



GIFT Unity SDK Documentation 2017-1

This site contains reference documentation for the GIFT Unity SDK that was released alongside the 2017-1 release of GIFT. For information on how to use the SDK visit the corresponding [GIFT Unity Embedded Application Developer Guide](#)



- + **Mil.Arl.Gift.Unity.Connectivity**
- + **Mil.Arl.Gift.Unity.Messaging**
- + **Mil.Arl.Gift.Unity.Messaging.Incoming**
- + **Mil.Arl.Gift.Unity.Messaging.Outgoing**

API Documentation

Click one of the namespaces in this page's table of contents to get started!

**- Mil.Arl.Gift.Unity.Connectivity**

AbstractGiftConnector
AbstractGiftConnector.Gift
ConnectorState
GiftConnection
GiftConnectorFactory
UnityWebGLGiftConnector

+ Mil.Arl.Gift.Unity.Messaging**+ Mil.Arl.Gift.Unity.Messaging.
Incoming****+ Mil.Arl.Gift.Unity.Messaging.
Outgoing**

Namespace Mil.Arl.Gift.Unity.Connectivity

Classes

[AbstractGiftConnector](#)

Represents a platform-agnostic point of connection to GIFT. Provides events that represents incoming messages as well as methods for sending common messages back to GIFT. Derived classes implement a way to communicate messages to GIFT as well as a way to serialize and deserialize messages to JSON

[GiftConnection](#)

Provides a target that the hosting HTML webpage can send messages to. Forwards incoming messages from GIFT to the GiftConnector and also performs logic for handling errors when the API consumer does not handle required messages.

[GiftConnectorFactory](#)

The factory class that constructs AbstractGiftConnectors. Responsible for ensuring that only one instance of a each type of connector exists at any given time.

[UnityWebGLGiftConnector](#)

The AbstractGiftConnector implementation for Unity WebGL applications. Used to communicate with GIFT through JavaScript postMessage events. Assumes the Unity WebGL application is hosted inside an iframe embedded in the Tutor.

Enums

[AbstractGiftConnector.GiftConnectorState](#)

The enumeration for all the states a GIFT Training application can be in

[Back to top](#)



+ Mil.Arl.Gift.Unity.Connectivity

- **Mil.Arl.Gift.Unity.Messaging**

Message

+ Mil.Arl.Gift.Unity.Messaging.
Incoming

+ Mil.Arl.Gift.Unity.Messaging.
Outgoing

Namespace

Mil.Arl.Gift.Unity.Messaging

Classes

[Message](#)

The class that represents a message to or from GIFT. An instance of Message is serialized as a JSON string to be between GIFT and the embedded training application who then deserializes Message and eventually deserializes payload as a specific type of object.



+ Mil.Arl.Gift.Unity.Connectivity

+ Mil.Arl.Gift.Unity.Messaging

- **Mil.Arl.Gift.Unity.Messaging.Incoming**

Siman

SingleStringPayload

+ Mil.Arl.Gift.Unity.Messaging.Outgoing

Namespace Mil.Arl.Gift.Unity.Messaging.Incoming

Classes

[Siman](#)

Represents a Siman message from Unity.

[SingleStringPayload](#)

Wrapper around a single string. Used only for deserializing a feedback request's feedback string.



+ [Mil.Arl.Gift.Unity.Connectivity](#)

+ [Mil.Arl.Gift.Unity.Messaging](#)

+ [Mil.Arl.Gift.Unity.Messaging.Incoming](#)

- [Mil.Arl.Gift.Unity.Messaging.Outgoing](#)

[SimpleExampleState](#)

[StopFreeze](#)

Namespace

Mil.Arl.Gift.Unity.Messaging.Outgoing

Classes

[SimpleExampleState](#)

Class that represents a SimpleExampleState message. Used to serialize to and deserialize from the JSON format that GIFT expects

[StopFreeze](#)

Class that represents a StopFreeze message. Used to tell GIFT to stop the scenario and advance to the next transition.



