Runtime Serialization

1.0

Generated by Doxygen 1.8.9.1

Thu May 14 2015 13:30:36

Contents

1 Namespace Index **1**

2 Hierarchical Index **3**

2.1 Class Hierarchy . . . 3

3 Class Index **5**

3.1 Class List . . . 5

4 Namespace Documentation **7**

4.1 Package UnityEngine . . . 7

4.2 Package VoxelBusters . . . 7

4.3 Package VoxelBusters.RuntimeSerialization . . . 7

4.3.1 Detailed Description . . . 8

4.3.2 Enumeration Type Documentation . . . 8

4.3.2.1 eSaveTarget . . . 8

4.4 Package VoxelBusters.RuntimeSerialization.Benchmark . . . 8

5 Class Documentation **9**

5.1 IRuntimeSerializable . . . 9

5.1.1 Detailed Description . . . 9

5.1.2 Member Function Documentation . . . 9

5.1.2.1 WriteSerializationData . . . 9

5.1.2.2 ReadSerializationData . . . 9

5.2 IRuntimeSerializationCallback . . . 10

5.2.1 Detailed Description . . . 10

5.2.2 Member Function Documentation . . . 10

5.2.2.1 OnAfterRuntimeSerialize . . . 10

5.2.2.2 OnAfterRuntimeDeserialize . . . 10

5.3 IRuntimeSerializationEventListener . . . 10

5.3.1 Detailed Description . . . 10

5.3.2 Member Function Documentation . . . 10

5.3.2.1 OnAfterRuntimeSerialize . . . 10

5.3.2.2 OnAfterRuntimeDeserialize . . . 11

5.4 IRuntimeSerializationExtension . . . 11
5.4.1 Detailed Description . . . 11
5.4.2 Member Function Documentation . . . 11
5.4.2.1 WriteSerializationData . . . 11
5.4.2.2 ReadSerializationData . . . 11
5.5 RSManager . . . 12
5.5.1 Detailed Description . . . 12
5.5.2 Member Function Documentation . . . 12
5.5.2.1 Serialize< T > . . . 12
5.5.2.2 Serialize< T > . . . 13
5.5.2.3 DeserializeData< T > . . . 13
5.5.2.4 Deserialize< T > . . . 13
5.5.2.5 GetSerializationData . . . 14
5.5.2.6 RestoreSerializationData . . . 14
5.5.2.7 RegisterEventListener . . . 14
5.5.2.8 UnRegisterEventListener . . . 14
5.5.2.9 Remove . . . 14
5.5.2.10 RemoveAll . . . 15
5.5.2.11 Save . . . 15
5.5.2.12 Purge . . . 15
5.6 RuntimeSerializableAttribute . . . 15
5.6.1 Detailed Description . . . 15
5.6.2 Constructor & Destructor Documentation . . . 15
5.6.2.1 RuntimeSerializableAttribute . . . 15
5.6.2.2 RuntimeSerializableAttribute . . . 15
5.6.2.3 RuntimeSerializableAttribute . . . 16
5.7 RuntimeSerializationInfo . . . 16
5.7.1 Detailed Description . . . 16
5.7.2 Member Function Documentation . . . 16
5.7.2.1 AddValue< T > . . . 16
5.7.2.2 AddValue . . . 17
5.7.2.3 GetValue< T > . . . 17
5.7.2.4 GetValue . . . 17
5.8 RuntimeSerializeFieldAttribute . . . 17
5.8.1 Detailed Description . . . 17

Index **19**

Chapter 1

Namespace Index

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

IRuntimeSerializable 9
IRuntimeSerializationCallback 10
IRuntimeSerializationEventListener 10
IRuntimeSerializationExtension 11
RSManager 12
RuntimeSerializableAttribute 15
RuntimeSerializationInfo 16
RuntimeSerializeFieldAttribute 17

Chapter 3

Class Index

3.1 Class List

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

IRuntimeSerializable
: Implement this interface to control serialization and deserialization of RuntimeSerializable↩ Attribute object. 9

IRuntimeSerializationCallback
: Implement this interface to receive implicit callbacks on RuntimeSerializableAttribute object. . . 10

IRuntimeSerializationEventListener
: Implement this interface to observe serialization process of RuntimeSerializableAttribute object. 10

