Namespace HotUpdate

Classes

UIButtons

Class UIButtons

Namespace: HotUpdate

Assembly: HotUpdate.dll

```
public class UIButtons : MonoBehaviour
```

Inheritance

object ← Object ← Component ← Behaviour ← MonoBehaviour ← UIButtons

Inherited Members

MonoBehaviour.IsInvoking() , MonoBehaviour.CancelInvoke() ,
MonoBehaviour.Invoke(string, float) , MonoBehaviour.InvokeRepeating(string, float, float) ,
MonoBehaviour.CancelInvoke(string) , MonoBehaviour.IsInvoking(string) ,
MonoBehaviour.StartCoroutine(string) , MonoBehaviour.StartCoroutine(string, object) ,
MonoBehaviour.StartCoroutine(IEnumerator) ,
MonoBehaviour.StartCoroutine_Auto(IEnumerator) ,
MonoBehaviour.StopCoroutine(IEnumerator) , MonoBehaviour.StopCoroutine(Coroutine) ,
MonoBehaviour.StopCoroutine(string) , MonoBehaviour.StopAllCoroutines() ,
MonoBehaviour.print(object) , MonoBehaviour.destroyCancellationToken ,
MonoBehaviour.useGUILayout , MonoBehaviour.runInEditMode , Behaviour.enabled ,
Behaviour.isActiveAndEnabled , Component.GetComponent(Type) ,
Component.GetComponent<T>() , Component.TryGetComponent(Type, out Component) ,
Component.TryGetComponent<T>(out T) , Component.GetComponent(string) ,
Component.GetComponentInChildren(Type, bool) ,
Component.GetComponentInChildren(Type) ,
Component.GetComponentInChildren<T>(bool) ,
Component.GetComponentInChildren<T>() ,
Component.GetComponentsInChildren(Type, bool) ,
Component.GetComponentsInChildren(Type) ,
Component.GetComponentsInChildren<T>(bool) ,
Component.GetComponentsInChildren<T>(bool, List<T>) ,
Component.GetComponentsInChildren<T>() ,
Component.GetComponentsInChildren<T>(List<T>) ,
Component.GetComponentInParent(Type, bool) , Component.GetComponentInParent(Type) ,
Component.GetComponentInParent<T>(bool) , Component.GetComponentInParent<T>() ,
Component.GetComponentsInParent(Type, bool) ,
Component.GetComponentsInParent(Type) ,
Component.GetComponentsInParent<T>(bool) ,

Component.GetComponentsInParent<T>(bool, List<T>) ,
Component.GetComponentsInParent<T>() , Component.GetComponents(Type) ,
Component.GetComponents(Type, List<Component>) ,
Component.GetComponents<T>(List<T>) , Component.GetComponents<T>() ,
Component.GetComponentIndex() , Component.CompareTag(string) ,
Component.SendMessageUpwards(string, object, SendMessageOptions) ,
Component.SendMessageUpwards(string, object) ,
Component.SendMessageUpwards(string) ,
Component.SendMessageUpwards(string, SendMessageOptions) ,
Component.SendMessage(string, object) , Component.SendMessage(string) ,
Component.SendMessage(string, object, SendMessageOptions) ,
Component.SendMessage(string, SendMessageOptions) ,
Component.BroadcastMessage(string, object, SendMessageOptions) ,
Component.BroadcastMessage(string, object) , Component.BroadcastMessage(string) ,
Component.BroadcastMessage(string, SendMessageOptions) , Component.transform ,
Component.gameObject , Component.tag , Object.GetInstanceID() , Object.GetHashCode() ,
Object.Equals(object) , Object.InstantiateAsync<T>(T) ,
Object.InstantiateAsync<T>(T, Transform) , Object.InstantiateAsync<T>(T, Vector3, Quaternion) ,
Object.InstantiateAsync<T>(T, Transform, Vector3, Quaternion) ,
Object.InstantiateAsync<T>(T, int) , Object.InstantiateAsync<T>(T, int, Transform) ,
Object.InstantiateAsync<T>(T, int, Vector3, Quaternion) ,
Object.InstantiateAsync<T>(T, int, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>) ,
Object.InstantiateAsync<T>(T, int, Transform, Vector3, Quaternion) ,
Object.InstantiateAsync<T>(T, int, Transform, ReadOnlySpan<Vector3>,
ReadOnlySpan<Quaternion>) ,
Object.Instantiate(Object, Vector3, Quaternion) ,
Object.Instantiate(Object, Vector3, Quaternion, Transform) , Object.Instantiate(Object) ,
Object.Instantiate(Object, Scene) , Object.Instantiate(Object, Transform) ,
Object.Instantiate(Object, Transform, bool) , Object.Instantiate<T>(T) ,
Object.Instantiate<T>(T, Vector3, Quaternion) ,
Object.Instantiate<T>(T, Vector3, Quaternion, Transform) , Object.Instantiate<T>(T, Transform) ,
Object.Instantiate<T>(T, Transform, bool) , Object.Destroy(Object, float) ,
Object.Destroy(Object) , Object.DestroyImmediate(Object, bool) ,
Object.DestroyImmediate(Object) , Object.FindObjectsOfType(Type) ,
Object.FindObjectsOfType(Type, bool) ,
Object.FindObjectsByType(Type, FindObjectsSortMode) ,
Object.FindObjectsByType(Type, FindObjectsInactive, FindObjectsSortMode) ,
Object.DontDestroyOnLoad(Object) , Object.DestroyObject(Object, float) ,
Object.DestroyObject(Object) , Object.FindSceneObjectsOfType(Type) ,
Object.FindObjectsOfTypeIncludingAssets(Type) , Object.FindObjectsOfType<T>() ,

Object.FindObjectsByType<T>(FindObjectsSortMode) , Object.FindObjectsOfType<T>(bool) , Object.FindObjectsByType<T>(FindObjectsInactive, FindObjectsSortMode) , Object.FindObjectOfType<T>() , Object.FindObjectOfType<T>(bool) , Object.FindFirstObjectByType<T>() , Object.FindAnyObjectByType<T>() , Object.FindFirstObjectByType<T>(FindObjectsInactive) , Object.FindAnyObjectByType<T>(FindObjectsInactive) , Object.FindObjectsOfTypeAll(Type) , Object.FindObjectOfType(Type) , Object.FindFirstObjectByType(Type) , Object.FindAnyObjectByType(Type) , Object.FindObjectOfType(Type, bool) , Object.FindFirstObjectByType(Type, FindObjectsInactive) , Object.FindAnyObjectByType(Type, FindObjectsInactive) , Object.ToString() , Object.name , Object.hideFlags , object.Equals(object, object) , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object)