- Mil.Arl.Gift.Unity.Connectivity

[AbstractGiftConnector](#)

[AbstractGiftConnector.Gift](#)

[ConnectorState](#)

[GiftConnection](#)

[GiftConnectorFactory](#)

[UnityWebGLGiftConnector](#)

+ Mil.Arl.Gift.Unity.Messaging

+ Mil.Arl.Gift.Unity.Messaging.Incoming

+ Mil.Arl.Gift.Unity.Messaging.Outgoing

Class AbstractGiftConnector

Represents a platform-agnostic point of connection to GIFT. Provides events that represents incoming messages as well as methods for sending common messages back to GIFT. Derived classes implement a way to communicate messages to GIFT as well as a way to serialize and deserialize messages to JSON

Inheritance

- System.Object
 - AbstractGiftConnector
- [UnityWebGLGiftConnector](#)

Inherited Members

System.Object.ToString()

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

Namespace: [Mil.Arl.Gift.Unity.Connectivity](#)

Assembly: Assembly-CSharp.dll

Syntax

```
public abstract class AbstractGiftConnector
```

Constructors

AbstractGiftConnector()

Constructs the connector and sends a handshake message to GIFT to let it know that the Application Author's code is now runnable

Declaration

```
public AbstractGiftConnector()
```

Properties

Status

The status of the connector/training application. Used in order to ensure the proper lifecycle messages are being sent at the appropriate times.

Declaration

```
public AbstractGiftConnector.GiftConnectorState Status {  
    get; protected set; }
```

Property Value

Type	Description
AbstractGiftConnector.GiftConnectorState	The current state of the connector/application

Methods

Deserialize<T>(String)

A method that converts a JSON string into a given type. Should be overridden by a derived connector so that whatever JSON Serialization library is available can be used.

Declaration

```
protected abstract T Deserialize<T>(string json)
```

Parameters

Type	Name	Description
System.String	<i>json</i>	The JSON string to deserialize

Returns

Type	Description
T	The deserialized object of the type that is supplied via the generic method's type parameter.

Type Parameters

Name	Description
<i>T</i>	

HasSimanHandler()

Evaluates whether or not there are any subscribers to the OnSimanReceived event.

Declaration

```
public bool HasSimanHandler()
```

Returns

Type	Description
System.Boolean	

PostIncomingMessage(String)

Deserializes an incoming message and raises the appropriate event

Declaration

```
public void PostIncomingMessage(string messageText)
```

Parameters

Type	Name	Description
System.String	<i>messageText</i>	The message serialized as JSON

SendErrorToGift(String)

Sends an error message to GIFT that will prematurely end the course. Use if the training application is in a state that cannot be recovered from.

Declaration

```
public void SendErrorToGift(string str)
```

Parameters

Type	Name	Description
System.String	<i>str</i>	The message to place into the details of the error dialog. Include any information that will help with debugging the issue.

SendFinishedMessage()

Sends a message that informs GIFT the application's scenario has ended.

Declaration

```
public void SendFinishedMessage()
```

SendHandshake()

Used to send a handshake message once the application is ready to receive messages from the Gift Tutor. Should be sent as soon as the AbstractGiftConnector is listening for messages from GIFT.

Declaration

```
public void SendHandshake()
```

SendLoadedMessage()

Used to send a loaded message once the application is ready to start. The application should be ready to start before this message is sent. The user should not yet be able to interact with the application until the start message is received.

Declaration

```
public void SendLoadedMessage()
```

SendMessageToGift(Message)

Sends the message to GIFT over the derived connector's chosen form of communication.

Declaration

```
public void SendMessageToGift(Message msg)
```

Parameters

Type	Name	Description
Message	<i>msg</i>	The message to send to GIFT

SendMessageToGift(String)

To be overridden by a derived connector. Should send the string to GIFT to via some form of communication (e.g. window.postMessage()).

Declaration

```
protected abstract void SendMessageToGift(string message )
```

Parameters

--	--	--

Type	Name	Description
System.String	<i>message</i>	The string to send to GIFT

SendMessageToGift(String, String)

Sends a message to GIFT with the given raw string and

Declaration

```
public void SendMessageToGift(string type, string payload)
```

Parameters

Type	Name	Description
System.String	<i>type</i>	
System.String	<i>payload</i>	

SendMessageToGift<T>(String, T)