IRuntimeSerializationExtension
: Implement this interface to support runtime serialization for classes which belong to external assembly. 11

RSManager
: RSManager class is responsible for serializing and deserializing objects at runtime. 12

RuntimeSerializableAttribute
: Indicates that a class can be serialized at runtime. 15

RuntimeSerializationInfo
: Stores all the data required to serialize or deserialize an object. 16

RuntimeSerializeFieldAttribute
: Indicates that a field of a RuntimeSerializableAttribute class should be serialized at runtime. . . 17

Chapter 4

Namespace Documentation

4.1 Package UnityEngine

4.2 Package VoxelBusters

Namespaces

- package RuntimeSerialization

  The VoxelBusters.RuntimeSerialization namespace contains classes that can be used for serializing and deserializing objects at runtime. Serialization is a process of converting an object into binary stream of data and storing it on disk. Deserialization is a process of reading stored data and recreating objects from it.

4.3 Package VoxelBusters.RuntimeSerialization

The VoxelBusters.RuntimeSerialization namespace contains classes that can be used for serializing and deserializing objects at runtime. Serialization is a process of converting an object into binary stream of data and storing it on disk. Deserialization is a process of reading stored data and recreating objects from it.

Namespaces

- package Benchmark

Classes

- interface IRuntimeSerializable

  Implement this interface to control serialization and deserialization of RuntimeSerializableAttribute object.
- interface IRuntimeSerializationCallback

  Implement this interface to receive implicit callbacks on RuntimeSerializableAttribute object.
- interface IRuntimeSerializationEventListener

  Implement this interface to observe serialization process of RuntimeSerializableAttribute object.
- interface IRuntimeSerializationExtension

  Implement this interface to support runtime serialization for classes which belong to external assembly.
- class RSManager

  RSManager class is responsible for serializing and deserializing objects at runtime.
- class RuntimeSerializableAttribute

  Indicates that a class can be serialized at runtime.

- class RuntimeSerializationInfo

  Stores all the data required to serialize or deserialize an object.
- class RuntimeSerializeFieldAttribute

  Indicates that a field of a RuntimeSerializableAttribute class should be serialized at runtime.

Enumerations

- enum eSaveTarget : byte {
  PLAYER_PREFS,
  FILE_SYSTEM }

  Saves serialization data to specified save target.

4.3.1 Detailed Description

The VoxelBusters.RuntimeSerialization namespace contains classes that can be used for serializing and deserializing objects at runtime. Serialization is a process of converting an object into binary stream of data and storing it on disk. Deserialization is a process of reading stored data and recreating objects from it.

The IRuntimeSerializable interface provides a way to control their own serialization behaviour. The IRuntime↩SerializationExtension interface provides a way to support runtime serialization for classes which belong to external assembly.

4.3.2 Enumeration Type Documentation

4.3.2.1 enum eSaveTarget : byte

Saves serialization data to specified save target.

Enumerator

PLAYER_PREFS Saves serialization data to PlayerPrefs.

FILE_SYSTEM Saves serialization data to files. This option is not available for Unity WebPlayer.

4.4 Package VoxelBusters.RuntimeSerialization.Benchmark

Chapter 5

Class Documentation

5.1 IRuntimeSerializable

Implement this interface to control serialization and deserialization of RuntimeSerializableAttribute object.

Public Member Functions

- void WriteSerializationData (RuntimeSerializationInfo _info)

  Populate RuntimeSerializationInfo with the properties required to serialize RuntimeSerializableAttribute object.
- void ReadSerializationData (RuntimeSerializationInfo _info)

  Retrieve properties of RuntimeSerializableAttribute object from RuntimeSerializationInfo.

5.1.1 Detailed Description

Implement this interface to control serialization and deserialization of RuntimeSerializableAttribute object.

5.1.2 Member Function Documentation

5.1.2.1 void WriteSerializationData (RuntimeSerializationInfo *_info*)

Populate RuntimeSerializationInfo with the properties required to serialize RuntimeSerializableAttribute object.

Parameters

_info	The RuntimeSerializationInfo provides interface to store properties of RuntimeSerializableAttribute object to be serialized.

5.1.2.2 void ReadSerializationData (RuntimeSerializationInfo *_info*)

Retrieve properties of RuntimeSerializableAttribute object from RuntimeSerializationInfo.

