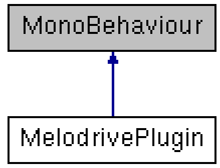
MelodrivePlugin Class Reference

Inheritance diagram for MelodrivePlugin:



Public Member Functions

delegate void	**DebugHandler** (string message)
delegate void	**NoteOnHandler** (string part, int num, int velocity)
delegate void	**NoteOffHandler** (string part, int num)
delegate void	**ParamChangeHandler** (string part, string param, float value)
delegate void	**BarHandler** (float bar)
delegate void	**BeatHandler** (float beat, float bar)
delegate void	**BeatTickHandler** (int tick, float beat, float bar)
delegate void	**TempoChangeHandler** (float bpm)
delegate void	**NewMusicalSeedHandler** (string name)
delegate void	**CueChangeHandler** (string cue, string seedName, string style)
delegate void	**ProjectLoadHandler** ()
void	**SetInstrumentsPathToStreamingAssets** ()
void	**SetInstrumentsPath** (string path)
void	**Init** (string style=DEFAULT_STYLE, string initEmotion=DEFAULT_EMOTION)
void	**SaveProject** (string filename)
void	**LoadProject** (string filename)
string []	**GetStyles** ()
void	**PreloadStyles** ()
void	**SetStyle** (string style)
string	**GetStyle** ()
void	**SetMusicalSeed** (string seedName)
string	**GetMusicalSeed** ()
void	**CreateMusicalSeed** (bool setActive=false)
void	**SaveMusicalSeed** (string seedName, string filename)
void	**LoadMusicalSeed** (string filename)
string []	**GetMusicalSeeds** ()
string []	**GetEnsembles** ()
string	**GetCurrentEnsemble** ()
void	**SetEnsemble** (string value)
void	**SetChiptuneMode** (bool value)

bool	**GetChiptuneMode** ()
string	**GetEmotionMode** ()
void	**SetEmotionMode** (string mode)
float	**GetEmotionalVelocity** ()
void	**SetEmotionalVelocity** (float value)
string	**GetEmotion** ()
void	**SetEmotion** (string emotion)
Vector2	**GetVA** ()
void	**SetVA** (Vector2 value)
int	**AddEmotionalPoint** (float x, float y, [MarshalAs(UnmanagedType.LPStr)] string mood)
void	**RemoveEmotionalPoint** (int id)
void	**SetEmotionalPointPosition** (int id, float x, float y)
void	**SetEmotionAtPoint** (int id, [MarshalAs(UnmanagedType.LPStr)] string emotion)
void	**ClearEmotionalPoints** ()
void	**SetListenerPosition** (float x, float y)
float	**GetEmotionalStrength** ()
void	**SetEmotionalStrength** (float value)
void	**SetTempoScale** (float value)
void	**Play** ()
void	**Pause** ()
void	**Stop** ()
void	**SetCue** (string cueName)
void	**SetStateOptions** (string style, string musicalSeed, string ensemble)
void	**SetMasterGain** (float value)
void	**SetLimiterEnabled** (bool value)
float []	**GetRMS** ()

Public Attributes

const string	**DEFAULT_STYLE** = "piano"
const string	**DEFAULT_EMOTION** = "neutral"
bool	**playOnStart** = false
string	**initStyle** = DEFAULT_STYLE
string	**initEmotion** = DEFAULT_EMOTION
bool	**chiptuneMode** = false

Events

DebugHandler	**DebugLog**
NoteOnHandler	**NoteOn**
NoteOffHandler	**NoteOff**

ParamChangeHandler	**ParamChange**
BarHandler	**Bar**
BeatHandler	**Beat**
BeatTickHandler	**BeatTick**
TempoChangeHandler	**TempoChange**
NewMusicalSeedHandler	**NewMusicalSeed**
CueChangeHandler	**CueChange**
ProjectLoadHandler	**ProjectLoad**

Detailed Description

This is the Melodrive controller class. Other Game objects in Unity should control this class, or listen to the events fired.

