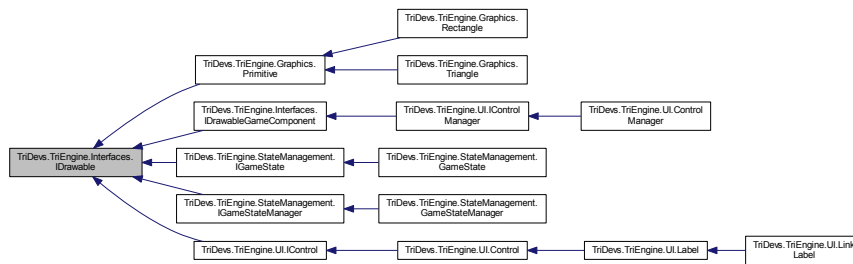
- [TriDevs.TriEngine/UI/IControlManager.cs](#)

## 7.21 TriDevs.TriEngine.Interfaces.IDrawable Interface Reference

Implements a simple draw method.



Inheritance diagram for TriDevs.TriEngine.Interfaces.IDrawable:



## Public Member Functions

- void [Draw](#) ()

*Draw the object to screen.*

### 7.21.1 Detailed Description

Implements a simple draw method.

### 7.21.2 Member Function Documentation

#### 7.21.2.1 void TriDevs.TriEngine.Interfaces.IDrawable.Draw ( )

Draw the object to screen.

Implemented in [TriDevs.TriEngine.UI.Label](#), [TriDevs.TriEngine.UI.Control](#), [TriDevs.TriEngine.Graphics.Primitive](#), [TriDevs.TriEngine.StateManagement.GameStateManager](#), [TriDevs.TriEngine.UI.ControlManager](#), and [TriDevs.TriEngine.StateManagement.GameState](#).

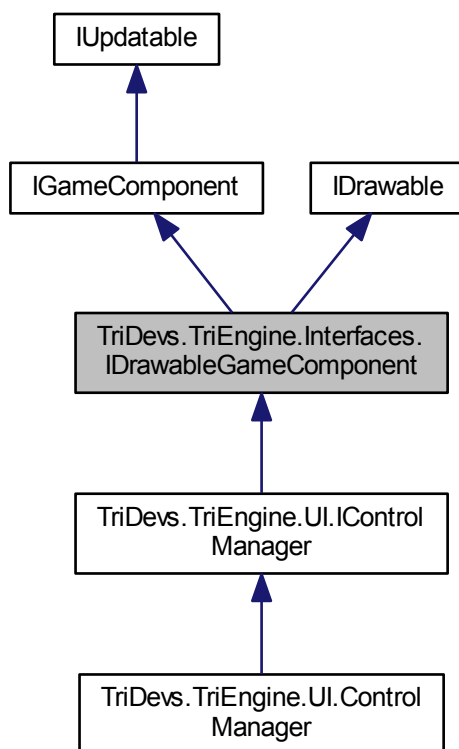
The documentation for this interface was generated from the following file:

- [TriDevs.TriEngine/Interfaces/IDrawable.cs](#)

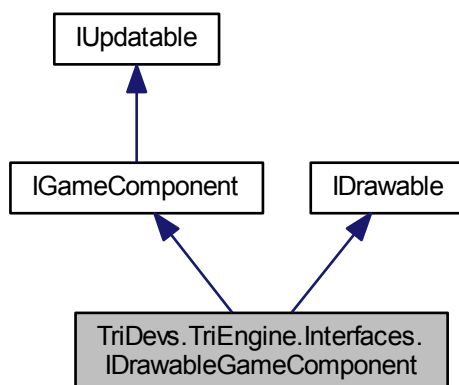
## 7.22 TriDevs.TriEngine.Interfaces.IDrawableGameComponent Interface Reference

A game component that can be added to GameState objects. Drawable game components also implement a draw method to draw themselves to screen.

Inheritance diagram for TriDevs.TriEngine.Interfaces.IDrawableGameComponent:



Collaboration diagram for TriDevs.TriEngine.Interfaces.IDrawableGameComponent:



## Additional Inherited Members

### 7.22.1 Detailed Description

A game component that can be added to GameState objects. Drawable game components also implement a draw method to draw themselves to screen.

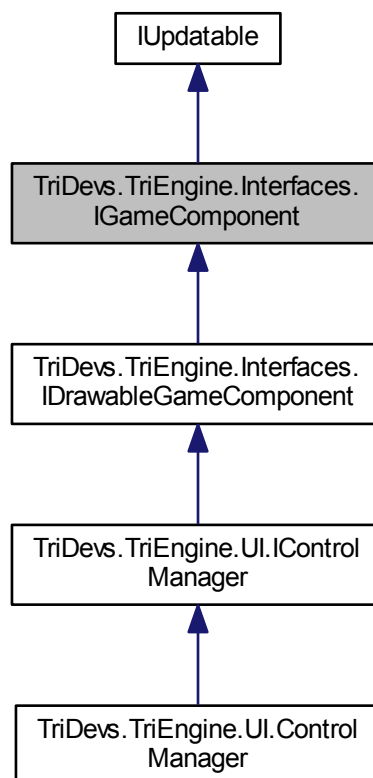
The documentation for this interface was generated from the following file:

- [TriDevs.TriEngine/Interfaces/IDrawableGameComponent.cs](#)

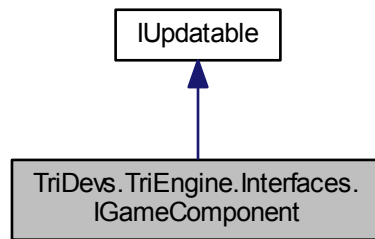
## 7.23 TriDevs.TriEngine.Interfaces.IGameComponent Interface Reference

A game component that can be added to IGameState objects.

Inheritance diagram for TriDevs.TriEngine.Interfaces.IGameComponent:



Collaboration diagram for TriDevs.TriEngine.Interfaces.IGameComponent:



### Public Member Functions

- void [Enable](#) ()  
*Enables this game component.*
- void [Disable](#) ()  
*Disables this game component.*

### Properties

- bool [Enabled](#) [get, set]  
*Gets or sets a value indicating whether this component is enabled or not.*

#### 7.23.1 Detailed Description

A game component that can be added to IGameState objects.

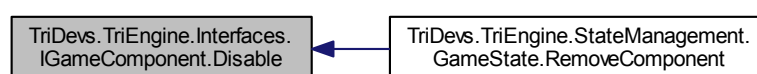
#### 7.23.2 Member Function Documentation

##### 7.23.2.1 void TriDevs.TriEngine.Interfaces.IGameComponent.Disable ( )

Disables this game component.

Implemented in [TriDevs.TriEngine.UI.ControlManager](#).

Here is the caller graph for this function:



### 7.23.2.2 void TriDevs.TriEngine.Interfaces.IGameComponent.Enable ( )

Enables this game component.

Implemented in [TriDevs.TriEngine.UI.ControlManager](#).

Here is the caller graph for this function:



## 7.23.3 Property Documentation

### 7.23.3.1 bool TriDevs.TriEngine.Interfaces.IGameComponent.Enabled [get], [set]

Gets or sets a value indicating whether this component is enabled or not.

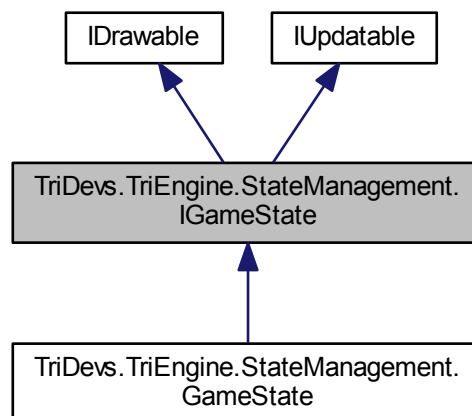
The documentation for this interface was generated from the following file:

- [TriDevs.TriEngine/Interfaces/IGameComponent.cs](#)

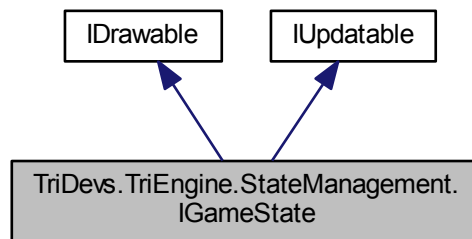
## 7.24 TriDevs.TriEngine.StateManagement.IGameState Interface Reference

A game state that can be used with the game state manager. Represent a specific state of the game, like main menu and options screen.

Inheritance diagram for `TriDevs.TriEngine.StateManagement.IGameState`:



Collaboration diagram for TriDevs.TriEngine.StateManagement.IGameState:



## Public Member Functions

- void [Load](#) ()  
*Loads resources associated with this game component.*
- void [Unload](#) ()  
*Unloads resources that were loaded in the [Load](#) method.*
- void [Pause](#) ()  
*Pauses the game state, preventing update calls from running.*
- void [Unpause](#) ()  
*Unpauses the game state, enabling update calls again.*
- [IGameComponent AddComponent](#) ([IGameComponent](#) component)  
*Adds a game component to this game state.*
- void [RemoveComponent](#) ([IGameComponent](#) component)  
*Removes the specified component from this game state.*
- void [RemoveAllComponents](#) ()  
*Removes all components from the game state.*
- void [RemoveAllComponents](#) (Type type)  
*Removes all components of the specified type from the game state.*
- void [RemoveAllComponents](#) (Predicate< [IGameComponent](#) > predicate)  
*Removes all components that match the supplied predicate function.*
- bool [HasComponent](#) ([IGameComponent](#) component)  
*Returns whether this game state contains the specified IGameComponent.*
- bool [HasComponent](#) (Type type)  
*Returns whether this game state contains a specific type of component.*
- bool [HasComponent](#) (Func< [IGameComponent](#), bool > func)  
*Returns whether this game state contains a component that matches the supplied predicate.*
- IEnumerable< [IGameComponent](#) > [GetAllComponents](#) ()  
*Returns a read-only collection of all components in this game state.*
- [IGameComponent GetComponent](#) (Type type)  
*Returns the specified component type if it exists in this game state.*
- IEnumerable< [IGameComponent](#) > [GetAllComponents](#) (Type type)  
*Returns all components of the specified type.*
- [IGameComponent GetComponent](#) (Func< [IGameComponent](#), bool > func)  
*Returns the first component that matches the supplied predicate function.*
- IEnumerable< [IGameComponent](#) > [GetAllComponents](#) (Func< [IGameComponent](#), bool > func)  
*Returns all components that matches the supplied predicate function.*

## Properties

- bool [Paused](#) [get, set]

*Gets or sets a value indicating whether this game state is currently paused.*

### 7.24.1 Detailed Description

A game state that can be used with the game state manager. Represent a specific state of the game, like main menu and options screen.

### 7.24.2 Member Function Documentation

#### 7.24.2.1 **IGameComponent** TriDevs.TriEngine.StateManagement.IGameState.AddComponent ( **IGameComponent** *component* )

Adds a game component to this game state.

##### Parameters

<i>component</i>	The component to add.
------------------	-----------------------

##### Returns

The component that was added.

Implemented in [TriDevs.TriEngine.StateManagement.GameState](#).

#### 7.24.2.2 **IEnumerable<IGameComponent>** TriDevs.TriEngine.StateManagement.IGameState.GetAllComponents ( )

Returns a read-only collection of all components in this game state.

##### Returns

Read-only collection of components.

Implemented in [TriDevs.TriEngine.StateManagement.GameState](#).

#### 7.24.2.3 **IEnumerable<IGameComponent>** TriDevs.TriEngine.StateManagement.IGameState.GetAllComponents ( **Type** *type* )

Returns all components of the specified type.

##### Parameters

<i>type</i>	The type of game component requested.
-------------	---------------------------------------

##### Returns

A collection of all components of matching type.

Implemented in [TriDevs.TriEngine.StateManagement.GameState](#).

#### 7.24.2.4 IEnumerable<IGameComponent> TriDevs.TriEngine.StateManagement.IGameState.GetAllComponents ( Func< IGameComponent, bool > func )

Returns all components that matches the supplied predicate function.

##### Parameters

<i>func</i>	The predicate function.
-------------	-------------------------

##### Returns

Collection of all matching components, empty collection if no matches were found.

Implemented in [TriDevs.TriEngine.StateManagement.GameState](#).

#### 7.24.2.5 IGameComponent TriDevs.TriEngine.StateManagement.IGameState.GetComponent ( Type type )

Returns the specified component type if it exists in this game state.

##### Parameters

<i>type</i>	The component type to get.
-------------	----------------------------

##### Returns

The component object, or null if it's not added to this game state.

Implemented in [TriDevs.TriEngine.StateManagement.GameState](#).

#### 7.24.2.6 IGameComponent TriDevs.TriEngine.StateManagement.IGameState.GetComponent ( Func< IGameComponent, bool > func )

Returns the first component that matches the supplied predicate function.

##### Parameters

<i>func</i>	The predicate function.
-------------	-------------------------

##### Returns

Component that matches the predicate, null if no matches were found.

Implemented in [TriDevs.TriEngine.StateManagement.GameState](#).

#### 7.24.2.7 bool TriDevs.TriEngine.StateManagement.IGameState.HasComponent ( IGameComponent component )

Returns whether this game state contains the specified IGameComponent.

##### Parameters

<i>component</i>	The component to check for.
------------------	-----------------------------



**Returns**

True if the component has been added to this game state, false otherwise.

Implemented in [TriDevs.TriEngine.StateManagement.GameState](#).

**7.24.2.8 bool TriDevs.TriEngine.StateManagement.IGameState.HasComponent ( Type *type* )**

Returns whether this game state contains a specific type of component.

**Parameters**

<i>type</i>	The type to check for.
-------------	------------------------

**Returns**

True if the type of component has been added to this game state, false otherwise.

Implemented in [TriDevs.TriEngine.StateManagement.GameState](#).

**7.24.2.9 bool TriDevs.TriEngine.StateManagement.IGameState.HasComponent ( Func< IGameComponent, bool > *func* )**

Returns whether this game state contains a component that matches the supplied predicate.

**Parameters**

<i>func</i>	Predicate function to use for search.
-------------	---------------------------------------

**Returns**

True if the game state contains a matching component, false otherwise.

Implemented in [TriDevs.TriEngine.StateManagement.GameState](#).

**7.24.2.10 void TriDevs.TriEngine.StateManagement.IGameState.Load ( )**

Loads resources associated with this game component.

Implemented in [TriDevs.TriEngine.StateManagement.GameState](#).

Here is the caller graph for this function:

**7.24.2.11 void TriDevs.TriEngine.StateManagement.IGameState.Pause ( )**

Pauses the game state, preventing update calls from running.

Implemented in [TriDevs.TriEngine.StateManagement.GameState](#).

Here is the caller graph for this function:



#### 7.24.2.12 void TriDevs.TriEngine.StateManagement.IGameState.RemoveAllComponents ( )

Removes all components from the game state.

Implemented in [TriDevs.TriEngine.StateManagement.GameState](#).

#### 7.24.2.13 void TriDevs.TriEngine.StateManagement.IGameState.RemoveAllComponents ( Type *type* )

Removes all components of the specified type from the game state.

##### Parameters

<i>type</i>	The type of component to remove.
-------------	----------------------------------

Implemented in [TriDevs.TriEngine.StateManagement.GameState](#).

#### 7.24.2.14 void TriDevs.TriEngine.StateManagement.IGameState.RemoveAllComponents ( Predicate< IGameComponent > *predicate* )

Removes all components that match the supplied predicate function.

##### Parameters

<i>predicate</i>	The predicate function.
------------------	-------------------------

Implemented in [TriDevs.TriEngine.StateManagement.GameState](#).

#### 7.24.2.15 void TriDevs.TriEngine.StateManagement.IGameState.RemoveComponent ( IGameComponent *component* )

Removes the specified component from this game state.

##### Parameters

<i>component</i>	
------------------	--

Implemented in [TriDevs.TriEngine.StateManagement.GameState](#).

#### 7.24.2.16 void TriDevs.TriEngine.StateManagement.IGameState.Unload ( )

Unloads resources that were loaded in the [Load](#) method.

Implemented in [TriDevs.TriEngine.StateManagement.GameState](#).

Here is the caller graph for this function:



#### 7.24.2.17 void TriDevs.TriEngine.StateManagement.IGameState.Unpause ( )

Unpauses the game state, enabling update calls again.

Implemented in [TriDevs.TriEngine.StateManagement.GameState](#).

### 7.24.3 Property Documentation

#### 7.24.3.1 bool TriDevs.TriEngine.StateManagement.IGameState.Paused [get],[set]

Gets or sets a value indicating whether this game state is currently paused.

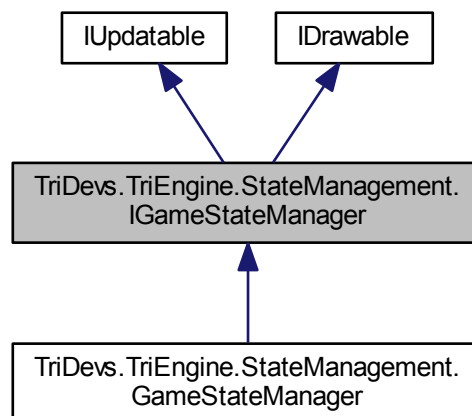
The documentation for this interface was generated from the following file:

- [TriDevs.TriEngine/StateManagement/IGameState.cs](#)

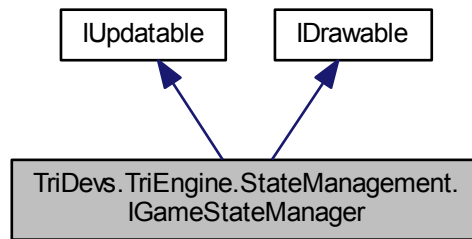
## 7.25 TriDevs.TriEngine.StateManagement.IGameStateManager Interface Reference

Game state manager that keeps track of the active game states and provides methods to control the states.

Inheritance diagram for `TriDevs.TriEngine.StateManagement.IGameStateManager`:



Collaboration diagram for `TriDevs.TriEngine.StateManagement.IGameStateManager`:



## Public Member Functions

- [IGameState Push](#) ([IGameState](#) state)  
*Pushes a new game state onto the stack, pausing the current one.*
- [IGameState Pop](#) ()  
*Pops the currently active state from the stack, unpausing the previous one.*
- [IGameState Peek](#) ()  
*Returns the game state at the top of the stack, without popping it.*
- [IGameState Switch](#) ([IGameState](#) state)  
*Switches to a new game state, discarding all previous ones in the stack.*

## Properties

- `int` [StateCount](#) [get]  
*Gets the number of game states currently in the stack.*
- [IGameState ActiveState](#) [get]  
*Gets the currently active game state.*

### 7.25.1 Detailed Description

Game state manager that keeps track of the active game states and provides methods to control the states.

### 7.25.2 Member Function Documentation

#### 7.25.2.1 [IGameState](#) `TriDevs.TriEngine.StateManagement.IGameStateManager.Peek` ( )

Returns the game state at the top of the stack, without popping it.

#### Returns

The state at the top of the stack.

Implemented in [TriDevs.TriEngine.StateManagement.GameStateManager](#).

**7.25.2.2 IGameState TriDevs.TriEngine.StateManagement.IGameStateManager.Pop ( )**

Pops the currently active state from the stack, unpausing the previous one.

**Returns**

The state that was popped.

Implemented in [TriDevs.TriEngine.StateManagement.GameStateManager](#).

**7.25.2.3 IGameState TriDevs.TriEngine.StateManagement.IGameStateManager.Push ( IGameState *state* )**

Pushes a new game state onto the stack, pausing the current one.

**Parameters**

<i>state</i>	The new game state to push onto the stack.
--------------	--

**Returns**

The game state that was pushed.

Implemented in [TriDevs.TriEngine.StateManagement.GameStateManager](#).

**7.25.2.4 IGameState TriDevs.TriEngine.StateManagement.IGameStateManager.Switch ( IGameState *state* )**

Switches to a new game state, discarding all previous ones in the stack.

**Parameters**

<i>state</i>	The new state to switch to.
--------------	-----------------------------

**Returns**

The state that was switched to.

Implemented in [TriDevs.TriEngine.StateManagement.GameStateManager](#).

**7.25.3 Property Documentation****7.25.3.1 IGameState TriDevs.TriEngine.StateManagement.IGameStateManager.ActiveState [get]**

Gets the currently active game state.

**7.25.3.2 int TriDevs.TriEngine.StateManagement.IGameStateManager.StateCount [get]**

Gets the number of game states currently in the stack.

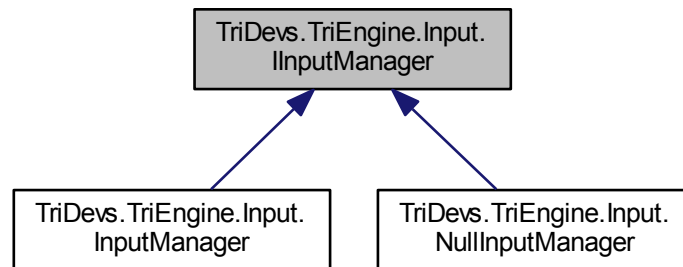
The documentation for this interface was generated from the following file:

- [TriDevs.TriEngine/StateManagement/IGameStateManager.cs](#)

**7.26 TriDevs.TriEngine.Input.IInputManager Interface Reference**

Provides various methods to query input devices like the keyboard.

Inheritance diagram for TriDevs.TriEngine.Input.IInputManager:



## Public Member Functions

- void [Update](#) ()  
*Updates the input manager, refreshing all current and previous states.*
- bool [IsKeyUp](#) (Key key)  
*Returns whether or not the specified key is currently unpressed.*
- bool [IsKeyDown](#) (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 [IsMouseUp](#) (MouseButton button)  
*Returns whether or not the specified mouse button is currently unpressed.*
- bool [IsMouseDown](#) (MouseButton button)  
*Returns whether or not the specified mouse button is currently being pressed.*
- bool [MousePressed](#) (MouseButton button)  
*Returns whether or not the specified mouse button has been pressed.*
- bool [MouseReleased](#) (MouseButton button)  
*Returns whether or not the specified mouse button has been released.*
- bool [IsWheelUp](#) ()  
*Returns whether the mouse wheel was scrolled up.*
- bool [IsWheelDown](#) ()  
*Returns whether the mouse wheel was scrolled down.*
- bool [IsWheelChanged](#) ()  
*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< int > [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.*

## Events

- KeyDownEventHandler [KeyDown](#)  
*Raised when a key is pressed down.*
- KeyUpEventHandler [KeyUp](#)  
*Raised when a key is released.*
- KeyPressEventHandler [KeyPress](#)  
*Raised when a character is typed.*
- MouseDownEventHandler [MouseDown](#)  
*Raised when a mouse button is pressed down.*
- MouseUpEventHandler [MouseUp](#)  
*Raised when a mouse button is released.*
- MouseWheelChangedEventHandler [WheelChanged](#)  
*Raised when the mouse wheel value changes.*
- MouseWheelDownEventHandler [WheelDown](#)  
*Raised when the mouse wheel is scrolled downwards.*
- MouseWheelUpEventHandler [WheelUp](#)  
*Raised when the mouse wheel is scrolled upwards.*

### 7.26.1 Detailed Description

Provides various methods to query input devices like the keyboard.

### 7.26.2 Member Function Documentation

#### 7.26.2.1 bool TriDevs.TriEngine.Input.IInputManager.IsKeyDown ( 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.TriEngine.Input.InputManager](#), and [TriDevs.TriEngine.Input.NullInputManager](#).

**7.26.2.2 bool TriDevs.TriEngine.Input.IInputManager.IsKeyUp ( 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.TriEngine.Input.InputManager](#), and [TriDevs.TriEngine.Input.NullInputManager](#).

**7.26.2.3 bool TriDevs.TriEngine.Input.IInputManager.IsMouseDown ( 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.TriEngine.Input.InputManager](#), and [TriDevs.TriEngine.Input.NullInputManager](#).

**7.26.2.4 bool TriDevs.TriEngine.Input.IInputManager.IsMouseUp ( 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.TriEngine.Input.InputManager](#), and [TriDevs.TriEngine.Input.NullInputManager](#).

**7.26.2.5 bool TriDevs.TriEngine.Input.IInputManager.IsWheelChanged ( )**

Returns whether the mouse wheel scrolled at all.

**Returns**

True if the mouse wheel scrolled, false otherwise.

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



**7.26.2.6** `bool TriDevs.TriEngine.Input.IInputManager.IsWheelDown ( )`

Returns whether the mouse wheel was scrolled down.

**Returns**

True if mouse wheel was scrolled down, false otherwise.

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

**7.26.2.7** `bool TriDevs.TriEngine.Input.IInputManager.IsWheelUp ( )`

Returns whether the mouse wheel was scrolled up.

**Returns**

True if mouse wheel was scrolled up, false otherwise.

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

**7.26.2.8** `bool TriDevs.TriEngine.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.TriEngine.Input.InputManager](#), and [TriDevs.TriEngine.Input.NullInputManager](#).

**7.26.2.9** `bool TriDevs.TriEngine.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.TriEngine.Input.InputManager](#), and [TriDevs.TriEngine.Input.NullInputManager](#).

**7.26.2.10** `bool TriDevs.TriEngine.Input.IInputManager.MousePressed ( 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.TriEngine.Input.InputManager](#), and [TriDevs.TriEngine.Input.NullInputManager](#).

#### 7.26.2.11 bool TriDevs.TriEngine.Input.IInputManager.MouseReleased ( 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.TriEngine.Input.InputManager](#), and [TriDevs.TriEngine.Input.NullInputManager](#).

#### 7.26.2.12 void TriDevs.TriEngine.Input.IInputManager.Update ( )

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

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

#### 7.26.2.13 int TriDevs.TriEngine.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.TriEngine.Input.InputManager](#), and [TriDevs.TriEngine.Input.NullInputManager](#).

### 7.26.3 Property Documentation

#### 7.26.3.1 Point<int> TriDevs.TriEngine.Input.IInputManager.MousePosition [get]

Gets a Point representing the position of the mouse pointer, in window pixel coordinates.

#### 7.26.3.2 int TriDevs.TriEngine.Input.IInputManager.MouseWheelValue [get]

Gets the current value of the mouse wheel.

#### 7.26.3.3 int TriDevs.TriEngine.Input.IInputManager.MouseX [get]

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

**7.26.3.4** `int TriDevs.TriEngine.Input.IInputManager.MouseY` `[get]`

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

**7.26.3.5** `bool TriDevs.TriEngine.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.

**7.26.3.6** `bool TriDevs.TriEngine.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.

**7.26.4 Event Documentation****7.26.4.1** `KeyDownEventHandler TriDevs.TriEngine.Input.IInputManager.KeyDown`

Raised when a key is pressed down.

**7.26.4.2** `KeyPressEventHandler TriDevs.TriEngine.Input.IInputManager.KeyPress`

Raised when a character is typed.

**7.26.4.3** `KeyUpEventHandler TriDevs.TriEngine.Input.IInputManager.KeyUp`

Raised when a key is released.

**7.26.4.4** `MouseDownEventHandler TriDevs.TriEngine.Input.IInputManager.MouseDown`

Raised when a mouse button is pressed down.

**7.26.4.5** `MouseUpEventHandler TriDevs.TriEngine.Input.IInputManager.MouseUp`

Raised when a mouse button is released.

#### 7.26.4.6 MouseWheelChangedEventHandler TriDevs.TriEngine.Input.IInputManager.WheelChanged

Raised when the mouse wheel value changes.

#### 7.26.4.7 MouseWheelDownEventHandler TriDevs.TriEngine.Input.IInputManager.WheelDown

Raised when the mouse wheel is scrolled downwards.

#### 7.26.4.8 MouseWheelUpEventHandler TriDevs.TriEngine.Input.IInputManager.WheelUp

Raised when the mouse wheel is scrolled upwards.

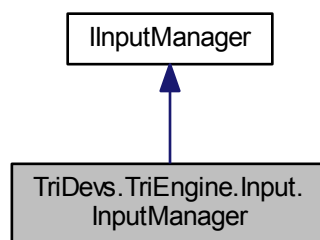
The documentation for this interface was generated from the following file:

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

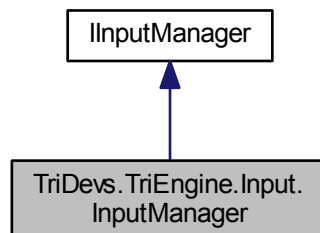
## 7.27 TriDevs.TriEngine.Input.InputManager Class Reference

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

Inheritance diagram for TriDevs.TriEngine.Input.InputManager:



Collaboration diagram for TriDevs.TriEngine.Input.InputManager:



## Public Member Functions

- [InputManager](#) ()  
*Creates a new [InputManager](#) with only basic low-level input support.*
- [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 [IsKeyUp](#) (Key key)  
*Returns whether or not the specified key is currently unpressed.*
- bool [IsKeyDown](#) (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 [IsMouseUp](#) (MouseButton button)  
*Returns whether or not the specified mouse button is currently unpressed.*
- bool [IsMouseDown](#) (MouseButton button)  
*Returns whether or not the specified mouse button is currently being pressed.*
- bool [MousePressed](#) (MouseButton button)  
*Returns whether or not the specified mouse button has been pressed.*
- bool [MouseReleased](#) (MouseButton button)  
*Returns whether or not the specified mouse button has been released.*
- bool [IsWheelUp](#) ()  
*Returns whether the mouse wheel was scrolled up.*
- bool [IsWheelDown](#) ()  
*Returns whether the mouse wheel was scrolled down.*
- bool [IsWheelChanged](#) ()  
*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< int > [MousePosition](#) [get]
- int [MouseWheelValue](#) [get]
- bool [this\[Key key\]](#) [get]
- bool [this\[MouseButton button\]](#) [get]

## Events

- KeyDownEventHandler [KeyDown](#)  
*Raised when a key is pressed down.*
- KeyUpEventHandler [KeyUp](#)  
*Raised when a key is released.*
- KeyPressEventHandler [KeyPress](#)  
*Raised when a character is typed.*
- MouseDownEventHandler [MouseDown](#)

*Raised when a mouse button is pressed down.*

- [MouseUpEventHandler](#) [MouseUp](#)

*Raised when a mouse button is released.*

- [MouseWheelChangedEventHandler](#) [WheelChanged](#)

*Raised when the mouse wheel value changes.*

- [MouseWheelDownEventHandler](#) [WheelDown](#)

*Raised when the mouse wheel is scrolled downwards.*

- [MouseWheelUpEventHandler](#) [WheelUp](#)

*Raised when the mouse wheel is scrolled upwards.*

## 7.27.1 Detailed Description

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

## 7.27.2 Constructor & Destructor Documentation

### 7.27.2.1 [TriDevs.TriEngine.Input.InputManager.InputManager](#) ( )

Creates a new [InputManager](#) with only basic low-level input support.