Parameters

_info	The RuntimeSerializationInfo provides interface to retrieve RuntimeSerializableAttribute object property values.

5.2 IRuntimeSerializationCallback

Implement this interface to receive implicit callbacks on RuntimeSerializableAttribute object.

Public Member Functions

- void OnAfterRuntimeSerialize ()

  Event triggered after RuntimeSerializableAttribute object is serialized.
- void OnAfterRuntimeDeserialize ()

  Event triggered after RuntimeSerializableAttribute object is deserialized.

5.2.1 Detailed Description

Implement this interface to receive implicit callbacks on RuntimeSerializableAttribute object.

5.2.2 Member Function Documentation

5.2.2.1 void OnAfterRuntimeSerialize ( )

Event triggered after RuntimeSerializableAttribute object is serialized.

5.2.2.2 void OnAfterRuntimeDeserialize ( )

Event triggered after RuntimeSerializableAttribute object is deserialized.

5.3 IRuntimeSerializationEventListener

Implement this interface to observe serialization process of RuntimeSerializableAttribute object.

Public Member Functions

- void OnAfterRuntimeSerialize (string _serializationID)

  Event triggered after RuntimeSerializableAttribute object is serialized.
- void OnAfterRuntimeDeserialize (string _serializationID, object _deserializedObject)

  Event triggered after RuntimeSerializableAttribute object is deserialized.

5.3.1 Detailed Description

Implement this interface to observe serialization process of RuntimeSerializableAttribute object.

5.3.2 Member Function Documentation

5.3.2.1 void OnAfterRuntimeSerialize (string *_serializationID*)

Event triggered after RuntimeSerializableAttribute object is serialized.

Parameters

_serializationID	Identifier associated with RuntimeSerializableAttribute object.

5.3.2.2 void OnAfterRuntimeDeserialize (string *_serializationID,* object *_deserializedObject*)

Event triggered after RuntimeSerializableAttribute object is deserialized.

Parameters

_serializationID	Identifier associated with RuntimeSerializableAttribute object.
_deserialized↩ Object	Deserialized RuntimeSerializableAttribute object.

5.4 IRuntimeSerializationExtension

Implement this interface to support runtime serialization for classes which belong to external assembly.

Inherited by BoundsRSExtension, and ObjectRSExtension.

Public Member Functions

- void WriteSerializationData (object _object, RuntimeSerializationInfo _info)

  Populate RuntimeSerializationInfo with the properties required to serialize target object.
- object ReadSerializationData (object _object, RuntimeSerializationInfo _info)

  Retrieve serialized properties of target object from RuntimeSerializationInfo.

5.4.1 Detailed Description

Implement this interface to support runtime serialization for classes which belong to external assembly.

5.4.2 Member Function Documentation

5.4.2.1 void WriteSerializationData (object *_object,* RuntimeSerializationInfo *_info*)

Populate RuntimeSerializationInfo with the properties required to serialize target object.

Parameters

_object	Target object to be serialized
_info	The RuntimeSerializationInfo provides interface to add properties to be serialized.

5.4.2.2 object ReadSerializationData (object *_object,* RuntimeSerializationInfo *_info*)

Retrieve serialized properties of target object from RuntimeSerializationInfo.

Returns

The deserialized object.

Parameters

_object	Target object to be deserialized.
_info	The RuntimeSerializationInfo provides interface to retrieve serialized values.

5.5 RSManager

RSManager class is responsible for serializing and deserializing objects at runtime.

Inherits SingletonPattern< RSManager >.

Static Public Member Functions

- static string Serialize< T > (T _object, string _serializationID=null)

  Returns serialization data after serializing target object. Note: Serialization data is not saved.
- static void Serialize< T > (T _object, string _serializationID, eSaveTarget _saveTarget)

  Serialize the specified object and save serialization data to specified target where it is associated with an identifier.
- static T DeserializeData< T > (string _serializationData, T _targetObject=default(T), string _serializationID=null)

  Deserializes the serialization data and recreates the object of specified type.
- static T Deserialize< T > (string _serializationID, T _targetObject=default(T))

  Deserializes the data serialization associated with identifier and recreates the object of specified type.
- static string GetSerializationData (string _serializationID)