Fields

playerIdInput

```
public InputField playerIdInput
```

Field Value

InputField

tokenInput

```
public InputField tokenInput
```

Field Value

InputField

worldIdInput

```
public InputField worldIdInput
```

Field Value

InputField

Methods

AddEntityBtnClick()

```
public void AddEntityBtnClick()
```

CreateWorldBtnClick()

```
public void CreateWorldBtnClick()
```

JoinGameBtnClick()

```
public void JoinGameBtnClick()
```

JoinWorldBtnClick()

```
public void JoinWorldBtnClick()
```

OnPlayerIdInputChanged()

```
public void OnPlayerIdInputChanged()
```

Namespace SyncerNet.Hotfix

Classes

Entity

Game

NetworkClient
网络客户端

NetworkLoop

NetworkMessage
这个类的数据结构和MemoryPack特性标签需要和服务端一致(函数除外)

Syncer
这个类的数据结构和MemoryPack特性标签需要和服务端一致(函数除外)

World

Structs

DataPack

Struct DataPack

Namespace: SyncerNet.Hotfix

Assembly: SyncerNet.Hotfix.dll

```
public struct DataPack
```

Inherited Members
ValueType.Equals(object) , ValueType.GetHashCode() , ValueType.ToString() , object.Equals(object, object) , object.GetType() , object.ReferenceEquals(object, object)

Constructors

DataPack(int, ArraySegment<byte>, KcpChannel)

```
public DataPack(int netId, ArraySegment<byte> data, KcpChannel channel)
```

Parameters

`netId` int

`data` ArraySegment<byte>

`channel` KcpChannel

Fields

Channel

```
public KcpChannel Channel
```

Field Value

KcpChannel

Data

```
public ArraySegment<byte> Data
```

Field Value

ArraySegment<byte>

NetId

```
public int NetId
```

Field Value

int

Class Entity

Namespace: SyncerNet.Hotfix

Assembly: SyncerNet.Hotfix.dll

```
[MemoryPackable(GenerateType.Object)]
public class Entity : IMemoryPackable<Entity>, IMemoryPackFormatterRegister
```

Inheritance

object ← Entity

Implements

IMemoryPackable<Entity>, IMemoryPackFormatterRegister

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

MemoryPack GenerateType: Object

```
uint EntityIduint WorldIduint OwnerIdstring
PrefabPathSystem.Collections.Concurrent.ConcurrentDictionary<System.Type,
SyncerNet.Hotfix.Syncer> Syncers
```

Constructors

Entity(uint, uint, string, World)

```
public Entity(uint entityId, uint ownerId, string prefabPath, World world)
```

Parameters

`entityId` uint

`ownerId` uint

`prefabPath` string

`world` World

Fields

Initialized

```
[MemoryPackIgnore]
public bool Initialized
```

Field Value

bool

World

```
[MemoryPackIgnore]
public World World
```

Field Value

World

Properties

EntityId

```
public uint EntityId { get; set; }
```

Property Value

uint

GameObject

```
[MemoryPackIgnore]
public GameObject? GameObject { get; set; }
```

Property Value

GameObject

IsLocal

```
[MemoryPackIgnore]
public bool IsLocal { get; }
```

Property Value

bool

OwnerId

```
public uint OwnerId { get; set; }
```

Property Value

uint

PrefabPath

```
public string PrefabPath { get; set; }
```

Property Value

string

Syncers

```
public ConcurrentDictionary<Type, Syncer> Syncers { get; }
```

Property Value

ConcurrentDictionary<Type, Syncer>

WorldId

```
public uint WorldId { get; set; }
```

Property Value

uint

Methods

AddOrSetSyncer<T>(T)

```
public void AddOrSetSyncer<T>(T tSyncer) where T : Syncer
```

Parameters

`tSyncer` T

Type Parameters

`T`

AddSyncer<T>()

```
public void AddSyncer<T>() where T : Syncer, new()
```

Type Parameters

T

Deserialize(ref MemoryPackReader, ref Entity?)

```
[Preserve]
public static void Deserialize(ref MemoryPackReader reader, ref Entity? value)
```

Parameters

reader MemoryPackReader

value Entity

Dispose()

```
public void Dispose()
```

GetSyncer<T>()

```
public T? GetSyncer<T>() where T : Syncer
```

Returns

T

Type Parameters

T

Initialize()

```
public void Initialize()
```

NetworkEarlyUpdate()

```
public void NetworkEarlyUpdate()
```

NetworkLateUpdate()

```
public void NetworkLateUpdate()
```

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

RemoveSyncer<T>()

```
public void RemoveSyncer<T>() where T : Syncer
```

Type Parameters

`T`

Reset()

```
public void Reset()
```

Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter>, ref Entity?)

```
[Preserve]
public static void Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter> writer, ref
```

```
  Entity? value) where TBufferWriter : class, IBufferWriter<byte>
```

Parameters

`writer` MemoryPackWriter<TBufferWriter>

`value` Entity

Type Parameters

`TBufferWriter`

Class Game

Namespace: SyncerNet.Hotfix

Assembly: SyncerNet.Hotfix.dll

```
public class Game
```

Inheritance

object ← Game

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Properties

Client

```
public NetworkClient Client { get; }
```

Property Value

NetworkClient

CurrentWorld

```
public World? CurrentWorld { get; }
```

Property Value

World

Instance

```
public static Game Instance { get; }
```

Property Value

Game

IsReady

```
public bool IsReady { get; }
```

Property Value

bool

StartCoroutineFunc

```
public Func<IEnumerator, Coroutine> StartCoroutineFunc { get; }
```

Property Value

Func<IEnumerator, Coroutine>

Timeout

```
public int Timeout { get; set; }
```

Property Value

int

Worlds

```
public Dictionary<uint, World> Worlds { get; set; }
```

Property Value

Dictionary<uint, World>

Methods

CreateWorld()

创建World

```
public Task<(bool, uint)> CreateWorld()
```

Returns

Task<(bool, uint)>

(isSuccess,WorldId)

GetWorld(uint)

获取World,前提是已经加入这个World

```
public World? GetWorld(uint worldId)
```

Parameters

`worldId` uint

WorldId

Returns

World

JoinWorld(uint)

加入World

```
public Task<bool> JoinWorld(uint worldId)
```

Parameters

`worldId` uint

Returns

Task<bool>

isSuccess

LoadSceneForWorld(string, uint)

异步加载场景同时激活World

```
public SceneHandle? LoadSceneForWorld(string sceneLocation, uint worldId)
```

Parameters

`sceneLocation` string

场景路径

`worldId` uint

Returns

SceneHandle

SceneHandle可用于获取场景加载进度

NetworkEarlyUpdate()

```
public void NetworkEarlyUpdate()
```

NetworkLateUpdate()

```
public void NetworkLateUpdate()
```

StartCoroutine(IEnumerator)

可有可无的协程，基本用不上

```
public Coroutine StartCoroutine(IEnumerator routine)
```

Parameters

`routine` IEnumerator

Returns

Coroutine

TryJoinGame()