Member Function Documentation

◆ AddEmotionalPoint()

int MelodrivePlugin.AddEmotionalPoint	(	float	x,
		float	y,
		[MarshalAs(UnmanagedType.LPStr)] string	mood
	)		

inline

Add an emotional point in game-space.

Parameters

float	x - the x co-ord
float	y - the y co-ord
string	mood - the mood of the point e.g. "happy"

Returns

int - the id of the created point

◆ ClearEmotionalPoints()

void MelodrivePlugin.ClearEmotionalPoints ( ) inline

Cleares the emotional points from the scene

◆ CreateMusicalSeed()

void MelodrivePlugin.CreateMusicalSeed (bool setActive = false) inline

Creates a new Musical Seed. This will trigger a NewMusicalSeed event when complete.

Parameters

bool setActive - set to true to trigger a Musical Seed change

◆ GetChiptuneMode()

bool MelodrivePlugin.GetChiptuneMode ( ) inline

Returns the current state of Chiptune mode.

◆ GetCurrentEnsemble()

string MelodrivePlugin.GetCurrentEnsemble ( ) inline

Returns the currently active Ensemble

◆ GetEmotion()

string MelodrivePlugin.GetEmotion ( ) inline

Returns the current emotion as a string e.g. "happy", "sad"

◆ GetEmotionalStrength()

float MelodrivePlugin.GetEmotionalStrength ( ) inline

Returns the emotional strength, which is how strong emotional points affect the emotion.

◆ GetEmotionalVelocity()

float MelodrivePlugin.GetEmotionalVelocity ( ) inline

Returns the emotional velocity, which is how fast emotion changes happen.

◆ GetEmotionMode()

string MelodrivePlugin.GetEmotionMode ( ) inline

Returns the current emotion control "mode". Options are "positional" (default), "discrete" or "direct". The mode changes automatically when you use the SetVA or SetEmotion methods, but you have to chnage back to "positional" mode if you'd like to use EmotionalPoints again.

◆ GetEnsembles()

string [] MelodrivePlugin.GetEnsembles ( ) inline

Returns a list of availiable Ensembles in a Style.

◆ GetMusicalSeeds()

string [] MelodrivePlugin.GetMusicalSeeds ( ) inline

Returns a list of the Musical Seeds.

◆ GetStyle()

string MelodrivePlugin.GetStyle ( ) inline

Returns the currently playing Style.

◆ GetStyles()

string [] MelodrivePlugin.GetStyles ( ) inline

Returns the list of supported Styles in Melodrive

◆ GetVA()

Vector2 MelodrivePlugin.GetVA ( ) inline

Returns the current Valence (x) Arousal (y) point

◆ Init()

```
void MelodrivePlugin.Init ( string  style = DEFAULT_STYLE,
                            string  initEmotion = DEFAULT_EMOTION
                          )
```

inline

Initialises Melodrive with a given style and emotion

Parameters

string style - the style

string initEmotion - the initial emotion

◆ LoadMusicalSeed()

void MelodrivePlugin.LoadMusicalSeed (string filename) inline

Loads a Musical Seed into the current project.

◆ LoadProject()

void MelodrivePlugin.LoadProject (string filename) inline

Loads a project from the specified path

◆ Pause()

void MelodrivePlugin.Pause ( ) inline

Pauses playback

◆ Play()

void MelodrivePlugin.Play ( ) inline

Starts playback

◆ PreloadStyles()

void MelodrivePlugin.PreloadStyles ( ) inline

Preloads all styles to avoid audio artifacts

◆ RemoveEmotionalPoint()

void MelodrivePlugin.RemoveEmotionalPoint (int id) inline

Remove an emotional point from Melodrive.

Parameters

int id - the id of the point to remove

◆ SaveMusicalSeed()

void MelodrivePlugin.SaveMusicalSeed (string seedName,
string filename
) inline

Saves a Given Musical Seed to a file for later loading.