Creating [InputManager](#) without a driver container will cause the events to be useless and never be raised, only the methods on this class will return any useful info. If you want event support, construct the [InputManager](#) with a [GameWindow](#) or other supported driver providers (NYI).

```

109     {
110         // We're assigning an empty mouse device.
111         // This will make position functions return a constant 0.
112         // Instead of being null and causing exceptions.
113         _mouse = new MouseDevice();
114         // We don't have to assign an empty keyboard device,
115         // since we don't have any code that directly relies on it being present.
116     }
```

### 7.27.2.2 [TriDevs.TriEngine.Input.InputManager.InputManager](#) ( [GameWindow window](#) )

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

#### Parameters

<i>window</i>	The <a href="#">GameWindow</a> this <a href="#">InputManager</a> will interface with.
---------------	---

```

123     {
124         _keyboard = window.Keyboard;
125         _mouse = window.Mouse;
126         _keyboard.KeyDown += OnKeyDown;
127         _keyboard.KeyUp += OnKeyUp;
128         window.KeyPress += OnKeyPress;
129         _mouse.ButtonDown += OnMouseDown;
130         _mouse.ButtonUp += OnMouseUp;
131         _mouse.WheelChanged += OnMouseWheelChanged;
132     }
```

## 7.27.3 Member Function Documentation

### 7.27.3.1 [bool TriDevs.TriEngine.Input.InputManager.IsKeyDown](#) ( [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.TriEngine.Input.IInputManager](#).

```

201     {
202         return _keyboardState[key];
203     }
```

## 7.27.3.2 bool TriDevs.TriEngine.Input.InputManager.IsKeyUp ( 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.TriEngine.Input.IInputManager](#).

```

196     {
197         return !_keyboardState[key];
198     }
```

7.27.3.3 bool TriDevs.TriEngine.Input.InputManager.IsMouseDown ( 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.TriEngine.Input.IInputManager](#).

```

221     {
222         return _mouseState[button];
223     }
```

7.27.3.4 bool TriDevs.TriEngine.Input.InputManager.IsMouseUp ( 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.TriEngine.Input.IInputManager](#).

```
216     {
217         return !_mouseState[button];
218     }
```

**7.27.3.5 bool TriDevs.TriEngine.Input.InputManager.IsWheelChanged ( )**

Returns whether the mouse wheel scrolled at all.

**Returns**

True if the mouse wheel scrolled, false otherwise.

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

```
246     {
247         return _mouseState.Wheel != _lastMouseState.Wheel;
248     }
```

**7.27.3.6 bool TriDevs.TriEngine.Input.InputManager.IsWheelDown ( )**

Returns whether the mouse wheel was scrolled down.

**Returns**

True if mouse wheel was scrolled down, false otherwise.

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

```
241     {
242         return _mouseState.Wheel < _lastMouseState.Wheel;
243     }
```

**7.27.3.7 bool TriDevs.TriEngine.Input.InputManager.IsWheelUp ( )**

Returns whether the mouse wheel was scrolled up.

**Returns**

True if mouse wheel was scrolled up, false otherwise.

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

```
236     {
237         return _mouseState.Wheel > _lastMouseState.Wheel;
238     }
```

**7.27.3.8 bool TriDevs.TriEngine.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.TriEngine.Input.IInputManager](#).

```
206     {  
207         return _keyboardState[key] && !_lastKeyboardState[key];  
208     }
```

#### 7.27.3.9 bool TriDevs.TriEngine.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.TriEngine.Input.IInputManager](#).

```
211     {  
212         return !_keyboardState[key] && _lastKeyboardState[key];  
213     }
```

#### 7.27.3.10 bool TriDevs.TriEngine.Input.InputManager.MousePressed ( 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.TriEngine.Input.IInputManager](#).

```
226     {  
227         return _mouseState[button] && !_lastMouseState[button];  
228     }
```

#### 7.27.3.11 bool TriDevs.TriEngine.Input.InputManager.MouseReleased ( 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.TriEngine.Input.IInputManager](#).

```

231     {
232         return !_mouseState[button] && _lastMouseState[button];
233     }
```

**7.27.3.12 void TriDevs.TriEngine.Input.InputManager.Update ( )**

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

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

```

187     {
188         _lastKeyboardState = _keyboardState;
189         _keyboardState = Keyboard.GetState();
190
191         _lastMouseState = _mouseState;
192         _mouseState = Mouse.GetState();
193     }
```

**7.27.3.13 int TriDevs.TriEngine.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.TriEngine.Input.IInputManager](#).

```

251     {
252         return _mouseState.Wheel - _lastMouseState.Wheel;
253     }
```

**7.27.4 Property Documentation**

**7.27.4.1** `Point<int> TriDevs.TriEngine.Input.InputManager.MousePosition` [get]

**7.27.4.2** `int TriDevs.TriEngine.Input.InputManager.MouseWheelValue` [get]

**7.27.4.3** `int TriDevs.TriEngine.Input.InputManager.MouseX` [get]

**7.27.4.4** `int TriDevs.TriEngine.Input.InputManager.MouseY` [get]

**7.27.4.5** `bool TriDevs.TriEngine.Input.InputManager.this[Key key]` [get]

**7.27.4.6** `bool TriDevs.TriEngine.Input.InputManager.this[MouseButton button]` [get]

**7.27.5 Event Documentation**

**7.27.5.1** `KeyDownEventHandler TriDevs.TriEngine.Input.InputManager.KeyDown`

Raised when a key is pressed down.

**7.27.5.2 KeyPressEventHandler TriDevs.TriEngine.Input.InputManager.KeyPress**

Raised when a character is typed.

**7.27.5.3 KeyUpEventHandler TriDevs.TriEngine.Input.InputManager.KeyUp**

Raised when a key is released.

**7.27.5.4 MouseDownEventHandler TriDevs.TriEngine.Input.InputManager.MouseDown**

Raised when a mouse button is pressed down.

**7.27.5.5 MouseUpEventHandler TriDevs.TriEngine.Input.InputManager.MouseUp**

Raised when a mouse button is released.

**7.27.5.6 MouseWheelChangedEventHandler TriDevs.TriEngine.Input.InputManager.WheelChanged**

Raised when the mouse wheel value changes.

**7.27.5.7 MouseWheelDownEventHandler TriDevs.TriEngine.Input.InputManager.WheelDown**

Raised when the mouse wheel is scrolled downwards.

**7.27.5.8 MouseWheelUpEventHandler TriDevs.TriEngine.Input.InputManager.WheelUp**

Raised when the mouse wheel is scrolled upwards.

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

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

## 7.28 TriDevs.TriEngine.Extensions.IntegerExtensions Class Reference

[Extensions](#) for System.Int16, System.UInt16, System.Int32, System.UInt32, System.Int64 and System.UInt64.

### Static Public Member Functions

- static short [Clamp](#) (this short value, short min, short max)  
*Clamps the specified short between a minimum and maximum value.*
- static ushort [Clamp](#) (this ushort value, ushort min, ushort max)  
*Clamps the specified unsigned short between a minimum and maximum value.*
- static int [Clamp](#) (this int value, int min, int max)  
*Clamps the specified integer between a minimum and maximum value.*
- static uint [Clamp](#) (this uint value, uint min, uint max)  
*Clamps the specified unsigned integer between a minimum and maximum value.*
- static long [Clamp](#) (this long value, long min, long max)  
*Clamps the specified 64-bit integer between a minimum and maximum value.*
- static ulong [Clamp](#) (this ulong value, ulong min, ulong max)  
*Clamps the specified 64-bit unsigned integer between a minimum and maximum value.*

### 7.28.1 Detailed Description

[Extensions](#) for System.Int16, System.UInt16, System.Int32, System.UInt32, System.Int64 and System.UInt64.

### 7.28.2 Member Function Documentation

7.28.2.1 **static short** TriDevs.TriEngine.Extensions.IntegerExtensions.Clamp ( this short *value*, short *min*, short *max* )  
[static]

Clamps the specified short between a minimum and maximum value.

#### Parameters

<i>value</i>	Value to clamp.
<i>min</i>	If the specified short is below this value, then this will be returned.
<i>max</i>	If the specified short is above this value, then this will be returned.

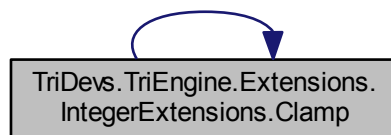
#### Returns

The clamped value of the short.

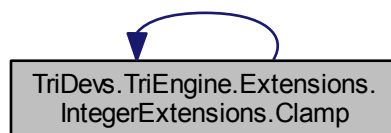
```

42     {
43         return Helpers.Math.Clamp(value, min, max);
44     }
```

Here is the call graph for this function:



Here is the caller graph for this function:



**7.28.2.2** `static ushort TriDevs.TriEngine.Extensions.IntegerExtensions.Clamp ( this ushort value, ushort min, ushort max )`  
`[static]`

Clamps the specified unsigned short between a minimum and maximum value.

#### Parameters

<i>value</i>	Value to clamp.
<i>min</i>	If the specified unsigned short is below this value, then this will be returned.
<i>max</i>	If the specified unsigned short is above this value, then this will be returned.

#### Returns

The clamped value of the unsigned short.

```

54         {
55             return Helpers.Math.Clamp(value, min, max);
56         }

```

**7.28.2.3** `static int TriDevs.TriEngine.Extensions.IntegerExtensions.Clamp ( this int value, int min, int max )` `[static]`

Clamps the specified integer between a minimum and maximum value.

#### Parameters

<i>value</i>	Value to clamp.
<i>min</i>	If the specified integer is below this value, then this will be returned.
<i>max</i>	If the specified integer is above this value, then this will be returned.

#### Returns

The clamped value of the integer.

```

66         {
67             return Helpers.Math.Clamp(value, min, max);
68         }

```

**7.28.2.4** `static uint TriDevs.TriEngine.Extensions.IntegerExtensions.Clamp ( this uint value, uint min, uint max )`  
`[static]`

Clamps the specified unsigned integer between a minimum and maximum value.

#### Parameters

<i>value</i>	Value to clamp.
<i>min</i>	If the specified unsigned integer is below this value, then this will be returned.
<i>max</i>	If the specified unsigned integer is above this value, then this will be returned.

#### Returns

The clamped value of the unsigned integer.

```

78         {
79             return Helpers.Math.Clamp(value, min, max);
80         }

```

#### 7.28.2.5 static long TriDevs.TriEngine.Extensions.IntegerExtensions.Clamp ( this long value, long min, long max ) [static]

Clamps the specified 64-bit integer between a minimum and maximum value.

##### Parameters

<i>value</i>	Value to clamp.
<i>min</i>	If the specified 64-bit integer is below this value, then this will be returned.
<i>max</i>	If the specified 64-bit integer is above this value, then this will be returned.

##### Returns

The clamped value of the 64-bit integer.

```

90         {
91             return Helpers.Math.Clamp(value, min, max);
92         }

```

#### 7.28.2.6 static ulong TriDevs.TriEngine.Extensions.IntegerExtensions.Clamp ( this ulong value, ulong min, ulong max ) [static]

Clamps the specified 64-bit unsigned integer between a minimum and maximum value.

##### Parameters

<i>value</i>	Value to clamp.
<i>min</i>	If the specified 64-bit unsigned integer is below this value, then this will be returned.
<i>max</i>	If the specified 64-bit unsigned integer is above this value, then this will be returned.

##### Returns

The clamped value of the 64-bit unsigned integer.

```

102         {
103             return Helpers.Math.Clamp(value, min, max);
104         }

```

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

- TriDevs.TriEngine/Extensions/[IntegerExtensions.cs](#)

## 7.29 TriDevs.TriEngine.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.*

#### 7.29.1 Detailed Description

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

## 7.29.2 Member Function Documentation

### 7.29.2.1 static string TriDevs.TriEngine.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         }

```

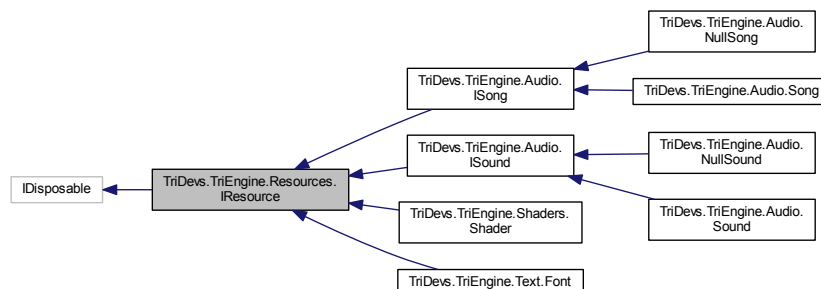
The documentation for this class was generated from the following file:

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

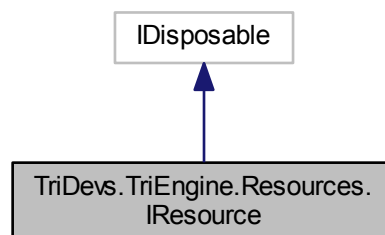
## 7.30 TriDevs.TriEngine.Resources.IResource Interface Reference

An external resource created from a file.

Inheritance diagram for TriDevs.TriEngine.Resources.IResource:



Collaboration diagram for TriDevs.TriEngine.Resources.IResource:



## Properties

- string [Name](#) [get]

*Gets the name associated with this resource.*

### 7.30.1 Detailed Description

An external resource created from a file.

### 7.30.2 Property Documentation

- 7.30.2.1 string `TriDevs.TriEngine.Resources.IResource.Name` [get]

Gets the name associated with this resource.

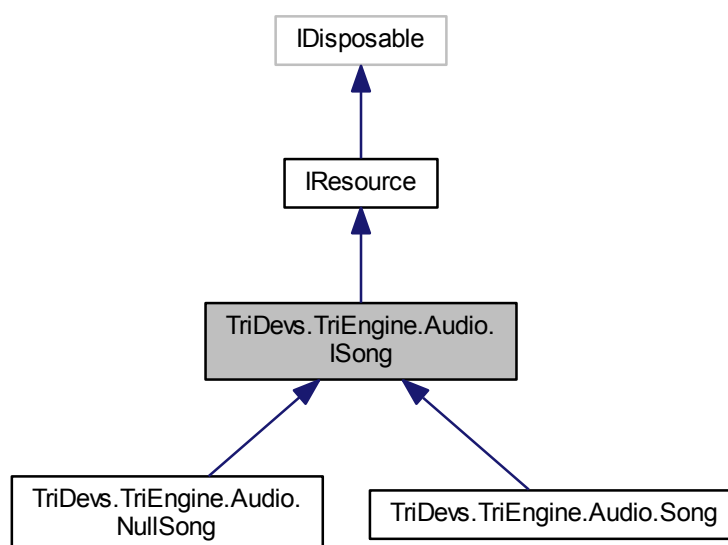
The documentation for this interface was generated from the following file:

- `TriDevs.TriEngine/Resources/IResource.cs`

## 7.31 TriDevs.TriEngine.Audio.ISong Interface Reference

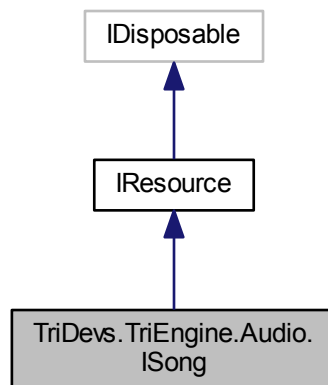
A song that will be streamed in the audio player.

Inheritance diagram for `TriDevs.TriEngine.Audio.ISong`:





Collaboration diagram for TriDevs.TriEngine.Audio.ISong:



### Public Member Functions

- void [Play](#) ()  
*Starts playback of the song.*
- void [Stop](#) ()  
*Stops playback of the song.*
- void [Pause](#) ()  
*Pauses playback of the song.*
- void [Resume](#) ()  
*Resumes playback of a paused song.*

### Properties

- string [File](#) [get]  
*Gets the file this song was loaded from.*
- float [Volume](#) [get, set]  
*Gets or sets the song volume.*
- bool [IsLooped](#) [get, set]  
*Gets or sets a value indicating that the song should be looped once it reaches the end.*

#### 7.31.1 Detailed Description

A song that will be streamed in the audio player.

#### 7.31.2 Member Function Documentation

##### 7.31.2.1 void TriDevs.TriEngine.Audio.ISong.Pause ( )

Pauses playback of the song.

Call the [Resume](#) or [Play](#) method to resume playback.

Implemented in [TriDevs.TriEngine.Audio.Song](#), and [TriDevs.TriEngine.Audio.NullSong](#).

#### 7.31.2.2 void TriDevs.TriEngine.Audio.ISong.Play ( )

Starts playback of the song.

Implemented in [TriDevs.TriEngine.Audio.Song](#), and [TriDevs.TriEngine.Audio.NullSong](#).

#### 7.31.2.3 void TriDevs.TriEngine.Audio.ISong.Resume ( )

Resumes playback of a paused song.

Implemented in [TriDevs.TriEngine.Audio.Song](#), and [TriDevs.TriEngine.Audio.NullSong](#).

#### 7.31.2.4 void TriDevs.TriEngine.Audio.ISong.Stop ( )

Stops playback of the song.

Implemented in [TriDevs.TriEngine.Audio.Song](#), and [TriDevs.TriEngine.Audio.NullSong](#).

### 7.31.3 Property Documentation

#### 7.31.3.1 string TriDevs.TriEngine.Audio.ISong.File [get]

Gets the file this song was loaded from.

#### 7.31.3.2 bool TriDevs.TriEngine.Audio.ISong.IsLooped [get], [set]

Gets or sets a value indicating that the song should be looped once it reaches the end.

#### 7.31.3.3 float TriDevs.TriEngine.Audio.ISong.Volume [get], [set]

Gets or sets the song volume.

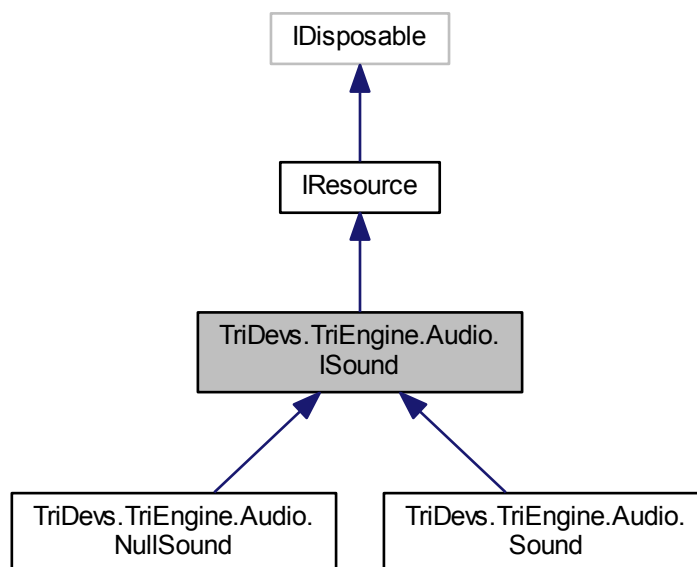
The documentation for this interface was generated from the following file:

- [TriDevs.TriEngine/Audio/ISong.cs](#)

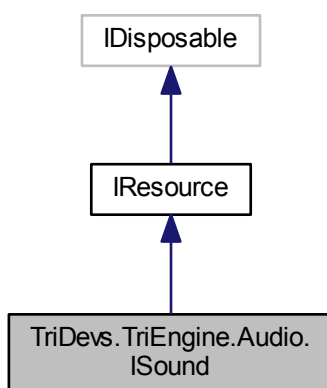
## 7.32 TriDevs.TriEngine.Audio.ISound Interface Reference

A sound file for use with the audio manager.

Inheritance diagram for TriDevs.TriEngine.Audio.ISound:



Collaboration diagram for TriDevs.TriEngine.Audio.ISound:



### Public Member Functions

- void `Play` ()  
*Plays the sound.*
- void `Stop` ()  
*Stops the sound.*

## Properties

- string [File](#) [get]  
*Gets the file this sound was created from.*

### 7.32.1 Detailed Description

A sound file for use with the audio manager.

### 7.32.2 Member Function Documentation

#### 7.32.2.1 void [TriDevs.TriEngine.Audio.ISound.Play](#) ( )

Plays the sound.

Implemented in [TriDevs.TriEngine.Audio.Sound](#), and [TriDevs.TriEngine.Audio.NullSound](#).

#### 7.32.2.2 void [TriDevs.TriEngine.Audio.ISound.Stop](#) ( )

Stops the sound.

Implemented in [TriDevs.TriEngine.Audio.Sound](#), and [TriDevs.TriEngine.Audio.NullSound](#).

### 7.32.3 Property Documentation

#### 7.32.3.1 string [TriDevs.TriEngine.Audio.ISound.File](#) [get]

Gets the file this sound was created from.

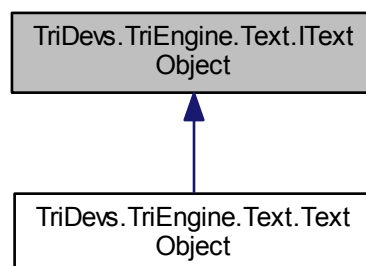
The documentation for this interface was generated from the following file:

- [TriDevs.TriEngine/Audio/ISound.cs](#)

## 7.33 TriDevs.TriEngine.Text.ITextObject Interface Reference

Implements methods to construct a text object and render it to screen.

Inheritance diagram for [TriDevs.TriEngine.Text.ITextObject](#):



## Public Member Functions

- void [Draw](#) ()  
*Draws this text object to screen with default parameters.*
- void [Draw](#) (Point< int > position)  
*Draws this text object to a specific position on the screen specified by the supplied Point struct.*
- void [Draw](#) (int x, int y)  
*Draws this text object to screen at the specified X/Y position.*

## Properties

- [Font](#) [Font](#) [get]  
*Gets the [Font](#) font instance associated with this text object.*
- string [Text](#) [get, set]  
*Gets or sets the text value of this text object.*
- [Color](#) [Color](#) [get, set]  
*Gets or sets the color of the text.*
- Point< int > [Position](#) [get, set]  
*Gets or sets the screen position of this text, in pixels.*
- [Rectangle](#) [Bounds](#) [get]  
*Gets the bounds of this text object.*
- QFontAlignment [Alignment](#) [get, set]  
*Gets or sets a QFontAlignment value to manage how this text is aligned on screen.*

### 7.33.1 Detailed Description

Implements methods to construct a text object and render it to screen.

### 7.33.2 Member Function Documentation

#### 7.33.2.1 void TriDevs.TriEngine.Text.ITextObject.Draw ( )

Draws this text object to screen with default parameters.

Implemented in [TriDevs.TriEngine.Text.TextObject](#).

#### 7.33.2.2 void TriDevs.TriEngine.Text.ITextObject.Draw ( Point< int > position )

Draws this text object to a specific position on the screen specified by the supplied Point struct.

#### Parameters

<i>position</i>	Point class with X/Y coordinates.
-----------------	-----------------------------------

Implemented in [TriDevs.TriEngine.Text.TextObject](#).

#### 7.33.2.3 void TriDevs.TriEngine.Text.ITextObject.Draw ( int x, int y )

Draws this text object to screen at the specified X/Y position.

## Parameters

<i>x</i>	
<i>y</i>	

Implemented in [TriDevs.TriEngine.Text.TextObject](#).

### 7.33.3 Property Documentation

#### 7.33.3.1 QFontAlignment TriDevs.TriEngine.Text.ITextObject.Alignment [get], [set]

Gets or sets a QFontAlignment value to manage how this text is aligned on screen.

#### 7.33.3.2 Rectangle TriDevs.TriEngine.Text.ITextObject.Bounds [get]

Gets the bounds of this text object.

#### 7.33.3.3 Color TriDevs.TriEngine.Text.ITextObject.Color [get], [set]

Gets or sets the color of the text.

#### 7.33.3.4 Font TriDevs.TriEngine.Text.ITextObject.Font [get]

Gets the [Font](#) font instance associated with this text object.

#### 7.33.3.5 Point<int> TriDevs.TriEngine.Text.ITextObject.Position [get], [set]

Gets or sets the screen position of this text, in pixels.

#### 7.33.3.6 string TriDevs.TriEngine.Text.ITextObject.Text [get], [set]

Gets or sets the text value of this text object.

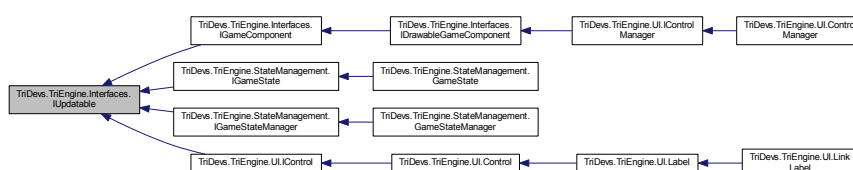
The documentation for this interface was generated from the following file:

- [TriDevs.TriEngine/Text/ITextObject.cs](#)

## 7.34 TriDevs.TriEngine.Interfaces.IUpdatable Interface Reference

Implements a simple update method.

Inheritance diagram for TriDevs.TriEngine.Interfaces.IUpdatable:



## Public Member Functions

- void [Update](#) ()

*Updates the object.*

### 7.34.1 Detailed Description

Implements a simple update method.

### 7.34.2 Member Function Documentation

#### 7.34.2.1 void TriDevs.TriEngine.Interfaces.IUpdatable.Update ( )

Updates the object.

Implemented in [TriDevs.TriEngine.UI.Label](#), [TriDevs.TriEngine.UI.Control](#), [TriDevs.TriEngine.UI.ControlManager](#), [TriDevs.TriEngine.StateManagement.GameStateManager](#), and [TriDevs.TriEngine.StateManagement.GameState](#).

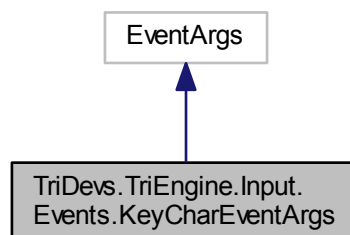
The documentation for this interface was generated from the following file:

- [TriDevs.TriEngine/Interfaces/IUpdatable.cs](#)

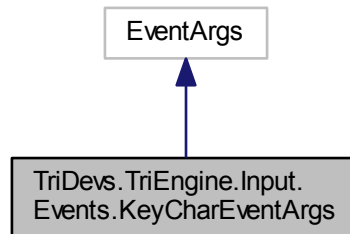
## 7.35 TriDevs.TriEngine.Input.Events.KeyCharEventArgs Class Reference

EventArgs class used for keychar-related events. Contains information about the character related with the event.

Inheritance diagram for TriDevs.TriEngine.Input.Events.KeyCharEventArgs:



Collaboration diagram for TriDevs.TriEngine.Input.Events.KeyCharEventArgs:



## Public Attributes

- readonly char [KeyChar](#)

*The char that was involved.*

### 7.35.1 Detailed Description

EventArgs class used for keychar-related events. Contains information about the character related with the event.

### 7.35.2 Member Data Documentation

#### 7.35.2.1 readonly char TriDevs.TriEngine.Input.Events.KeyCharEventArgs.KeyChar

The char that was involved.

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

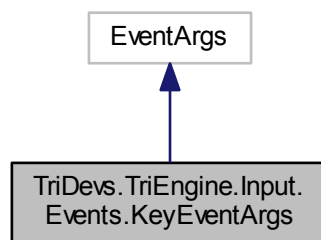
- TriDevs.TriEngine/Input/Events/[Key.cs](#)

## 7.36 TriDevs.TriEngine.Input.Events.KeyEventArgs Class Reference

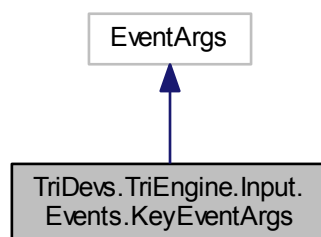
EventArgs class used for key-related events. Contains information about the key related with the event.



Inheritance diagram for TriDevs.TriEngine.Input.Events.KeyEventArgs:



Collaboration diagram for TriDevs.TriEngine.Input.Events.KeyEventArgs:



## Public Attributes

- readonly Key [Key](#)  
*The [Key](#) that was involved.*

### 7.36.1 Detailed Description

EventArgs class used for key-related events. Contains information about the key related with the event.

### 7.36.2 Member Data Documentation

#### 7.36.2.1 readonly Key TriDevs.TriEngine.Input.Events.KeyEventArgs.Key

The [Key](#) that was involved.

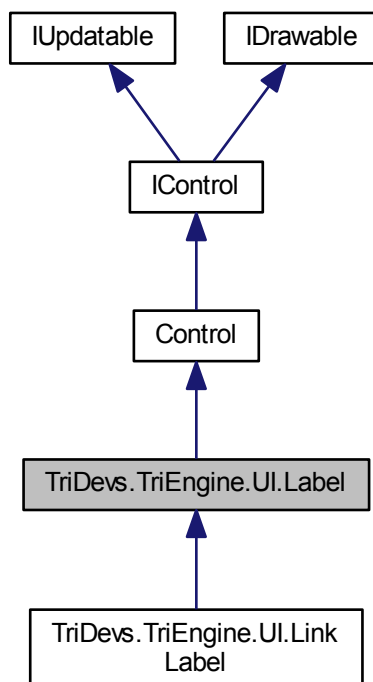
The documentation for this class was generated from the following file:

- TriDevs.TriEngine/Input/Events/[Key.cs](#)

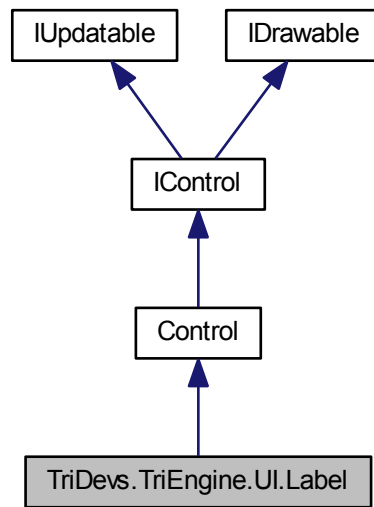
### 7.37 TriDevs.TriEngine.UI.Label Class Reference

A simple label to display text on the screen.

Inheritance diagram for TriDevs.TriEngine.UI.Label:



Collaboration diagram for TriDevs.TriEngine.UI.Label:



### Public Member Functions

- virtual void `SetFont` (`Font` font)  
*Sets the font that this label uses.*
- override void `Update` ()  
*Updates the object.*
- override void `Draw` ()  
*Draw the object to screen.*

### Protected Member Functions

- virtual void `UpdateTextObject` ()

### Properties

- override string `Text` [get, set]
- override `Point< int >` `Position` [get, set]
- virtual `QFontAlignment` `Alignment` [get, set]  
*Gets or sets the QFontAlignment of this label's text.*

### Additional Inherited Members

#### 7.37.1 Detailed Description

A simple label to display text on the screen.

## 7.37.2 Member Function Documentation

### 7.37.2.1 override void TriDevs.TriEngine.UI.Label.Draw ( ) [virtual]

Draw the object to screen.

Reimplemented from [TriDevs.TriEngine.UI.Control](#).