加入游戏。进行任何操作之前都应该先加入游戏。

```
public Task<bool> TryJoinGame()
```

Returns

Task<bool>

Class NetworkClient

Namespace: SyncerNet.Hotfix

Assembly: SyncerNet.Hotfix.dll

网络客户端

```
public class NetworkClient : MonoBehaviour
```

Inheritance

object ← Object ← Component ← Behaviour ← MonoBehaviour ← NetworkClient

Inherited Members

MonoBehaviour.IsInvoking() , MonoBehaviour.CancelInvoke() ,
MonoBehaviour.Invoke(string, float) , MonoBehaviour.InvokeRepeating(string, float, float) ,
MonoBehaviour.CancelInvoke(string) , MonoBehaviour.IsInvoking(string) ,
MonoBehaviour.StartCoroutine(string) , MonoBehaviour.StartCoroutine(string, object) ,
MonoBehaviour.StartCoroutine_Auto(IEnumerator) ,
MonoBehaviour.StopCoroutine(IEnumerator) , MonoBehaviour.StopCoroutine(Coroutine) ,
MonoBehaviour.StopCoroutine(string) , MonoBehaviour.StopAllCoroutines() ,
MonoBehaviour.print(object) , MonoBehaviour.destroyCancellationToken ,
MonoBehaviour.useGUILayout , MonoBehaviour.runInEditMode , Behaviour.enabled ,
Behaviour.isActiveAndEnabled , Component.GetComponent(Type) ,
Component.GetComponent<T>() , Component.TryGetComponent(Type, out Component) ,
Component.TryGetComponent<T>(out T) , Component.GetComponent(string) ,
Component.GetComponentInChildren(Type, bool) ,
Component.GetComponentInChildren(Type) ,
Component.GetComponentInChildren<T>(bool) ,
Component.GetComponentInChildren<T>() ,
Component.GetComponentsInChildren(Type, bool) ,
Component.GetComponentsInChildren(Type) ,
Component.GetComponentsInChildren<T>(bool) ,
Component.GetComponentsInChildren<T>(bool, List<T>) ,
Component.GetComponentsInChildren<T>() ,
Component.GetComponentsInChildren<T>(List<T>) ,
Component.GetComponentInParent(Type, bool) , Component.GetComponentInParent(Type) ,
Component.GetComponentInParent<T>(bool) , Component.GetComponentInParent<T>() ,
Component.GetComponentsInParent(Type, bool) ,
Component.GetComponentsInParent(Type) ,

Component.GetComponentsInParent<T>(bool) ,
Component.GetComponentsInParent<T>(bool, List<T>) ,
Component.GetComponentsInParent<T>() , Component.GetComponents(Type) ,
Component.GetComponents(Type, List<Component>) ,
Component.GetComponents<T>(List<T>) , Component.GetComponents<T>() ,
Component.GetComponentIndex() , Component.CompareTag(string) ,
Component.SendMessageUpwards(string, object, SendMessageOptions) ,
Component.SendMessageUpwards(string, object) ,
Component.SendMessageUpwards(string) ,
Component.SendMessageUpwards(string, SendMessageOptions) ,
Component.SendMessage(string, object) , Component.SendMessage(string) ,
Component.SendMessage(string, object, SendMessageOptions) ,
Component.SendMessage(string, SendMessageOptions) ,
Component.BroadcastMessage(string, object, SendMessageOptions) ,
Component.BroadcastMessage(string, object) , Component.BroadcastMessage(string) ,
Component.BroadcastMessage(string, SendMessageOptions) , Component.transform ,
Component.gameObject , Component.tag , Object.GetInstanceID() , Object.GetHashCode() ,
Object.Equals(object) , Object.InstantiateAsync<T>(T) ,
Object.InstantiateAsync<T>(T, Transform) , Object.InstantiateAsync<T>(T, Vector3, Quaternion) ,
Object.InstantiateAsync<T>(T, Transform, Vector3, Quaternion) ,
Object.InstantiateAsync<T>(T, int) , Object.InstantiateAsync<T>(T, int, Transform) ,
Object.InstantiateAsync<T>(T, int, Vector3, Quaternion) ,
Object.InstantiateAsync<T>(T, int, ReadOnlySpan<Vector3>, ReadOnlySpan<Quaternion>) ,
Object.InstantiateAsync<T>(T, int, Transform, Vector3, Quaternion) ,
Object.InstantiateAsync<T>(T, int, Transform, ReadOnlySpan<Vector3>,
ReadOnlySpan<Quaternion>) ,
Object.Instantiate(Object, Vector3, Quaternion) ,
Object.Instantiate(Object, Vector3, Quaternion, Transform) , Object.Instantiate(Object) ,
Object.Instantiate(Object, Scene) , Object.Instantiate(Object, Transform) ,
Object.Instantiate(Object, Transform, bool) , Object.Instantiate<T>(T) ,
Object.Instantiate<T>(T, Vector3, Quaternion) ,
Object.Instantiate<T>(T, Vector3, Quaternion, Transform) , Object.Instantiate<T>(T, Transform) ,
Object.Instantiate<T>(T, Transform, bool) , Object.Destroy(Object, float) ,
Object.Destroy(Object) , Object.DestroyImmediate(Object, bool) ,
Object.DestroyImmediate(Object) , Object.FindObjectsOfType(Type) ,
Object.FindObjectsOfType(Type, bool) ,
Object.FindObjectsByType(Type, FindObjectsSortMode) ,
Object.FindObjectsByType(Type, FindObjectsInactive, FindObjectsSortMode) ,
Object.DontDestroyOnLoad(Object) , Object.DestroyObject(Object, float) ,
Object.DestroyObject(Object) , Object.FindSceneObjectsOfType(Type) ,

Object.FindObjectsOfTypeIncludingAssets(Type) , Object.FindObjectsOfType<T>() , Object.FindObjectsByType<T>(FindObjectsSortMode) , Object.FindObjectsOfType<T>(bool) , Object.FindObjectsByType<T>(FindObjectsInactive, FindObjectsSortMode) , Object.FindObjectOfType<T>() , Object.FindObjectOfType<T>(bool) , Object.FindFirstObjectByType<T>() , Object.FindAnyObjectByType<T>() , Object.FindFirstObjectByType<T>(FindObjectsInactive) , Object.FindAnyObjectByType<T>(FindObjectsInactive) , Object.FindObjectsOfTypeAll(Type) , Object.FindObjectOfType(Type) , Object.FindFirstObjectByType(Type) , Object.FindAnyObjectByType(Type) , Object.FindObjectOfType(Type, bool) , Object.FindFirstObjectByType(Type, FindObjectsInactive) , Object.FindAnyObjectByType(Type, FindObjectsInactive) , Object.ToString() , Object.name , Object.hideFlags , object.Equals(object, object) , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object)

Fields

FastResend

```
[Tooltip("KCP fastresend parameter. Faster resend for the cost of higher bandwidth. 0 in normal mode, 2 in turbo mode.")]
public int FastResend
```

Field Value

int

Instance

```
public static NetworkClient? Instance
```

Field Value

NetworkClient

Interval

```
[Tooltip("KCP internal update interval. 100ms is KCP default, but a lower interval is
recommended to minimize latency and to scale to more networked entities.")]
public uint Interval
```