Sends a message to GIFT with the specified type as well as a serialized payload

Declaration

```
public void SendMessageToGift<T>(string type, T payload)
```

Parameters

Type	Name	Description
System.String	<i>type</i>	The type of the message. Provides a hint to the receiver of the message on how to deserialize the payload
T	<i>payload</i>	The object to serialize as the payload of the message. Should be a type found in the Mil.Arl.Gift.Unity.Messaging

Type Parameters

Name	Description
<i>T</i>	

SendSimpleExampleState(String)

Sends a simple example state message to GIFT

Declaration

```
public void SendSimpleExampleState(string var)
```

Parameters

Type	Name	Description
System.String	<i>var</i>	The string value to use within the message

SendStartedMessage()

Used to send a started message once the application has started. Only use this once the user can begin to interact with the application so that GIFT's timekeeping for the simulation is accurate.

Declaration

```
public void SendStartedMessage()
```

SendStoppedMessage()

Used to send a stopped message once the application successfully stopped. Should not be called until the application has successfully cleaned itself up since GIFT will be able to force quit the application once this message is received.

Declaration

```
public void SendStoppedMessage()
```

Serialize<T>(T)

A method that converts a .NET object into a JSON string. Should be overridden by a derived connector so that whatever JSON Serialization library is available can be used.

Declaration

```
protected abstract string Serialize<T>(T obj)
```

Parameters

Type	Name	Description
T	<i>obj</i>	The object to serialize into JSON

Returns

Type	Description

System.String	The object represented as a JSON string
---------------	---

Type Parameters

Name	Description
<i>T</i>	

Events

OnFeedbackReceived

The event that is raised when the connector receives a request to display feedback in the training application from GIFT.

Declaration

```
public event Action<string> OnFeedbackReceived
```

Event Type

Type	Description
System.Action<System.String>	

OnMessageReceived

The event that is raised when the connector receives a message from GIFT that is not currently supported by the API.

Declaration

```
public event Action<Message> OnMessageReceived
```

Event Type

Type	Description
System.Action<Mil.Arl.Gift.Unity.Messaging.Message>	

OnSimanReceived

The event that is raised when the connector receives a Siman message from GIFT

Declaration

```
public event Action<Siman> OnSimanReceived
```

Event Type

Type	Description
System.Action<Mil.Arl.Gift.Unity.Messaging.Incoming.Siman>	

[Back to top](#)



- **Mil.Arl.Gift.Unity.Connectivity**

- AbstractGiftConnector
- AbstractGiftConnector.Gift
- ConnectorState
- [GiftConnection](#)
- GiftConnectorFactory
- UnityWebGLGiftConnector

+ **Mil.Arl.Gift.Unity.Messaging**

+ **Mil.Arl.Gift.Unity.Messaging.Incoming**

+ **Mil.Arl.Gift.Unity.Messaging.Outgoing**

Class GiftConnection

Provides a target that the hosting HTML webpage can send messages to. Forwards incoming messages from GIFT to the GiftConnector and also performs logic for handling errors when the API consumer does not handle required messages.

Inheritance

- System.Object
 - UnityEngine.Object
 - UnityEngine.Component
 - UnityEngine.Behaviour
 - UnityEngine.MonoBehaviour
 - GiftConnection

Inherited Members

- UnityEngine.MonoBehaviour.Invoke(System.String, System.Single)
- UnityEngine.MonoBehaviour.InvokeRepeating(System.String, System.Single, System.Single)
- UnityEngine.MonoBehaviour.CancelInvoke()
- UnityEngine.MonoBehaviour.CancelInvoke(System.String)
- UnityEngine.MonoBehaviour.IsInvoking(System.String)
- UnityEngine.MonoBehaviour.IsInvoking()
- UnityEngine.MonoBehaviour.StartCoroutine(System.Collections.IEnumerator)
- UnityEngine.MonoBehaviour.StartCoroutine_Auto(System.Collections.IEnumerator)
- UnityEngine.MonoBehaviour.StartCoroutine(System.String, System.Object)
- UnityEngine.MonoBehaviour.StartCoroutine(System.String)
- UnityEngine.MonoBehaviour.StopCoroutine(System.String)
- UnityEngine.MonoBehaviour.StopCoroutine(System.Collections.IEnumerator)
- UnityEngine.MonoBehaviour.StopCoroutine(UnityEngine.Coroutine)
- UnityEngine.MonoBehaviour.StopAllCoroutines()
- UnityEngine.MonoBehaviour.print(System.Object)
- UnityEngine.MonoBehaviour.useGUILayout
- UnityEngine.MonoBehaviour.runInEditMode
- UnityEngine.Behaviour.enabled
- UnityEngine.Behaviour.isActiveAndEnabled
- UnityEngine.Component.GetComponent(System.Type)