```

135     {
136         base.Draw(_drawPosition);
137
138         if (_textObject == null)
139             return;
140
141         _textObject.Draw();
142     }
```

### 7.37.2.2 virtual void TriDevs.TriEngine.UI.Label.SetFont ( Font font ) [virtual]

Sets the font that this label uses.

#### Parameters

<i>font</i>	The new font instance to use.
-------------	-------------------------------

```

82     {
83         _font = font;
84         UpdateTextObject();
85     }
```

### 7.37.2.3 override void TriDevs.TriEngine.UI.Label.Update ( ) [virtual]

Updates the object.

Reimplemented from [TriDevs.TriEngine.UI.Control](#).

```

121     {
122         // Override update logic to translate mouse click
123         // positions when label is aligned in a certain way
124
125         if (!Services.Input.MouseReleased(MouseButton.Left))
126             return;
127
128         var mousePos = Services.Input.MousePosition;
129         if ((mousePos.X >= _drawPosition.X && mousePos.X <= (_drawPosition.X +
Rectangle.Width))
130             && (mousePos.Y >= _drawPosition.Y && mousePos.Y <= (_drawPosition.Y +
Rectangle.Height)))
131             OnClicked();
132     }
```

### 7.37.2.4 virtual void TriDevs.TriEngine.UI.Label.UpdateTextObject ( ) [protected],[virtual]

```

88     {
89         if (_font == null)
90             return;
91
92         if (_textObject == null)
93         {
94             _textObject = new TextObject(_text, _font, Position,
Alignment);
95         }
96         else
97         {
98             _textObject.Text = Text;
99             _textObject.Font = _font;
100             _textObject.Position = Position;
101             _textObject.Alignment = Alignment;
```

```

102         }
103
104         Size = new Point<int>(_textObject.Bounds.Width, _textObject.
Bounds.Height);
105
106         switch (Alignment)
107         {
108             case QFontAlignment.Centre:
109                 _drawPosition = new Point<int>(Position.X - Size.X / 2,
Position.Y);
110                 break;
111             case QFontAlignment.Right:
112                 _drawPosition = new Point<int>(Position.X - Size.X,
Position.Y);
113                 break;
114             default:
115                 _drawPosition = Position;
116                 break;
117         }
118     }

```

### 7.37.3 Property Documentation

**7.37.3.1** virtual QFontAlignment TriDevs.TriEngine.UI.Label.Alignment [get], [set]

Gets or sets the QFontAlignment of this label's text.

**7.37.3.2** override Point<int> TriDevs.TriEngine.UI.Label.Position [get], [set]

**7.37.3.3** override string TriDevs.TriEngine.UI.Label.Text [get], [set]

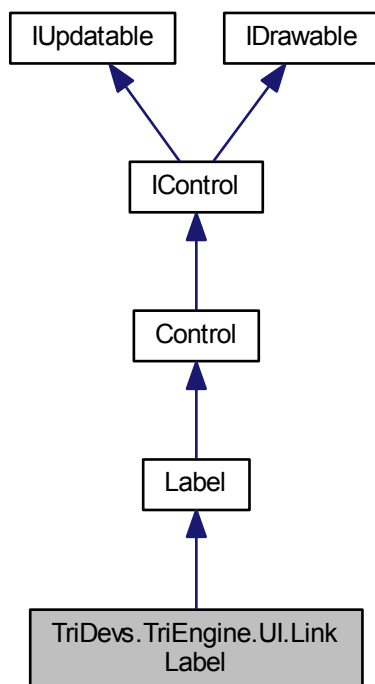
The documentation for this class was generated from the following file:

- TriDevs.TriEngine/UI/[Label.cs](#)

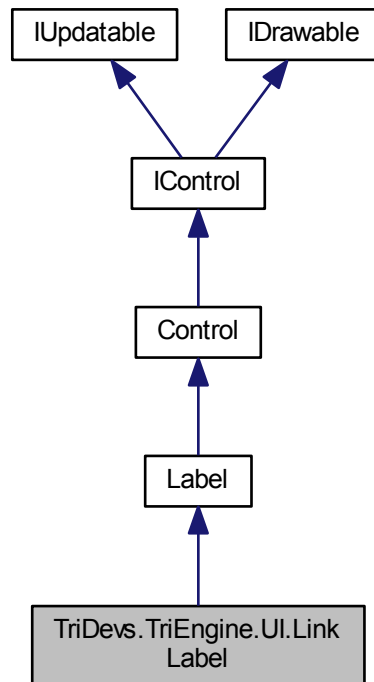
## 7.38 TriDevs.TriEngine.UI.LinkLabel Class Reference

A label that, when clicked, will open a URL.

Inheritance diagram for TriDevs.TriEngine.UI.LinkLabel:



Collaboration diagram for TriDevs.TriEngine.UI.LinkLabel:



### Protected Member Functions

- override void [OnClicked](#) ()

### Properties

- virtual string [Url](#) [get, set]  
*Gets or sets the URL that will open when this label is clicked.*

### Additional Inherited Members

#### 7.38.1 Detailed Description

A label that, when clicked, will open a URL.

#### 7.38.2 Member Function Documentation

7.38.2.1 override void [TriDevs.TriEngine.UI.LinkLabel.OnClicked](#) ( ) [protected], [virtual]

Reimplemented from [TriDevs.TriEngine.UI.Control](#).

```

39         {
40             if (!string.IsNullOrEmpty(Url))

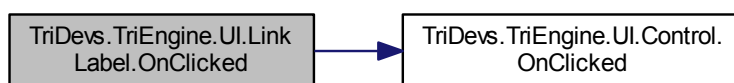
```

```

41         {
42             try
43             {
44                 System.Diagnostics.Process.Start (Url);
45             }
46             catch (Win32Exception)
47             {
48             }
49         }
50     }
51 }
52
53     base.OnClicked();
54 }

```

Here is the call graph for this function:



### 7.38.3 Property Documentation

7.38.3.1 virtual string TriDevs.TriEngine.UI.LinkLabel.Url [get], [set]

Gets or sets the URL that will open when this label is clicked.

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

- TriDevs.TriEngine/UI/[LinkLabel.cs](#)

## 7.39 TriDevs.TriEngine.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.*



### 7.39.1 Detailed Description

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

### 7.39.2 Member Function Documentation

#### 7.39.2.1 static void TriDevs.TriEngine.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     }

```

#### 7.39.2.2 static void TriDevs.TriEngine.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     }

```

#### 7.39.2.3 static ILog TriDevs.TriEngine.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
)sender : sender.GetType());
94     }

```

#### 7.39.2.4 static void TriDevs.TriEngine.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         _loaded = true;
76     }
77 }

```

#### 7.39.2.5 static void TriDevs.TriEngine.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     }

```

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

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

## 7.40 TriDevs.TriEngine.Helpers.Math Class Reference

Various helper methods for working with math.

### Static Public Member Functions

- static byte [Clamp](#) (byte value, byte min, byte max)  
*Clamps the specified byte between a minimum and maximum value.*
- static short [Clamp](#) (short value, short min, short max)  
*Clamps the specified short between a minimum and maximum value.*
- static ushort [Clamp](#) (ushort value, ushort min, ushort max)  
*Clamps the specified unsigned short between a minimum and maximum value.*
- static int [Clamp](#) (int value, int min, int max)  
*Clamps the specified integer between a minimum and maximum value.*
- static uint [Clamp](#) (uint value, uint min, uint max)  
*Clamps the specified unsigned integer between a minimum and maximum value.*
- static long [Clamp](#) (long value, long min, long max)  
*Clamps the specified 64-bit integer between a minimum and maximum value.*
- static ulong [Clamp](#) (ulong value, ulong min, ulong max)  
*Clamps the specified 64-bit unsigned integer between a minimum and maximum value.*
- static float [Clamp](#) (float value, float min, float max)  
*Clamps the specified float between a minimum and maximum value.*
- static double [Clamp](#) (double value, double min, double max)  
*Clamps the specified double between a minimum and maximum value.*
- static decimal [Clamp](#) (decimal value, decimal min, decimal max)  
*Clamps the specified decimal between a minimum and maximum value.*

### 7.40.1 Detailed Description

Various helper methods for working with math.

### 7.40.2 Member Function Documentation

#### 7.40.2.1 static byte TriDevs.TriEngine.Helpers.Math.Clamp ( byte value, byte min, byte max ) [static]

Clamps the specified byte between a minimum and maximum value.

#### Parameters

<i>value</i>	Value to clamp.
<i>min</i>	If the specified byte is below this value, then this will be returned.
<i>max</i>	If the specified byte is above this value, then this will be returned.

**Returns**

The clamped value of the byte.

```

41     {
42         if (min > max)
43             throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");
44
45         return value < min ? min : (value > max ? max : value);
46     }

```

**7.40.2.2 static short TriDevs.TriEngine.Helpers.Math.Clamp ( short value, short min, short max ) [static]**

Clamps the specified short between a minimum and maximum value.

**Parameters**

<i>value</i>	Value to clamp.
<i>min</i>	If the specified short is below this value, then this will be returned.
<i>max</i>	If the specified short is above this value, then this will be returned.

**Returns**

The clamped value of the short.

```

56     {
57         if (min > max)
58             throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");
59
60         return value < min ? min : (value > max ? max : value);
61     }

```

**7.40.2.3 static ushort TriDevs.TriEngine.Helpers.Math.Clamp ( ushort value, ushort min, ushort max ) [static]**

Clamps the specified unsigned short between a minimum and maximum value.

**Parameters**

<i>value</i>	Value to clamp.
<i>min</i>	If the specified unsigned short is below this value, then this will be returned.
<i>max</i>	If the specified unsigned short is above this value, then this will be returned.

**Returns**

The clamped value of the unsigned short.

```

71     {
72         if (min > max)
73             throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");
74
75         return value < min ? min : (value > max ? max : value);
76     }

```

**7.40.2.4 static int TriDevs.TriEngine.Helpers.Math.Clamp ( int value, int min, int max ) [static]**

Clamps the specified integer between a minimum and maximum value.

**Parameters**

<i>value</i>	Value to clamp.
<i>min</i>	If the specified integer is below this value, then this will be returned.
<i>max</i>	If the specified integer is above this value, then this will be returned.

**Returns**

The clamped value of the integer.

```

86         {
87             if (min > max)
88                 throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");
89             return value < min ? min : (value > max ? max : value);
90         }
91     }

```

**7.40.2.5 static uint TriDevs.TriEngine.Helpers.Math.Clamp ( uint value, uint min, uint max ) [static]**

Clamps the specified unsigned integer between a minimum and maximum value.

**Parameters**

<i>value</i>	Value to clamp.
<i>min</i>	If the specified unsigned integer is below this value, then this will be returned.
<i>max</i>	If the specified unsigned integer is above this value, then this will be returned.

**Returns**

The clamped value of the unsigned integer.

```

101         {
102             if (min > max)
103                 throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");
104             return value < min ? min : (value > max ? max : value);
105         }
106     }

```

**7.40.2.6 static long TriDevs.TriEngine.Helpers.Math.Clamp ( long value, long min, long max ) [static]**

Clamps the specified 64-bit integer between a minimum and maximum value.

**Parameters**

<i>value</i>	Value to clamp.
<i>min</i>	If the specified 64-bit integer is below this value, then this will be returned.
<i>max</i>	If the specified 64-bit integer is above this value, then this will be returned.

**Returns**

The clamped value of the 64-bit integer.

```

116         {
117             if (min > max)
118                 throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");
119             return value < min ? min : (value > max ? max : value);
120         }
121     }

```

**7.40.2.7 static ulong TriDevs.TriEngine.Helpers.Math.Clamp ( ulong value, ulong min, ulong max ) [static]**

Clamps the specified 64-bit unsigned integer between a minimum and maximum value.

**Parameters**

<i>value</i>	Value to clamp.
<i>min</i>	If the specified 64-bit unsigned integer is below this value, then this will be returned.
<i>max</i>	If the specified 64-bit unsigned integer is above this value, then this will be returned.

**Returns**

The clamped value of the 64-bit unsigned integer.

```

131     {
132         if (min > max)
133             throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");
134
135         return value < min ? min : (value > max ? max : value);
136     }

```

**7.40.2.8 static float TriDevs.TriEngine.Helpers.Math.Clamp ( float *value*, float *min*, float *max* ) [static]**

Clamps the specified float between a minimum and maximum value.

**Parameters**

<i>value</i>	Value to clamp.
<i>min</i>	If the specified float is below this value, then this will be returned.
<i>max</i>	If the specified float is above this value, then this will be returned.

**Returns**

The clamped value of the float.

```

146     {
147         if (min > max)
148             throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");
149
150         return value < min ? min : (value > max ? max : value);
151     }

```

**7.40.2.9 static double TriDevs.TriEngine.Helpers.Math.Clamp ( double *value*, double *min*, double *max* ) [static]**

Clamps the specified double between a minimum and maximum value.

**Parameters**

<i>value</i>	Value to clamp.
<i>min</i>	If the specified double is below this value, then this will be returned.
<i>max</i>	If the specified double is above this value, then this will be returned.

**Returns**

The clamped value of the double.

```

161     {
162         if (min > max)
163             throw new ArgumentException("Minimum value cannot be greater than maximum value.", "min");
164
165         return value < min ? min : (value > max ? max : value);
166     }

```

**7.40.2.10 static decimal TriDevs.TriEngine.Helpers.Math.Clamp ( decimal *value*, decimal *min*, decimal *max* ) [static]**

Clamps the specified decimal between a minimum and maximum value.

**Parameters**

<i>value</i>	Value to clamp.
<i>min</i>	If the specified decimal is below this value, then this will be returned.
<i>max</i>	If the specified decimal is above this value, then this will be returned.

### Returns

The clamped value of the decimal.

```
176     {  
177         if (min > max)  
178             throw new ArgumentException("Minimum value cannot be grater than maximum value.", "min");  
179         return value < min ? min : (value > max ? max : value);  
180     }  
181 }
```

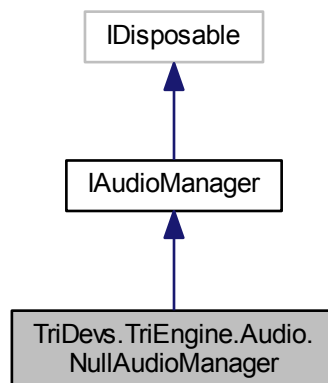
The documentation for this class was generated from the following file:

- TriDevs.TriEngine/Helpers/[Math.cs](#)

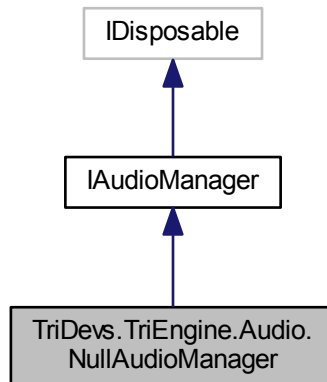
## 7.41 TriDevs.TriEngine.Audio.NullAudioManager Class Reference

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

Inheritance diagram for TriDevs.TriEngine.Audio.NullAudioManager:



Collaboration diagram for TriDevs.TriEngine.Audio.NullAudioManager:



## Public Member Functions

- void [Dispose](#) ()
- void [StopAll](#) ()  
*Immediately stops all playback of sounds and songs.*
- void [StopAllSounds](#) ()  
*Immediately stops playback of all sounds.*
- void [StopAllSongs](#) ()  
*Immediately stops playback of all songs.*

### 7.41.1 Detailed Description

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

### 7.41.2 Member Function Documentation

#### 7.41.2.1 void TriDevs.TriEngine.Audio.NullAudioManager.Dispose ( )

```
35     {  
36  
37     }
```

#### 7.41.2.2 void TriDevs.TriEngine.Audio.NullAudioManager.StopAll ( )

Immediately stops all playback of sounds and songs.

Implements [TriDevs.TriEngine.Audio.IAudioManager](#).

```
40     {  
41  
42     }
```



### 7.41.2.3 void TriDevs.TriEngine.Audio.NullAudioManager.StopAllSongs ( )

Immediately stops playback of all songs.

Implements [TriDevs.TriEngine.Audio.IAudioManager](#).

```
50      {  
51  
52      }
```

### 7.41.2.4 void TriDevs.TriEngine.Audio.NullAudioManager.StopAllSounds ( )

Immediately stops playback of all sounds.

Implements [TriDevs.TriEngine.Audio.IAudioManager](#).

```
45      {  
46  
47      }
```

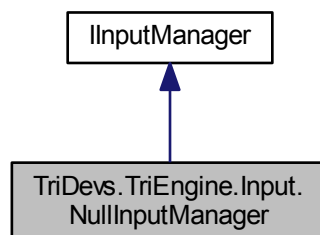
The documentation for this class was generated from the following file:

- [TriDevs.TriEngine/Audio/NullAudioManager.cs](#)

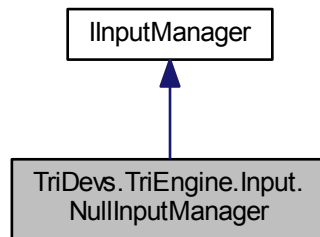
## 7.42 TriDevs.TriEngine.Input.NullInputManager Class Reference

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

Inheritance diagram for TriDevs.TriEngine.Input.NullInputManager:



Collaboration diagram for TriDevs.TriEngine.Input.NullInputManager:



## Public Member Functions

- void [Update](#) ()  
*Updates the input manager, refreshing all current and previous states.*
- bool [IsKeyUp](#) (Key key)  
*Returns whether or not the specified key is currently unpressed.*
- bool [IsKeyDown](#) (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 [IsMouseUp](#) (MouseButton button)  
*Returns whether or not the specified mouse button is currently unpressed.*
- bool [IsMouseDown](#) (MouseButton button)  
*Returns whether or not the specified mouse button is currently being pressed.*
- bool [MousePressed](#) (MouseButton button)  
*Returns whether or not the specified mouse button has been pressed.*
- bool [MouseReleased](#) (MouseButton button)  
*Returns whether or not the specified mouse button has been released.*
- bool [IsWheelUp](#) ()  
*Returns whether the mouse wheel was scrolled up.*
- bool [IsWheelDown](#) ()  
*Returns whether the mouse wheel was scrolled down.*
- bool [IsWheelChanged](#) ()  
*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< int > [MousePosition](#) [get]
- int [MouseWheelValue](#) [get]
- bool [this\[Key key\]](#) [get]
- bool [this\[MouseButton button\]](#) [get]

## Events

- [KeyDownEventHandler](#) [KeyDown](#)  
*Raised when a key is pressed down.*
- [KeyUpEventHandler](#) [KeyUp](#)  
*Raised when a key is released.*
- [KeyPressEventHandler](#) [KeyPress](#)  
*Raised when a character is typed.*
- [MouseDownEventHandler](#) [MouseDown](#)  
*Raised when a mouse button is pressed down.*
- [MouseUpEventHandler](#) [MouseUp](#)  
*Raised when a mouse button is released.*
- [MouseWheelChangedEventHandler](#) [WheelChanged](#)  
*Raised when the mouse wheel value changes.*
- [MouseWheelDownEventHandler](#) [WheelDown](#)  
*Raised when the mouse wheel is scrolled downwards.*
- [MouseWheelUpEventHandler](#) [WheelUp](#)  
*Raised when the mouse wheel is scrolled upwards.*

### 7.42.1 Detailed Description

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

### 7.42.2 Member Function Documentation

#### 7.42.2.1 `bool TriDevs.TriEngine.Input.NullInputManager.IsKeyDown ( 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.TriEngine.Input.IInputManager](#).

```

132         {
133             return false;
134         }
```

#### 7.42.2.2 `bool TriDevs.TriEngine.Input.NullInputManager.IsKeyUp ( 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.TriEngine.Input.IInputManager](#).

```
127     {
128         return true;
129     }
```

**7.42.2.3 bool TriDevs.TriEngine.Input.NullInputManager.IsMouseDown ( 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.TriEngine.Input.IInputManager](#).

```
152     {
153         return false;
154     }
```

**7.42.2.4 bool TriDevs.TriEngine.Input.NullInputManager.IsMouseUp ( 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.TriEngine.Input.IInputManager](#).

```
147     {
148         return true;
149     }
```

**7.42.2.5 bool TriDevs.TriEngine.Input.NullInputManager.IsWheelChanged ( )**

Returns whether the mouse wheel scrolled at all.

**Returns**

True if the mouse wheel scrolled, false otherwise.

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

```
177     {
178         return false;
179     }
```

#### 7.42.2.6 bool TriDevs.TriEngine.Input.NullInputManager.IsWheelDown ( )

Returns whether the mouse wheel was scrolled down.

##### Returns

True if mouse wheel was scrolled down, false otherwise.

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

```
172     {  
173         return false;  
174     }
```

#### 7.42.2.7 bool TriDevs.TriEngine.Input.NullInputManager.IsWheelUp ( )

Returns whether the mouse wheel was scrolled up.

##### Returns

True if mouse wheel was scrolled up, false otherwise.

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

```
167     {  
168         return false;  
169     }
```

#### 7.42.2.8 bool TriDevs.TriEngine.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.TriEngine.Input.IInputManager](#).

```
137     {  
138         return false;  
139     }
```

#### 7.42.2.9 bool TriDevs.TriEngine.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.TriEngine.Input.IInputManager](#).

```
142     {
143         return false;
144     }
```

**7.42.2.10 bool TriDevs.TriEngine.Input.NullInputManager.MousePressed ( 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.TriEngine.Input.IInputManager](#).

```
157     {
158         return false;
159     }
```

**7.42.2.11 bool TriDevs.TriEngine.Input.NullInputManager.MouseReleased ( 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.TriEngine.Input.IInputManager](#).

```
162     {
163         return false;
164     }
```

**7.42.2.12 void TriDevs.TriEngine.Input.NullInputManager.Update ( )**

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

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

```
122     {
123         // Do nothing
124     }
```

**7.42.2.13** `int TriDevs.TriEngine.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.TriEngine.Input.InputManager](#).

```

182         {
183             return 0;
184         }

```

**7.42.3 Property Documentation**

**7.42.3.1** `Point<int> TriDevs.TriEngine.Input.NullInputManager.MousePosition` `[get]`

**7.42.3.2** `int TriDevs.TriEngine.Input.NullInputManager.MouseWheelValue` `[get]`

**7.42.3.3** `int TriDevs.TriEngine.Input.NullInputManager.MouseX` `[get]`

**7.42.3.4** `int TriDevs.TriEngine.Input.NullInputManager.MouseY` `[get]`

**7.42.3.5** `bool TriDevs.TriEngine.Input.NullInputManager.this[Key key]` `[get]`

**7.42.3.6** `bool TriDevs.TriEngine.Input.NullInputManager.this[MouseButton button]` `[get]`

**7.42.4 Event Documentation**

**7.42.4.1** `KeyDownEventHandler TriDevs.TriEngine.Input.NullInputManager.KeyDown`

Raised when a key is pressed down.

This particular event is never raised, it's merely a placeholder in case a proper [InputManager](#) was not supplied.

**7.42.4.2** `KeyPressEventHandler TriDevs.TriEngine.Input.NullInputManager.KeyPress`

Raised when a character is typed.

This particular event is never raised, it's merely a placeholder in case a proper [InputManager](#) was not supplied.

**7.42.4.3** `KeyUpEventHandler TriDevs.TriEngine.Input.NullInputManager.KeyUp`

Raised when a key is released.

This particular event is never raised, it's merely a placeholder in case a proper [InputManager](#) was not supplied.

**7.42.4.4** `MouseDownEventHandler TriDevs.TriEngine.Input.NullInputManager.MouseDown`

Raised when a mouse button is pressed down.

This particular event is never raised, it's merely a placeholder in case a proper [InputManager](#) was not supplied.

#### 7.42.4.5 `MouseUpEventHandler` `TriDevs.TriEngine.Input.NullInputManager.MouseUp`

Raised when a mouse button is released.

This particular event is never raised, it's merely a placeholder in case a proper [InputManager](#) was not supplied.

#### 7.42.4.6 `MouseWheelChangedEventHandler` `TriDevs.TriEngine.Input.NullInputManager.WheelChanged`

Raised when the mouse wheel value changes.

This particular event is never raised, it's merely a placeholder in case a proper [InputManager](#) was not supplied.

#### 7.42.4.7 `MouseWheelDownEventHandler` `TriDevs.TriEngine.Input.NullInputManager.WheelDown`

Raised when the mouse wheel is scrolled downwards.

This particular event is never raised, it's merely a placeholder in case a proper [InputManager](#) was not supplied.

#### 7.42.4.8 `MouseWheelUpEventHandler` `TriDevs.TriEngine.Input.NullInputManager.WheelUp`

Raised when the mouse wheel is scrolled upwards.

This particular event is never raised, it's merely a placeholder in case a proper [InputManager](#) was not supplied.

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

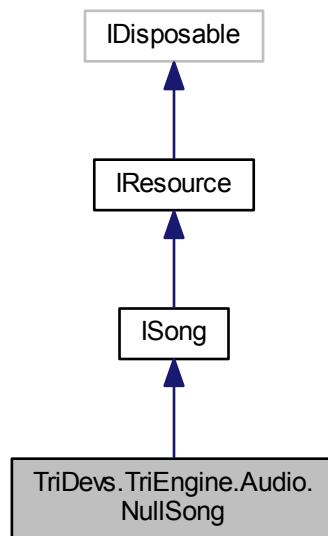
- `TriDevs.TriEngine/Input/NullInputManager.cs`

## 7.43 `TriDevs.TriEngine.Audio.NullSong` Class Reference

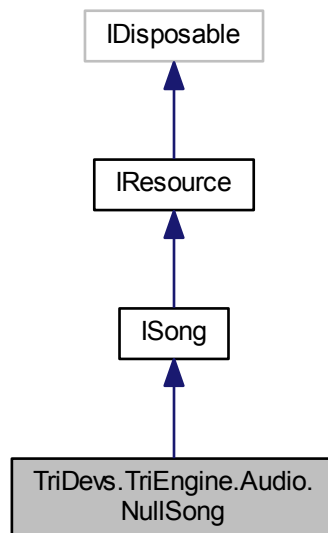
Fallback song class used in [NullAudioManager](#).



Inheritance diagram for TriDevs.TriEngine.Audio.NullSong:



Collaboration diagram for TriDevs.TriEngine.Audio.NullSong:



## Public Member Functions

- void [Dispose](#) ()

- void [Play](#) ()  
*Starts playback of the song.*
- void [Stop](#) ()  
*Stops playback of the song.*
- void [Pause](#) ()  
*Pauses playback of the song.*
- void [Resume](#) ()  
*Resumes playback of a paused song.*

## Properties

- string [Name](#) [get]
- string [File](#) [get]
- float [Volume](#) [get, set]
- bool [IsLooped](#) [get, set]

### 7.43.1 Detailed Description

Fallback song class used in [NullAudioManager](#).

### 7.43.2 Member Function Documentation

#### 7.43.2.1 void [TriDevs.TriEngine.Audio.NullSong.Dispose](#) ( )

```

37         {
38
39     }
```

#### 7.43.2.2 void [TriDevs.TriEngine.Audio.NullSong.Pause](#) ( )

Pauses playback of the song.

Call the [Resume](#) or [Play](#) method to resume playback.