Field Value

uint

MaxRetransmits

```
[Tooltip("KCP will try to retransmit lost messages up to MaxRetransmit (aka dead_link)
before disconnecting.")]
public uint MaxRetransmits
```

Field Value

uint

NoDelay

```
[Tooltip("NoDelay is recommended to reduce latency. This also scales better without buffers
getting full.")]
public bool NoDelay
```

Field Value

bool

ReceiveWindowSize

```
[Tooltip("KCP window size can be modified to support higher loads. This also increases max
message size.")]
public uint ReceiveWindowSize
```

Field Value

uint

RecvBufferSize

```
[Tooltip("Socket receive buffer size. Large buffer helps support more connections. Increase operating system socket buffer size limits if needed.")]
public int RecvBufferSize
```

Field Value

int

SendBufferSize

```
[Tooltip("Socket send buffer size. Large buffer helps support more connections. Increase operating system socket buffer size limits if needed.")]
public int SendBufferSize
```

Field Value

int

SendWindowSize

```
[Tooltip("KCP window size can be modified to support higher loads.")]
public uint SendWindowSize
```

Field Value

uint

Timeout

```
[Tooltip("KCP timeout in milliseconds. Note that KCP sends a ping automatically.")]
```

```
public int Timeout
```

Field Value

int

address

```
public string address
```

Field Value

string

port

```
public ushort port
```

Field Value

ushort

Properties

Client

```
public KcpClient? Client { get; }
```

Property Value

KcpClient

Methods

Disconnect()

```
public void Disconnect()
```

NetworkEarlyUpdate()

在Unity的Update之前调用

```
public void NetworkEarlyUpdate()
```

NetworkLateUpdate()

在Unity的LateUpdate之后调用

```
public void NetworkLateUpdate()
```

Reconnect()

```
public void Reconnect()
```

Send(NetworkMessage, bool, int)

发送Message消息

```
public Task<NetworkMessage?> Send(NetworkMessage message, bool needToResponse = false, int
timeout = 10000)
```

Parameters

`message` NetworkMessage

消息

`needToResponse` bool

是否需要响应

`timeout` int

响应超时时间

Returns

Task<NetworkMessage>

如果不需要响应，则返回null。如果需要响应，则在收到响应时返回响应消息，否则返回null。

StartCoroutine(IEnumerator)

```
public Coroutine StartCoroutine(IEnumerator routine)
```

Parameters

`routine` IEnumerator

Returns

Coroutine

Class NetworkLoop

Namespace: SyncerNet.Hotfix

Assembly: SyncerNet.Hotfix.dll

```
public static class NetworkLoop
```

Inheritance

object ← NetworkLoop

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Class NetworkMessage

Namespace: SyncerNet.Hotfix

Assembly: SyncerNet.Hotfix.dll

这个类的数据结构和MemoryPack特性标签需要和服务端一致(函数除外)

```
[MemoryPackable(GenerateType.Object)]
[MemoryPackUnion(0, typeof(AddEntityMessage))]
[MemoryPackUnion(1, typeof(AddEntityReqMessage))]
[MemoryPackUnion(2, typeof(AddEntityRespMessage))]
[MemoryPackUnion(3, typeof(CreateWorldReqMessage))]
[MemoryPackUnion(4, typeof(CreateWorldRespMessage))]
[MemoryPackUnion(5, typeof(JoinGameReqMessage))]
[MemoryPackUnion(6, typeof(JoinGameRespMessage))]
[MemoryPackUnion(7, typeof(JoinWorldReqMessage))]
[MemoryPackUnion(8, typeof(JoinWorldRespMessage))]
[MemoryPackUnion(9, typeof(RemoveEntityReqMessage))]
[MemoryPackUnion(10, typeof(RemoveEntityRespMessage))]
[MemoryPackUnion(11, typeof(SyncerMessage))]
public abstract class NetworkMessage : IMemoryPackFormatterRegister
```

Inheritance

object ← NetworkMessage

Implements

IMemoryPackFormatterRegister

Derived

AddEntityMessage, AddEntityReqMessage, AddEntityRespMessage, CreateWorldReqMessage, CreateWorldRespMessage, JoinGameReqMessage, JoinGameRespMessage, JoinWorldReqMessage, JoinWorldRespMessage, RemoveEntityReqMessage, RemoveEntityRespMessage, SyncerMessage

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Fields

Id

```
public uint Id
```

Field Value

uint

IsResponse

```
public bool IsResponse
```

Field Value

bool

PlayerId

```
public uint PlayerId
```

Field Value

uint

Token

```
public string Token
```

Field Value

string

Methods

Process(Game, KcpChannel)

```
public abstract void Process(Game game, KcpChannel channel)
```

Parameters

`game` Game

`channel` KcpChannel

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

Class Syncer

Namespace: SyncerNet.Hotfix

Assembly: SyncerNet.Hotfix.dll

这个类的数据结构和MemoryPack特性标签需要和服务端一致(函数除外)

```
[MemoryPackable(GenerateType.Object)]
[MemoryPackUnion(0, typeof(AnimatorSyncer))]
[MemoryPackUnion(1, typeof(TransformSyncer))]
public abstract class Syncer : IMemoryPackFormatterRegister
```

Inheritance

object ← Syncer

Implements

IMemoryPackFormatterRegister

Derived

AnimatorSyncer, TransformSyncer

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Fields

IsChanged

```
[MemoryPackIgnore]
public bool IsChanged
```

Field Value

bool

Methods

NetworkEarlyUpdate(Entity)

```
public virtual void NetworkEarlyUpdate(Entity entity)
```

Parameters

`entity` Entity

NetworkLateUpdate(Entity)

```
public virtual void NetworkLateUpdate(Entity entity)
```

Parameters

`entity` Entity

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

UpdateSyncer(Entity)

```
public abstract void UpdateSyncer(Entity entity)
```

Parameters

`entity` Entity

Class World

Namespace: SyncerNet.Hotfix

Assembly: SyncerNet.Hotfix.dll

```
[MemoryPackable(GenerateType.Object)]
public class World : IMemoryPackable<World>, IMemoryPackFormatterRegister
```

Inheritance

object ← World

Implements

IMemoryPackable<World>, IMemoryPackFormatterRegister

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

MemoryPack GenerateType: Object

```
uint WorldIdSystem.Collections.Concurrent.ConcurrentDictionary<uint,
SyncerNet.Hotfix.Entity> Entities
```

Properties

Entities

```
public ConcurrentDictionary<uint, Entity> Entities { get; set; }
```

Property Value

ConcurrentDictionary <uint, Entity>

IsActive

```
[MemoryPackIgnore]
public bool IsActive { get; }
```

Property Value

bool

WorldId

```
public uint WorldId { get; set; }
```

Property Value

uint

Methods

Deserialize(ref MemoryPackReader, ref World?)

```
[Preserve]
public static void Deserialize(ref MemoryPackReader reader, ref World? value)
```