UnityEngine.Component.GetComponent<T>()
UnityEngine.Component.GetComponent(System.String)
UnityEngine.Component.GetComponentInChildren(System.Type,
System.Boolean)
UnityEngine.Component.GetComponentInChildren(System.Type)
UnityEngine.Component.GetComponentInChildren<T>()
UnityEngine.Component.GetComponentInChildren<T>(System.Boolean)
UnityEngine.Component.GetComponentsInChildren(System.Type)
UnityEngine.Component.GetComponentsInChildren(System.Type,
System.Boolean)
UnityEngine.Component.GetComponentsInChildren<T>
(System.Boolean)
UnityEngine.Component.GetComponentsInChildren<T>
(System.Boolean, System.Collections.Generic.List<T>)
UnityEngine.Component.GetComponentsInChildren<T>()
UnityEngine.Component.GetComponentsInChildren<T>
(System.Collections.Generic.List<T>)
UnityEngine.Component.GetComponentInParent(System.Type)
UnityEngine.Component.GetComponentInParent<T>()
UnityEngine.Component.GetComponentsInParent(System.Type)
UnityEngine.Component.GetComponentsInParent(System.Type,
System.Boolean)
UnityEngine.Component.GetComponentsInParent<T>(System.Boolean)
UnityEngine.Component.GetComponentsInParent<T>(System.Boolean,
System.Collections.Generic.List<T>)
UnityEngine.Component.GetComponentsInParent<T>()
UnityEngine.Component.GetComponents(System.Type)
UnityEngine.Component.GetComponents(System.Type,
System.Collections.Generic.List<UnityEngine.Component>)
UnityEngine.Component.GetComponents<T>
(System.Collections.Generic.List<T>)
UnityEngine.Component.GetComponents<T>()
UnityEngine.Component.CompareTag(System.String)
UnityEngine.Component.SendMessageUpwards(System.String,
System.Object, UnityEngine.SendMessageOptions)
UnityEngine.Component.SendMessageUpwards(System.String,
System.Object)
UnityEngine.Component.SendMessageUpwards(System.String)
UnityEngine.Component.SendMessageUpwards(System.String,
UnityEngine.SendMessageOptions)
UnityEngine.Component.SendMessage(System.String, System.Object,
UnityEngine.SendMessageOptions)
UnityEngine.Component.SendMessage(System.String, System.Object)
UnityEngine.Component.SendMessage(System.String)
UnityEngine.Component.SendMessage(System.String,
UnityEngine.SendMessageOptions)
UnityEngine.Component.BroadcastMessage(System.String,
System.Object, UnityEngine.SendMessageOptions)

UnityEngine.Component.BroadcastMessage(System.String, System.Object)
UnityEngine.Component.BroadcastMessage(System.String)
UnityEngine.Component.BroadcastMessage(System.String, UnityEngine.SendMessageOptions)
UnityEngine.Component.transform
UnityEngine.Component.gameObject
UnityEngine.Component.tag
UnityEngine.Component.rigidbody
UnityEngine.Component.rigidbody2D
UnityEngine.Component.camera
UnityEngine.Component.light
UnityEngine.Component.animation
UnityEngine.Component.constantForce
UnityEngine.Component.renderer
UnityEngine.Component.audio
UnityEngine.Component.guiText
UnityEngine.Component.networkView
UnityEngine.Component.guiElement
UnityEngine.Component.guiTexture
UnityEngine.Component.collider
UnityEngine.Component.collider2D
UnityEngine.Component.hingeJoint
UnityEngine.Component.particleEmitter
UnityEngine.Component.particleSystem
UnityEngine.Object.Destroy(UnityEngine.Object, System.Single)
UnityEngine.Object.Destroy(UnityEngine.Object)
UnityEngine.Object.DestroyImmediate(UnityEngine.Object, System.Boolean)
UnityEngine.Object.DestroyImmediate(UnityEngine.Object)
UnityEngine.Object.FindObjectsOfType(System.Type)
UnityEngine.Object.DontDestroyOnLoad(UnityEngine.Object)
UnityEngine.Object.DestroyObject(UnityEngine.Object, System.Single)
UnityEngine.Object.DestroyObject(UnityEngine.Object)
UnityEngine.Object.FindSceneObjectsOfType(System.Type)
UnityEngine.Object.FindObjectsOfTypeIncludingAssets(System.Type)
UnityEngine.Object.FindObjectsOfTypeAll(System.Type)
UnityEngine.Object.ToString()
UnityEngine.Object.GetInstanceID()
UnityEngine.Object.GetHashCode()
UnityEngine.Object.Equals(System.Object)
UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Vector3, UnityEngine.Quaternion)
UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Vector3, UnityEngine.Quaternion, UnityEngine.Transform)
UnityEngine.Object.Instantiate(UnityEngine.Object)
UnityEngine.Object.Instantiate(UnityEngine.Object, UnityEngine.Transform)