Implements [TriDevs.TriEngine.Audio.ISong](#).

```

52         {
53
54     }
```

#### 7.43.2.3 void [TriDevs.TriEngine.Audio.NullSong.Play](#) ( )

Starts playback of the song.

Implements [TriDevs.TriEngine.Audio.ISong](#).

```

42         {
43
44     }
```

#### 7.43.2.4 void TriDevs.TriEngine.Audio.NullSong.Resume ( )

Resumes playback of a paused song.

Implements [TriDevs.TriEngine.Audio.ISong](#).

```
57         {  
58  
59     }
```

#### 7.43.2.5 void TriDevs.TriEngine.Audio.NullSong.Stop ( )

Stops playback of the song.

Implements [TriDevs.TriEngine.Audio.ISong](#).

```
47         {  
48  
49     }
```

### 7.43.3 Property Documentation

7.43.3.1 string TriDevs.TriEngine.Audio.NullSong.File [get]

7.43.3.2 bool TriDevs.TriEngine.Audio.NullSong.IsLooped [get], [set]

7.43.3.3 string TriDevs.TriEngine.Audio.NullSong.Name [get]

7.43.3.4 float TriDevs.TriEngine.Audio.NullSong.Volume [get], [set]

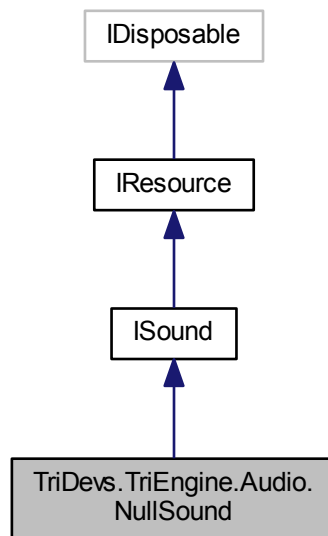
The documentation for this class was generated from the following file:

- TriDevs.TriEngine/Audio/[NullSong.cs](#)

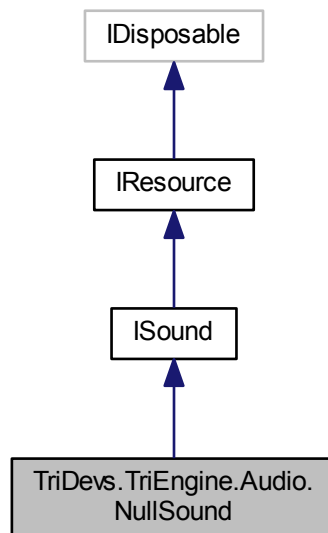
## 7.44 TriDevs.TriEngine.Audio.NullSound Class Reference

Fallback sound class used in [NullAudioManager](#).

Inheritance diagram for TriDevs.TriEngine.Audio.NullSound:



Collaboration diagram for TriDevs.TriEngine.Audio.NullSound:



## Public Member Functions

- void [Play](#) ()

- Plays the sound.*
- void [Stop](#) ()
- Stops the sound.*
- void [Dispose](#) ()

## Properties

- string [Name](#) [get]
- string [File](#) [get]

### 7.44.1 Detailed Description

Fallback sound class used in [NullAudioManager](#).

### 7.44.2 Member Function Documentation

#### 7.44.2.1 void TriDevs.TriEngine.Audio.NullSound.Dispose ( )

```
45         {  
46  
47         }
```

#### 7.44.2.2 void TriDevs.TriEngine.Audio.NullSound.Play ( )

Plays the sound.

Implements [TriDevs.TriEngine.Audio.ISound](#).

```
35         {  
36  
37         }
```

#### 7.44.2.3 void TriDevs.TriEngine.Audio.NullSound.Stop ( )

Stops the sound.

Implements [TriDevs.TriEngine.Audio.ISound](#).

```
40         {  
41  
42         }
```

### 7.44.3 Property Documentation

#### 7.44.3.1 string TriDevs.TriEngine.Audio.NullSound.File [get]

#### 7.44.3.2 string TriDevs.TriEngine.Audio.NullSound.Name [get]

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

- [TriDevs.TriEngine/Audio/NullSound.cs](#)

## 7.45 TriDevs.TriEngine.Point< T > Struct Template Reference

A struct representing an X/Y/Z coordinate.

### Public Member Functions

- [Point](#) (T x, T y, T z=default(T))  
*Creates a new Point<T> with the specified X and Y values.*

### Public Attributes

- [T X](#)  
*The X value of the coordinate.*
- [T Y](#)  
*The Y value of the coordinate.*
- [T Z](#)  
*The Z value of the coordinate.*

#### 7.45.1 Detailed Description

A struct representing an X/Y/Z coordinate.

#### Template Parameters

<b>T</b>	The type used for the X, Y and Z members.
----------	---

#### Type Constraints

**T : struct**

#### 7.45.2 Constructor & Destructor Documentation

##### 7.45.2.1 TriDevs.TriEngine.Point< T >.Point ( T x, T y, T z = default (T) )

Creates a new Point<T> with the specified X and Y values.

#### Parameters

<b>x</b>	The X value.
<b>y</b>	The Y value.
<b>z</b>	The Z value.

```

54     {
55         X = x;
56         Y = y;
57         Z = z;
58     }
```

#### 7.45.3 Member Data Documentation

##### 7.45.3.1 T TriDevs.TriEngine.Point< T >.X

The X value of the coordinate.

### 7.45.3.2 T TriDevs.TriEngine.Point< T >.Y

The Y value of the coordinate.

### 7.45.3.3 T TriDevs.TriEngine.Point< T >.Z

The Z value of the coordinate.

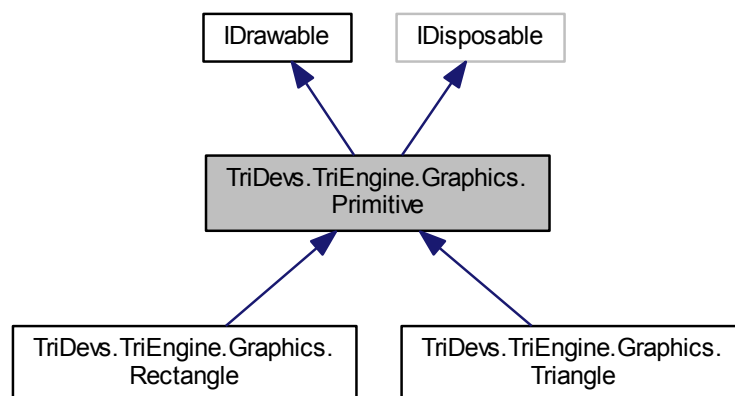
The documentation for this struct was generated from the following file:

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

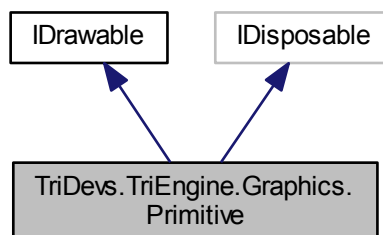
## 7.46 TriDevs.TriEngine.Graphics.Primitive Class Reference

Represents a primitive 2D shape composed of triangles.

Inheritance diagram for TriDevs.TriEngine.Graphics.Primitive:



Collaboration diagram for TriDevs.TriEngine.Graphics.Primitive:



## Public Member Functions

- void [Draw](#) ()  
*Draw the object to screen.*
- void [Dispose](#) ()

## Protected Member Functions

- [Primitive](#) (ushort[] indices, Vector3[] vectors, [Color](#)[] colors=null)
- [Primitive](#) (ushort[] indices, float[] vertices, [Color](#)[] colors=null)

## Protected Attributes

- uint[] [Ids](#)
- uint [ColorId](#)
- ushort[] [Indices](#)
- float[] [Vertices](#)
- int[] [Colors](#)

## Properties

- uint [IndicesID](#) [get]  
*Indices buffer ID assigned to this primitive by GL.BindBuffer.*
- uint [VerticesID](#) [get]  
*Vertices buffer ID assigned to this primitive by GL.BindBuffer.*
- uint [ColorID](#) [get]  
*Color buffer ID assigned to this primitive by GL.BindBuffer.*

### 7.46.1 Detailed Description

Represents a primitive 2D shape composed of triangles.

### 7.46.2 Constructor & Destructor Documentation

#### 7.46.2.1 TriDevs.TriEngine.Graphics.Primitive.Primitive ( ushort[] indices, Vector3[] vectors, Color[] colors = null ) [protected]

```
60         : this(indices, vectors.ToFloatArray(), colors)
61         {
62         }
```

#### 7.46.2.2 TriDevs.TriEngine.Graphics.Primitive.Primitive ( ushort[] indices, float[] vertices, Color[] colors = null ) [protected]

```
65         {
66             if (indices.Length % 3 != 0)
67                 throw new EngineException(
68                     "Primitives can only be composed of a series of triangles. Expected n*3 points, got " +
69                     vertices.Length + ".",
70                     new ArgumentException("Unexpected number of array items.", "indices"));
71
72             Indices = indices;
73             Vertices = vertices;
74
75             Ids = new uint[2];
76         }
```



```

77         GL.GenBuffers(2, Ids);
78
79         GL.BindBuffer(BufferTarget.ElementArrayBuffer, IndicesID);
80         GL.BufferData(BufferTarget.ElementArrayBuffer, new IntPtr(Indices.Length * sizeof(ushort)), Indices, BufferUsageHint.StaticDraw);
81         GL.BindBuffer(BufferTarget.ElementArrayBuffer, 0);
82
83         GL.BindBuffer(BufferTarget.ArrayBuffer, VerticesID);
84         GL.BufferData(BufferTarget.ArrayBuffer, new IntPtr(Vertices.Length * sizeof(float)), Vertices, BufferUsageHint.StaticDraw);
85         GL.BindBuffer(BufferTarget.ArrayBuffer, 0);
86
87         if (colors != null)
88         {
89             Colors = new int[colors.Length];
90             for (int i = 0; i < colors.Length; i++)
91                 Colors[i] = colors[i].ToColor4().ToArgb();
92
93             GL.GenBuffers(1, out ColorID);
94             GL.BindBuffer(BufferTarget.ArrayBuffer, ColorID);
95             GL.BufferData(BufferTarget.ArrayBuffer, new IntPtr(Colors.Length * sizeof(int)), Colors, BufferUsageHint.StaticDraw);
96             GL.BindBuffer(BufferTarget.ArrayBuffer, 0);
97         }
98     }

```

## 7.46.3 Member Function Documentation

### 7.46.3.1 void TriDevs.TriEngine.Graphics.Primitive.Dispose ( )

```

123     {
124         GL.DeleteBuffers(2, Ids);
125
126         if (ColorID != 0)
127             GL.DeleteBuffers(1, ref ColorID);
128     }

```

### 7.46.3.2 void TriDevs.TriEngine.Graphics.Primitive.Draw ( )

Draw the object to screen.

Implements [TriDevs.TriEngine.Interfaces.IDrawable](#).

```

101     {
102         GL.PushClientAttrib(ClientAttribMask.ClientVertexArrayBit);
103
104         GL.EnableClientState(ArrayCap.VertexArray);
105
106         if (ColorID != 0)
107         {
108             GL.EnableClientState(ArrayCap.ColorArray);
109             GL.BindBuffer(BufferTarget.ArrayBuffer, ColorID);
110             GL.ColorPointer(sizeof(int), ColorPointerType.UnsignedByte, 0, 0);
111         }
112
113         GL.BindBuffer(BufferTarget.ArrayBuffer, VerticesID);
114         GL.VertexPointer(3, VertexPointerType.Float, 0, 0);
115
116         GL.BindBuffer(BufferTarget.ElementArrayBuffer, IndicesID);
117         GL.DrawElements(BeginMode.Triangles, Indices.Length, DrawElementsType.UnsignedShort, 0);
118
119         GL.PopClientAttrib();
120     }

```

## 7.46.4 Member Data Documentation

7.46.4.1 uint TriDevs.TriEngine.Graphics.Primitive.ColorId [protected]

7.46.4.2 int [] TriDevs.TriEngine.Graphics.Primitive.Colors [protected]

7.46.4.3 uint [] TriDevs.TriEngine.Graphics.Primitive.Ids [protected]

7.46.4.4 `ushort [] TriDevs.TriEngine.Graphics.Primitive.Indices` `[protected]`

7.46.4.5 `float [] TriDevs.TriEngine.Graphics.Primitive.Vertices` `[protected]`

## 7.46.5 Property Documentation

7.46.5.1 `uint TriDevs.TriEngine.Graphics.Primitive.ColorID` `[get]`

[Color](#) buffer ID assigned to this primitive by `GL.BindBuffer`.

7.46.5.2 `uint TriDevs.TriEngine.Graphics.Primitive.IndicesID` `[get]`

Indices buffer ID assigned to this primitive by `GL.BindBuffer`.

7.46.5.3 `uint TriDevs.TriEngine.Graphics.Primitive.VerticesID` `[get]`

Vertices buffer ID assigned to this primitive by `GL.BindBuffer`.

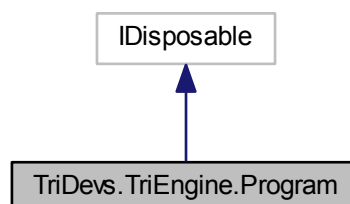
The documentation for this class was generated from the following file:

- `TriDevs.TriEngine/Graphics/Primitive.cs`

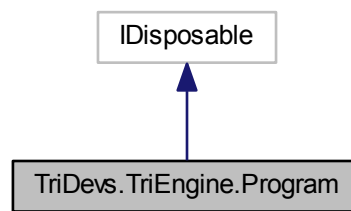
## 7.47 TriDevs.TriEngine.Program Class Reference

An OpenGL program.

Inheritance diagram for `TriDevs.TriEngine.Program`:



Collaboration diagram for TriDevs.TriEngine.Program:



## Public Member Functions

- `Program` (params `Shader[]` shaders)  
*Initializes a new `Program` as a shader program.*
- `void Dispose ()`

## Public Attributes

- readonly int `ID`  
*The ID of this program.*

### 7.47.1 Detailed Description

An OpenGL program.

### 7.47.2 Constructor & Destructor Documentation

#### 7.47.2.1 TriDevs.TriEngine.Program.Program ( params Shader[] shaders )

Initializes a new `Program` as a shader program.

#### Parameters

<i>shaders</i>	The shaders to attach.
----------------	------------------------

```

45     {
46         ID = GL.CreateProgram();
47
48         foreach (var shader in shaders)
49         {
50             GL.AttachShader(ID, shader.ID);
51         }
52
53         GL.LinkProgram(ID);
54     }
  
```

### 7.47.3 Member Function Documentation

### 7.47.3.1 void TriDevs.TriEngine.Program.Dispose ( )

```

57         {
58             GL.DeleteProgram(ID);
59         }

```

## 7.47.4 Member Data Documentation

### 7.47.4.1 readonly int TriDevs.TriEngine.Program.ID

The ID of this program.

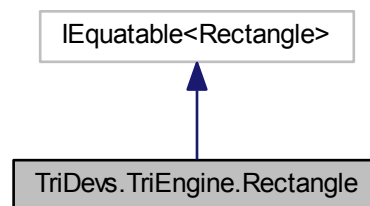
The documentation for this class was generated from the following file:

- TriDevs.TriEngine/[Program.cs](#)

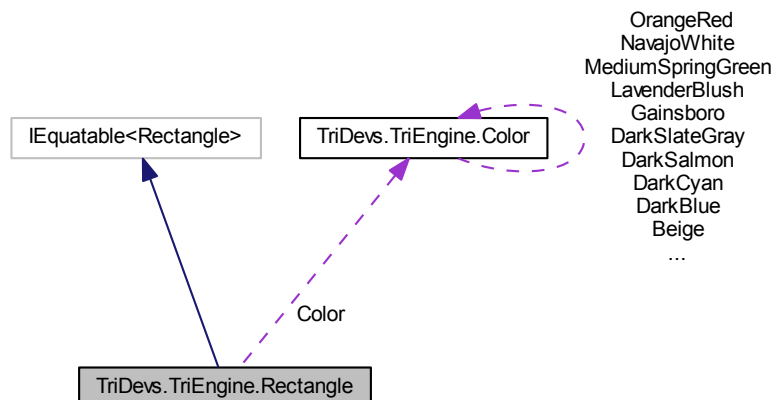
## 7.48 TriDevs.TriEngine.Rectangle Struct Reference

A rectangle representing an area in 2D space.

Inheritance diagram for TriDevs.TriEngine.Rectangle:



Collaboration diagram for TriDevs.TriEngine.Rectangle:



## Public Member Functions

- [Rectangle](#) (Point< int > position, Point< int > size, [Color](#)?color=null)  
*Initializes a new [Rectangle](#) with position and size based on two point objects.*
- [Rectangle](#) (int x, int y, int width, int height, [Color](#)?color=null)  
*Initializes a new rectangle with specified position and size.*
- bool [Intersects](#) ([Rectangle](#) other)  
*Returns whether this rectangle is intersecting with another rectangle.*
- bool [Equals](#) ([Rectangle](#) other)

## Public Attributes

- readonly int [X](#)  
*The X position of this rectangle, in screen pixels.*
- readonly int [Y](#)  
*The Y position of this rectangle, in screen pixels.*
- readonly int [Width](#)  
*The width of this rectangle in pixels.*
- readonly int [Height](#)  
*The height of this rectangle in pixels.*
- readonly [Color](#) [Color](#)  
*Optional color of this rectangle, if it is to be drawn onto the screen.*

### 7.48.1 Detailed Description

A rectangle representing an area in 2D space.

### 7.48.2 Constructor & Destructor Documentation

#### 7.48.2.1 TriDevs.TriEngine.Rectangle.Rectangle ( Point< int > position, Point< int > size, [Color](#)? color = null )

Initializes a new [Rectangle](#) with position and size based on two point objects.

##### Parameters

<i>position</i>	The point to get position from.
<i>size</i>	The point to get width and height from.
<i>color</i>	<a href="#">Color</a> of this rectangle, set to null for default color of black.

```

66         : this(position.X, position.Y, size.X, size.Y, color)
67     {
68
69     }
```

#### 7.48.2.2 TriDevs.TriEngine.Rectangle.Rectangle ( int x, int y, int width, int height, [Color](#)? color = null )

Initializes a new rectangle with specified position and size.

##### Parameters

<i>x</i>	X-position of this rectangle, in screen pixels.
<i>y</i>	Y-position of this rectangle, in screen pixels.
<i>width</i>	Width of this rectangle, in pixels.
<i>height</i>	Height of this rectangle, in pixels.
<i>color</i>	<a href="#">Color</a> of this rectangle, set to null for default color of black.

```

80     {
81         X = x;
82         Y = y;
83         Width = width;
84         Height = height;
85         Color = color.HasValue ? color.Value : Color.Black;
86     }

```

### 7.48.3 Member Function Documentation

#### 7.48.3.1 bool TriDevs.TriEngine.Rectangle.Equals ( Rectangle *other* )

```

114     {
115         return X == other.X && Y == other.Y && Width == other.Width &&
Height == other.Height;
116     }

```

#### 7.48.3.2 bool TriDevs.TriEngine.Rectangle.Intersects ( Rectangle *other* )

Returns whether this rectangle is intersecting with another rectangle.

##### Parameters

<i>other</i>	Rectangle to check against.
--------------	-----------------------------

##### Returns

True if this rectangle is intersecting with the other rectangle, false otherwise.

```

94     {
95         if (Equals(other))
96             return true;
97
98         if (X >= other.X && (X + Width) <= (other.X + other.Width))
99             return true;
100
101         if (X < other.X && (X + Width) >= other.X)
102             return true;
103
104         if (Y >= other.Y && (Y + Height) <= (other.Y + other.Height))
105             return true;
106
107         if (Y < other.Y && (Y + Height) >= other.Y)
108             return true;
109
110         return false;
111     }

```

### 7.48.4 Member Data Documentation

#### 7.48.4.1 readonly Color TriDevs.TriEngine.Rectangle.Color

Optional color of this rectangle, if it is to be drawn onto the screen.

#### 7.48.4.2 readonly int TriDevs.TriEngine.Rectangle.Height

The height of this rectangle in pixels.

#### 7.48.4.3 readonly int TriDevs.TriEngine.Rectangle.Width

The width of this rectangle in pixels.

#### 7.48.4.4 readonly int TriDevs.TriEngine.Rectangle.X

The X position of this rectangle, in screen pixels.

#### 7.48.4.5 readonly int TriDevs.TriEngine.Rectangle.Y

The Y position of this rectangle, in screen pixels.

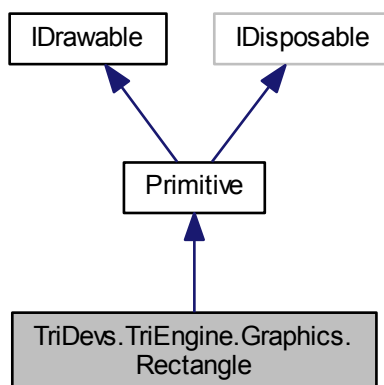
The documentation for this struct was generated from the following file:

- TriDevs.TriEngine/[Rectangle.cs](#)

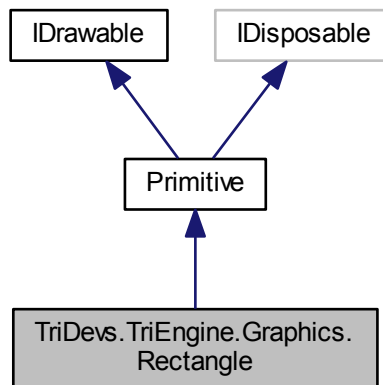
## 7.49 TriDevs.TriEngine.Graphics.Rectangle Class Reference

A primitive 2D [Rectangle](#) shape.

Inheritance diagram for TriDevs.TriEngine.Graphics.Rectangle:



Collaboration diagram for TriDevs.TriEngine.Graphics.Rectangle:



## Public Member Functions

- [Rectangle](#) ([TriEngine.Rectangle](#) rect)  
Creates a new 2D [Rectangle](#) with the specified settings.

## Additional Inherited Members

### 7.49.1 Detailed Description

A primitive 2D [Rectangle](#) shape.

### 7.49.2 Constructor & Destructor Documentation

#### 7.49.2.1 TriDevs.TriEngine.Graphics.Rectangle.Rectangle ( [TriEngine.Rectangle](#) rect )

Creates a new 2D [Rectangle](#) with the specified settings.

#### Parameters

<i>rect</i>	The rectangle defining the position and size of this primitive.
-------------	---

```

38         : base(new ushort[]
39         {
40             // First triangle
41             0, 1, 2,
42
43             // Second triangle
44             0, 2, 3
45         },

```

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

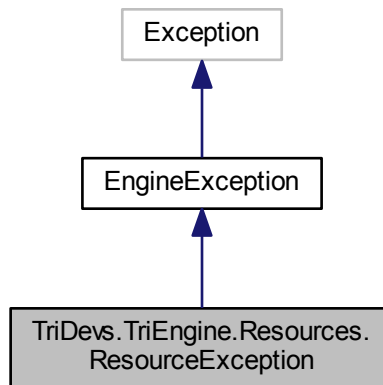
- TriDevs.TriEngine/Graphics/[Rectangle.cs](#)



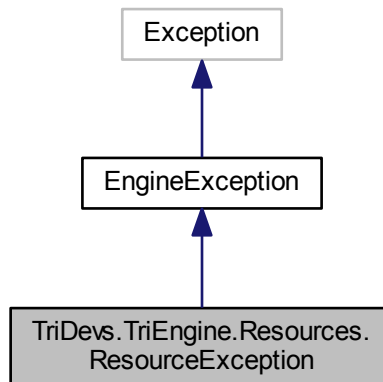
## 7.50 TriDevs.TriEngine.Resources.ResourceException Class Reference

Exception for resource-related issues. Thrown when there is a problem in [ResourceManager](#).

Inheritance diagram for TriDevs.TriEngine.Resources.ResourceException:



Collaboration diagram for TriDevs.TriEngine.Resources.ResourceException:



### 7.50.1 Detailed Description

Exception for resource-related issues. Thrown when there is a problem in [ResourceManager](#).

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

- TriDevs.TriEngine/Resources/[ResourceException.cs](#)

## 7.51 TriDevs.TriEngine.Resources.ResourceManager Class Reference

Static class to manage resources.

### Static Public Member Functions

- static [Font LoadFont](#) (string name, string file, int size, bool dropShadow=false, [FontType](#) type=FontType.TTF)  
*Loads a font file from the default resources path into the resources.*
- static [Font LoadFont](#) (string name, string file, int size, [FontType](#) type, [FontConstructionConfig](#) config)  
*Loads a font file from the default resources path into the resources.*
- static [Shader LoadShader](#) (string name, string file, ShaderType type)  
*Loads a shader file from the default resources path into the resources.*
- static [ISound LoadSound](#) (string name, string file, [AudioFormat](#) format=AudioFormat.Wav)  
*Loads a sound file from the default resources path into the resources.*
- static [ISong LoadSong](#) (string name, string file, [AudioFormat](#) format=AudioFormat.Ogg)  
*Loads a song file from the default resources path into the resources.*
- static bool [Has](#) (string name)  
*Checks if the resource with the specified name has been added to the resource collection.*
- static bool [Has< T >](#) (string name)  
*Checks if the resource with the specified name and type has been added to the resource collection.*
- static [IResource Get](#) (string name)  
*Gets the resource with the specified name.*
- static T [Get< T >](#) (string name)  
*Gets the resource with the specified name and casts it to the specified type.*
- static IEnumerable< T > [GetAll< T >](#) ()  
*Gets all resources of the specified type.*
- static void [Add](#) ([IResource](#) resource)  
*Adds a resource to the resource collection.*

### Static Public Attributes

- static string [BasePath](#) = "Resources"  
*Base path to the resources directory, relative to the current working directory.*
- static string [FontPath](#) = "Fonts"  
*Path to the fonts directory, relative to [BasePath](#).*
- static string [ShaderPath](#) = "Shaders"  
*Path to the shaders directory, relative to [BasePath](#).*
- static string [SoundPath](#) = "Sounds"  
*Path to the sounds directory, relative to [BasePath](#).*
- static string [SongPath](#) = "Songs"  
*Path to the songs directory, relative to [BasePath](#).*

#### 7.51.1 Detailed Description

Static class to manage resources.

## 7.51.2 Member Function Documentation

### 7.51.2.1 static void TriDevs.TriEngine.Resources.ResourceManager.Add ( IResource *resource* ) [static]

Adds a resource to the resource collection.

#### Parameters

<i>resource</i>	Resource to add.
-----------------	------------------

#### Exceptions

<a href="#">ResourceException</a>	Thrown if the collection already contains a resource with the same name.
-----------------------------------	--

```

259         {
260             if (Has(resource.Name))
261                 throw new ResourceException("Attempted to add resource that already exists: " + resource.
Name);
262
263             Resources.Add(resource.Name, resource);
264         }
```

### 7.51.2.2 static IResource TriDevs.TriEngine.Resources.ResourceManager.Get ( string *name* ) [static]

Gets the resource with the specified name.

#### Parameters

<i>name</i>	Name of resource to get.
-------------	--------------------------

#### Exceptions

<a href="#">ResourceException</a>	Thrown if the resource does not exist.
-----------------------------------	--

#### Returns

The [IResource](#) object with the specified name.

```

212         {
213             if (!Has(name))
214                 throw new ResourceException("Attempted to get non-existing resource \"" + name + "\"!");
215
216             return Resources[name];
217         }
```

### 7.51.2.3 static T TriDevs.TriEngine.Resources.ResourceManager.Get< T > ( string *name* ) [static]

Gets the resource with the specified name and casts it to the specified type.

#### Template Parameters

<i>T</i>	Type to cast to.
----------	------------------

#### Parameters

<i>name</i>	Name of resource to get.
-------------	--------------------------

## Exceptions

<a href="#"><i>ResourceException</i></a>	Thrown if the resource could not be found or if the cast failed.
--	--

## Returns

The resource object of type T with the specified name.

## Type Constraints

**T : class**

**T : *IResource***

```

227                                     : class, IResource
228     {
229         if (!Has(name))
230             throw new ResourceException("Attempted to get non-existing resource \"" + name + "\"!");
231
232         T resource = Get(name) as T;
233
234         if (resource == null)
235             throw new ResourceException("Resource with name \"" + name +
236                                         "\" is not of the requested type: " + typeof(T));
237
238         return resource;
239     }

```

#### 7.51.2.4 static IEnumerable<T> TriDevs.TriEngine.Resources.ResourceManager.GetAll<T>( ) [static]

Gets all resources of the specified type.

## Template Parameters

<b>T</b>	Type of resource to get.
----------	--------------------------

## Returns

An IEnumerable containing the relevant resources.

Returned collection will be empty if no matching resources were found.

## Type Constraints

**T : class**

**T : *IResource***

```

247                                     : class, IResource
248     {
249         return Resources.Values.Where(r => r is T).Cast<T>();
250     }

```

#### 7.51.2.5 static bool TriDevs.TriEngine.Resources.ResourceManager.Has( string name ) [static]

Checks if the resource with the specified name has been added to the resource collection.

## Parameters

<i>name</i>	Name to search for.
-------------	---------------------

## Returns

True if the resource has been added, false otherwise.

```

190     {
191         return Resources.ContainsKey(name);
192     }

```

### 7.51.2.6 static bool TriDevs.TriEngine.Resources.ResourceManager.Has< T > ( string name ) [static]

Checks if the resource with the specified name and type has been added to the resource collection.

## Template Parameters

<i>T</i>	Type of resource to search for.
----------	---------------------------------

## Parameters

<i>name</i>	Name to search for.
-------------	---------------------

## Returns

True if the resource has been added, false otherwise.