Parameters

`reader` MemoryPackReader

`value` World

GetEntity(uint)

获取Entity

```
public Entity? GetEntity(uint id)
```

Parameters

`id` uint

Returns

Entity

NetworkEarlyUpdate()

```
public void NetworkEarlyUpdate()
```

NetworkLateUpdate()

```
public void NetworkLateUpdate()
```

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

Reset()

重置World，移除所有已创建的Unity游戏对象（物体），不会移除Entity

```
public void Reset()
```

Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter>, ref World?)

```
[Preserve]
public static void Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter> writer, ref
```

```
World? value) where TBufferWriter : class, IBufferWriter<byte>
```

Parameters

`writer` MemoryPackWriter<TBufferWriter>

`value` World

Type Parameters

`TBufferWriter`

SetActive(bool)

设置World的启用状态。只有启用的World会进行网络同步。

```
public void SetActive(bool active)
```

Parameters

`active` bool

TryAddEntity(string)

添加Entity

```
public Task<(bool, uint)> TryAddEntity(string prefabPath = "")
```

Parameters

`prefabPath` string

Prefab路径，留空则不创建游戏对象

Returns

Task<(bool, uint)>

(isSuccess,EntityId)

TryRemoveEntity(uint)

移除Entity

```
public Task<bool> TryRemoveEntity(uint entityId)
```

Parameters

`entityId` uint

Returns

Task <bool>

isSuccess

Namespace SyncerNet.Hotfix.Messages

Classes

AddEntityMessage

AddEntityReqMessage

AddEntityRespMessage

CreateWorldReqMessage

CreateWorldRespMessage

JoinGameReqMessage

JoinGameRespMessage

JoinWorldReqMessage

JoinWorldRespMessage

RemoveEntityReqMessage

RemoveEntityRespMessage

SyncerMessage

Class AddEntityMessage

Namespace: SyncerNet.Hotfix.Messages

Assembly: SyncerNet.Hotfix.dll

```
[MemoryPackable(GenerateType.Object)]
public class AddEntityMessage : NetworkMessage, IMemoryPackable<AddEntityMessage>,
IMemoryPackFormatterRegister
```

Inheritance

object ← NetworkMessage ← AddEntityMessage

Implements

IMemoryPackable<AddEntityMessage>, IMemoryPackFormatterRegister

Inherited Members

NetworkMessage.Id , NetworkMessage.PlayerId , NetworkMessage.Token , NetworkMessage.IsResponse , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

MemoryPack GenerateType: Object

```
uint Iduint PlayerIdstring Tokenbool IsResponseuint WorldIduint EntityIduint
OwnerIdstring PrefabPath
```

Constructors

AddEntityMessage(uint, uint, uint, string)

```
public AddEntityMessage(uint worldId, uint entityId, uint ownerId, string prefabPath)
```

Parameters

`worldId` uint

`entityId` uint

`ownerId` uint

`prefabPath` string

Fields

EntityId

```
public uint EntityId
```

Field Value

uint

OwnerId

```
public uint OwnerId
```

Field Value

uint

PrefabPath

```
public string PrefabPath
```

Field Value

string

WorldId

```
public uint WorldId
```

Field Value

uint

Methods

Deserialize(ref MemoryPackReader, ref AddEntity Message?)

```
[Preserve]
public static void Deserialize(ref MemoryPackReader reader, ref AddEntityMessage? value)
```

Parameters

`reader` MemoryPackReader

`value` AddEntityMessage

Process(Game, KcpChannel)

```
public override void Process(Game game, KcpChannel channel)
```

Parameters

`game` Game

`channel` KcpChannel

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter>, ref AddEntityMessage?)

```
[Preserve]
public static void Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter> writer, ref
AddEntityMessage? value) where TBufferWriter : class, IBufferWriter<byte>
```

Parameters

`writer` MemoryPackWriter<TBufferWriter>

`value` AddEntityMessage

Type Parameters

`TBufferWriter`

Class AddEntityReqMessage

Namespace: SyncerNet.Hotfix.Messages

Assembly: SyncerNet.Hotfix.dll

```
[MemoryPackable(GenerateType.Object)]
public class AddEntityReqMessage : NetworkMessage, IMemoryPackable<AddEntityReqMessage>,
IMemoryPackFormatterRegister
```

Inheritance

object ← NetworkMessage ← AddEntityReqMessage

Implements

IMemoryPackable<AddEntityReqMessage>, IMemoryPackFormatterRegister

Inherited Members

NetworkMessage.Id , NetworkMessage.PlayerId , NetworkMessage.Token , NetworkMessage.IsResponse , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

MemoryPack GenerateType: Object

```
uint Iduint PlayerIdstring Tokenbool IsResponseuint WorldIduint OwnerIdstring PrefabPath
```

Constructors

AddEntityReqMessage(uint, uint, string)

```
public AddEntityReqMessage(uint worldId, uint ownerId, string prefabPath)
```

Parameters

`worldId` uint

`ownerId` uint

prefabPath string

Fields

OwnerId

```
public uint OwnerId
```

Field Value

uint

PrefabPath

```
public string PrefabPath
```

Field Value

string

WorldId

```
public uint WorldId
```

Field Value

uint

Methods

Deserialize(ref MemoryPackReader, ref AddEntityReq Message?)

```
[Preserve]
public static void Deserialize(ref MemoryPackReader reader, ref AddEntityReqMessage? value)
```

Parameters

`reader` MemoryPackReader

`value` AddEntityReqMessage

Process(Game, KcpChannel)

```
public override void Process(Game game, KcpChannel channel)
```

Parameters

`game` Game

`channel` KcpChannel

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter>, ref AddEntityReqMessage?)

```
[Preserve]
public static void Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter> writer, ref
AddEntityReqMessage? value) where TBufferWriter : class, IBufferWriter<byte>
```

Parameters

`writer` MemoryPackWriter<TBufferWriter>

`value` AddEntityReqMessage

Type Parameters

`TBufferWriter`

Class AddEntityRespMessage

Namespace: SyncerNet.Hotfix.Messages

Assembly: SyncerNet.Hotfix.dll

```
[MemoryPackable(GenerateType.Object)]
public class AddEntityRespMessage : NetworkMessage, IMemoryPackable<AddEntityRespMessage>,
IMemoryPackFormatterRegister
```

Inheritance

object ← NetworkMessage ← AddEntityRespMessage

Implements

IMemoryPackable<AddEntityRespMessage>, IMemoryPackFormatterRegister

Inherited Members

NetworkMessage.Id , NetworkMessage.PlayerId , NetworkMessage.Token , NetworkMessage.IsResponse , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

MemoryPack GenerateType: Object

```
uint Iduint PlayerIdstring Tokenbool IsResponsebool Successuint EntityId
```

Constructors

AddEntityRespMessage(bool, uint)

```
public AddEntityRespMessage(bool success, uint entityId)
```