  Returns serialization data associated with given identifier. Ideal for supporting multi device login, wherein you can remotely save serialization data and restore it on other devices using RSManager.RestoreSerializationData.
- static bool RestoreSerializationData (string _serializationData, string _serializationID, eSaveTarget _saveTarget)

  Saves Base64String format serialization data to specified target location where it is associated with an identifier.
- static void RegisterEventListener (string _serializationID, IRuntimeSerializationEventListener _newListener)

  Register object to receive serialization events.
- static void UnRegisterEventListener (string _serializationID, IRuntimeSerializationEventListener _listener)

  Unregister object from receiving serialization events.
- static void Remove (string _serializationID)

  Clears existing serialization data associated with serialization identifier.
- static void RemoveAll ()

  Clears all existing serialization data.
- static void Save ()

  Writes all serialization data to disk. By default RSManager writes serialization data to PlayerPrefs/File on Application Pause and on Application Quit.
- static void Purge (Type _objectType)

  Clears all the cached information of specified object type.

5.5.1 Detailed Description

RSManager class is responsible for serializing and deserializing objects at runtime.

5.5.2 Member Function Documentation

5.5.2.1 static string Serialize< T > (T *_object*, string *_serializationID* = `null`) [static]

Returns serialization data after serializing target object. Note: Serialization data is not saved.

Parameters

_object	Object to serialize.
_serializationID	Identifier associated with serialization data.

Template Parameters

T	Type of the object to serialize.

5.5.2.2 static void Serialize< T >(T *_object*, string *_serializationID*, eSaveTarget *_saveTarget*) [static]

Serialize the specified object and save serialization data to specified target where it is associated with an identifier.

Parameters

_object	Object to serialize.
_serializationID	Identifier associated with serialization data of the target object.
_saveTarget	Save target for serialization data.

Template Parameters

T	Type of the object to serialize.

5.5.2.3 static T DeserializeData< T >(string *_serializationData*, T *_targetObject* = default(T), string *_serializationID* = null) [static]

Deserializes the serialization data and recreates the object of specified type.

Returns

The deserialized object of specified type.

Parameters

_serialization↩ Data	Serialization data in Base64String format.
_targetObject	Deserialized value is assigned to this instance, if value is not null.
_serializationID	Identifier associated with serialization data.

Template Parameters

T	The Type of the value to deserialize.

5.5.2.4 static T Deserialize< T >(string *_serializationID*, T *_targetObject* = default(T)) [static]

Deserializes the data serialization associated with identifier and recreates the object of specified type.

Parameters

_serializationID	Identifier associated with serialization data.
_targetObject	Deserialized value is assigned to this instance, if value is not null.

Template Parameters

T	The Type of the value to deserialize.

5.5.2.5 static string GetSerializationData (string *_serializationID*) [static]

Returns serialization data associated with given identifier. Ideal for supporting multi device login, wherein you can remotely save serialization data and restore it on other devices using RSManager.RestoreSerializationData.

Returns

Serialization data as Base64String.

Parameters

_serializationID	Identifier associated with serialization data.

5.5.2.6 static bool RestoreSerializationData (string *_serializationData,* string *_serializationID,* eSaveTarget *_saveTarget*) [static]

Saves Base64String format serialization data to specified target location where it is associated with an identifier.

Parameters

_serialization↩ Data	Serialization data in Base64String format.
_serializationID	Identifier associated with serialization data.
_saveTarget	Serialization data save target.

5.5.2.7 static void RegisterEventListener (string *_serializationID,* IRuntimeSerializationEventListener *_newListener*) [static]

Register object to receive serialization events.

Parameters

_serializationID	Identifier associated with serialization.
_newListener	Instance to be registered as listener.

5.5.2.8 static void UnRegisterEventListener (string *_serializationID,* IRuntimeSerializationEventListener *_listener*) [static]

Unregister object from receiving serialization events.

Parameters

_serializationID	Identifier associated with serialization data.
_callback	Target object to be unregistered from serialization callbacks.

5.5.2.9 static void Remove (string *_serializationID*) [static]

Clears existing serialization data associated with serialization identifier.

Parameters

_serializationID	Identifier associated with serialization data.

5.5.2.10 static void RemoveAll ( ) [static]

Clears all existing serialization data.

5.5.2.11 static void Save ( ) [static]

Writes all serialization data to disk. By default RSManager writes serialization data to PlayerPrefs/File on Application Pause and on Application Quit.