## Type Constraints

***T* : class**

***T* : IResource**

```

200                                     : class, IResource
201     {
202         return Has(name) && Resources[name].GetType() == typeof (T);
203     }

```

### 7.51.2.7 static Font TriDevs.TriEngine.Resources.ResourceManager.LoadFont ( string name, string file, int size, bool dropShadow = false, FontType type = FontType.TTF ) [static]

Loads a font file from the default resources path into the resources.

## Parameters

<i>name</i>	Name to assign the font, or null to auto-generate one.
<i>file</i>	Font file to load.
<i>size</i>	Size (in points) to use for the font.
<i>dropShadow</i>	Whether or not the font should have shadows.
<i>type</i>	The font filetype.

## Returns

The newly loaded font object, or existing font object if one with matching name was found.

```

84     {
85         return LoadFont(name, file, size, type,
86             new FontConstructionConfig(new QFontBuilderConfiguration(
dropShadow),
87                                     new QFontLoaderConfiguration(dropShadow)));
88     }

```

### 7.51.2.8 static Font TriDevs.TriEngine.Resources.ResourceManager.LoadFont ( string name, string file, int size, FontType type, FontConstructionConfig config ) [static]

Loads a font file from the default resources path into the resources.

#### Parameters

<i>name</i>	Name to assign the font, or null to auto-generate one.
<i>file</i>	Font file to load.
<i>size</i>	Size (in points) to use for the font.
<i>type</i>	The font filetype.
<i>config</i>	The relevant font construction configs.

#### Returns

The newly loaded font object, or existing font object if one with matching name was found.

```

100     {
101         if (Has<Font>(name))
102             return Get<Font>(name);
103
104         file = Path.Combine(BasePath, FontPath, file);
105
106         var font = new Font(name, file, size, type, config);
107         Add(font);
108         return font;
109     }
```

### 7.51.2.9 static Shader TriDevs.TriEngine.Resources.ResourceManager.LoadShader ( string name, string file, ShaderType type ) [static]

Loads a shader file from the default resources path into the resources.

#### Parameters

<i>name</i>	Name to assign the shader, or null to auto-generate one.
<i>file</i>	File to load shader code from.
<i>type</i>	The type of shader.

#### Returns

The newly loaded shader object, or existing shader object if one with matching name was found.

```

123     {
124         if (Has<Shader>(name))
125             return Get<Shader>(name);
126
127         file = Path.Combine(BasePath, ShaderPath, file);
128
129         var shader = new Shader(name, file, type);
130         Add(shader);
131         return shader;
132     }
```

### 7.51.2.10 static ISong TriDevs.TriEngine.Resources.ResourceManager.LoadSong ( string name, string file, AudioFormat format = AudioFormat.Ogg ) [static]

Loads a song file from the default resources path into the resources.

#### Parameters

<i>name</i>	Name to assign the song.
<i>file</i>	File to load song from.
<i>format</i>	The audio format of the song.

**Returns**

The newly loaded song object, or existing song object if one with matching name was found.

```

169     {
170         if (Has<ISong>(name))
171             return Get<ISong>(name);
172
173         file = Path.Combine(BasePath, SongPath, file);
174
175         var song = new Song(name, file, format);
176         Add(song);
177         return song;
178     }

```

#### 7.51.2.11 static ISound TriDevs.TriEngine.Resources.ResourceManager.LoadSound ( string name, string file, AudioFormat format = AudioFormat.Wav ) [static]

Loads a sound file from the default resources path into the resources.

**Parameters**

<i>name</i>	Name to assign the sound.
<i>file</i>	File to load sound from.
<i>format</i>	The audio format of the sound.

**Returns**

The newly loaded sound object, or existing sound object if one with matching name was found.

```

146     {
147         if (Has<ISound>(name))
148             return Get<ISound>(name);
149
150         file = Path.Combine(BasePath, SoundPath, file);
151
152         var sound = new Sound(name, file, format);
153         Add(sound);
154         return sound;
155     }

```

**7.51.3 Member Data Documentation**

#### 7.51.3.1 string TriDevs.TriEngine.Resources.ResourceManager.BasePath = "Resources" [static]

Base path to the resources directory, relative to the current working directory.

#### 7.51.3.2 string TriDevs.TriEngine.Resources.ResourceManager.FontPath = "Fonts" [static]

Path to the fonts directory, relative to [BasePath](#).

#### 7.51.3.3 string TriDevs.TriEngine.Resources.ResourceManager.ShaderPath = "Shaders" [static]

Path to the shaders directory, relative to [BasePath](#).

#### 7.51.3.4 string TriDevs.TriEngine.Resources.ResourceManager.SongPath = "Songs" [static]

Path to the songs directory, relative to [BasePath](#).

7.51.3.5 `string TriDevs.TriEngine.Resources.ResourceManager.SoundPath = "Sounds" [static]`

Path to the sounds directory, relative to [BasePath](#).

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

- TriDevs.TriEngine/Resources/[ResourceManager.cs](#)

## 7.52 TriDevs.TriEngine.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.*

### 7.52.1 Detailed Description

Provides serialization methods.

### 7.52.2 Member Function Documentation

7.52.2.1 `static T TriDevs.TriEngine.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     }

```

### 7.52.2.2 static string TriDevs.TriEngine.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     }

```

### 7.52.2.3 static void TriDevs.TriEngine.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.TriEngine/Serializing/[Serializer.cs](#)

## 7.53 TriDevs.TriEngine.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.*
- static void [Provide](#) ([IAudioManager](#) audio)  
*Specifies an audio manager service to provide.*
- static void [Provide](#) ([IInputManager](#) input, [IAudioManager](#) audio)  
*Specifies what services to provide.*

### Properties

- static [IInputManager](#) [Input](#) [get]  
*The input manager service.*
- static [IAudioManager](#) [Audio](#) [get]  
*The audio manager service.*

#### 7.53.1 Detailed Description

Provides different game-related service interfaces.

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

#### 7.53.2 Member Function Documentation

##### 7.53.2.1 static void TriDevs.TriEngine.Services.Provide ( [IInputManager](#) *input* ) [static]

Specifies an input manager service to provide.

##### Parameters

<i>input</i>	An object implementing the <a href="#">IInputManager</a> interface.
--------------	---

```

57     {
58         _input = input;
59     }
```

##### 7.53.2.2 static void TriDevs.TriEngine.Services.Provide ( [IAudioManager](#) *audio* ) [static]

Specifies an audio manager service to provide.

##### Parameters

<i>audio</i>	An object implementing the <a href="#">IAudioManager</a> interface.
--------------	---

```

66     {
67         _audio = audio;
68     }
```

7.53.2.3 `static void TriDevs.TriEngine.Services.Provide ( IInputManager input, IAudioManager audio )` `[static]`

Specifies what services to provide.

#### Parameters

<i>input</i>	The input service to provide.
<i>audio</i>	The audio service to provide.

```

76      {
77          Provide(input);
78          Provide(audio);
79      }

```

### 7.53.3 Property Documentation

7.53.3.1 `IAudioManager TriDevs.TriEngine.Services.Audio` `[static], [get]`

The audio manager service.

7.53.3.2 `IInputManager TriDevs.TriEngine.Services.Input` `[static], [get]`

The input manager service.

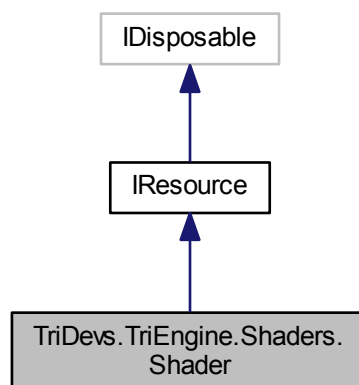
The documentation for this class was generated from the following file:

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

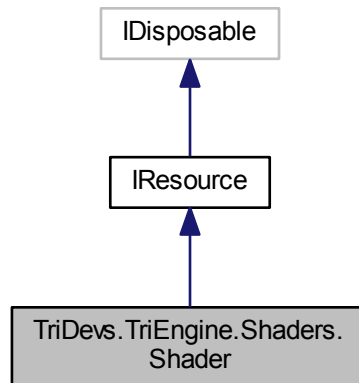
## 7.54 TriDevs.TriEngine.Shaders.Shader Class Reference

GLSL shader object loaded and compiled from a \*.glsl shader file.

Inheritance diagram for TriDevs.TriEngine.Shaders.Shader:



Collaboration diagram for TriDevs.TriEngine.Shaders.Shader:



### Public Member Functions

- [Shader](#) (string name, string file, ShaderType type)  
*Creates a new shader from specified GLSL source file.*
- void [Dispose](#) ()

### Static Public Member Functions

- static string [GetDefaultName](#) (string file)  
*Returns an auto-generated shader name based on the file name.*

### Public Attributes

- readonly int [ID](#)  
*ID of the shader compiled by OpenGL.*

### Properties

- string [Name](#) [get]  
*The name of this shader object.*
- string [File](#) [get]  
*The file containing the source for this shader.*

#### 7.54.1 Detailed Description

GLSL shader object loaded and compiled from a \*.glsl shader file.

## 7.54.2 Constructor & Destructor Documentation

### 7.54.2.1 TriDevs.TriEngine.Shaders.Shader.Shader ( string *name*, string *file*, ShaderType *type* )

Creates a new shader from specified GLSL source file.

#### Parameters

<i>name</i>	The name to give to this shader, or null to let constructor auto-generate a name based on the file name.
<i>file</i>	GLSL source to use.
<i>type</i>	The type of shader to create.

```

63     {
64         _file = file;
65         _name = name ?? GetDefaultName(_file);
66         ID = GL.CreateShader(type);
67         var source = System.IO.File.ReadAllText(_file);
68         GL.ShaderSource(ID, source);
69         GL.CompileShader(ID);
70     }

```

## 7.54.3 Member Function Documentation

### 7.54.3.1 void TriDevs.TriEngine.Shaders.Shader.Dispose ( )

```

83     {
84         GL.DeleteShader(ID);
85     }

```

### 7.54.3.2 static string TriDevs.TriEngine.Shaders.Shader.GetDefaultName ( string *file* ) [static]

Returns an auto-generated shader name based on the file name.

#### Parameters

<i>file</i>	The file name.
-------------	----------------

#### Returns

The auto-generated shader name.

```

78     {
79         return Path.GetFileNameWithoutExtension(file);
80     }

```

## 7.54.4 Member Data Documentation

### 7.54.4.1 readonly int TriDevs.TriEngine.Shaders.Shader.ID

ID of the shader compiled by OpenGL.

## 7.54.5 Property Documentation

### 7.54.5.1 string TriDevs.TriEngine.Shaders.Shader.File [get]

The file containing the source for this shader.

#### 7.54.5.2 `string TriDevs.TriEngine.Shaders.Shader.Name` [get]

The name of this shader object.

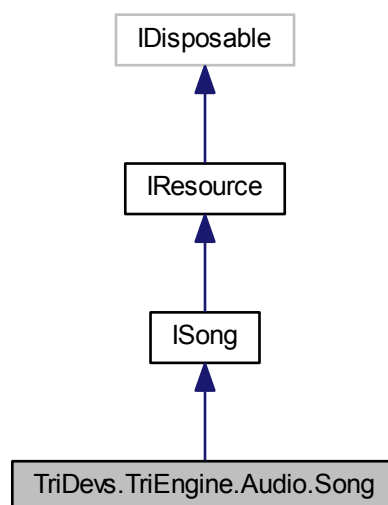
The documentation for this class was generated from the following file:

- `TriDevs.TriEngine/Shaders/Shader.cs`

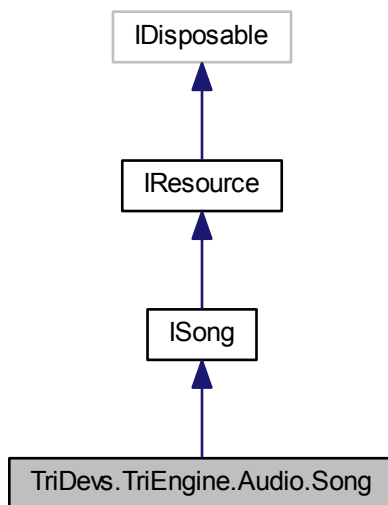
## 7.55 `TriDevs.TriEngine.Audio.Song` Class Reference

`Song` class that can be used with `AudioManager`.

Inheritance diagram for `TriDevs.TriEngine.Audio.Song`:



Collaboration diagram for TriDevs.TriEngine.Audio.Song:



### Public Member Functions

- void [Dispose](#) ()
- void [Play](#) ()  
*Starts playback of the song.*
- void [Stop](#) ()  
*Stops playback of the song.*
- void [Pause](#) ()  
*Pauses playback of the song.*
- void [Resume](#) ()  
*Resumes playback of a paused song.*

### Properties

- string [Name](#) [get]
- string [File](#) [get]
- float [Volume](#) [get, set]
- bool [IsLooped](#) [get, set]

#### 7.55.1 Detailed Description

[Song](#) class that can be used with [AudioManager](#).

The song class will stream from a specified file, instead of loading it into memory.

## 7.55.2 Member Function Documentation

### 7.55.2.1 void TriDevs.TriEngine.Audio.Song.Dispose ( )

```
80      {
81          if (_stream == null)
82              return;
83
84          Stop();
85          _stream.Dispose();
86          _stream = null;
87      }
```

### 7.55.2.2 void TriDevs.TriEngine.Audio.Song.Pause ( )

Pauses playback of the song.

Call the [Resume](#) or [Play](#) method to resume playback.

Implements [TriDevs.TriEngine.Audio.ISong](#).

```
108      {
109          _stream.Pause();
110      }
```

### 7.55.2.3 void TriDevs.TriEngine.Audio.Song.Play ( )

Starts playback of the song.

Implements [TriDevs.TriEngine.Audio.ISong](#).

```
90      {
91          try
92          {
93              _stream.Play();
94          }
95          catch (InvalidDataException)
96          {
97              Stop();
98              Play();
99          }
100      }
```

### 7.55.2.4 void TriDevs.TriEngine.Audio.Song.Resume ( )

Resumes playback of a paused song.

Implements [TriDevs.TriEngine.Audio.ISong](#).

```
113      {
114          _stream.Resume();
115      }
```

### 7.55.2.5 void TriDevs.TriEngine.Audio.Song.Stop ( )

Stops playback of the song.

Implements [TriDevs.TriEngine.Audio.ISong](#).

```
103      {
104          _stream.Stop();
105      }
```



### 7.55.3 Property Documentation

7.55.3.1 `string TriDevs.TriEngine.Audio.Song.File` `[get]`

7.55.3.2 `bool TriDevs.TriEngine.Audio.Song.IsLooped` `[get]`, `[set]`

7.55.3.3 `string TriDevs.TriEngine.Audio.Song.Name` `[get]`

7.55.3.4 `float TriDevs.TriEngine.Audio.Song.Volume` `[get]`, `[set]`

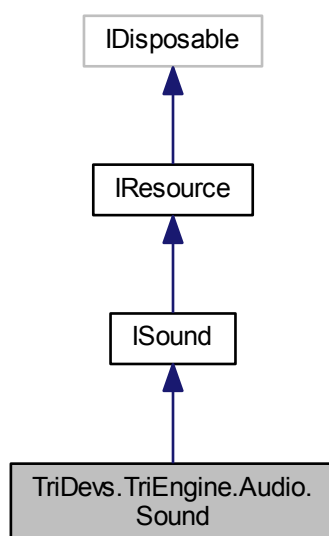
The documentation for this class was generated from the following file:

- `TriDevs.TriEngine/Audio/Song.cs`

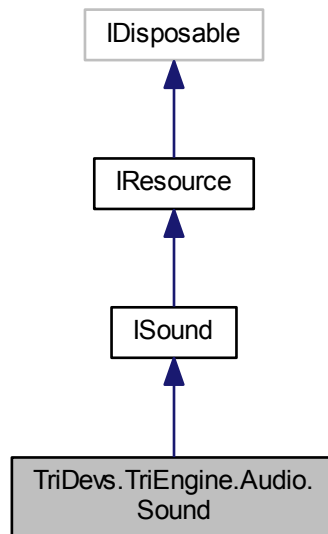
## 7.56 TriDevs.TriEngine.Audio.Sound Class Reference

[Sound](#) class that can be used with the [AudioManager](#).

Inheritance diagram for `TriDevs.TriEngine.Audio.Sound`:



Collaboration diagram for TriDevs.TriEngine.Audio.Sound:



## Public Member Functions

- void [Play](#) ()  
*Plays the sound.*
- void [Stop](#) ()  
*Stops the sound.*
- void [Dispose](#) ()

## Properties

- string [Name](#) [get]
- string [File](#) [get]

### 7.56.1 Detailed Description

[Sound](#) class that can be used with the [AudioManager](#).

### 7.56.2 Member Function Documentation

#### 7.56.2.1 void TriDevs.TriEngine.Audio.Sound.Dispose ( )

```
187     {  
188         Stop();  
189         AL.DeleteSources(_sources);  
190         AL.DeleteBuffer(_buffer);  
191     }
```

**7.56.2.2 void TriDevs.TriEngine.Audio.Sound.Play ( )**

Plays the sound.

Implements [TriDevs.TriEngine.Audio.ISound](#).

```

169         {
170             for (var i = 0; i < SourceCount; i++)
171             {
172                 if (GetSourceState(_sources[i]) != ALSourceState.Playing)
173                 {
174                     AL.SourcePlay(_sources[i]);
175                     return;
176                 }
177             }
178         }

```

**7.56.2.3 void TriDevs.TriEngine.Audio.Sound.Stop ( )**

Stops the sound.

Implements [TriDevs.TriEngine.Audio.ISound](#).

```

181         {
182             for (var i = 0; i < SourceCount; i++)
183                 AL.SourceStop(_sources[i]);
184         }

```

**7.56.3 Property Documentation**

**7.56.3.1 string TriDevs.TriEngine.Audio.Sound.File** [get]

**7.56.3.2 string TriDevs.TriEngine.Audio.Sound.Name** [get]

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

- TriDevs.TriEngine/Audio/[Sound.cs](#)

**7.57 TriDevs.TriEngine.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.*

**7.57.1 Detailed Description**

[Extensions](#) for System.String

## 7.57.2 Member Function Documentation

### 7.57.2.1 `static string TriDevs.TriEngine.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

<code>s</code>	String to modify.
<code>search</code>	String to search for.
<code>replace</code>	String to replace the match(es) with.
<code>count</code>	Number of occurrences to replace.
<code>caseInsensitive</code>	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         }
```

### 7.57.2.2 `static string TriDevs.TriEngine.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

<code>s</code>	String to modify.
<code>search</code>	String to search for.
<code>replace</code>	String to replace the match(es) with.
<code>caseInsensitive</code>	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         }
```

### 7.57.2.3 `static string TriDevs.TriEngine.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

<code>s</code>	String to modify.
<code>search</code>	String to search for.
<code>replace</code>	String to replace the match with.
<code>caseInsensitive</code>	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     }
```

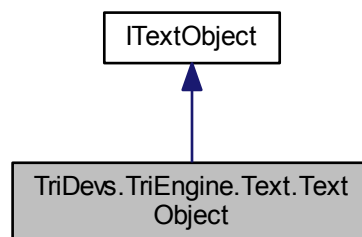
The documentation for this class was generated from the following file:

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

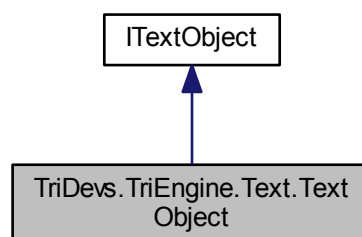
## 7.58 TriDevs.TriEngine.Text.TextObject Class Reference

Implements the [ITextObject](#) interface.

Inheritance diagram for TriDevs.TriEngine.Text.TextObject:



Collaboration diagram for TriDevs.TriEngine.Text.TextObject:



### Public Member Functions

- [TextObject](#) (string text, string fontName, Point< int > position=new Point< int >(), QFontAlignment alignment=QFontAlignment.Centre)

- **TextObject** (string text, **Font** font, Point< int > position=new Point< int >(), QFontAlignment alignment=QFontAlignment.Centre)  
*Initializes a new **TextObject** instance.*
- void **Draw** ()  
*Draws this text object to screen with default parameters.*
- void **Draw** (Point< int > position)  
*Draws this text object to a specific position on the screen specified by the supplied Point struct.*
- void **Draw** (int x, int y)  
*Draws this text object to screen at the specified X/Y position.*

## Properties

- **Font** **Font** [get, set]
- **Color** **Color** [get, set]
- string **Text** [get, set]
- Point< int > **Position** [get, set]
- **Rectangle** **Bounds** [get]
- QFontAlignment **Alignment** [get, set]

## 7.58.1 Detailed Description

Implements the **ITextObject** interface.

## 7.58.2 Constructor & Destructor Documentation

**7.58.2.1** TriDevs.TriEngine.Text.TextObject.TextObject ( string text, string fontName, Point< int > position = new Point<int>(), QFontAlignment alignment = QFontAlignment.Centre )

```

73         : this(text, Resources.ResourceManager.Get<Font>(fontName), position, alignment)
74     {
75
76     }
```

**7.58.2.2** TriDevs.TriEngine.Text.TextObject.TextObject ( string text, **Font** font, Point< int > position = new Point<int>(), QFontAlignment alignment = QFontAlignment.Centre )

Initializes a new **TextObject** instance.

### Parameters

<i>text</i>	The initial text to set for this text object.
<i>font</i>	The font to use for this text object.
<i>position</i>	The initial position of this text object.
<i>alignment</i>	The initial alignment of the text in this text object.

```

87     {
88         if (font == null)
89             throw new EngineException("Font supplied for TextObject is null!", new
ArgumentNullException("font"));
90
91         Text = text;
92         Font = font;
93         Position = position;
94         Alignment = alignment;
95     }
```

### 7.58.3 Member Function Documentation

#### 7.58.3.1 void TriDevs.TriEngine.Text.TextObject.Draw ( )

Draws this text object to screen with default parameters.

Implements [TriDevs.TriEngine.Text.ITextObject](#).

```

104         {
105             Draw(_vectorPos);
106         }
```

#### 7.58.3.2 void TriDevs.TriEngine.Text.TextObject.Draw ( Point< int > *position* )

Draws this text object to a specific position on the screen specified by the supplied Point struct.

##### Parameters

<i>position</i>	Point class with X/Y coordinates.
-----------------	-----------------------------------

Implements [TriDevs.TriEngine.Text.ITextObject](#).

```

109         {
110             Draw(position.X, position.Y);
111         }
```

#### 7.58.3.3 void TriDevs.TriEngine.Text.TextObject.Draw ( int x, int y )

Draws this text object to screen at the specified X/Y position.

##### Parameters

<i>x</i>	
<i>y</i>	

Implements [TriDevs.TriEngine.Text.ITextObject](#).

```

114         {
115             Draw(new Vector2(x, y));
116         }
```

### 7.58.4 Property Documentation

7.58.4.1 QFontAlignment TriDevs.TriEngine.Text.TextObject.Alignment [get], [set]

7.58.4.2 Rectangle TriDevs.TriEngine.Text.TextObject.Bounds [get]

7.58.4.3 Color TriDevs.TriEngine.Text.TextObject.Color [get], [set]

7.58.4.4 Font TriDevs.TriEngine.Text.TextObject.Font [get], [set]

7.58.4.5 Point<int> TriDevs.TriEngine.Text.TextObject.Position [get], [set]

7.58.4.6 string TriDevs.TriEngine.Text.TextObject.Text [get], [set]

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

- [TriDevs.TriEngine/Text/TextObject.cs](#)

## 7.59 TriDevs.TriEngine.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.*

#### 7.59.1 Detailed Description

Provides various helper functions for doing threading operations.

#### 7.59.2 Member Function Documentation

**7.59.2.1** static void [TriDevs.TriEngine.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:

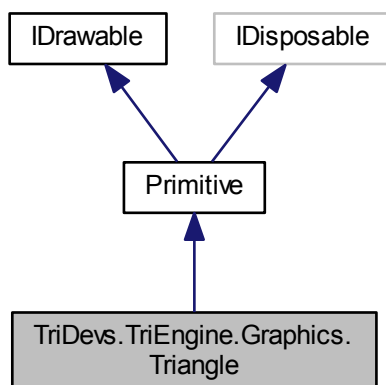
- [TriDevs.TriEngine/Helpers/Threading.cs](#)

## 7.60 TriDevs.TriEngine.Graphics.Triangle Class Reference

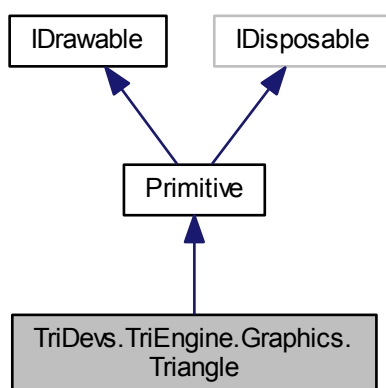
A primitive 2D [Triangle](#) shape.



Inheritance diagram for TriDevs.TriEngine.Graphics.Triangle:



Collaboration diagram for TriDevs.TriEngine.Graphics.Triangle:



## Public Member Functions

- [Triangle](#) (Point< int > top, Point< int > left, Point< int > right)  
*Creates a new 2D [Triangle](#) with the specified points.*

## Additional Inherited Members

### 7.60.1 Detailed Description

A primitive 2D [Triangle](#) shape.

## 7.60.2 Constructor & Destructor Documentation

### 7.60.2.1 TriDevs.TriEngine.Graphics.Triangle.Triangle ( Point< int > top, Point< int > left, Point< int > right )

Creates a new 2D [Triangle](#) with the specified points.

#### Parameters

<i>top</i>	Coordinate of the top edge of this triangle.
<i>left</i>	Coordinate of the lower left edge of this triangle.
<i>right</i>	Coordinate of the lower right edge of this triangle.

```
40         : base(new ushort[] {0, 1, 2},
```

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

- TriDevs.TriEngine/Graphics/[Triangle.cs](#)

## 7.61 TriDevs.TriEngine.Extensions.Vector3Extensions Class Reference

[Extensions](#) for Vector3.

### Static Public Member Functions

- static float[] [ToFloatArray](#) (this Vector3 vector)  
*Converts a Vector3 into a float array (3 floats for XYZ).*
- static float[] [ToFloatArray](#) (this Vector3[] vectors)  
*Converts an array of Vector3 into a float array (3 floats per vector).*

### 7.61.1 Detailed Description

[Extensions](#) for Vector3.

### 7.61.2 Member Function Documentation

#### 7.61.2.1 static float [] TriDevs.TriEngine.Extensions.Vector3Extensions.ToFloatArray ( this Vector3 vector ) [static]

Converts a Vector3 into a float array (3 floats for XYZ).

#### Parameters

<i>vector</i>	The vector to convert.
---------------	------------------------

#### Returns

A float array representation of the vector.

```
40         {
41             return new[] {vector.X, vector.Y, vector.Z};
42         }
```

#### 7.61.2.2 static float [] TriDevs.TriEngine.Extensions.Vector3Extensions.ToFloatArray ( this Vector3[] vectors ) [static]

Converts an array of Vector3 into a float array (3 floats per vector).

## Parameters

<i>vectors</i>	The vector array to convert.
----------------	------------------------------

## Returns

A float array representation of the vectors.

```

51     {
52         var result = new float[3 * vectors.Length];
53
54         for (var i = 0; i < vectors.Length; i++)
55         {
56             var index = i * 3;
57
58             result[index] = vectors[i].X;
59             result[index + 1] = vectors[i].Y;
60             result[index + 2] = vectors[i].Z;
61         }
62
63         return result;
64     }

```

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

- TriDevs.TriEngine/Extensions/[Vector3Extensions.cs](#)

## 7.62 TriDevs.TriEngine.Extensions.Vector4Extensions Class Reference

[Extensions](#) for Vector4.

### Static Public Member Functions

- static float[] [ToFloatArray](#) (this Vector4 vector)  
*Converts a Vector4 into a float array (4 floats for XYZW).*
- static float[] [ToFloatArray](#) (this Vector4[] vectors)  
*Converts an array of Vector4 into a float array (4 floats per vector).*

### 7.62.1 Detailed Description

[Extensions](#) for Vector4.

### 7.62.2 Member Function Documentation

**7.62.2.1** static float [] TriDevs.TriEngine.Extensions.Vector4Extensions.ToFloatArray ( this Vector4 *vector* ) [static]

Converts a Vector4 into a float array (4 floats for XYZW).

## Parameters

<i>vector</i>	The vector to convert.
---------------	------------------------

## Returns

A float array representation of the vector.

```

40     {
41         return new[] {vector.X, vector.Y, vector.Z, vector.W};
42     }

```

7.62.2.2 `static float [] TriDevs.TriEngine.Extensions.Vector4Extensions.ToFloatArray ( this Vector4[] vectors ) [static]`

Converts an array of Vector4 into a float array (4 floats per vector).

#### Parameters

<code>vectors</code>	The vector array to convert.
----------------------	------------------------------

#### Returns

A float array representation of the vectors.