UnityEngine.Object.Instantiate(UnityEngine.Object,
UnityEngine.Transform, System.Boolean)
UnityEngine.Object.Instantiate<T>(T)
UnityEngine.Object.Instantiate<T>(T, UnityEngine.Vector3,
UnityEngine.Quaternion)
UnityEngine.Object.Instantiate<T>(T, UnityEngine.Vector3,
UnityEngine.Quaternion, UnityEngine.Transform)
UnityEngine.Object.Instantiate<T>(T, UnityEngine.Transform)
UnityEngine.Object.Instantiate<T>(T, UnityEngine.Transform,
System.Boolean)
UnityEngine.Object.FindObjectsOfType<T>()
UnityEngine.Object.FindObjectOfType<T>()
UnityEngine.Object.FindObjectOfType(System.Type)
UnityEngine.Object.name
UnityEngine.Object.hideFlags
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: [Mil.Arl.Gift.Unity.Connectivity](#)

Assembly: Assembly-CSharp.dll

Syntax

```
public class GiftConnection : MonoBehaviour
```

Methods

OnExternalMessageReceived(String)

Method that is called externally by the hosting HTML page when a message is sent from GIFT, to the Unity WebGL application.

Declaration

```
public IEnumerator OnExternalMessageReceived(string message)
```

Parameters

Type	Name	Description
System.String	<i>message</i>	The JSON string that is being received from GIFT

Returns

Type	Description
System.Collections.IEnumerator	An IEnumerator representing the coroutine

[Back to top](#)



- **Mil.Arl.Gift.Unity.Connectivity**

- AbstractGiftConnector
- AbstractGiftConnector.Gift
- ConnectorState
- GiftConnection
- [GiftConnectorFactory](#)
- UnityWebGLGiftConnector

+ **Mil.Arl.Gift.Unity.Messaging**

+ **Mil.Arl.Gift.Unity.Messaging.Incoming**

+ **Mil.Arl.Gift.Unity.Messaging.Outgoing**

Class GiftConnectorFactory

The factory class that constructs AbstractGiftConnectors. Responsible for ensuring that only one instance of a each type of connector exists at any given time.

Inheritance

- System.Object
 - GiftConnectorFactory

Inherited Members

- System.Object.ToString()
- System.Object.Equals(System.Object)
- System.Object.Equals(System.Object, System.Object)
- System.Object.ReferenceEquals(System.Object, System.Object)
- System.Object.GetHashCode()
- System.Object.GetType()
- System.Object.MemberwiseClone()

Namespace: [Mil.Arl.Gift.Unity.Connectivity](#)

Assembly: Assembly-CSharp.dll

Syntax

```
public class GiftConnectorFactory
```

Methods

CreateGiftConnector(Type)

Creates or fetches the connector of a specified type

Declaration

```
public static AbstractGiftConnector CreateGiftConnector(
    Type connectorType)
```

Parameters

Type	Name	Description
System.Type	<i>connectorType</i>	The type of connector to return

Returns

Type	Description
AbstractGiftConnector	The connector of the specified type. If one has been previously constructed it is returned

[Back to top](#)