5.5.2.12 static void Purge (Type *_objectType*) [static]

Clears all the cached information of specified object type.

5.6 RuntimeSerializableAttribute

Indicates that a class can be serialized at runtime.

Inherits Attribute.

Public Member Functions

- RuntimeSerializableAttribute ()

  Initializes a new instance of the RuntimeSerializableAttribute class.
- RuntimeSerializableAttribute (bool _serializeAllPublicVariables)

  Initializes a new instance of the RuntimeSerializableAttribute class.
- RuntimeSerializableAttribute (bool _serializeAllPublicVariables, bool _serializeAllNonPublicVariables)

  Initializes a new instance of the RuntimeSerializableAttribute class.

5.6.1 Detailed Description

Indicates that a class can be serialized at runtime.

5.6.2 Constructor & Destructor Documentation

5.6.2.1 RuntimeSerializableAttribute ( )

Initializes a new instance of the RuntimeSerializableAttribute class.

5.6.2.2 RuntimeSerializableAttribute (bool *_serializeAllPublicVariables*)

Initializes a new instance of the RuntimeSerializableAttribute class.

Parameters

_serializeAll↩ PublicVariables	If set to `true` all public variables will be serialized at runtime.

5.6.2.3 RuntimeSerializableAttribute (bool *_serializeAllPublicVariables,* bool *_serializeAllNonPublicVariables*)

Initializes a new instance of the RuntimeSerializableAttribute class.

Parameters

_serializeAll↩ PublicVariables	If set to `true` all public variables will be serialized at runtime.
_serializeAll↩ NonPublic↩ Variables	If set to `true` all non public variables will be serialized at runtime.

5.7 RuntimeSerializationInfo

Stores all the data required to serialize or deserialize an object.

Public Member Functions

- void AddValue< T > (string _name, T _value)

  Adds the specified object into the RuntimeSerializationInfo for serialization, where it is associated with name.
- void AddValue (string _name, object _value, Type _valueType)

  Adds the specified object into the RuntimeSerializationInfo for serialization, where it is associated with name.
- T GetValue< T > (string _name)

  Retrieves value from RuntimeSerializationInfo.
- object GetValue (string _name, Type _type)

  Retrieves value from RuntimeSerializationInfo.

5.7.1 Detailed Description

Stores all the data required to serialize or deserialize an object.

5.7.2 Member Function Documentation

5.7.2.1 void AddValue< T > (string *_name,* T *_value*)

Adds the specified object into the RuntimeSerializationInfo for serialization, where it is associated with name.

Parameters

_name	The name to associate with the value, so it can be deserialized later.
_value	The value to be serialized. Any children of this object will automatically be serialized.

Template Parameters

T	The Type associated with the current object. This must always be the type of the object itself.

5.7.2.2 void AddValue (string *_name,* object *_value,* Type *_valueType*)

Adds the specified object into the RuntimeSerializationInfo for serialization, where it is associated with name.

Parameters

_name	The name to associate with the value, so it can be deserialized later.
_value	The value to be serialized. Any children of this object will automatically be serialized.
_valueType	The Type associated with the current object. This must always be the type of the object itself.

5.7.2.3 T GetValue< T > (string *_name*)

Retrieves value from RuntimeSerializationInfo.

Returns

The object of specified type associated with name.

Parameters

_name	The name associated with the value to retrieve.

Template Parameters

T	The Type of the value to retrieve.

5.7.2.4 object GetValue (string *_name,* Type *_type*)

Retrieves value from RuntimeSerializationInfo.

Returns

The object of specified type associated with name.

Parameters

_name	The name associated with the value to retrieve.

<param name=="_type">The Type of the value to retrieve.

5.8 RuntimeSerializeFieldAttribute

Indicates that a field of a RuntimeSerializableAttribute class should be serialized at runtime.

Inherits Attribute.

5.8.1 Detailed Description

Indicates that a field of a RuntimeSerializableAttribute class should be serialized at runtime.