```

51     {
52         var result = new float[4 * vectors.Length];
53
54         for (var i = 0; i < vectors.Length; i++)
55         {
56             var index = i * 4;
57
58             result[index] = vectors[i].X;
59             result[index + 1] = vectors[i].Y;
60             result[index + 2] = vectors[i].Z;
61             result[index + 3] = vectors[i].W;
62         }
63
64         return result;
65     }
```

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

- [TriDevs.TriEngine/Extensions/Vector4Extensions.cs](#)

## 7.63 TriDevs.TriEngine.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 = 16`  
*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.*

### 7.63.1 Detailed Description

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

### 7.63.2 Member Data Documentation

**7.63.2.1** `const int TriDevs.TriEngine.Version.Major = 0`

Major version of the project.

**7.63.2.2** `const int TriDevs.TriEngine.Version.Minor = 0`

Minor version of the project.

**7.63.2.3** `const int TriDevs.TriEngine.Version.Patch = 16`

Patch version of the project.

**7.63.2.4** `const string TriDevs.TriEngine.Version.Suffix = ""`

Optional suffix, empty if no suffix for this version.

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

**7.63.2.5** `const string TriDevs.TriEngine.Version.VersionStringFormat = "{0}.{1}.{2}"`

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

**7.63.2.6** `const string TriDevs.TriEngine.Version.VersionStringFormatWithSuffix = VersionStringFormat + "-{3}"`

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

### 7.63.3 Property Documentation

**7.63.3.1** `string TriDevs.TriEngine.Version.VersionString` `[static], [get]`

String representation of the current project version.

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

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



## Chapter 8

# File Documentation

### 8.1 README.md File Reference

### 8.2 TriDevs.TriEngine/Audio/AudioFormat.cs File Reference

#### Namespaces

- package [TriDevs.TriEngine.Audio](#)

#### Enumerations

- enum [TriDevs.TriEngine.Audio.AudioFormat](#) { [TriDevs.TriEngine.Audio.Ogg](#), [TriDevs.TriEngine.Audio.Wav](#) }  
*Different audio formats usable with audio managers.*

### 8.3 TriDevs.TriEngine/Audio/AudioManager.cs File Reference

#### Classes

- class [TriDevs.TriEngine.Audio.AudioManager](#)  
*Class to manage engine audio.*

#### Namespaces

- package [TriDevs.TriEngine.Audio](#)

### 8.4 TriDevs.TriEngine/Audio/IAudioManager.cs File Reference

#### Classes

- interface [TriDevs.TriEngine.Audio.IAudioManager](#)  
*Provides various methods to manipulate audio.*

#### Namespaces

- package [TriDevs.TriEngine.Audio](#)

## 8.5 TriDevs.TriEngine/Audio/ISong.cs File Reference

### Classes

- interface [TriDevs.TriEngine.Audio.ISong](#)  
*A song that will be streamed in the audio player.*

### Namespaces

- package [TriDevs.TriEngine.Audio](#)

## 8.6 TriDevs.TriEngine/Audio/ISound.cs File Reference

### Classes

- interface [TriDevs.TriEngine.Audio.ISound](#)  
*A sound file for use with the audio manager.*

### Namespaces

- package [TriDevs.TriEngine.Audio](#)

## 8.7 TriDevs.TriEngine/Audio/NullAudioManager.cs File Reference

### Classes

- class [TriDevs.TriEngine.Audio.NullAudioManager](#)  
*Used as a fallback [AudioManager](#) object when the service locator fails to find one.*

### Namespaces

- package [TriDevs.TriEngine.Audio](#)

## 8.8 TriDevs.TriEngine/Audio/NullSong.cs File Reference

### Classes

- class [TriDevs.TriEngine.Audio.NullSong](#)  
*Fallback song class used in [NullAudioManager](#).*

### Namespaces

- package [TriDevs.TriEngine.Audio](#)



## 8.9 TriDevs.TriEngine/Audio/NullSound.cs File Reference

### Classes

- class [TriDevs.TriEngine.Audio.NullSound](#)  
*Fallback sound class used in [NullAudioManager](#).*

### Namespaces

- package [TriDevs.TriEngine.Audio](#)

## 8.10 TriDevs.TriEngine/Audio/Song.cs File Reference

### Classes

- class [TriDevs.TriEngine.Audio.Song](#)  
*[Song](#) class that can be used with [AudioManager](#).*

### Namespaces

- package [TriDevs.TriEngine.Audio](#)

## 8.11 TriDevs.TriEngine/Audio/Sound.cs File Reference

### Classes

- class [TriDevs.TriEngine.Audio.Sound](#)  
*[Sound](#) class that can be used with the [AudioManager](#).*

### Namespaces

- package [TriDevs.TriEngine.Audio](#)

## 8.12 TriDevs.TriEngine/Color.cs File Reference

### Classes

- struct [TriDevs.TriEngine.Color](#)  
*Represents an RGBA color that can be used with [TriEngine2D](#).*

### Namespaces

- package [TriDevs.TriEngine](#)

## 8.13 TriDevs.TriEngine/EngineException.cs File Reference

### Classes

- class [TriDevs.TriEngine.EngineException](#)

*Base exception class for all engine-related exceptions. The inner exception will contain more info as to what actually happened.*

### Namespaces

- package [TriDevs.TriEngine](#)

## 8.14 TriDevs.TriEngine/Extensions/ByteExtensions.cs File Reference

### Classes

- class [TriDevs.TriEngine.Extensions.ByteExtensions](#)

*[Extensions](#) for `System.Byte`.*

### Namespaces

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

## 8.15 TriDevs.TriEngine/Extensions/DecimalExtensions.cs File Reference

### Classes

- class [TriDevs.TriEngine.Extensions.DecimalExtensions](#)

*[Extensions](#) for `System.Decimal`.*

### Namespaces

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

## 8.16 TriDevs.TriEngine/Extensions/DoubleExtensions.cs File Reference

### Classes

- class [TriDevs.TriEngine.Extensions.DoubleExtensions](#)

*[Extensions](#) for `System.Double`.*

### Namespaces

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

## 8.17 TriDevs.TriEngine/Extensions/EnumerationExtensions.cs File Reference

### Classes

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

### Namespaces

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

## 8.18 TriDevs.TriEngine/Extensions/FloatExtensions.cs File Reference

### Classes

- class [TriDevs.TriEngine.Extensions.FloatExtensions](#)  
*Extensions for System.Single.*

### Namespaces

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

## 8.19 TriDevs.TriEngine/Extensions/IntegerExtensions.cs File Reference

### Classes

- class [TriDevs.TriEngine.Extensions.IntegerExtensions](#)  
*Extensions for System.Int16, System.UInt16, System.Int32, System.UInt32, System.Int64 and System.UInt64.*

### Namespaces

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

## 8.20 TriDevs.TriEngine/Extensions/StringExtensions.cs File Reference

### Classes

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

### Namespaces

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

## 8.21 TriDevs.TriEngine/Extensions/Vector3Extensions.cs File Reference

### Classes

- class [TriDevs.TriEngine.Extensions.Vector3Extensions](#)  
*Extensions for Vector3.*

### Namespaces

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

## 8.22 TriDevs.TriEngine/Extensions/Vector4Extensions.cs File Reference

### Classes

- class [TriDevs.TriEngine.Extensions.Vector4Extensions](#)  
*Extensions for Vector4.*

### Namespaces

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

## 8.23 TriDevs.TriEngine/GameWindow2D.cs File Reference

### Classes

- class [TriDevs.TriEngine.GameWindow2D](#)  
*Game window class specialized for drawing 2D graphics.*

### Namespaces

- package [TriDevs.TriEngine](#)

## 8.24 TriDevs.TriEngine/Graphics/Primitive.cs File Reference

### Classes

- class [TriDevs.TriEngine.Graphics.Primitive](#)  
*Represents a primitive 2D shape composed of triangles.*

### Namespaces

- package [TriDevs.TriEngine.Graphics](#)

## 8.25 TriDevs.TriEngine/Graphics/Rectangle.cs File Reference

### Classes

- class [TriDevs.TriEngine.Graphics.Rectangle](#)  
*A primitive 2D [Rectangle](#) shape.*

### Namespaces

- package [TriDevs.TriEngine.Graphics](#)

## 8.26 TriDevs.TriEngine/Rectangle.cs File Reference

### Classes

- struct [TriDevs.TriEngine.Rectangle](#)  
*A rectangle representing an area in 2D space.*

### Namespaces

- package [TriDevs.TriEngine](#)

## 8.27 TriDevs.TriEngine/Graphics/Triangle.cs File Reference

### Classes

- class [TriDevs.TriEngine.Graphics.Triangle](#)  
*A primitive 2D [Triangle](#) shape.*

### Namespaces

- package [TriDevs.TriEngine.Graphics](#)

## 8.28 TriDevs.TriEngine/Helpers/Exceptions.cs File Reference

### Classes

- class [TriDevs.TriEngine.Helpers.Exceptions](#)  
*Provides helper methods for dealing with exceptions.*

### Namespaces

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

## 8.29 TriDevs.TriEngine/Helpers/IO.cs File Reference

### Classes

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

### Namespaces

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

## 8.30 TriDevs.TriEngine/Helpers/Math.cs File Reference

### Classes

- class [TriDevs.TriEngine.Helpers.Math](#)  
*Various helper methods for working with math.*

### Namespaces

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

## 8.31 TriDevs.TriEngine/Helpers/Threading.cs File Reference

### Classes

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

### Namespaces

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

## 8.32 TriDevs.TriEngine/Input/Events/Key.cs File Reference

### Classes

- class [TriDevs.TriEngine.Input.Events.KeyEventArgs](#)  
*EventArgs class used for key-related events. Contains information about the key related with the event.*
- class [TriDevs.TriEngine.Input.Events.KeyCharEventArgs](#)  
*EventArgs class used for keychar-related events. Contains information about the character related with the event.*

### Namespaces

- package [TriDevs.TriEngine.Input.Events](#)

## Functions

- delegate void [TriDevs.TriEngine.Input.Events.KeyDownEventHandler](#) (object sender, KeyEventArgs e)  
*Event handler delegate for the KeyDown event.*
- delegate void [TriDevs.TriEngine.Input.Events.KeyUpEventHandler](#) (object sender, KeyEventArgs e)  
*Event handler delegate for the KeyUp event.*
- delegate void [TriDevs.TriEngine.Input.Events.KeyPressEventHandler](#) (object sender, KeyCharEventArgs e)  
*Event handler delegate for the KeyPress event.*

## 8.33 TriDevs.TriEngine/Input/Events/Mouse.cs File Reference

### Namespaces

- package [TriDevs.TriEngine.Input.Events](#)

### Functions

- delegate void [TriDevs.TriEngine.Input.Events.MouseDownEventHandler](#) (object sender, MouseButtonEventArgs e)  
*Event handler delegate for the MouseDown event.*
- delegate void [TriDevs.TriEngine.Input.Events.MouseUpEventHandler](#) (object sender, MouseButtonEventArgs e)  
*Event handler delegate for the MouseUp event.*
- delegate void [TriDevs.TriEngine.Input.Events.MouseWheelChangedEventHandler](#) (object sender, MouseWheelEventArgs e)  
*Event handler delegate for the MouseWheelChanged event.*
- delegate void [TriDevs.TriEngine.Input.Events.MouseWheelDownEventHandler](#) (object sender, MouseWheelEventArgs e)  
*Event handler delegate for the MouseWheelDown event.*
- delegate void [TriDevs.TriEngine.Input.Events.MouseWheelUpEventHandler](#) (object sender, MouseWheelEventArgs e)  
*Event handler delegate for the MouseWheelUp event.*

## 8.34 TriDevs.TriEngine/Input/IInputManager.cs File Reference

### Classes

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

### Namespaces

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

## 8.35 TriDevs.TriEngine/Input/InputManager.cs File Reference

### Classes

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

## Namespaces

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

## 8.36 TriDevs.TriEngine/Input/NullInputManager.cs File Reference

### Classes

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

## Namespaces

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

## 8.37 TriDevs.TriEngine/Interfaces/IDrawable.cs File Reference

### Classes

- interface [TriDevs.TriEngine.Interfaces.IDrawable](#)  
*Implements a simple draw method.*

## Namespaces

- package [TriDevs.TriEngine.Interfaces](#)

## 8.38 TriDevs.TriEngine/Interfaces/IDrawableGameComponent.cs File Reference

### Classes

- interface [TriDevs.TriEngine.Interfaces.IDrawableGameComponent](#)  
*A game component that can be added to GameState objects. Drawable game components also implement a draw method to draw themselves to screen.*

## Namespaces

- package [TriDevs.TriEngine.Interfaces](#)

## 8.39 TriDevs.TriEngine/Interfaces/IGameComponent.cs File Reference

### Classes

- interface [TriDevs.TriEngine.Interfaces.IGameComponent](#)  
*A game component that can be added to IGameState objects.*

## Namespaces

- package [TriDevs.TriEngine.Interfaces](#)



## 8.40 TriDevs.TriEngine/Interfaces/IUpdatable.cs File Reference

### Classes

- interface [TriDevs.TriEngine.Interfaces.IUpdatable](#)  
*Implements a simple update method.*

### Namespaces

- package [TriDevs.TriEngine.Interfaces](#)

## 8.41 TriDevs.TriEngine/Logging/LogManager.cs File Reference

### Classes

- class [TriDevs.TriEngine.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.TriEngine.Logging](#)

## 8.42 TriDevs.TriEngine/Native/Helpers.cs File Reference

### Classes

- class [TriDevs.TriEngine.Native.Helpers](#)  
*Helper class with various methods to help native coding and debugging.*

### Namespaces

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

## 8.43 TriDevs.TriEngine/Native/WinAPI.cs File Reference

## 8.44 TriDevs.TriEngine/Point.cs File Reference

### Classes

- struct [TriDevs.TriEngine.Point< T >](#)  
*A struct representing an X/Y/Z coordinate.*

### Namespaces

- package [TriDevs.TriEngine](#)

## 8.45 TriDevs.TriEngine/Program.cs File Reference

### Classes

- class [TriDevs.TriEngine.Program](#)  
*An OpenGL program.*

### Namespaces

- package [TriDevs.TriEngine](#)

## 8.46 TriDevs.TriEngine/Properties/AssemblyInfo.cs File Reference

## 8.47 TriDevs.TriEngine/Resources/IResource.cs File Reference

### Classes

- interface [TriDevs.TriEngine.Resources.IResource](#)  
*An external resource created from a file.*

### Namespaces

- package [TriDevs.TriEngine.Resources](#)

## 8.48 TriDevs.TriEngine/Resources/ResourceException.cs File Reference

### Classes

- class [TriDevs.TriEngine.Resources.ResourceException](#)  
*Exception for resource-related issues. Thrown when there is a problem in [ResourceManager](#).*

### Namespaces

- package [TriDevs.TriEngine.Resources](#)

## 8.49 TriDevs.TriEngine/Resources/ResourceManager.cs File Reference

### Classes

- class [TriDevs.TriEngine.Resources.ResourceManager](#)  
*Static class to manage resources.*

### Namespaces

- package [TriDevs.TriEngine.Resources](#)

## 8.50 TriDevs.TriEngine/Serializing/Serializer.cs File Reference

### Classes

- class [TriDevs.TriEngine.Serializing.Serializer](#)

*Provides serialization methods.*

### Namespaces

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

## 8.51 TriDevs.TriEngine/Services.cs File Reference

### Classes

- class [TriDevs.TriEngine.Services](#)

*Provides different game-related service interfaces.*

### Namespaces

- package [TriDevs.TriEngine](#)

## 8.52 TriDevs.TriEngine/Shaders/Shader.cs File Reference

### Classes

- class [TriDevs.TriEngine.Shaders.Shader](#)

*GLSL shader object loaded and compiled from a \*.glsl shader file.*

### Namespaces

- package [TriDevs.TriEngine.Shaders](#)

## 8.53 TriDevs.TriEngine/StateManagement/GameState.cs File Reference

### Classes

- class [TriDevs.TriEngine.StateManagement.GameState](#)

*Base [GameState](#) class that all other game states derive from, defines basic [GameState](#) behaviour.*

### Namespaces

- package [TriDevs.TriEngine.StateManagement](#)

## 8.54 TriDevs.TriEngine/StateManagement/GameStateManager.cs File Reference

### Classes

- class [TriDevs.TriEngine.StateManagement.GameStateManager](#)  
*Game state manager that keeps track of the active game states and provides methods to control the states.*

### Namespaces

- package [TriDevs.TriEngine.StateManagement](#)

## 8.55 TriDevs.TriEngine/StateManagement/IGameState.cs File Reference

### Classes

- interface [TriDevs.TriEngine.StateManagement.IGameState](#)  
*A game state that can be used with the game state manager. Represent a specific state of the game, like main menu and options screen.*

### Namespaces

- package [TriDevs.TriEngine.StateManagement](#)

## 8.56 TriDevs.TriEngine/StateManagement/IGameStateManager.cs File Reference

### Classes

- interface [TriDevs.TriEngine.StateManagement.IGameStateManager](#)  
*Game state manager that keeps track of the active game states and provides methods to control the states.*

### Namespaces

- package [TriDevs.TriEngine.StateManagement](#)

## 8.57 TriDevs.TriEngine/Text/Font.cs File Reference

### Classes

- class [TriDevs.TriEngine.Text.Font](#)  
*Holds a specific font type.*

### Namespaces

- package [TriDevs.TriEngine.Text](#)

## 8.58 TriDevs.TriEngine/Text/FontConstructionConfig.cs File Reference

### Classes

- class [TriDevs.TriEngine.Text.FontConstructionConfig](#)  
*Container class for different QFont configurations for use with the [Font](#) constructor.*

### Namespaces

- package [TriDevs.TriEngine.Text](#)

## 8.59 TriDevs.TriEngine/Text/FontType.cs File Reference

### Namespaces

- package [TriDevs.TriEngine.Text](#)

### Enumerations

- enum [TriDevs.TriEngine.Text.FontType](#) { [TriDevs.TriEngine.Text.TTF](#), [TriDevs.TriEngine.Text.QFont](#), [TriDevs.TriEngine.Text.Unsupported](#) }  
*Different font types supported by TriEngine2D.*

## 8.60 TriDevs.TriEngine/Text/ITextObject.cs File Reference

### Classes

- interface [TriDevs.TriEngine.Text.ITextObject](#)  
*Implements methods to construct a text object and render it to screen.*

### Namespaces

- package [TriDevs.TriEngine.Text](#)

## 8.61 TriDevs.TriEngine/Text/TextObject.cs File Reference

### Classes

- class [TriDevs.TriEngine.Text.TextObject](#)  
*Implements the [ITextObject](#) interface.*

### Namespaces

- package [TriDevs.TriEngine.Text](#)

## 8.62 TriDevs.TriEngine/UI/Control.cs File Reference

### Classes

- class [TriDevs.TriEngine.UI.Control](#)

*Base control class that all other controls inherits from. Defines basic [UI](#) control behaviour.*

### Namespaces

- package [TriDevs.TriEngine.UI](#)

## 8.63 TriDevs.TriEngine/UI/ControlManager.cs File Reference

### Classes

- class [TriDevs.TriEngine.UI.ControlManager](#)

*[Control](#) manager to manage various [UI](#) controls for a game.*

### Namespaces

- package [TriDevs.TriEngine.UI](#)

## 8.64 TriDevs.TriEngine/UI/Events/ControlClicked.cs File Reference

### Namespaces

- package [TriDevs.TriEngine.UI.Events](#)

### Functions

- delegate void [TriDevs.TriEngine.UI.Events.ControlClickedEventHandler](#) (object sender, EventArgs e)

*Delegate handler for the control clicked event.*

## 8.65 TriDevs.TriEngine/UI/IControl.cs File Reference

### Classes

- interface [TriDevs.TriEngine.UI.IControl](#)

*A [UI](#) control that can be drawn on screen and interacted with.*

### Namespaces

- package [TriDevs.TriEngine.UI](#)

## 8.66 TriDevs.TriEngine/UI/IControlManager.cs File Reference

### Classes

- interface [TriDevs.TriEngine.UI.IControlManager](#)  
*Manages various [UI](#) controls, automatically updating and drawing them to the screen.*

### Namespaces

- package [TriDevs.TriEngine.UI](#)

## 8.67 TriDevs.TriEngine/UI/Label.cs File Reference

### Classes

- class [TriDevs.TriEngine.UI.Label](#)  
*A simple label to display text on the screen.*

### Namespaces

- package [TriDevs.TriEngine.UI](#)

## 8.68 TriDevs.TriEngine/UI/LinkLabel.cs File Reference

### Classes

- class [TriDevs.TriEngine.UI.LinkLabel](#)  
*A label that, when clicked, will open a URL.*

### Namespaces

- package [TriDevs.TriEngine.UI](#)

## 8.69 TriDevs.TriEngine/Version.cs File Reference

### Classes

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

### Namespaces

- package [TriDevs.TriEngine](#)

# Index

## A

TriDevs::TriEngine::Color, [39](#)

## ActiveState

TriDevs::TriEngine::StateManagement::Game-  
StateManager, [89](#)

TriDevs::TriEngine::StateManagement::IGame-  
StateManager, [115](#)

## Add

TriDevs::TriEngine::Resources::ResourceManager,  
[185](#)

## AddComponent

TriDevs::TriEngine::StateManagement::Game-  
State, [78](#)

TriDevs::TriEngine::StateManagement::IGame-  
State, [109](#)

## AddControl

TriDevs::TriEngine::UI::ControlManager, [58](#)

TriDevs::TriEngine::UI::IControlManager, [101](#)

## AliceBlue

TriDevs::TriEngine::Color, [39](#)

## Alignment

TriDevs::TriEngine::Text::ITextObject, [140](#)

TriDevs::TriEngine::Text::TextObject, [205](#)

TriDevs::TriEngine::UI::Label, [147](#)

## AntiqueWhite

TriDevs::TriEngine::Color, [39](#)

## Aqua

TriDevs::TriEngine::Color, [39](#)

## Aquamarine

TriDevs::TriEngine::Color, [39](#)

## Audio

TriDevs::TriEngine::Services, [193](#)

## AudioFormat

TriDevs::TriEngine::Audio, [14](#)

## AudioManager

TriDevs::TriEngine::Audio::AudioManager, [24](#)

## Azure

TriDevs::TriEngine::Color, [39](#)

## B

TriDevs::TriEngine::Color, [39](#)

## BasePath

TriDevs::TriEngine::Resources::ResourceManager,  
[189](#)

## Beige

TriDevs::TriEngine::Color, [39](#)

## Bisque

TriDevs::TriEngine::Color, [40](#)

## Black

TriDevs::TriEngine::Color, [40](#)

## BlanchedAlmond

TriDevs::TriEngine::Color, [40](#)

## Blue

TriDevs::TriEngine::Color, [40](#)

## BlueViolet

TriDevs::TriEngine::Color, [40](#)

## Bounds

TriDevs::TriEngine::Text::ITextObject, [140](#)

TriDevs::TriEngine::Text::TextObject, [205](#)

## Brown

TriDevs::TriEngine::Color, [40](#)

## BuildConfig

TriDevs::TriEngine::Text::FontConstructionConfig,  
[76](#)

## BurlyWood

TriDevs::TriEngine::Color, [40](#)

## CadetBlue

TriDevs::TriEngine::Color, [40](#)

## Chartreuse

TriDevs::TriEngine::Color, [40](#)

## Chocolate

TriDevs::TriEngine::Color, [40](#)

## Clamp

TriDevs::TriEngine::Extensions::ByteExtensions,  
[26](#)

TriDevs::TriEngine::Extensions::DecimalExtensions,  
[62](#)

TriDevs::TriEngine::Extensions::DoubleExtensions,  
[64](#)

TriDevs::TriEngine::Extensions::FloatExtensions,  
[69](#)

TriDevs::TriEngine::Extensions::IntegerExtensions,  
[130–132](#)

TriDevs::TriEngine::Helpers::Math, [153–156](#)

## ClearColor

TriDevs::TriEngine::GameWindow2D, [91](#)

## ClearOldLogs

TriDevs::TriEngine::Logging::LogManager, [151](#)

## Clicked

TriDevs::TriEngine::UI::Control, [56](#)

TriDevs::TriEngine::UI::IControl, [98](#)

## Color

TriDevs::TriEngine::Color, [34, 35](#)

TriDevs::TriEngine::Rectangle, [180](#)

TriDevs::TriEngine::Text::Font, [74](#)

TriDevs::TriEngine::Text::ITextObject, [140](#)

TriDevs::TriEngine::Text::TextObject, [205](#)

TriDevs::TriEngine::UI::Control, [55](#)

TriDevs::TriEngine::UI::IControl, [97](#)



- ColorID
  - TriDevs::TriEngine::Graphics::Primitive, [176](#)
- ColorId
  - TriDevs::TriEngine::Graphics::Primitive, [175](#)
- Colors
  - TriDevs::TriEngine::Graphics::Primitive, [175](#)
- Components
  - TriDevs::TriEngine::StateManagement::Game-  
State, [84](#)
- ControlClickedEventHandler
  - TriDevs::TriEngine::UI::Events, [21](#)
- ControlManager