**- Mil.Arl.Gift.Unity.Connectivity**

[AbstractGiftConnector](#)
[AbstractGiftConnector.Gift](#)
[ConnectorState](#)
[GiftConnection](#)
[GiftConnectorFactory](#)
[UnityWebGLGiftConnector](#)

+ Mil.Arl.Gift.Unity.Messaging**+ Mil.Arl.Gift.Unity.Messaging.
Incoming****+ Mil.Arl.Gift.Unity.Messaging.
Outgoing**

Class

UnityWebGLGiftConnector

The [AbstractGiftConnector](#) implementation for Unity WebGL applications. Used to communicate with GIFT through JavaScript `postMessage` events. Assumes the Unity WebGL application is hosted inside an `iframe` embedded in the Tutor.

Inheritance

- [System.Object](#)
 - [AbstractGiftConnector](#)
 - [UnityWebGLGiftConnector](#)

Inherited Members

[AbstractGiftConnector.PostIncomingMessage\(String\)](#)
[AbstractGiftConnector.HasSimanHandler\(\)](#)
[AbstractGiftConnector.SendHandshake\(\)](#)
[AbstractGiftConnector.SendLoadedMessage\(\)](#)
[AbstractGiftConnector.SendStartedMessage\(\)](#)
[AbstractGiftConnector.SendStoppedMessage\(\)](#)
[AbstractGiftConnector.SendFinishedMessage\(\)](#)
[AbstractGiftConnector.SendSimpleExampleState\(String\)](#)
[AbstractGiftConnector.SendMessageToGift<T>\(String, T\)](#)
[AbstractGiftConnector.SendMessageToGift\(String, String\)](#)
[AbstractGiftConnector.SendMessageToGift\(Message\)](#)
[AbstractGiftConnector.SendErrorToGift\(String\)](#)
[AbstractGiftConnector.Status](#)
[AbstractGiftConnector.OnSimanReceived](#)
[AbstractGiftConnector.OnMessageReceived](#)
[AbstractGiftConnector.OnFeedbackReceived](#)
[System.Object.ToString\(\)](#)
[System.Object.Equals\(System.Object\)](#)
[System.Object.Equals\(System.Object, System.Object\)](#)
[System.Object.ReferenceEquals\(System.Object, System.Object\)](#)
[System.Object.GetHashCode\(\)](#)
[System.Object.GetType\(\)](#)
[System.Object.MemberwiseClone\(\)](#)

Namespace: [Mil.Arl.Gift.Unity.Connectivity](#)

Assembly: [Assembly-CSharp.dll](#)

Syntax

```
public class UnityWebGLGiftConnector : AbstractGiftConnector
```

Constructors

UnityWebGLGiftConnector()

Declaration

```
public UnityWebGLGiftConnector()
```

Methods

Deserialize<T>(String)

Deserializes a JSON string using Unity's JSONUtility

Declaration

```
protected override T Deserialize<T>(string json)
```

Parameters

Type	Name	Description
System.String	<i>json</i>	The JSON string to deserialize

Returns

Type	Description
T	The deserialized JSON string as the type specified by the type parameter T

Type Parameters

Name	Description
<i>T</i>	

Overrides

[AbstractGiftConnector.Deserialize<T>\(String\)](#)

SendMessageToGift(String)

Sends a string as a message to GIFT using the postMessage JavaScript function. Assumes that the HTML page hosting the Unity WebGL application has specific global scripts present. These scripts are injected

when the Unity application is uploaded through the GAT

Declaration

```
protected override void SendMessageToGift(string message)
```

Parameters

Type	Name	Description
System.String	<i>message</i>	The message to send to GIFT. Should be a JSON string

Overrides

[AbstractGiftConnector.SendMessageToGift\(String\)](#)

Serialize<T>(T)

Converts a given object to a JSON string using Unity's JsonUtility class

Declaration

```
protected override string Serialize<T>(T obj)
```

Parameters

Type	Name	Description
T	<i>obj</i>	The object to serialize to JSON

Returns

Type	Description
System.String	The passed object as a JSON string

Type Parameters

Name	Description
<i>T</i>	

Overrides

[Mil.Arl.Gift.Unity.Connectivity.AbstractGiftConnector.Serialize<T>\(T\)](#)

[Back to top](#)