Parameters

`success` bool

`entityId` uint

Fields

EntityId

```
public uint EntityId
```

Field Value

uint

Success

```
public bool Success
```

Field Value

bool

Methods

Deserialize(ref MemoryPackReader, ref AddEntityResp Message?)

```
[Preserve]
public static void Deserialize(ref MemoryPackReader reader, ref AddEntityRespMessage? value)
```

Parameters

`reader` MemoryPackReader

`value` AddEntityRespMessage

Process(Game, KcpChannel)

```
public override void Process(Game game, KcpChannel channel)
```

Parameters

`game` Game

`channel` KcpChannel

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter>, ref AddEntityRespMessage?)

```
[Preserve]
public static void Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter> writer, ref
AddEntityRespMessage? value) where TBufferWriter : class, IBufferWriter<byte>
```

Parameters

`writer` MemoryPackWriter<TBufferWriter>

`value` AddEntityRespMessage

Type Parameters

`TBufferWriter`

Class CreateWorldReqMessage

Namespace: SyncerNet.Hotfix.Messages

Assembly: SyncerNet.Hotfix.dll

```
[MemoryPackable(GenerateType.Object)]
public class CreateWorldReqMessage : NetworkMessage, IMemoryPackable<CreateWorldReqMessage>,
IMemoryPackFormatterRegister
```

Inheritance

object ← NetworkMessage ← CreateWorldReqMessage

Implements

IMemoryPackable<CreateWorldReqMessage>, IMemoryPackFormatterRegister

Inherited Members

NetworkMessage.Id , NetworkMessage.PlayerId , NetworkMessage.Token , NetworkMessage.IsResponse , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

MemoryPack GenerateType: Object

```
uint Iduint PlayerIdstring Tokenbool IsResponse
```

Methods

Deserialize(ref MemoryPackReader, ref CreateWorldReqMessage?)

```
[Preserve]
public static void Deserialize(ref MemoryPackReader reader, ref
CreateWorldReqMessage? value)
```

Parameters

`reader` MemoryPackReader

`value` CreateWorldReqMessage

Process(Game, KcpChannel)

```
public override void Process(Game game, KcpChannel channel)
```

Parameters

`game` Game

`channel` KcpChannel

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter>, ref CreateWorldReqMessage?)

```
[Preserve]
public static void Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter> writer, ref
CreateWorldReqMessage? value) where TBufferWriter : class, IBufferWriter<byte>
```

Parameters

`writer` MemoryPackWriter<TBufferWriter>

`value` CreateWorldReqMessage

Type Parameters

`TBufferWriter`

Class CreateWorldRespMessage

Namespace: SyncerNet.Hotfix.Messages

Assembly: SyncerNet.Hotfix.dll

```
[MemoryPackable(GenerateType.Object)]
public class CreateWorldRespMessage : NetworkMessage,
IMemoryPackable<CreateWorldRespMessage>, IMemoryPackFormatterRegister
```

Inheritance

object ← NetworkMessage ← CreateWorldRespMessage

Implements

IMemoryPackable<CreateWorldRespMessage>, IMemoryPackFormatterRegister

Inherited Members

NetworkMessage.Id , NetworkMessage.PlayerId , NetworkMessage.Token , NetworkMessage.IsResponse , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

MemoryPack GenerateType: Object

```
uint Iduint PlayerIdstring Tokenbool IsResponsebool Successuint WorldId
```

Constructors

CreateWorldRespMessage(bool, uint)

```
public CreateWorldRespMessage(bool success, uint worldId)
```

Parameters

`success` bool

`worldId` uint

Fields

Success

```
public bool Success
```

Field Value

bool

WorldId

```
public uint WorldId
```

Field Value

uint

Methods

Deserialize(ref MemoryPackReader, ref CreateWorldResp Message?)

```
[Preserve]
public static void Deserialize(ref MemoryPackReader reader, ref
CreateWorldRespMessage? value)
```

Parameters

`reader` MemoryPackReader

`value` CreateWorldRespMessage

Process(Game, KcpChannel)

```
public override void Process(Game game, KcpChannel channel)
```

Parameters

`game` Game

`channel` KcpChannel

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter>, ref CreateWorldRespMessage?)

```
[Preserve]
public static void Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter> writer, ref
CreateWorldRespMessage? value) where TBufferWriter : class, IBufferWriter<byte>
```

Parameters

`writer` MemoryPackWriter<TBufferWriter>

`value` CreateWorldRespMessage

Type Parameters

`TBufferWriter`

Class JoinGameReqMessage

Namespace: SyncerNet.Hotfix.Messages

Assembly: SyncerNet.Hotfix.dll

```
[MemoryPackable(GenerateType.Object)]
public class JoinGameReqMessage : NetworkMessage, IMemoryPackable<JoinGameReqMessage>,
IMemoryPackFormatterRegister
```

Inheritance

object ← NetworkMessage ← JoinGameReqMessage

Implements

IMemoryPackable<JoinGameReqMessage>, IMemoryPackFormatterRegister

Inherited Members

NetworkMessage.Id , NetworkMessage.PlayerId , NetworkMessage.Token , NetworkMessage.IsResponse , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

MemoryPack GenerateType: Object

```
uint Iduint PlayerIdstring Tokenbool IsResponse
```

Methods

Deserialize(ref MemoryPackReader, ref JoinGameReq Message?)

```
[Preserve]
public static void Deserialize(ref MemoryPackReader reader, ref JoinGameReqMessage? value)
```

Parameters

`reader` MemoryPackReader

`value` JoinGameReqMessage

Process(Game, KcpChannel)

```
public override void Process(Game game, KcpChannel channel)
```

Parameters

`game` Game

`channel` KcpChannel

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter>, ref JoinGameReqMessage?)

```
[Preserve]
public static void Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter> writer, ref
JoinGameReqMessage? value) where TBufferWriter : class, IBufferWriter<byte>
```

Parameters

`writer` MemoryPackWriter<TBufferWriter>

`value` JoinGameReqMessage

Type Parameters

`TBufferWriter`

Class JoinGameRespMessage

Namespace: SyncerNet.Hotfix.Messages

Assembly: SyncerNet.Hotfix.dll

```
[MemoryPackable(GenerateType.Object)]
public class JoinGameRespMessage : NetworkMessage, IMemoryPackable<JoinGameRespMessage>,
IMemoryPackFormatterRegister
```

Inheritance

object ← NetworkMessage ← JoinGameRespMessage

Implements

IMemoryPackable<JoinGameRespMessage>, IMemoryPackFormatterRegister

Inherited Members

NetworkMessage.Id , NetworkMessage.PlayerId , NetworkMessage.Token , NetworkMessage.IsResponse , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

MemoryPack GenerateType: Object

```
uint Iduint PlayerIdstring Tokenbool IsResponse
```

Constructors

JoinGameRespMessage()

```
public JoinGameRespMessage()
```

Methods

Deserialize(ref MemoryPackReader, ref JoinGameResp Message?)

```
[Preserve]
public static void Deserialize(ref MemoryPackReader reader, ref JoinGameRespMessage? value)
```