Index

AddValue
    VoxelBusters::RuntimeSerialization::Runtime↩
        SerializationInfo, 17
AddValue< T >
    VoxelBusters::RuntimeSerialization::Runtime↩
        SerializationInfo, 16

Deserialize< T >
    VoxelBusters::RuntimeSerialization::RSManager,
        13
DeserializeData< T >
    VoxelBusters::RuntimeSerialization::RSManager,
        13

eSaveTarget
    VoxelBusters::RuntimeSerialization, 8

FILE_SYSTEM
    VoxelBusters::RuntimeSerialization, 8

GetSerializationData
    VoxelBusters::RuntimeSerialization::RSManager,
        14
GetValue
    VoxelBusters::RuntimeSerialization::Runtime↩
        SerializationInfo, 17
GetValue< T >
    VoxelBusters::RuntimeSerialization::Runtime↩
        SerializationInfo, 17

IRuntimeSerializable, 9
IRuntimeSerializationCallback, 10
IRuntimeSerializationEventListener, 10
IRuntimeSerializationExtension, 11

OnAfterRuntimeDeserialize
    VoxelBusters::RuntimeSerialization::IRuntime↩
        SerializationCallback, 10
    VoxelBusters::RuntimeSerialization::IRuntime↩
        SerializationEventListener, 11
OnAfterRuntimeSerialize
    VoxelBusters::RuntimeSerialization::IRuntime↩
        SerializationCallback, 10
    VoxelBusters::RuntimeSerialization::IRuntime↩
        SerializationEventListener, 10

PLAYER_PREFS
    VoxelBusters::RuntimeSerialization, 8
Purge
    VoxelBusters::RuntimeSerialization::RSManager,
        15

RSManager, 12
ReadSerializationData
    VoxelBusters::RuntimeSerialization::IRuntime↩
        Serializable, 9
    VoxelBusters::RuntimeSerialization::IRuntime↩
        SerializationExtension, 11
RegisterEventListener
    VoxelBusters::RuntimeSerialization::RSManager,
        14
Remove
    VoxelBusters::RuntimeSerialization::RSManager,
        14
RemoveAll
    VoxelBusters::RuntimeSerialization::RSManager,
        15
RestoreSerializationData
    VoxelBusters::RuntimeSerialization::RSManager,
        14
RuntimeSerializableAttribute, 15
    VoxelBusters::RuntimeSerialization::Runtime↩
        SerializableAttribute, 15, 16
RuntimeSerializationInfo, 16
RuntimeSerializeFieldAttribute, 17

Save
    VoxelBusters::RuntimeSerialization::RSManager,
        15
Serialize< T >
    VoxelBusters::RuntimeSerialization::RSManager,
        12, 13

UnRegisterEventListener
    VoxelBusters::RuntimeSerialization::RSManager,
        14
UnityEngine, 7

VoxelBusters, 7
VoxelBusters.RuntimeSerialization, 7
VoxelBusters.RuntimeSerialization.Benchmark, 8
VoxelBusters::RuntimeSerialization
    eSaveTarget, 8
    FILE_SYSTEM, 8
    PLAYER_PREFS, 8
VoxelBusters::RuntimeSerialization::IRuntimeSerializable
    ReadSerializationData, 9
    WriteSerializationData, 9
VoxelBusters::RuntimeSerialization::IRuntimeSerialization↩
        Callback
    OnAfterRuntimeDeserialize, 10
    OnAfterRuntimeSerialize, 10

VoxelBusters::RuntimeSerialization::IRuntimeSerialization↩
EventListener
OnAfterRuntimeDeserialize, 11
OnAfterRuntimeSerialize, 10
VoxelBusters::RuntimeSerialization::IRuntimeSerialization↩
Extension
ReadSerializationData, 11
WriteSerializationData, 11
VoxelBusters::RuntimeSerialization::RSManager
Deserialize< T >, 13
DeserializeData< T >, 13
GetSerializationData, 14
Purge, 15
RegisterEventListener, 14
Remove, 14
RemoveAll, 15
RestoreSerializationData, 14
Save, 15
Serialize< T >, 12, 13
UnRegisterEventListener, 14
VoxelBusters::RuntimeSerialization::RuntimeSerializable↩
Attribute
RuntimeSerializableAttribute, 15, 16
VoxelBusters::RuntimeSerialization::RuntimeSerialization↩
Info
AddValue, 17
AddValue< T >, 16
GetValue, 17
GetValue< T >, 17

WriteSerializationData
VoxelBusters::RuntimeSerialization::IRuntime↩
Serializable, 9
VoxelBusters::RuntimeSerialization::IRuntime↩
SerializationExtension, 11