- Mil.Arl.Gift.Unity.Connectivity

AbstractGiftConnector
AbstractGiftConnector.GiftConnectorState
GiftConnection
GiftConnectorFactory
UnityWebGLGiftConnector

+ Mil.Arl.Gift.Unity.Messaging

+ Mil.Arl.Gift.Unity.Messaging.Incoming

+ Mil.Arl.Gift.Unity.Messaging.Outgoing

Enum

AbstractGiftConnector.GiftConnectorState

The enumeration for all the states a GIFT Training application can be in

Namespace: [Mil.Arl.Gift.Unity.Connectivity](#)

Assembly: Assembly-CSharp.dll

Syntax

```
public enum GiftConnectorState
```

Fields

Name	Description
Error	
Loaded	
Loading	
Paused	
Pausing	
PreHandshake	
PreLoad	
Started	
Starting	
Stopped	
Stopping	



- + [Mil.Arl.Gift.Unity.Connectivity](#)
- [Mil.Arl.Gift.Unity.Messaging](#)
 - [Message](#)
- + [Mil.Arl.Gift.Unity.Messaging.Incoming](#)
- + [Mil.Arl.Gift.Unity.Messaging.Outgoing](#)

Class Message

The class that represents a message to or from GIFT. An instance of Message is serialized as a JSON string to be between GIFT and the embedded training application who then deserializes Message and eventually deserializes payload as a specific type of object.

Inheritance

- System.Object
 - Message

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: [Mil.Arl.Gift.Unity.Messaging](#)

Assembly: Assembly-CSharp.dll

Syntax

```
[Serializable]  
public class Message
```

Fields

Feedback

Declaration

```
public const string Feedback = "Feedback"
```

Field Value

Type	Description
System.String	

GenericJSONState

Declaration

```
public const string GenericJSONState = "GenericJSONState"
```

Field Value

Type	Description
System.String	

payload

The message content itself as a JSON string. Uses the above field 'type' as a hint on what object to deserialize the object as.

Declaration

```
public string payload
```

Field Value

Type	Description
System.String	

Siman

Declaration

```
public const string Siman = "Siman"
```

Field Value

Type	Description
System.String	

SimanResponse

Declaration

```
public const string SimanResponse = "SimanResponse"
```

Field Value

Type	Description
System.String	

SimpleExampleState

Declaration

```
public const string SimpleExampleState = "SimpleExampleState"
```

Field Value

Type	Description
System.String	

StopFreeze

Declaration

```
public const string StopFreeze = "StopFreeze"
```

Field Value

Type	Description
System.String	

type

The type of the sibling payload field. Used as a hint on how to deserialize the payload for the receiver of this message. Should be assigned one of the public const strings of the Message class.

Declaration

```
public string type
```

Field Value

Type	Description
System.String	

[Back to top](#)



+ Mil.Arl.Gift.Unity.Connectivity

+ Mil.Arl.Gift.Unity.Messaging

- Mil.Arl.Gift.Unity.Messaging.Incoming

Siman

SingleStringPayload

+ Mil.Arl.Gift.Unity.Messaging.Outgoing

Class Siman

Represents a Siman message from Unity.

Inheritance

- System.Object
- Siman

Inherited Members

- System.Object.ToString()
- System.Object.Equals(System.Object)
- System.Object.Equals(System.Object, System.Object)
- System.Object.ReferenceEquals(System.Object, System.Object)
- System.Object.GetHashCode()
- System.Object.GetType()
- System.Object.MemberwiseClone()

Namespace: [Mil.Arl.Gift.Unity.Messaging.Incoming](#)

Assembly: Assembly-CSharp.dll

Syntax

```
[Serializable]
public class Siman
```

Fields

Load

Declaration

```
public const string Load = "Load"
```

Field Value

Type	Description
System.String	

LoadArgs

Key Value pairs that are used as inputs for Siman load messages. Null for all other message types

Declaration

```
public Dictionary<string, string> LoadArgs
```

Field Value

Type	Description
System.Collections.Generic.Dictionary<System.String, System.String>	

Pause

Declaration

```
public const string Pause = "Pause"
```

Field Value

Type	Description
System.String	

Restart

Declaration

```
public const string Restart = "Restart"
```

Field Value

Type	Description
System.String	

Resume

Declaration

```
public const string Resume = "Resume"
```

Field Value

Type	Description
System.String	

RouteType

The type of routing the Siman message uses. Should always be embedded application. Included here for complete representation of the Siman Message

Declaration

```
public string RouteType
```

Field Value

Type	Description
System.String	

Siman_Type

The type of Siman message that this object represents. One of the public const strigs in the Siman class should be used for this value.