  - TriDevs::TriEngine::UI::ControlManager, [58](#)
- Coral
  - TriDevs::TriEngine::Color, [40](#)
- CornflowerBlue
  - TriDevs::TriEngine::Color, [40](#)
- Cornsilk
  - TriDevs::TriEngine::Color, [41](#)
- Crimson
  - TriDevs::TriEngine::Color, [41](#)
- Cyan
  - TriDevs::TriEngine::Color, [41](#)
- DarkBlue
  - TriDevs::TriEngine::Color, [41](#)
- DarkCyan
  - TriDevs::TriEngine::Color, [41](#)
- DarkGoldenrod
  - TriDevs::TriEngine::Color, [41](#)
- DarkGray
  - TriDevs::TriEngine::Color, [41](#)
- DarkGreen
  - TriDevs::TriEngine::Color, [41](#)
- DarkKhaki
  - TriDevs::TriEngine::Color, [41](#)
- DarkMagenta
  - TriDevs::TriEngine::Color, [41](#)
- DarkOliveGreen
  - TriDevs::TriEngine::Color, [41](#)
- DarkOrange
  - TriDevs::TriEngine::Color, [41](#)
- DarkOrchid
  - TriDevs::TriEngine::Color, [42](#)
- DarkRed
  - TriDevs::TriEngine::Color, [42](#)
- DarkSalmon
  - TriDevs::TriEngine::Color, [42](#)
- DarkSeaGreen
  - TriDevs::TriEngine::Color, [42](#)
- DarkSlateBlue
  - TriDevs::TriEngine::Color, [42](#)
- DarkSlateGray
  - TriDevs::TriEngine::Color, [42](#)
- DarkTurquoise
  - TriDevs::TriEngine::Color, [42](#)
- DarkViolet
  - TriDevs::TriEngine::Color, [42](#)
- DeepPink
  - TriDevs::TriEngine::Color, [42](#)
- DeepSkyBlue
  - TriDevs::TriEngine::Color, [42](#)
- Deserialize< T >
  - TriDevs::TriEngine::Serializing::Serializer, [190](#)
- DestroyConsole
  - TriDevs::TriEngine::Logging::LogManager, [151](#)
- DimGray
  - TriDevs::TriEngine::Color, [42](#)
- Disable
  - TriDevs::TriEngine::Interfaces::IGameComponent, [106](#)
  - TriDevs::TriEngine::UI::Control, [53](#)
  - TriDevs::TriEngine::UI::ControlManager, [59](#)
  - TriDevs::TriEngine::UI::IControl, [96](#)
- Dispose
  - TriDevs::TriEngine::Audio::AudioManager, [24](#)
  - TriDevs::TriEngine::Audio::NullAudioManager, [158](#)
  - TriDevs::TriEngine::Audio::NullSong, [168](#)
  - TriDevs::TriEngine::Audio::NullSound, [171](#)
  - TriDevs::TriEngine::Audio::Song, [198](#)
  - TriDevs::TriEngine::Audio::Sound, [200](#)
  - TriDevs::TriEngine::Graphics::Primitive, [175](#)
  - TriDevs::TriEngine::Program, [177](#)
  - TriDevs::TriEngine::Shaders::Shader, [195](#)
  - TriDevs::TriEngine::Text::Font, [73](#)
- DodgerBlue
  - TriDevs::TriEngine::Color, [42](#)
- Draw
  - TriDevs::TriEngine::Graphics::Primitive, [175](#)
  - TriDevs::TriEngine::Interfaces::IDrawable, [103](#)
  - TriDevs::TriEngine::StateManagement::Game-  
State, [79](#)
  - TriDevs::TriEngine::StateManagement::Game-  
StateManager, [86](#)
  - TriDevs::TriEngine::Text::ITextObject, [139](#)
  - TriDevs::TriEngine::Text::TextObject, [205](#)
  - TriDevs::TriEngine::UI::Control, [53](#), [54](#)
  - TriDevs::TriEngine::UI::ControlManager, [59](#)
  - TriDevs::TriEngine::UI::Label, [146](#)
- Enable
  - TriDevs::TriEngine::Interfaces::IGameComponent, [106](#)
  - TriDevs::TriEngine::UI::Control, [54](#)
  - TriDevs::TriEngine::UI::ControlManager, [59](#)
  - TriDevs::TriEngine::UI::IControl, [96](#)
- Enabled
  - TriDevs::TriEngine::Interfaces::IGameComponent, [107](#)
  - TriDevs::TriEngine::UI::Control, [55](#)
  - TriDevs::TriEngine::UI::ControlManager, [62](#)
  - TriDevs::TriEngine::UI::IControl, [97](#)
- Equals
  - TriDevs::TriEngine::Rectangle, [180](#)
- File
  - TriDevs::TriEngine::Audio::ISong, [136](#)
  - TriDevs::TriEngine::Audio::ISound, [138](#)

- TriDevs::TriEngine::Audio::NullSong, 169
- TriDevs::TriEngine::Audio::NullSound, 171
- TriDevs::TriEngine::Audio::Song, 199
- TriDevs::TriEngine::Audio::Sound, 201
- TriDevs::TriEngine::Shaders::Shader, 195
- TriDevs::TriEngine::Text::Font, 74
- Firebrick
  - TriDevs::TriEngine::Color, 43
- FloralWhite
  - TriDevs::TriEngine::Color, 43
- Font
  - TriDevs::TriEngine::Text::Font, 72
  - TriDevs::TriEngine::Text::ITextObject, 140
  - TriDevs::TriEngine::Text::TextObject, 205
- FontConstructionConfig
  - TriDevs::TriEngine::Text::FontConstructionConfig, 75
- FontPath
  - TriDevs::TriEngine::Resources::ResourceManager, 189
- FontType
  - TriDevs::TriEngine::Text, 20
- ForestGreen
  - TriDevs::TriEngine::Color, 43
- FromAbgr
  - TriDevs::TriEngine::Color, 36
- FromBgra
  - TriDevs::TriEngine::Color, 36
- FromRgba
  - TriDevs::TriEngine::Color, 37
- Fuchsia
  - TriDevs::TriEngine::Color, 43
- G
  - TriDevs::TriEngine::Color, 43
- Gainsboro
  - TriDevs::TriEngine::Color, 43
- GameState
  - TriDevs::TriEngine::StateManagement::GameState, 78
- GameStateManager
  - TriDevs::TriEngine::StateManagement::GameStateManager, 86
- GameWindow2D
  - TriDevs::TriEngine::GameWindow2D, 90
- Get
  - TriDevs::TriEngine::Resources::ResourceManager, 185
- Get< T >
  - TriDevs::TriEngine::Resources::ResourceManager, 185
- GetAbsolutePath
  - TriDevs::TriEngine::Helpers::IO, 133
- GetAll< T >
  - TriDevs::TriEngine::Resources::ResourceManager, 186
- GetAllComponents
  - TriDevs::TriEngine::StateManagement::GameState, 79, 80
- TriDevs::TriEngine::StateManagement::IGameState, 109
- GetComponent
  - TriDevs::TriEngine::StateManagement::GameState, 80, 81
  - TriDevs::TriEngine::StateManagement::IGameState, 110
- GetDefaultName
  - TriDevs::TriEngine::Shaders::Shader, 195
  - TriDevs::TriEngine::Text::Font, 74
- GetErrorMessage
  - TriDevs::TriEngine::Native::Helpers, 92
- GetLastError
  - TriDevs::TriEngine::Native::Helpers, 92
- GetLastErrorInfo
  - TriDevs::TriEngine::Native::Helpers, 92
- GetLastErrorMessage
  - TriDevs::TriEngine::Native::Helpers, 92
- GetLogger
  - TriDevs::TriEngine::Logging::LogManager, 151
- GhostWhite
  - TriDevs::TriEngine::Color, 43
- Gold
  - TriDevs::TriEngine::Color, 43
- Goldenrod
  - TriDevs::TriEngine::Color, 43
- Gray
  - TriDevs::TriEngine::Color, 43
- Green
  - TriDevs::TriEngine::Color, 43
- GreenYellow
  - TriDevs::TriEngine::Color, 43
- Has
  - TriDevs::TriEngine::Resources::ResourceManager, 186
- Has< T >
  - TriDevs::TriEngine::Extensions::EnumerationExtensions, 66
  - TriDevs::TriEngine::Resources::ResourceManager, 187
- HasComponent
  - TriDevs::TriEngine::StateManagement::GameState, 81, 82
  - TriDevs::TriEngine::StateManagement::IGameState, 110, 111
- HasControl
  - TriDevs::TriEngine::UI::ControlManager, 59, 60
  - TriDevs::TriEngine::UI::IControlManager, 101
- Height
  - TriDevs::TriEngine::Rectangle, 180
- Hide
  - TriDevs::TriEngine::UI::Control, 54
  - TriDevs::TriEngine::UI::IControl, 97
- HoneyDew
  - TriDevs::TriEngine::Color, 44
- HotPink
  - TriDevs::TriEngine::Color, 44

- ID
  - TriDevs::TriEngine::Program, [178](#)
  - TriDevs::TriEngine::Shaders::Shader, [195](#)
- Ids
  - TriDevs::TriEngine::Graphics::Primitive, [175](#)
- Include< T >
  - TriDevs::TriEngine::Extensions::Enumeration-Extensions, [66](#)
- IndianRed
  - TriDevs::TriEngine::Color, [44](#)
- Indices
  - TriDevs::TriEngine::Graphics::Primitive, [175](#)
- IndicesID
  - TriDevs::TriEngine::Graphics::Primitive, [176](#)
- Indigo
  - TriDevs::TriEngine::Color, [44](#)
- Initialize
  - TriDevs::TriEngine::GameWindow2D, [90](#)
- Input
  - TriDevs::TriEngine::Services, [193](#)
- InputManager
  - TriDevs::TriEngine::Input::InputManager, [124](#)
- Intersects
  - TriDevs::TriEngine::Rectangle, [180](#)
- IsKeyDown
  - TriDevs::TriEngine::Input::IInputManager, [117](#)
  - TriDevs::TriEngine::Input::InputManager, [124](#)
  - TriDevs::TriEngine::Input::NullInputManager, [161](#)
- IsKeyUp
  - TriDevs::TriEngine::Input::IInputManager, [118](#)
  - TriDevs::TriEngine::Input::InputManager, [125](#)
  - TriDevs::TriEngine::Input::NullInputManager, [161](#)
- IsLooped
  - TriDevs::TriEngine::Audio::ISong, [136](#)
  - TriDevs::TriEngine::Audio::NullSong, [169](#)
  - TriDevs::TriEngine::Audio::Song, [199](#)
- IsMouseDown
  - TriDevs::TriEngine::Input::IInputManager, [118](#)
  - TriDevs::TriEngine::Input::InputManager, [125](#)
  - TriDevs::TriEngine::Input::NullInputManager, [162](#)
- IsMouseUp
  - TriDevs::TriEngine::Input::IInputManager, [118](#)
  - TriDevs::TriEngine::Input::InputManager, [125](#)
  - TriDevs::TriEngine::Input::NullInputManager, [162](#)
- IsWheelChanged
  - TriDevs::TriEngine::Input::IInputManager, [118](#)
  - TriDevs::TriEngine::Input::InputManager, [126](#)
  - TriDevs::TriEngine::Input::NullInputManager, [162](#)
- IsWheelDown
  - TriDevs::TriEngine::Input::IInputManager, [118](#)
  - TriDevs::TriEngine::Input::InputManager, [126](#)
  - TriDevs::TriEngine::Input::NullInputManager, [162](#)
- IsWheelUp
  - TriDevs::TriEngine::Input::IInputManager, [119](#)
  - TriDevs::TriEngine::Input::InputManager, [126](#)
  - TriDevs::TriEngine::Input::NullInputManager, [163](#)
- Ivory
  - TriDevs::TriEngine::Color, [44](#)
- Key
  - TriDevs::TriEngine::Input::Events::KeyEventArgs, [143](#)
- KeyChar
  - TriDevs::TriEngine::Input::Events::KeyCharEventArgs, [142](#)
- KeyDown
  - TriDevs::TriEngine::Input::IInputManager, [121](#)
  - TriDevs::TriEngine::Input::InputManager, [128](#)
  - TriDevs::TriEngine::Input::NullInputManager, [165](#)
- KeyDownEventHandler
  - TriDevs::TriEngine::Input::Events, [16](#)
- KeyPress
  - TriDevs::TriEngine::Input::IInputManager, [121](#)
  - TriDevs::TriEngine::Input::InputManager, [128](#)
  - TriDevs::TriEngine::Input::NullInputManager, [165](#)
- KeyPressEventHandler
  - TriDevs::TriEngine::Input::Events, [17](#)
- KeyPressed
  - TriDevs::TriEngine::Input::IInputManager, [119](#)
  - TriDevs::TriEngine::Input::InputManager, [126](#)
  - TriDevs::TriEngine::Input::NullInputManager, [163](#)
- KeyReleased
  - TriDevs::TriEngine::Input::IInputManager, [119](#)
  - TriDevs::TriEngine::Input::InputManager, [127](#)
  - TriDevs::TriEngine::Input::NullInputManager, [163](#)
- KeyUp
  - TriDevs::TriEngine::Input::IInputManager, [121](#)
  - TriDevs::TriEngine::Input::InputManager, [129](#)
  - TriDevs::TriEngine::Input::NullInputManager, [165](#)
- KeyUpEventHandler
  - TriDevs::TriEngine::Input::Events, [17](#)
- Khaki
  - TriDevs::TriEngine::Color, [44](#)
- Lavender
  - TriDevs::TriEngine::Color, [44](#)
- LavenderBlush
  - TriDevs::TriEngine::Color, [44](#)
- LawnGreen
  - TriDevs::TriEngine::Color, [44](#)
- LemonChiffon
  - TriDevs::TriEngine::Color, [44](#)
- LightBlue
  - TriDevs::TriEngine::Color, [44](#)
- LightCoral
  - TriDevs::TriEngine::Color, [44](#)
- LightCyan
  - TriDevs::TriEngine::Color, [45](#)
- LightGoldenrodYellow
  - TriDevs::TriEngine::Color, [45](#)
- LightGray
  - TriDevs::TriEngine::Color, [45](#)
- LightGreen
  - TriDevs::TriEngine::Color, [45](#)
- LightPink
  - TriDevs::TriEngine::Color, [45](#)
- LightSalmon
  - TriDevs::TriEngine::Color, [45](#)

- LightSeaGreen
  - TriDevs::TriEngine::Color, [45](#)
- LightSkyBlue
  - TriDevs::TriEngine::Color, [45](#)
- LightSlateGray
  - TriDevs::TriEngine::Color, [45](#)
- LightSteelBlue
  - TriDevs::TriEngine::Color, [45](#)
- LightYellow
  - TriDevs::TriEngine::Color, [45](#)
- Lime
  - TriDevs::TriEngine::Color, [45](#)
- LimeGreen
  - TriDevs::TriEngine::Color, [46](#)
- Linen
  - TriDevs::TriEngine::Color, [46](#)
- Load
  - TriDevs::TriEngine::StateManagement::Game-  
State, [82](#)
  - TriDevs::TriEngine::StateManagement::IGame-  
State, [111](#)
- LoadConfig
  - TriDevs::TriEngine::Logging::LogManager, [152](#)
  - TriDevs::TriEngine::Text::FontConstructionConfig,  
[76](#)
- LoadFont
  - TriDevs::TriEngine::Resources::ResourceManager,  
[187](#)
- LoadShader
  - TriDevs::TriEngine::Resources::ResourceManager,  
[188](#)
- LoadSong
  - TriDevs::TriEngine::Resources::ResourceManager,  
[188](#)
- LoadSound
  - TriDevs::TriEngine::Resources::ResourceManager,  
[189](#)
- LogException
  - TriDevs::TriEngine::Helpers::Exceptions, [68](#)
- Magenta
  - TriDevs::TriEngine::Color, [46](#)
- Major
  - TriDevs::TriEngine::Version, [211](#)
- Maroon
  - TriDevs::TriEngine::Color, [46](#)
- MediumAquamarine
  - TriDevs::TriEngine::Color, [46](#)
- MediumBlue
  - TriDevs::TriEngine::Color, [46](#)
- MediumOrchid
  - TriDevs::TriEngine::Color, [46](#)
- MediumPurple
  - TriDevs::TriEngine::Color, [46](#)
- MediumSeaGreen
  - TriDevs::TriEngine::Color, [46](#)
- MediumSlateBlue
  - TriDevs::TriEngine::Color, [46](#)
- MediumSpringGreen
  - TriDevs::TriEngine::Color, [46](#)
- MediumTurquoise
  - TriDevs::TriEngine::Color, [46](#)
- MediumVioletRed
  - TriDevs::TriEngine::Color, [47](#)
- MidnightBlue
  - TriDevs::TriEngine::Color, [47](#)
- Minor
  - TriDevs::TriEngine::Version, [211](#)
- MintCream
  - TriDevs::TriEngine::Color, [47](#)
- Missing < T >
  - TriDevs::TriEngine::Extensions::Enumeration-  
Extensions, [67](#)
- MistyRose
  - TriDevs::TriEngine::Color, [47](#)
- Moccasin
  - TriDevs::TriEngine::Color, [47](#)
- MouseDown
  - TriDevs::TriEngine::Input::InputManager, [121](#)
  - TriDevs::TriEngine::Input::InputManager, [129](#)
  - TriDevs::TriEngine::Input::NullInputManager, [165](#)
- MouseDownEventHandler
  - TriDevs::TriEngine::Input::Events, [17](#)
- MousePosition
  - TriDevs::TriEngine::Input::InputManager, [120](#)
  - TriDevs::TriEngine::Input::InputManager, [128](#)
  - TriDevs::TriEngine::Input::NullInputManager, [165](#)
- MousePressed
  - TriDevs::TriEngine::Input::InputManager, [119](#)
  - TriDevs::TriEngine::Input::InputManager, [127](#)
  - TriDevs::TriEngine::Input::NullInputManager, [164](#)
- MouseReleased
  - TriDevs::TriEngine::Input::InputManager, [120](#)
  - TriDevs::TriEngine::Input::InputManager, [127](#)
  - TriDevs::TriEngine::Input::NullInputManager, [164](#)
- MouseUp
  - TriDevs::TriEngine::Input::InputManager, [121](#)
  - TriDevs::TriEngine::Input::InputManager, [129](#)
  - TriDevs::TriEngine::Input::NullInputManager, [165](#)
- MouseUpEventHandler
  - TriDevs::TriEngine::Input::Events, [17](#)
- MouseWheelChangedEventHandler
  - TriDevs::TriEngine::Input::Events, [17](#)
- MouseWheelDownEventHandler
  - TriDevs::TriEngine::Input::Events, [17](#)
- MouseWheelUpEventHandler
  - TriDevs::TriEngine::Input::Events, [18](#)
- MouseWheelValue
  - TriDevs::TriEngine::Input::InputManager, [120](#)
  - TriDevs::TriEngine::Input::InputManager, [128](#)
  - TriDevs::TriEngine::Input::NullInputManager, [165](#)
- MouseX
  - TriDevs::TriEngine::Input::InputManager, [120](#)
  - TriDevs::TriEngine::Input::InputManager, [128](#)
  - TriDevs::TriEngine::Input::NullInputManager, [165](#)
- MouseY
  - TriDevs::TriEngine::Input::InputManager, [120](#)

- TriDevs::TriEngine::Input::InputManager, [128](#)
- TriDevs::TriEngine::Input::NullInputManager, [165](#)
- Name
  - TriDevs::TriEngine::Audio::NullSong, [169](#)
  - TriDevs::TriEngine::Audio::NullSound, [171](#)
  - TriDevs::TriEngine::Audio::Song, [199](#)
  - TriDevs::TriEngine::Audio::Sound, [201](#)
  - TriDevs::TriEngine::Resources::IResource, [134](#)
  - TriDevs::TriEngine::Shaders::Shader, [195](#)
  - TriDevs::TriEngine::Text::Font, [74](#)
- NavajoWhite
  - TriDevs::TriEngine::Color, [47](#)
- Navy
  - TriDevs::TriEngine::Color, [47](#)
- Ogg
  - TriDevs::TriEngine::Audio, [14](#)
- OldLace
  - TriDevs::TriEngine::Color, [47](#)
- Olive
  - TriDevs::TriEngine::Color, [47](#)
- OliveDrab
  - TriDevs::TriEngine::Color, [47](#)
- OnClicked
  - TriDevs::TriEngine::UI::Control, [54](#)
  - TriDevs::TriEngine::UI::LinkLabel, [149](#)
- OnDraw
  - TriDevs::TriEngine::GameWindow2D, [90](#)
- OnLoad
  - TriDevs::TriEngine::GameWindow2D, [90](#)
- OnRenderFrame
  - TriDevs::TriEngine::GameWindow2D, [90](#)
- OnResize
  - TriDevs::TriEngine::GameWindow2D, [91](#)
- Orange
  - TriDevs::TriEngine::Color, [47](#)
- OrangeRed
  - TriDevs::TriEngine::Color, [47](#)
- Orchid
  - TriDevs::TriEngine::Color, [48](#)
- PaleGoldenrod
  - TriDevs::TriEngine::Color, [48](#)
- PaleGreen
  - TriDevs::TriEngine::Color, [48](#)
- PaleTurquoise
  - TriDevs::TriEngine::Color, [48](#)
- PaleVioletRed
  - TriDevs::TriEngine::Color, [48](#)
- PapayaWhip
  - TriDevs::TriEngine::Color, [48](#)
- Patch
  - TriDevs::TriEngine::Version, [211](#)
- Pause
  - TriDevs::TriEngine::Audio::ISong, [135](#)
  - TriDevs::TriEngine::Audio::NullSong, [168](#)
  - TriDevs::TriEngine::Audio::Song, [198](#)
- TriDevs::TriEngine::StateManagement::Game-  
State, [82](#)
- TriDevs::TriEngine::StateManagement::IGame-  
State, [111](#)
- Paused
  - TriDevs::TriEngine::StateManagement::Game-  
State, [84](#)
  - TriDevs::TriEngine::StateManagement::IGame-  
State, [113](#)
- PeachPuff
  - TriDevs::TriEngine::Color, [48](#)
- Peek
  - TriDevs::TriEngine::StateManagement::Game-  
StateManager, [87](#)
  - TriDevs::TriEngine::StateManagement::IGame-  
StateManager, [114](#)
- Peru
  - TriDevs::TriEngine::Color, [48](#)
- Pink
  - TriDevs::TriEngine::Color, [48](#)
- Play
  - TriDevs::TriEngine::Audio::ISong, [136](#)
  - TriDevs::TriEngine::Audio::ISound, [138](#)
  - TriDevs::TriEngine::Audio::NullSong, [168](#)
  - TriDevs::TriEngine::Audio::NullSound, [171](#)
  - TriDevs::TriEngine::Audio::Song, [198](#)
  - TriDevs::TriEngine::Audio::Sound, [200](#)
- Plum
  - TriDevs::TriEngine::Color, [48](#)
- Point
  - TriDevs::TriEngine::Point< T >, [172](#)
- Pop
  - TriDevs::TriEngine::StateManagement::Game-  
StateManager, [87](#)
  - TriDevs::TriEngine::StateManagement::IGame-  
StateManager, [114](#)
- Position
  - TriDevs::TriEngine::Text::ITextObject, [140](#)
  - TriDevs::TriEngine::Text::TextObject, [205](#)
  - TriDevs::TriEngine::UI::Control, [55](#)
  - TriDevs::TriEngine::UI::IControl, [98](#)
  - TriDevs::TriEngine::UI::Label, [147](#)
- PowderBlue
  - TriDevs::TriEngine::Color, [48](#)
- Primitive
  - TriDevs::TriEngine::Graphics::Primitive, [174](#)
- Program
  - TriDevs::TriEngine::Program, [177](#)
- Provide
  - TriDevs::TriEngine::Services, [192](#)
- Purple
  - TriDevs::TriEngine::Color, [48](#)
- Push
  - TriDevs::TriEngine::StateManagement::Game-  
StateManager, [87](#)
  - TriDevs::TriEngine::StateManagement::IGame-  
StateManager, [115](#)
- QFont

- TriDevs::TriEngine::Text, [20](#)
- QFont
  - TriDevs::TriEngine::Text::Font, [74](#)
- R
  - TriDevs::TriEngine::Color, [49](#)
- README.md, [213](#)
- Rectangle
  - TriDevs::TriEngine::Graphics::Rectangle, [182](#)
  - TriDevs::TriEngine::Rectangle, [179](#)
  - TriDevs::TriEngine::UI::Control, [56](#)
  - TriDevs::TriEngine::UI::IControl, [98](#)
- Red
  - TriDevs::TriEngine::Color, [49](#)
- Remove< T >
  - TriDevs::TriEngine::Extensions::Enumeration-Extensions, [67](#)
- RemoveAllComponents
  - TriDevs::TriEngine::StateManagement::Game-State, [82](#), [83](#)
  - TriDevs::TriEngine::StateManagement::IGame-State, [112](#)
- RemoveAllControls
  - TriDevs::TriEngine::UI::ControlManager, [60](#), [61](#)
  - TriDevs::TriEngine::UI::IControlManager, [102](#)
- RemoveComponent
  - TriDevs::TriEngine::StateManagement::Game-State, [83](#)
  - TriDevs::TriEngine::StateManagement::IGame-State, [112](#)
- RemoveControl
  - TriDevs::TriEngine::UI::ControlManager, [61](#)
  - TriDevs::TriEngine::UI::IControlManager, [102](#)
- Replace
  - TriDevs::TriEngine::Extensions::StringExtensions, [202](#)
- ReplaceFirst
  - TriDevs::TriEngine::Extensions::StringExtensions, [202](#)
- Resume
  - TriDevs::TriEngine::Audio::ISong, [136](#)
  - TriDevs::TriEngine::Audio::NullSong, [168](#)
  - TriDevs::TriEngine::Audio::Song, [198](#)
- RosyBrown
  - TriDevs::TriEngine::Color, [49](#)
- RoyalBlue
  - TriDevs::TriEngine::Color, [49](#)
- SaddleBrown
  - TriDevs::TriEngine::Color, [49](#)
- Salmon
  - TriDevs::TriEngine::Color, [49](#)
- SandyBrown
  - TriDevs::TriEngine::Color, [49](#)
- SeaGreen
  - TriDevs::TriEngine::Color, [49](#)
- SeaShell
  - TriDevs::TriEngine::Color, [49](#)
- Serialize< T >
  - TriDevs::TriEngine::Serializing::Serializer, [191](#)
- SetCurrentThreadName
  - TriDevs::TriEngine::Helpers::Threading, [206](#)
- SetFont
  - TriDevs::TriEngine::UI::Label, [146](#)
- SetupConsole
  - TriDevs::TriEngine::Logging::LogManager, [152](#)
- Shader
  - TriDevs::TriEngine::Shaders::Shader, [195](#)
- ShaderPath
  - TriDevs::TriEngine::Resources::ResourceManager, [189](#)
- Show
  - TriDevs::TriEngine::UI::Control, [55](#)
  - TriDevs::TriEngine::UI::IControl, [97](#)
- Sienna
  - TriDevs::TriEngine::Color, [49](#)
- Silver
  - TriDevs::TriEngine::Color, [49](#)
- Size
  - TriDevs::TriEngine::Text::Font, [74](#)
  - TriDevs::TriEngine::UI::Control, [56](#)
  - TriDevs::TriEngine::UI::IControl, [98](#)
- SkyBlue
  - TriDevs::TriEngine::Color, [49](#)
- SlateBlue
  - TriDevs::TriEngine::Color, [50](#)
- SlateGray
  - TriDevs::TriEngine::Color, [50](#)
- Snow
  - TriDevs::TriEngine::Color, [50](#)
- SongPath
  - TriDevs::TriEngine::Resources::ResourceManager, [189](#)
- SoundPath
  - TriDevs::TriEngine::Resources::ResourceManager, [189](#)
- SpringGreen
  - TriDevs::TriEngine::Color, [50](#)
- StateCount
  - TriDevs::TriEngine::StateManagement::Game-StateManager, [89](#)
  - TriDevs::TriEngine::StateManagement::IGame-StateManager, [115](#)
- SteelBlue
  - TriDevs::TriEngine::Color, [50](#)
- Stop
  - TriDevs::TriEngine::Audio::ISong, [136](#)
  - TriDevs::TriEngine::Audio::ISound, [138](#)
  - TriDevs::TriEngine::Audio::NullSong, [169](#)
  - TriDevs::TriEngine::Audio::NullSound, [171](#)
  - TriDevs::TriEngine::Audio::Song, [198](#)
  - TriDevs::TriEngine::Audio::Sound, [201](#)
- StopAll
  - TriDevs::TriEngine::Audio::AudioManager, [25](#)
  - TriDevs::TriEngine::Audio::IAudioManager, [94](#)
  - TriDevs::TriEngine::Audio::NullAudioManager, [158](#)
- StopAllSongs



- TriDevs::TriEngine::Audio::AudioManager, [25](#)
- TriDevs::TriEngine::Audio::IAudioManager, [94](#)
- TriDevs::TriEngine::Audio::NullAudioManager, [158](#)
- StopAllSounds
  - TriDevs::TriEngine::Audio::AudioManager, [25](#)
  - TriDevs::TriEngine::Audio::IAudioManager, [94](#)
  - TriDevs::TriEngine::Audio::NullAudioManager, [159](#)
- Suffix
  - TriDevs::TriEngine::Version, [211](#)
- Switch
  - TriDevs::TriEngine::StateManagement::Game-  
StateManager, [88](#)
  - TriDevs::TriEngine::StateManagement::IGame-  
StateManager, [115](#)
- TTF
  - TriDevs::TriEngine::Text, [20](#)
- Tan
  - TriDevs::TriEngine::Color, [50](#)
- Teal
  - TriDevs::TriEngine::Color, [50](#)
- Text
  - TriDevs::TriEngine::Text::ITextObject, [140](#)
  - TriDevs::TriEngine::Text::TextObject, [205](#)
  - TriDevs::TriEngine::UI::Control, [56](#)
  - TriDevs::TriEngine::UI::IControl, [98](#)
  - TriDevs::TriEngine::UI::Label, [147](#)
- TextObject
  - TriDevs::TriEngine::Text::TextObject, [204](#)
- Thistle
  - TriDevs::TriEngine::Color, [50](#)
- ToAbgr
  - TriDevs::TriEngine::Color, [37](#)
- ToArgb
  - TriDevs::TriEngine::Color, [37](#)
- ToBgra
  - TriDevs::TriEngine::Color, [37](#)
- ToColor4
  - TriDevs::TriEngine::Color, [38](#)
- ToFloatArray
  - TriDevs::TriEngine::Extensions::Vector3Extensions, [208](#)
  - TriDevs::TriEngine::Extensions::Vector4Extensions, [209](#)
- ToRgba
  - TriDevs::TriEngine::Color, [38](#)
- ToString
  - TriDevs::TriEngine::Text::Font, [74](#)
- ToVector3
  - TriDevs::TriEngine::Color, [38](#)
- ToVector4
  - TriDevs::TriEngine::Color, [39](#)
- Tomato
  - TriDevs::TriEngine::Color, [50](#)
- TransparentBlack
  - TriDevs::TriEngine::Color, [50](#)
- TransparentWhite
  - TriDevs::TriEngine::Color, [50](#)
- TriDevs::TriEngine::Audio
  - Ogg, [14](#)
  - Wav, [14](#)
- TriDevs::TriEngine::Text
  - QFont, [20](#)
  - TTF, [20](#)
  - Unsupported, [20](#)
- TriDevs, [13](#)