Parameters

`reader` MemoryPackReader

`value` JoinGameRespMessage

Process(Game, KcpChannel)

```
public override void Process(Game game, KcpChannel channel)
```

Parameters

`game` Game

`channel` KcpChannel

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter>, ref JoinGameRespMessage?)

```
[Preserve]
public static void Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter> writer, ref
JoinGameRespMessage? value) where TBufferWriter : class, IBufferWriter<byte>
```

Parameters

`writer` MemoryPackWriter<TBufferWriter>

`value` JoinGameRespMessage

Type Parameters

`TBufferWriter`

Class JoinWorldReqMessage

Namespace: SyncerNet.Hotfix.Messages

Assembly: SyncerNet.Hotfix.dll

```
[MemoryPackable(GenerateType.Object)]
public class JoinWorldReqMessage : NetworkMessage, IMemoryPackable<JoinWorldReqMessage>,
IMemoryPackFormatterRegister
```

Inheritance

object ← NetworkMessage ← JoinWorldReqMessage

Implements

IMemoryPackable<JoinWorldReqMessage>, IMemoryPackFormatterRegister

Inherited Members

NetworkMessage.Id , NetworkMessage.PlayerId , NetworkMessage.Token , NetworkMessage.IsResponse , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

MemoryPack GenerateType: Object

```
uint Iduint PlayerIdstring Tokenbool IsResponseuint WorldId
```

Constructors

JoinWorldReqMessage(uint)

```
public JoinWorldReqMessage(uint worldId)
```

Parameters

`worldId` uint

Fields

WorldId

```
public uint WorldId
```

Field Value

uint

Methods

Deserialize(ref MemoryPackReader, ref JoinWorldReq Message?)

```
[Preserve]
public static void Deserialize(ref MemoryPackReader reader, ref JoinWorldReqMessage? value)
```

Parameters

`reader` MemoryPackReader

`value` JoinWorldReqMessage

Process(Game, KcpChannel)

```
public override void Process(Game game, KcpChannel channel)
```

Parameters

`game` Game

`channel` KcpChannel

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter>, ref JoinWorldReqMessage?)

```
[Preserve]
public static void Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter> writer, ref
JoinWorldReqMessage? value) where TBufferWriter : class, IBufferWriter<byte>
```

Parameters

`writer` MemoryPackWriter<TBufferWriter>

`value` JoinWorldReqMessage

Type Parameters

`TBufferWriter`

Class JoinWorldRespMessage

Namespace: SyncerNet.Hotfix.Messages

Assembly: SyncerNet.Hotfix.dll

```
[MemoryPackable(GenerateType.Object)]
public class JoinWorldRespMessage : NetworkMessage, IMemoryPackable<JoinWorldRespMessage>,
IMemoryPackFormatterRegister
```

Inheritance

object ← NetworkMessage ← JoinWorldRespMessage

Implements

IMemoryPackable<JoinWorldRespMessage>, IMemoryPackFormatterRegister

Inherited Members

NetworkMessage.Id , NetworkMessage.PlayerId , NetworkMessage.Token , NetworkMessage.IsResponse , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

MemoryPack GenerateType: Object

```
uint Iduint PlayerIdstring Tokenbool IsResponsebool SuccessSyncerNet.Hotfix.World World
```

Constructors

JoinWorldRespMessage(bool, World?)

```
public JoinWorldRespMessage(bool success, World? world)
```

Parameters

`success` bool

`world` World

Fields

Success

```
public bool Success
```

Field Value

bool

World

```
public World? World
```

Field Value

World

Methods

Deserialize(ref MemoryPackReader, ref JoinWorldResp Message?)

```
[Preserve]
public static void Deserialize(ref MemoryPackReader reader, ref JoinWorldRespMessage? value)
```

Parameters

`reader` MemoryPackReader

`value` JoinWorldRespMessage

Process(Game, KcpChannel)

```
public override void Process(Game game, KcpChannel channel)
```

Parameters

`game` Game

`channel` KcpChannel

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter>, ref JoinWorldRespMessage?)

```
[Preserve]
public static void Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter> writer, ref
JoinWorldRespMessage? value) where TBufferWriter : class, IBufferWriter<byte>
```

Parameters

`writer` MemoryPackWriter<TBufferWriter>

`value` JoinWorldRespMessage

Type Parameters

`TBufferWriter`

Class RemoveEntityReqMessage

Namespace: SyncerNet.Hotfix.Messages

Assembly: SyncerNet.Hotfix.dll

```
[MemoryPackable(GenerateType.Object)]
public class RemoveEntityReqMessage : NetworkMessage,
IMemoryPackable<RemoveEntityReqMessage>, IMemoryPackFormatterRegister
```

Inheritance

object ← NetworkMessage ← RemoveEntityReqMessage

Implements

IMemoryPackable<RemoveEntityReqMessage>, IMemoryPackFormatterRegister

Inherited Members

NetworkMessage.Id , NetworkMessage.PlayerId , NetworkMessage.Token , NetworkMessage.IsResponse , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

MemoryPack GenerateType: Object

```
uint Iduint PlayerIdstring Tokenbool IsResponseuint WorldIduint EntityId
```

Constructors

RemoveEntityReqMessage(uint, uint)

```
public RemoveEntityReqMessage(uint worldId, uint entityId)
```

Parameters

`worldId` uint

`entityId` uint

Fields

EntityId

```
public uint EntityId
```

Field Value

uint

WorldId

```
public uint WorldId
```

Field Value

uint

Methods

Deserialize(ref MemoryPackReader, ref RemoveEntityReqMessage?)

```
[Preserve]
public static void Deserialize(ref MemoryPackReader reader, ref
RemoveEntityReqMessage? value)
```

Parameters

`reader` MemoryPackReader

`value` RemoveEntityReqMessage

Process(Game, KcpChannel)

```
public override void Process(Game game, KcpChannel channel)
```

Parameters

`game` Game

`channel` KcpChannel

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter>, ref RemoveEntityReqMessage?)

```
[Preserve]
public static void Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter> writer, ref
RemoveEntityReqMessage? value) where TBufferWriter : class, IBufferWriter<byte>
```

Parameters

`writer` MemoryPackWriter<TBufferWriter>

`value` RemoveEntityReqMessage

Type Parameters

`TBufferWriter`

Class RemoveEntityRespMessage

Namespace: SyncerNet.Hotfix.Messages

Assembly: SyncerNet.Hotfix.dll

```
[MemoryPackable(GenerateType.Object)]
public class RemoveEntityRespMessage : NetworkMessage,
IMemoryPackable<RemoveEntityRespMessage>, IMemoryPackFormatterRegister
```

Inheritance

object ← NetworkMessage ← RemoveEntityRespMessage

Implements

IMemoryPackable<RemoveEntityRespMessage>, IMemoryPackFormatterRegister

Inherited Members

NetworkMessage.Id , NetworkMessage.PlayerId , NetworkMessage.Token , NetworkMessage.IsResponse , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