◆ SaveProject()

void MelodrivePlugin.SaveProject (string filename) inline

Saves a project to the specified path

◆ SetChiptuneMode()

void MelodrivePlugin.SetChiptuneMode (bool value) inline

Activates/deactivates Melodrive's Chiptune mode. While active, all the ensembles will be Chiptune until deactivated. Note Chiptune mode is deactivated if SetEnsemble is called with a non-chiptune ensemble.

◆ SetCue()

void MelodrivePlugin.SetCue (string cueName) inline

Activates a Cue in Melodrive

◆ SetEmotion()

void MelodrivePlugin.SetEmotion (string emotion) inline

Sets the current emotionas a string e.g. "happy", "sad". This will also set the emotion mode to "discrete".

◆ SetEmotionalPointPosition()

```
void MelodrivePlugin.SetEmotionalPointPosition ( int   id,
                                                 float x,
                                                 float y
                                               )
```
inline

Update an emotional point's position in game-space.4

Parameters

int	id - Melodrives ID for the point
float	x - the x co-ord
float	y - the y co-ord

◆ SetEmotionalStrength()

```
void MelodrivePlugin.SetEmotionalStrength ( float value )
```
inline

Sets the emotional strength, which is how strong emotional points affect the emotion.

◆ SetEmotionalVelocity()

```
void MelodrivePlugin.SetEmotionalVelocity ( float value )
```
inline

Sets the emotional velocity, which is how fast emotion changes happen.

◆ SetEmotionAtPoint()

```
void MelodrivePlugin.SetEmotionAtPoint ( int                                     id,
                                         [MarshalAs(UnmanagedType.LPStr)] string emotion
                                       )
```
inline

Updates an emotional points mood

Parameters

int	id - Melodrive's ID for the point
string	mood - the new mood

◆ SetEmotionMode()

void MelodrivePlugin.SetEmotionMode (string mode) inline

Sets the current emotion mode. Options: "positional" - will listen to the listener position and the emotional points to determine emotion "discrete" - set when using SetEmotion. Emotion values map to discrete points "direct" - set when using SetVA. Allows direct control over the VA space

◆ SetEnsemble()

void MelodrivePlugin.SetEnsemble (string value) inline

Sets the active Ensemble. This will Trigger a CueChange.

◆ SetInstrumentsPath()

void MelodrivePlugin.SetInstrumentsPath (string path) inline

Sets the Melodrive AudioPlugin to load instruments from the specified folder

◆ SetInstrumentsPathToStreamingAssets()

void MelodrivePlugin.SetInstrumentsPathToStreamingAssets ( ) inline

Sets the Melodrive AudioPlugin to load instruments from the StreamingAssets folder

◆ SetLimiterEnabled()

void MelodrivePlugin.SetLimiterEnabled (bool value) inline

Turns the limiter on/off

◆ SetListenerPosition()

void MelodrivePlugin.SetListenerPosition (float x,
float y
) inline

Updates the position of the listener in world space.

Parameters

float x

float y

◆ SetMasterGain()

void MelodrivePlugin.SetMasterGain (float value) inline

Sets the gain of the Master bus

◆ SetMusicalSeed()

void MelodrivePlugin.SetMusicalSeed (string seedName) inline

Sets the Musical Seed in Melodrive. This will trigger a CueChange event.

◆ SetStateOptions()

void MelodrivePlugin.SetStateOptions (string style,
string musicalSeed,
string ensemble
) inline

Sets the playback state in Melodrive

◆ SetStyle()

void MelodrivePlugin.SetStyle (string style) inline

Sets the current style

◆ SetTempoScale()

void MelodrivePlugin.SetTempoScale (float value) inline

Sets the tempo "scale", which is a multiplier on the base tempo that Melodrive chooses.

◆ SetVA()

void MelodrivePlugin.SetVA (Vector2 value) inline

Set the Valence/Arousal point. This will also set the emotion mode to "direct".

◆ Stop()

void MelodrivePlugin.Stop ( ) inline

Stops playback

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

- unity/MelodrivePackage/Assets/Melodrive/Scripts/MelodrivePlugin.cs