- TriDevs.TriEngine, [13](#)
- TriDevs.TriEngine.Audio, [14](#)
- TriDevs.TriEngine.Audio.AudioManager, [23](#)
- TriDevs.TriEngine.Audio.IAudioManager, [93](#)
- TriDevs.TriEngine.Audio.ISong, [134](#)
- TriDevs.TriEngine.Audio.ISound, [136](#)
- TriDevs.TriEngine.Audio.NullAudioManager, [157](#)
- TriDevs.TriEngine.Audio.NullSong, [166](#)
- TriDevs.TriEngine.Audio.NullSound, [169](#)
- TriDevs.TriEngine.Audio.Song, [196](#)
- TriDevs.TriEngine.Audio.Sound, [199](#)
- TriDevs.TriEngine.Color, [27](#)
- TriDevs.TriEngine.EngineException, [64](#)
- TriDevs.TriEngine.Extensions, [15](#)
- TriDevs.TriEngine.Extensions.ByteExtensions, [25](#)
- TriDevs.TriEngine.Extensions.DecimalExtensions, [62](#)
- TriDevs.TriEngine.Extensions.DoubleExtensions, [63](#)
- TriDevs.TriEngine.Extensions.EnumerationExtensions, [65](#)
- TriDevs.TriEngine.Extensions.FloatExtensions, [69](#)
- TriDevs.TriEngine.Extensions.IntegerExtensions, [129](#)
- TriDevs.TriEngine.Extensions.StringExtensions, [201](#)
- TriDevs.TriEngine.Extensions.Vector3Extensions, [208](#)
- TriDevs.TriEngine.Extensions.Vector4Extensions, [209](#)
- TriDevs.TriEngine.GameWindow2D, [89](#)
- TriDevs.TriEngine.Graphics, [15](#)
- TriDevs.TriEngine.Graphics.Primitive, [173](#)
- TriDevs.TriEngine.Graphics.Rectangle, [181](#)
- TriDevs.TriEngine.Graphics.Triangle, [206](#)
- TriDevs.TriEngine.Helpers, [15](#)
- TriDevs.TriEngine.Helpers.Exceptions, [68](#)
- TriDevs.TriEngine.Helpers.IO, [132](#)
- TriDevs.TriEngine.Helpers.Math, [153](#)
- TriDevs.TriEngine.Helpers.Threading, [206](#)
- TriDevs.TriEngine.Input, [16](#)
- TriDevs.TriEngine.Input.Events, [16](#)
- TriDevs.TriEngine.Input.Events.KeyCharEventArgs, [141](#)
- TriDevs.TriEngine.Input.Events.KeyEventArgs, [142](#)
- TriDevs.TriEngine.Input.IInputManager, [115](#)
- TriDevs.TriEngine.Input.InputManager, [122](#)
- TriDevs.TriEngine.Input.NullInputManager, [159](#)
- TriDevs.TriEngine.Interfaces, [18](#)
- TriDevs.TriEngine.Interfaces.IDrawable, [102](#)
- TriDevs.TriEngine.Interfaces.IDrawableGameComponent, [103](#)
- TriDevs.TriEngine.Interfaces.IGameComponent, [105](#)
- TriDevs.TriEngine.Interfaces.IUpdatable, [140](#)
- TriDevs.TriEngine.Logging, [18](#)
- TriDevs.TriEngine.Logging.LogManager, [150](#)
- TriDevs.TriEngine.Native, [18](#)
- TriDevs.TriEngine.Native.Helpers, [91](#)

- TriDevs.TriEngine.Point< T >, 172
- TriDevs.TriEngine.Program, 176
- TriDevs.TriEngine.Rectangle, 178
- TriDevs.TriEngine.Resources, 19
- TriDevs.TriEngine.Resources.IResource, 133
- TriDevs.TriEngine.Resources.ResourceException, 183
- TriDevs.TriEngine.Resources.ResourceManager, 184
- TriDevs.TriEngine.Serializing, 19
- TriDevs.TriEngine.Serializing.Serializer, 190
- TriDevs.TriEngine.Services, 192
- TriDevs.TriEngine.Shaders, 19
- TriDevs.TriEngine.Shaders.Shader, 193
- TriDevs.TriEngine.StateManagement, 19
- TriDevs.TriEngine.StateManagement.GameState, 76
- TriDevs.TriEngine.StateManagement.GameState-Manager, 85
- TriDevs.TriEngine.StateManagement.IGameState, 107
- TriDevs.TriEngine.StateManagement.IGameState-Manager, 113
- TriDevs.TriEngine.Text, 19
- TriDevs.TriEngine.Text.Font, 70
- TriDevs.TriEngine.Text.FontConstructionConfig, 75
- TriDevs.TriEngine.Text.ITextObject, 138
- TriDevs.TriEngine.Text.TextObject, 203
- TriDevs.TriEngine.UI, 20
- TriDevs.TriEngine.UI.Control, 51
- TriDevs.TriEngine.UI.ControlManager, 56
- TriDevs.TriEngine.UI.Events, 21
- TriDevs.TriEngine.UI.IControl, 94
- TriDevs.TriEngine.UI.IControlManager, 98
- TriDevs.TriEngine.UI.Label, 144
- TriDevs.TriEngine.UI.LinkLabel, 147
- TriDevs.TriEngine.Version, 210
- TriDevs.TriEngine/Audio/AudioFormat.cs, 213
- TriDevs.TriEngine/Audio/AudioManager.cs, 213
- TriDevs.TriEngine/Audio/IAudioManager.cs, 213
- TriDevs.TriEngine/Audio/ISong.cs, 214
- TriDevs.TriEngine/Audio/ISound.cs, 214
- TriDevs.TriEngine/Audio/NullAudioManager.cs, 214
- TriDevs.TriEngine/Audio/NullSong.cs, 214
- TriDevs.TriEngine/Audio/NullSound.cs, 215
- TriDevs.TriEngine/Audio/Song.cs, 215
- TriDevs.TriEngine/Audio/Sound.cs, 215
- TriDevs.TriEngine/Color.cs, 215
- TriDevs.TriEngine/EngineException.cs, 216
- TriDevs.TriEngine/Extensions/ByteExtensions.cs, 216
- TriDevs.TriEngine/Extensions/DecimalExtensions.cs, 216
- TriDevs.TriEngine/Extensions/DoubleExtensions.cs, 216
- TriDevs.TriEngine/Extensions/EnumerationExtensions.-cs, 217
- TriDevs.TriEngine/Extensions/FloatExtensions.cs, 217
- TriDevs.TriEngine/Extensions/IntegerExtensions.cs, 217
- TriDevs.TriEngine/Extensions/StringExtensions.cs, 217
- TriDevs.TriEngine/Extensions/Vector3Extensions.cs, 218
- TriDevs.TriEngine/Extensions/Vector4Extensions.cs, 218
- TriDevs.TriEngine/GameWindow2D.cs, 218
- TriDevs.TriEngine/Graphics/Primitive.cs, 218
- TriDevs.TriEngine/Graphics/Rectangle.cs, 219
- TriDevs.TriEngine/Graphics/Triangle.cs, 219
- TriDevs.TriEngine/Helpers/Exceptions.cs, 219
- TriDevs.TriEngine/Helpers/IO.cs, 220
- TriDevs.TriEngine/Helpers/Math.cs, 220
- TriDevs.TriEngine/Helpers/Threading.cs, 220
- TriDevs.TriEngine/Input/Events/Key.cs, 220
- TriDevs.TriEngine/Input/Events/Mouse.cs, 221
- TriDevs.TriEngine/Input/IInputManager.cs, 221
- TriDevs.TriEngine/Input/InputManager.cs, 221
- TriDevs.TriEngine/Input/NullInputManager.cs, 222
- TriDevs.TriEngine/Interfaces/IDrawable.cs, 222
- TriDevs.TriEngine/Interfaces/IDrawableGameComponent.-cs, 222
- TriDevs.TriEngine/Interfaces/IGameComponent.cs, 222
- TriDevs.TriEngine/Interfaces/IUpdatable.cs, 223
- TriDevs.TriEngine/Logging/LogManager.cs, 223
- TriDevs.TriEngine/Native/Helpers.cs, 223
- TriDevs.TriEngine/Native/WinAPI.cs, 223
- TriDevs.TriEngine/Point.cs, 223
- TriDevs.TriEngine/Program.cs, 224
- TriDevs.TriEngine/Properties/AssemblyInfo.cs, 224
- TriDevs.TriEngine/Rectangle.cs, 219
- TriDevs.TriEngine/Resources/IResource.cs, 224
- TriDevs.TriEngine/Resources/ResourceException.cs, 224
- TriDevs.TriEngine/Resources/ResourceManager.cs, 224
- TriDevs.TriEngine/Serializing/Serializer.cs, 225
- TriDevs.TriEngine/Services.cs, 225
- TriDevs.TriEngine/Shaders/Shader.cs, 225
- TriDevs.TriEngine/StateManagement/GameState.cs, 225
- TriDevs.TriEngine/StateManagement/GameState-Manager.cs, 226
- TriDevs.TriEngine/StateManagement/IGameState.cs, 226
- TriDevs.TriEngine/StateManagement/IGameState-Manager.cs, 226
- TriDevs.TriEngine/Text/Font.cs, 226
- TriDevs.TriEngine/Text/FontConstructionConfig.cs, 227
- TriDevs.TriEngine/Text/FontType.cs, 227
- TriDevs.TriEngine/Text/ITextObject.cs, 227
- TriDevs.TriEngine/Text/TextObject.cs, 227
- TriDevs.TriEngine/UI/Control.cs, 228
- TriDevs.TriEngine/UI/ControlManager.cs, 228
- TriDevs.TriEngine/UI/Events/ControlClicked.cs, 228
- TriDevs.TriEngine/UI/IControl.cs, 228
- TriDevs.TriEngine/UI/IControlManager.cs, 229
- TriDevs.TriEngine/UI/Label.cs, 229
- TriDevs.TriEngine/UI/LinkLabel.cs, 229
- TriDevs.TriEngine/Version.cs, 229
- TriDevs::TriEngine::Audio
  - AudioFormat, 14
- TriDevs::TriEngine::Audio::AudioManager
  - AudioManager, 24
  - Dispose, 24



- StopAll, [25](#)
- StopAllSongs, [25](#)
- StopAllSounds, [25](#)
- TriDevs::TriEngine::Audio::IAudioManager
  - StopAll, [94](#)
  - StopAllSongs, [94](#)
  - StopAllSounds, [94](#)
- TriDevs::TriEngine::Audio::ISong
  - File, [136](#)
  - IsLooped, [136](#)
  - Pause, [135](#)
  - Play, [136](#)
  - Resume, [136](#)
  - Stop, [136](#)
  - Volume, [136](#)
- TriDevs::TriEngine::Audio::ISound
  - File, [138](#)
  - Play, [138](#)
  - Stop, [138](#)
- TriDevs::TriEngine::Audio::NullAudioManager
  - Dispose, [158](#)
  - StopAll, [158](#)
  - StopAllSongs, [158](#)
  - StopAllSounds, [159](#)
- TriDevs::TriEngine::Audio::NullSong
  - Dispose, [168](#)
  - File, [169](#)
  - IsLooped, [169](#)
  - Name, [169](#)
  - Pause, [168](#)
  - Play, [168](#)
  - Resume, [168](#)
  - Stop, [169](#)
  - Volume, [169](#)
- TriDevs::TriEngine::Audio::NullSound
  - Dispose, [171](#)
  - File, [171](#)
  - Name, [171](#)
  - Play, [171](#)
  - Stop, [171](#)
- TriDevs::TriEngine::Audio::Song
  - Dispose, [198](#)
  - File, [199](#)
  - IsLooped, [199](#)
  - Name, [199](#)
  - Pause, [198](#)
  - Play, [198](#)
  - Resume, [198](#)
  - Stop, [198](#)
  - Volume, [199](#)
- TriDevs::TriEngine::Audio::Sound
  - Dispose, [200](#)
  - File, [201](#)
  - Name, [201](#)
  - Play, [200](#)
  - Stop, [201](#)
- TriDevs::TriEngine::Color
  - A, [39](#)
  - AliceBlue, [39](#)
  - AntiqueWhite, [39](#)
  - Aqua, [39](#)
  - Aquamarine, [39](#)
  - Azure, [39](#)
  - B, [39](#)
  - Beige, [39](#)
  - Bisque, [40](#)
  - Black, [40](#)
  - BlanchedAlmond, [40](#)
  - Blue, [40](#)
  - BlueViolet, [40](#)
  - Brown, [40](#)
  - BurlyWood, [40](#)
  - CadetBlue, [40](#)
  - Chartreuse, [40](#)
  - Chocolate, [40](#)
  - Color, [34](#), [35](#)
  - Coral, [40](#)
  - CornflowerBlue, [40](#)
  - Cornsilk, [41](#)
  - Crimson, [41](#)
  - Cyan, [41](#)
  - DarkBlue, [41](#)
  - DarkCyan, [41](#)
  - DarkGoldenrod, [41](#)
  - DarkGray, [41](#)
  - DarkGreen, [41](#)
  - DarkKhaki, [41](#)
  - DarkMagenta, [41](#)
  - DarkOliveGreen, [41](#)
  - DarkOrange, [41](#)
  - DarkOrchid, [42](#)
  - DarkRed, [42](#)
  - DarkSalmon, [42](#)
  - DarkSeaGreen, [42](#)
  - DarkSlateBlue, [42](#)
  - DarkSlateGray, [42](#)
  - DarkTurquoise, [42](#)
  - DarkViolet, [42](#)
  - DeepPink, [42](#)
  - DeepSkyBlue, [42](#)
  - DimGray, [42](#)
  - DodgerBlue, [42](#)
  - Firebrick, [43](#)
  - FloralWhite, [43](#)
  - ForestGreen, [43](#)
  - FromAbgr, [36](#)
  - FromBgra, [36](#)
  - FromRgba, [37](#)
  - Fuchsia, [43](#)
  - G, [43](#)
  - Gainsboro, [43](#)
  - GhostWhite, [43](#)
  - Gold, [43](#)
  - Goldenrod, [43](#)
  - Gray, [43](#)
  - Green, [43](#)

- GreenYellow, [43](#)
- HoneyDew, [44](#)
- HotPink, [44](#)
- IndianRed, [44](#)
- Indigo, [44](#)
- Ivory, [44](#)
- Khaki, [44](#)
- Lavender, [44](#)
- LavenderBlush, [44](#)
- LawnGreen, [44](#)
- LemonChiffon, [44](#)
- LightBlue, [44](#)
- LightCoral, [44](#)
- LightCyan, [45](#)
- LightGoldenrodYellow, [45](#)
- LightGray, [45](#)
- LightGreen, [45](#)
- LightPink, [45](#)
- LightSalmon, [45](#)
- LightSeaGreen, [45](#)
- LightSkyBlue, [45](#)
- LightSlateGray, [45](#)
- LightSteelBlue, [45](#)
- LightYellow, [45](#)
- Lime, [45](#)
- LimeGreen, [46](#)
- Linen, [46](#)
- Magenta, [46](#)
- Maroon, [46](#)
- MediumAquamarine, [46](#)
- MediumBlue, [46](#)
- MediumOrchid, [46](#)
- MediumPurple, [46](#)
- MediumSeaGreen, [46](#)
- MediumSlateBlue, [46](#)
- MediumSpringGreen, [46](#)
- MediumTurquoise, [46](#)
- MediumVioletRed, [47](#)
- MidnightBlue, [47](#)
- MintCream, [47](#)
- MistyRose, [47](#)
- Moccasin, [47](#)
- NavajoWhite, [47](#)
- Navy, [47](#)
- OldLace, [47](#)
- Olive, [47](#)
- OliveDrab, [47](#)
- Orange, [47](#)
- OrangeRed, [47](#)
- Orchid, [48](#)
- PaleGoldenrod, [48](#)
- PaleGreen, [48](#)
- PaleTurquoise, [48](#)
- PaleVioletRed, [48](#)
- PapayaWhip, [48](#)
- PeachPuff, [48](#)
- Peru, [48](#)
- Pink, [48](#)
- Plum, [48](#)
- PowderBlue, [48](#)
- Purple, [48](#)
- R, [49](#)
- Red, [49](#)
- RosyBrown, [49](#)
- RoyalBlue, [49](#)
- SaddleBrown, [49](#)
- Salmon, [49](#)
- SandyBrown, [49](#)
- SeaGreen, [49](#)
- SeaShell, [49](#)
- Sienna, [49](#)
- Silver, [49](#)
- SkyBlue, [49](#)
- SlateBlue, [50](#)
- SlateGray, [50](#)
- Snow, [50](#)
- SpringGreen, [50](#)
- SteelBlue, [50](#)
- Tan, [50](#)
- Teal, [50](#)
- Thistle, [50](#)
- ToArgb, [37](#)
- ToArgb, [37](#)
- ToBgra, [37](#)
- ToColor4, [38](#)
- ToRgba, [38](#)
- ToVector3, [38](#)
- ToVector4, [39](#)
- Tomato, [50](#)
- TransparentBlack, [50](#)
- TransparentWhite, [50](#)
- Turquoise, [50](#)
- Violet, [51](#)
- Wheat, [51](#)
- White, [51](#)
- WhiteSmoke, [51](#)
- Yellow, [51](#)
- YellowGreen, [51](#)
- TriDevs::TriEngine::Extensions::ByteExtensions
  - Clamp, [26](#)
- TriDevs::TriEngine::Extensions::DecimalExtensions
  - Clamp, [62](#)
- TriDevs::TriEngine::Extensions::DoubleExtensions
  - Clamp, [64](#)
- TriDevs::TriEngine::Extensions::EnumerationExtensions
  - Has< T >, [66](#)
  - Include< T >, [66](#)
  - Missing< T >, [67](#)
  - Remove< T >, [67](#)
- TriDevs::TriEngine::Extensions::FloatExtensions
  - Clamp, [69](#)
- TriDevs::TriEngine::Extensions::IntegerExtensions
  - Clamp, [130–132](#)
- TriDevs::TriEngine::Extensions::StringExtensions
  - Replace, [202](#)
  - ReplaceFirst, [202](#)

- TriDevs::TriEngine::Extensions::Vector3Extensions
  - ToFloatArray, [208](#)
- TriDevs::TriEngine::Extensions::Vector4Extensions
  - ToFloatArray, [209](#)
- TriDevs::TriEngine::GameWindow2D
  - ClearColor, [91](#)
  - GameWindow2D, [90](#)
  - Initialize, [90](#)
  - OnDraw, [90](#)
  - OnLoad, [90](#)
  - OnRenderFrame, [90](#)
  - OnResize, [91](#)
- TriDevs::TriEngine::Graphics::Primitive
  - ColorID, [176](#)
  - ColorId, [175](#)
  - Colors, [175](#)
  - Dispose, [175](#)
  - Draw, [175](#)
  - Ids, [175](#)
  - Indices, [175](#)
  - IndicesID, [176](#)
  - Primitive, [174](#)
  - Vertices, [176](#)
  - VerticesID, [176](#)
- TriDevs::TriEngine::Graphics::Rectangle
  - Rectangle, [182](#)
- TriDevs::TriEngine::Graphics::Triangle
  - Triangle, [208](#)
- TriDevs::TriEngine::Helpers::Exceptions
  - LogException, [68](#)
- TriDevs::TriEngine::Helpers::IO
  - GetAbsolutePath, [133](#)
- TriDevs::TriEngine::Helpers::Math
  - Clamp, [153–156](#)
- TriDevs::TriEngine::Helpers::Threading
  - SetCurrentThreadName, [206](#)
- TriDevs::TriEngine::Input::Events
  - KeyDownEventHandler, [16](#)
  - KeyPressEventHandler, [17](#)
  - KeyUpEventHandler, [17](#)
  - MouseDownEventHandler, [17](#)
  - MouseUpEventHandler, [17](#)
  - MouseWheelChangedEventHandler, [17](#)
  - MouseWheelDownEventHandler, [17](#)
  - MouseWheelUpEventHandler, [18](#)
- TriDevs::TriEngine::Input::Events::KeyCharEventArgs
  - KeyChar, [142](#)
- TriDevs::TriEngine::Input::Events::KeyEventArgs
  - Key, [143](#)
- TriDevs::TriEngine::Input::IInputManager
  - IsKeyDown, [117](#)
  - IsKeyUp, [118](#)
  - IsMouseDown, [118](#)
  - IsMouseUp, [118](#)
  - IsWheelChanged, [118](#)
  - IsWheelDown, [118](#)
  - IsWheelUp, [119](#)
  - KeyDown, [121](#)
  - KeyPress, [121](#)
  - KeyPressed, [119](#)
  - KeyReleased, [119](#)
  - KeyUp, [121](#)
  - MouseDown, [121](#)
  - MousePosition, [120](#)
  - MousePressed, [119](#)
  - MouseReleased, [120](#)
  - MouseUp, [121](#)
  - MouseWheelValue, [120](#)
  - MouseX, [120](#)
  - MouseY, [120](#)
  - Update, [120](#)
  - WheelChange, [120](#)
  - WheelChanged, [121](#)
  - WheelDown, [122](#)
  - WheelUp, [122](#)
- TriDevs::TriEngine::Input::InputManager
  - InputManager, [124](#)
  - IsKeyDown, [124](#)
  - IsKeyUp, [125](#)
  - IsMouseDown, [125](#)
  - IsMouseUp, [125](#)
  - IsWheelChanged, [126](#)
  - IsWheelDown, [126](#)
  - IsWheelUp, [126](#)
  - KeyDown, [128](#)
  - KeyPress, [128](#)
  - KeyPressed, [126](#)
  - KeyReleased, [127](#)
  - KeyUp, [129](#)
  - MouseDown, [129](#)
  - MousePosition, [128](#)
  - MousePressed, [127](#)
  - MouseReleased, [127](#)
  - MouseUp, [129](#)
  - MouseWheelValue, [128](#)
  - MouseX, [128](#)
  - MouseY, [128](#)
  - Update, [128](#)
  - WheelChange, [128](#)
  - WheelChanged, [129](#)
  - WheelDown, [129](#)
  - WheelUp, [129](#)
- TriDevs::TriEngine::Input::NullInputManager
  - IsKeyDown, [161](#)
  - IsKeyUp, [161](#)
  - IsMouseDown, [162](#)
  - IsMouseUp, [162](#)
  - IsWheelChanged, [162](#)
  - IsWheelDown, [162](#)
  - IsWheelUp, [163](#)
  - KeyDown, [165](#)
  - KeyPress, [165](#)
  - KeyPressed, [163](#)
  - KeyReleased, [163](#)
  - KeyUp, [165](#)
  - MouseDown, [165](#)

- MousePosition, 165
- MousePressed, 164
- MouseReleased, 164
- MouseUp, 165
- MouseWheelValue, 165
- MouseX, 165
- MouseY, 165
- Update, 164
- WheelChange, 164
- WheelChanged, 166
- WheelDown, 166
- WheelUp, 166
- TriDevs::TriEngine::Interfaces::IDrawable
  - Draw, 103
- TriDevs::TriEngine::Interfaces::IGameComponent
  - Disable, 106
  - Enable, 106
  - Enabled, 107
- TriDevs::TriEngine::Interfaces::IUpdatable
  - Update, 141
- TriDevs::TriEngine::Logging::LogManager
  - ClearOldLogs, 151
  - DestroyConsole, 151
  - GetLogger, 151
  - LoadConfig, 152
  - SetupConsole, 152
- TriDevs::TriEngine::Native::Helpers
  - GetErrorMessage, 92
  - GetLastError, 92
  - GetLastErrorInfo, 92
  - GetLastErrorMessage, 92
- TriDevs::TriEngine::Point< T >
  - Point, 172
  - X, 172
  - Y, 172
  - Z, 173
- TriDevs::TriEngine::Program
  - Dispose, 177
  - ID, 178
  - Program, 177
- TriDevs::TriEngine::Rectangle
  - Color, 180
  - Equals, 180
  - Height, 180
  - Intersects, 180
  - Rectangle, 179
  - Width, 180
  - X, 180
  - Y, 181
- TriDevs::TriEngine::Resources::IResource
  - Name, 134
- TriDevs::TriEngine::Resources::ResourceManager
  - Add, 185
  - BasePath, 189
  - FontPath, 189
  - Get, 185
  - Get< T >, 185
  - GetAll< T >, 186
- Has, 186
- Has< T >, 187
- LoadFont, 187
- LoadShader, 188
- LoadSong, 188
- LoadSound, 189
- ShaderPath, 189
- SongPath, 189
- SoundPath, 189
- TriDevs::TriEngine::Serializing::Serializer
  - Deserialize< T >, 190
  - Serialize< T >, 191
- TriDevs::TriEngine::Services
  - Audio, 193
  - Input, 193
  - Provide, 192
- TriDevs::TriEngine::Shaders::Shader
  - Dispose, 195
  - File, 195
  - GetDefaultName, 195
  - ID, 195
  - Name, 195
  - Shader, 195
- TriDevs::TriEngine::StateManagement::GameState
  - AddComponent, 78
  - Components, 84
  - Draw, 79
  - GameState, 78
  - GetAllComponents, 79, 80
  - GetComponent, 80, 81
  - HasComponent, 81, 82
  - Load, 82
  - Pause, 82
  - Paused, 84
  - RemoveAllComponents, 82, 83
  - RemoveComponent, 83
  - Unload, 84
  - Unpause, 84
  - Update, 84
- TriDevs::TriEngine::StateManagement::GameState-Manager
  - ActiveState, 89
  - Draw, 86
  - GameStateManager, 86
  - Peek, 87
  - Pop, 87
  - Push, 87
  - StateCount, 89
  - Switch, 88
  - Update, 88
- TriDevs::TriEngine::StateManagement::IGameState
  - AddComponent, 109
  - GetAllComponents, 109
  - GetComponent, 110
  - HasComponent, 110, 111
  - Load, 111
  - Pause, 111
  - Paused, 113

- RemoveAllComponents, 112
- RemoveComponent, 112
- Unload, 112
- Unpause, 113
- TriDevs::TriEngine::StateManagement::IGameState-Manager
  - ActiveState, 115
  - Peek, 114
  - Pop, 114
  - Push, 115
  - StateCount, 115
  - Switch, 115
- TriDevs::TriEngine::Text
  - FontType, 20
- TriDevs::TriEngine::Text::Font
  - Color, 74
  - Dispose, 73
  - File, 74
  - Font, 72
  - GetDefaultName, 74
  - Name, 74
  - QFont, 74
  - Size, 74
  - ToString, 74
  - Type, 74
- TriDevs::TriEngine::Text::FontConstructionConfig
  - BuildConfig, 76
  - FontConstructionConfig, 75
  - LoadConfig, 76
- TriDevs::TriEngine::Text::ITextObject
  - Alignment, 140
  - Bounds, 140
  - Color, 140
  - Draw, 139
  - Font, 140
  - Position, 140
  - Text, 140
- TriDevs::TriEngine::Text::TextObject
  - Alignment, 205
  - Bounds, 205
  - Color, 205
  - Draw, 205
  - Font, 205
  - Position, 205
  - Text, 205
  - TextObject, 204
- TriDevs::TriEngine::UI::Control
  - Clicked, 56
  - Color, 55
  - Disable, 53
  - Draw, 53, 54
  - Enable, 54
  - Enabled, 55
  - Hide, 54
  - OnClicked, 54
  - Position, 55
  - Rectangle, 56
  - Show, 55
  - Size, 56
  - Text, 56
  - Update, 55
  - Visible, 56
- TriDevs::TriEngine::UI::ControlManager
  - AddControl, 58
  - ControlManager, 58
  - Disable, 59
  - Draw, 59
  - Enable, 59
  - Enabled, 62
  - HasControl, 59, 60
  - RemoveAllControls, 60, 61
  - RemoveControl, 61
  - Update, 61
- TriDevs::TriEngine::UI::Events
  - ControlClickedEventHandler, 21
- TriDevs::TriEngine::UI::IControl
  - Clicked, 98
  - Color, 97
  - Disable, 96
  - Enable, 96
  - Enabled, 97
  - Hide, 97
  - Position, 98
  - Rectangle, 98
  - Show, 97
  - Size, 98
  - Text, 98
  - Visible, 98
- TriDevs::TriEngine::UI::IControlManager
  - AddControl, 101
  - HasControl, 101
  - RemoveAllControls, 102
  - RemoveControl, 102
- TriDevs::TriEngine::UI::Label
  - Alignment, 147
  - Draw, 146
  - Position, 147
  - SetFont, 146
  - Text, 147
  - Update, 146
  - UpdateTextObject, 146
- TriDevs::TriEngine::UI::LinkLabel
  - OnClicked, 149
  - Url, 150
- TriDevs::TriEngine::Version
  - Major, 211
  - Minor, 211
  - Patch, 211
  - Suffix, 211
  - VersionString, 211
  - VersionStringFormat, 211
  - VersionStringFormatWithSuffix, 211
- Triangle
  - TriDevs::TriEngine::Graphics::Triangle, 208
- Turquoise
  - TriDevs::TriEngine::Color, 50

- Type
  - TriDevs::TriEngine::Text::Font, [74](#)
- Unload
  - TriDevs::TriEngine::StateManagement::Game-State, [84](#)
  - TriDevs::TriEngine::StateManagement::IGame-State, [112](#)
- Unpause
  - TriDevs::TriEngine::StateManagement::Game-State, [84](#)
  - TriDevs::TriEngine::StateManagement::IGame-State, [113](#)
- Unsupported
  - TriDevs::TriEngine::Text, [20](#)
- Update
  - TriDevs::TriEngine::Input::IInputManager, [120](#)
  - TriDevs::TriEngine::Input::InputManager, [128](#)
  - TriDevs::TriEngine::Input::NullInputManager, [164](#)
  - TriDevs::TriEngine::Interfaces::IUpdatable, [141](#)
  - TriDevs::TriEngine::StateManagement::Game-State, [84](#)
  - TriDevs::TriEngine::StateManagement::Game-StateManager, [88](#)
  - TriDevs::TriEngine::UI::Control, [55](#)
  - TriDevs::TriEngine::UI::ControlManager, [61](#)
  - TriDevs::TriEngine::UI::Label, [146](#)
- UpdateTextObject
  - TriDevs::TriEngine::UI::Label, [146](#)
- Url
  - TriDevs::TriEngine::UI::LinkLabel, [150](#)
- VersionString
  - TriDevs::TriEngine::Version, [211](#)
- VersionStringFormat
  - TriDevs::TriEngine::Version, [211](#)
- VersionStringFormatWithSuffix
  - TriDevs::TriEngine::Version, [211](#)
- Vertices
  - TriDevs::TriEngine::Graphics::Primitive, [176](#)
- VerticesID
  - TriDevs::TriEngine::Graphics::Primitive, [176](#)
- Violet
  - TriDevs::TriEngine::Color, [51](#)
- Visible
  - TriDevs::TriEngine::UI::Control, [56](#)
  - TriDevs::TriEngine::UI::IControl, [98](#)
- Volume
  - TriDevs::TriEngine::Audio::ISong, [136](#)
  - TriDevs::TriEngine::Audio::NullSong, [169](#)
  - TriDevs::TriEngine::Audio::Song, [199](#)
- Wav
  - TriDevs::TriEngine::Audio, [14](#)
- Wheat
  - TriDevs::TriEngine::Color, [51](#)
- WheelChange
  - TriDevs::TriEngine::Input::IInputManager, [120](#)
  - TriDevs::TriEngine::Input::InputManager, [128](#)
  - TriDevs::TriEngine::Input::NullInputManager, [164](#)
- WheelChanged
  - TriDevs::TriEngine::Input::IInputManager, [121](#)
  - TriDevs::TriEngine::Input::InputManager, [129](#)
  - TriDevs::TriEngine::Input::NullInputManager, [166](#)
- WheelDown
  - TriDevs::TriEngine::Input::IInputManager, [122](#)
  - TriDevs::TriEngine::Input::InputManager, [129](#)
  - TriDevs::TriEngine::Input::NullInputManager, [166](#)
- WheelUp
  - TriDevs::TriEngine::Input::IInputManager, [122](#)
  - TriDevs::TriEngine::Input::InputManager, [129](#)
  - TriDevs::TriEngine::Input::NullInputManager, [166](#)
- White
  - TriDevs::TriEngine::Color, [51](#)
- WhiteSmoke
  - TriDevs::TriEngine::Color, [51](#)
- Width
  - TriDevs::TriEngine::Rectangle, [180](#)
- X
  - TriDevs::TriEngine::Point< T >, [172](#)
  - TriDevs::TriEngine::Rectangle, [180](#)
- Y
  - TriDevs::TriEngine::Point< T >, [172](#)
  - TriDevs::TriEngine::Rectangle, [181](#)
- Yellow
  - TriDevs::TriEngine::Color, [51](#)
- YellowGreen
  - TriDevs::TriEngine::Color, [51](#)
- Z
  - TriDevs::TriEngine::Point< T >, [173](#)