MemoryPack GenerateType: Object

```
uint Iduint PlayerIdstring Tokenbool IsResponsebool Success
```

Constructors

RemoveEntityRespMessage(bool)

```
public RemoveEntityRespMessage(bool success)
```

Parameters

`success` bool

Fields

Success

```
public bool Success
```

Field Value

bool

Methods

Deserialize(ref MemoryPackReader, ref RemoveEntityRespMessage?)

```
[Preserve]
public static void Deserialize(ref MemoryPackReader reader, ref RemoveEntityRespMessage?
value)
```

Parameters

`reader` MemoryPackReader

`value` RemoveEntityRespMessage

Process(Game, KcpChannel)

```
public override void Process(Game game, KcpChannel channel)
```

Parameters

`game` Game

`channel` KcpChannel

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter>, ref RemoveEntityRespMessage?)

```
[Preserve]
public static void Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter> writer, ref
RemoveEntityRespMessage? value) where TBufferWriter : class, IBufferWriter<byte>
```

Parameters

`writer` MemoryPackWriter<TBufferWriter>

`value` RemoveEntityRespMessage

Type Parameters

`TBufferWriter`

Class SyncerMessage

Namespace: SyncerNet.Hotfix.Messages

Assembly: SyncerNet.Hotfix.dll

```
[MemoryPackable(GenerateType.Object)]
public class SyncerMessage : NetworkMessage, IMemoryPackable<SyncerMessage>,
IMemoryPackFormatterRegister
```

Inheritance

object ← NetworkMessage ← SyncerMessage

Implements

IMemoryPackable<SyncerMessage>, IMemoryPackFormatterRegister

Inherited Members

NetworkMessage.Id , NetworkMessage.PlayerId , NetworkMessage.Token , NetworkMessage.IsResponse , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

MemoryPack GenerateType: Object

```
uint Iduint PlayerIdstring Tokenbool IsResponseuint WorldIduint
EntityIdSyncerNet.Hotfix.Syncer Syncer
```

Constructors

SyncerMessage(uint, uint, Syncer)

```
public SyncerMessage(uint worldId, uint entityId, Syncer syncer)
```

Parameters

`worldId` uint

`entityId` uint

`syncer` Syncer

Fields

EntityId

```
public uint EntityId
```

Field Value

uint

Syncer

```
public Syncer Syncer
```

Field Value

Syncer

WorldId

```
public uint WorldId
```

Field Value

uint

Methods

Deserialize(ref MemoryPackReader, ref SyncerMessage?)

```
[Preserve]
public static void Deserialize(ref MemoryPackReader reader, ref SyncerMessage? value)
```

Parameters

`reader` MemoryPackReader

`value` SyncerMessage

Process(Game, KcpChannel)

```
public override void Process(Game game, KcpChannel channel)
```

Parameters

`game` Game

`channel` KcpChannel

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter>, ref SyncerMessage?)

```
[Preserve]
public static void Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter> writer, ref
SyncerMessage? value) where TBufferWriter : class, IBufferWriter<byte>
```

Parameters

`writer` MemoryPackWriter<TBufferWriter>

`value` SyncerMessage

Type Parameters

`TBufferWriter`

Namespace SyncerNet.Hotfix.Syncers

Classes

AnimatorSyncer

TransformSyncer

Class AnimatorSyncer

Namespace: SyncerNet.Hotfix.Syncers

Assembly: SyncerNet.Hotfix.dll

```
[MemoryPackable(GenerateType.Object)]
public class AnimatorSyncer : Syncer, IMemoryPackable<AnimatorSyncer>,
IMemoryPackFormatterRegister
```

Inheritance

object ← Syncer ← AnimatorSyncer

Implements

IMemoryPackable<AnimatorSyncer>, IMemoryPackFormatterRegister

Inherited Members

Syncer.IsChanged , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

MemoryPack GenerateType: Object

```
float _animatorSpeedint[] _intParametersfloat[] _floatParametersbool[] _boolParametersint[]
_stateHashfloat[] _normalizedTimefloat[] _layerWeight
```

Methods

Deserialize(ref MemoryPackReader, ref AnimatorSyncer?)

```
[Preserve]
public static void Deserialize(ref MemoryPackReader reader, ref AnimatorSyncer? value)
```

Parameters

`reader` MemoryPackReader

`value` AnimatorSyncer

NetworkEarlyUpdate(Entity)

```
public override void NetworkEarlyUpdate(Entity entity)
```

Parameters

`entity` Entity

NetworkLateUpdate(Entity)

```
public override void NetworkLateUpdate(Entity entity)
```

Parameters

`entity` Entity

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter>, ref AnimatorSyncer?)

```
[Preserve]
public static void Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter> writer, ref
AnimatorSyncer? value) where TBufferWriter : class, IBufferWriter<byte>
```

Parameters

`writer` MemoryPackWriter<TBufferWriter>

`value` AnimatorSyncer

Type Parameters

`TBufferWriter`

UpdateSyncer(Entity)

```
public override void UpdateSyncer(Entity entity)
```

Parameters

`entity` Entity

Class TransformSyncer

Namespace: SyncerNet.Hotfix.Syncers

Assembly: SyncerNet.Hotfix.dll

```
[MemoryPackable(GenerateType.Object)]
public class TransformSyncer : Syncer, IMemoryPackable<TransformSyncer>,
IMemoryPackFormatterRegister
```

Inheritance

object ← Syncer ← TransformSyncer

Implements

IMemoryPackable<TransformSyncer>, IMemoryPackFormatterRegister

Inherited Members

Syncer.IsChanged , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Remarks

MemoryPack GenerateType: Object

```
UnityEngine.Vector3 _positionUnityEngine.Vector3 _eulerAnglesUnityEngine.Vector3 _localScale
```

Methods

Deserialize(ref MemoryPackReader, ref TransformSyncer?)

```
[Preserve]
public static void Deserialize(ref MemoryPackReader reader, ref TransformSyncer? value)
```

Parameters

`reader` MemoryPackReader

`value` TransformSyncer

NetworkEarlyUpdate(Entity)

```
public override void NetworkEarlyUpdate(Entity entity)
```

Parameters

`entity` Entity

NetworkLateUpdate(Entity)

```
public override void NetworkLateUpdate(Entity entity)
```

Parameters

`entity` Entity

RegisterFormatter()

```
[Preserve]
public static void RegisterFormatter()
```

Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter>, ref TransformSyncer?)

```
[Preserve]
public static void Serialize<TBufferWriter>(ref MemoryPackWriter<TBufferWriter> writer, ref
TransformSyncer? value) where TBufferWriter : class, IBufferWriter<byte>
```

Parameters

`writer` MemoryPackWriter<TBufferWriter>

`value` TransformSyncer

Type Parameters

`TBufferWriter`

UpdateSyncer(Entity)

```
public override void UpdateSyncer(Entity entity)
```

Parameters

`entity` Entity