Declaration

```
public string Siman_Type
```

Field Value

Type	Description
System.String	

Start

Declaration

```
public const string Start = "Start"
```

Field Value

Type	Description
System.String	

Stop

Declaration

```
public const string Stop = "Stop"
```

Field Value

Type	Description
System.String	



+ Mil.Arl.Gift.Unity.Connectivity

+ Mil.Arl.Gift.Unity.Messaging

- Mil.Arl.Gift.Unity.Messaging.
Incoming

Siman

SingleStringPayload

+ Mil.Arl.Gift.Unity.Messaging.
Outgoing

Class SingleStringPayload

Wrapper around a single string. Used only for deserializing a feedback request's feedback string.

Inheritance

- System.Object
 - SingleStringPayload

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: Mil.Arl.Gift.Unity.Messaging.Incoming

Assembly: Assembly-CSharp.dll

Syntax

```
[Serializable]  
public class SingleStringPayload
```

Constructors

SingleStringPayload(String)

Initializes a SingleStringPayload's string field with the given value

Declaration

```
public SingleStringPayload(string var)
```

Parameters

Type	Name	Description
System.String	var	The value to initialize the string field with

Fields

StringPayload

The string value

Declaration

```
public string StringPayload
```

Field Value

Type	Description
System.String	

[Back to top](#)

[+ Mil.Arl.Gift.Unity.Connectivity](#)[+ Mil.Arl.Gift.Unity.Messaging](#)[+ Mil.Arl.Gift.Unity.Messaging.Incoming](#)[- **Mil.Arl.Gift.Unity.Messaging.Outgoing**](#)[SimpleExampleState](#)[StopFreeze](#)

Class SimpleExampleState

Class that represents a SimpleExampleState message. Used to serialize to and deserialize from the JSON format that GIFT expects

Inheritance

- System.Object
 - SimpleExampleState

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: [Mil.Arl.Gift.Unity.Messaging.Outgoing](#)

Assembly: Assembly-CSharp.dll

Syntax

```
[Serializable]  
public class SimpleExampleState
```

Constructors

SimpleExampleState(String)

Constructs a SimpleExampleState and initializes its VAR payload to a given value

Declaration

```
public SimpleExampleState(string var)
```

Parameters

Type	Name	Description
System.String	<i>var</i>	The value to initialize VAR to

Fields

VAR

Declaration

```
public string VAR
```

Field Value

Type	Description
System.String	

[Back to top](#)

[+ Mil.Arl.Gift.Unity.Connectivity](#)[+ Mil.Arl.Gift.Unity.Messaging](#)[+ Mil.Arl.Gift.Unity.Messaging.Incoming](#)[- Mil.Arl.Gift.Unity.Messaging.Outgoing](#)[SimpleExampleState](#)[StopFreeze](#)

Class StopFreeze

Class that represents a StopFreeze message. Used to tell GIFT to stop the scenario and advance to the next transition.

Inheritance

- System.Object
 - StopFreeze

Inherited Members

System.Object.ToString()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()

Namespace: [Mil.Arl.Gift.Unity.Messaging.Outgoing](#)

Assembly: Assembly-CSharp.dll

Syntax

```
[Serializable]  
public class StopFreeze
```

Fields

frozenBehavior

ID for the frozen behavior, that will indicate how the entity or exercise acts while frozen

Declaration

```
public int frozenBehavior
```

Field Value

Type	Description
System.Int32	

realWorldTime

The real-world time (UTC, milliseconds since midnight Jan 1, 1970) at which the entity is to start/resume in the exercise.

Declaration

```
public long realWorldTime
```

Field Value

Type	Description
System.Int64	

reason

ID for the reason that an entity or exercise was stopped/frozen

Declaration

```
public int reason
```

Field Value

Type	Description
System.Int32	

requestID

Unique ID for this request

Declaration

```
public long requestID
```

Field Value

Type	Description
System.Int64	

[Back to top](#)