Runtime Model Importer

1.0.2

Generated by Doxygen 1.8.5

Thu Dec 12 2013 21:05:26

Contents

Chapter 1

Main Page

1.1 About Runtime Model Importer

This plugin can import model at runtime.

This is developed to expand the possibilities of a 3d game.

Development is based on FBXSDK. You can use for windows standalone only.

Target formats are simple animationFBX or OBJ, others are not recommended.

The development is now underway, improvements will continue.

Supported elements

- Vertex, uv, normals, tangents.
- Triangulate meshes.
- Texture mapping (Base, Alpha, Specular, Normal).
- Material mapping (Main, Specular, Emissive, Shininess).
- Position, Rotation, Scale, Parent.
- Bind pose.
- Process animation data (transform only).
- Multiple takes of animation.

Not supported elements

- Animation curve interpolation.
- Multi animation layer.
- Camera and light.
- Texture tiling and offset.
- BlendShape.

File format

- FBX files compatible with FBX file format versions 7.3, 7.2, 7.1, 7.0, 6.1, and 6.0
- Autodesk AutoCAD DXF (.dxf) Version 13 and earlier.
- Collada DAE (.dae). Version 1.5 and earlier.
- Alias OBJ (.obj) All versions.
- 3D Studio 3DS (.3ds) All versions.

Rules of animation model

- Use 'Bake keys' and 'Resample All' to the animation.
- If you want to adjust the start or end frame, use not the export dialog but the time setting.

Chapter 2

Namespace Index

2.1 Namespace List

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

Chapter 3

Class Index

3.1 Class List

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

Chapter 4

Namespace Documentation

4.1 Package RMI

Classes

- class **RMIManagerHelper**

  *Represents foundation for **RMI** (p. 7).*
- class **RMITask**

  Represents one model import process.

Chapter 5

Class Documentation

5.1 RMI.RMITask.AnimationSetting Class Reference

Represents import setting of the animation.

Public Attributes

- bool **isReadAnimation** = true

  Read animation curves and attach Animation component to root.

5.1.1 Detailed Description

Represents import setting of the animation.

5.1.2 Member Data Documentation

5.1.2.1 bool RMI.RMITask.AnimationSetting.isReadAnimation = true

Read animation curves and attach Animation component to root.

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

- Assets/RMI/RMI/Scripts/RMITask.cs

5.2 RMI.RMITask.BoneSetting Class Reference

Represents import setting of the bone.

Public Attributes

- bool **isReadSkinning** = true

  Read skinning and attach Skinned Mesh Renderer component.

5.2.1 Detailed Description

Represents import setting of the bone.

5.2.2 Member Data Documentation

5.2.2.1 bool RMI.RMITask.BoneSetting.isReadSkinning = true

Read skinning and attach Skinned Mesh Renderer component.

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

- Assets/RMI/RMI/Scripts/RMITask.cs

5.3 RMI.RMITask.MeshSetting Class Reference

Represents import setting of the mesh.

Public Attributes

- bool **isReadMesh** = true

  Read mesh and attach Mesh Renderer component.
- bool **isReadMaterials** = true

  Read material and texture.
- bool **isRecalculateNormals** = false

  Apply RecalculateNormals method.
- bool **isRecalculateTangents** = true

  Calculate tangents auto.
- bool **isHideProcess** = false

  Hide or show object in process.

5.3.1 Detailed Description

Represents import setting of the mesh.

5.3.2 Member Data Documentation

5.3.2.1 bool RMI.RMITask.MeshSetting.isHideProcess = false

Hide or show object in process.

5.3.2.2 bool RMI.RMITask.MeshSetting.isReadMaterials = true

Read material and texture.

5.3.2.3 bool RMI.RMITask.MeshSetting.isReadMesh = true

Read mesh and attach Mesh Renderer component.

5.3.2.4 bool RMI.RMITask.MeshSetting.isRecalculateNormals = false

Apply RecalculateNormals method.

5.3.2.5 bool RMI.RMITask.MeshSetting.isRecalculateTangents = true

Calculate tangents auto.

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

- Assets/RMI/RMI/Scripts/RMITask.cs

5.4 RMI.RMIManagerHelper Class Reference

Represents foundation for **RMI** (p. 7).

Inherits MonoBehaviour.

Public Member Functions

- static void **RMIInitalize** ()
  *Initalize **RMI** (p. 7).*

5.4.1 Detailed Description

Represents foundation for **RMI** (p. 7).

Use **RMITask** (p. 11) from a class that inherits from this.

5.4.2 Member Function Documentation

5.4.2.1 static void RMI.RMIManagerHelper.RMIInitalize ( )

Initalize **RMI** (p. 7).

Warning

Make sure that you call before and after use of **RMI** (p. 7).

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

- Assets/RMI/RMI/Scripts/RMIManagerHelper.cs

5.5 RMI.RMITask Class Reference

Represents one model import process.

Inherits MonoBehaviour.

Classes

- class **AnimationSetting**
  Represents import setting of the animation.
- class **BoneSetting**
  Represents import setting of the bone.
- class **MeshSetting**
  Represents import setting of the mesh.

Public Member Functions

- delegate void **RMIEndCallback** (**RMITask** inRMITask)

  Called when import is finished.
- void **Load** (string inPath, Vector3 inPosition, Quaternion inRotation, **RMITask.RMIEndCallback** inEndCallback)

  Start model import process.

Public Attributes

- **RMITask.MeshSetting meshSetting**

  Import setting of Mesh.
- **RMITask.BoneSetting boneSetting**

  Import setting of Bone.
- **RMITask.AnimationSetting animationSetting**

  Import setting of Animation.

Properties

- string **FilePath** `[get, set]`

  Gets the import file path.
- GameObject **ImportedObject** `[get, set]`

  Gets the imported object.

5.5.1 Detailed Description

Represents one model import process.

You can use some import settings.

5.5.2 Member Function Documentation

5.5.2.1 void RMI.RMITask.Load (string *inPath*, Vector3 *inPosition*, Quaternion *inRotation*, RMITask.RMIEndCallback *inEndCallback*) `[inline]`

Start model import process.

Object is generated in scene when succeed.

This method process async in part. You need callback function.

Empty object which have name is always applied to root.

Parallel processing is not possible.

Instead it will be a wait state and start import automatically.

Warning

- Don't use wide character texture path.
- Supported texture format are jpg and png.

Parameters

inPath	Import file path.
inPosition	Generate position.
inRotation	Generate rotation.
inEndCallback	Function called at the end of the process.

5.5.2.2 delegate void RMI.RMITask.RMIEndCallback (RMITask *inRMITask*)

Called when import is finished.

This callback is occurred regardless of success/failure of import. Generated object is inactive after import, so you will use SetActive method. And you need to destroy RMITask.gameObject in callback.

5.5.3 Property Documentation

5.5.3.1 GameObject RMI.RMITask.ImportedObject [get],[set]

Gets the imported object.

On failure, a null is returned.

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

- Assets/RMI/RMI/Scripts/RMITask.cs