

Getting Started with the IRC

Created with reference to version 2.0.0

Authentication

TwitchIRCCClient is dependent on TwitchAPIClient for authentication and chat room information. Before trying to run the IRC please make sure you have read the Getting Started with StreamLinked guide so you have an instance of TwitchAPIClient in your scene. If you encounter a Login error with the IRC, make sure you include scopes for your tokens that include chat_read and chat_edit.

Setup

The IRC is built into the GameObject TwitchIRCCClient class. A gameobject is provided with the script attached, just place this object in the scene and run the scene, it will try to connect to the authenticators twitch chat.

To extract the data received by the IRC, UnityEvent listeners are provided for every response sent to it from Twitch.

These can be found on the Events page of the inspector. For example Twitch chat messages sent by users are received under PRIVMSG which contains all of the information contained in the class TwitchMessage.

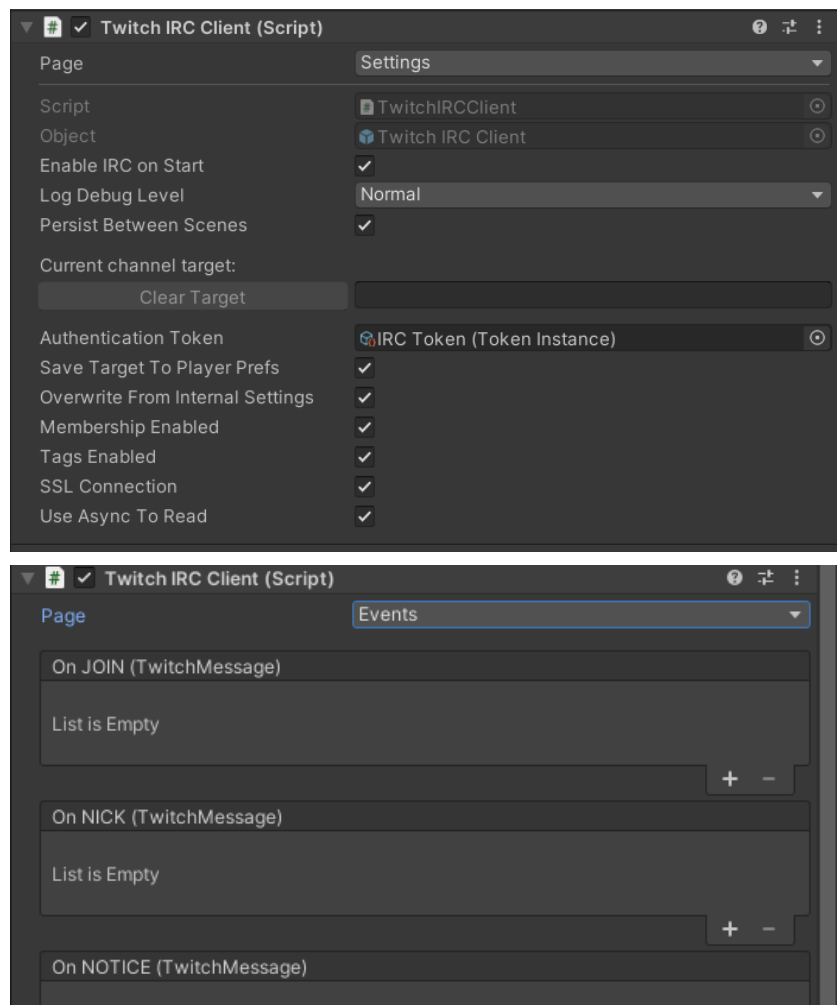
You can find the reference page for Twitch IRC [here](#).

Example Scene

Chatbox Example is the example scene provided with IRC built into it. Using the emote and badge builders provided the scene works to translate the incoming messages from the IRC into a presentable format just like Twitch chat with the images provided for a user's badges and any emotes sent in the message.

The scene requires TextMeshPro to be imported to work.

The script that reads and displays the message is in the class TwitchMessageGameObjectProvider and adds the listeners to the IRC programmatically.



The GameObject that holds this script and subsequent prefabs displaying the messages is 'Twitch Message Provider'. This object listens to the IRC, receives any processed PRIVMSG received from Twitch, requests any objects that need to be generated such as the TextMeshPro Atlas, request the text to display all the images in the message and finally creates the TextMeshPro complete object to then be placed in the message list.

Built into the construction of TwitchMessage, if any badges or emotes are found in the IRC tags, it will send the information (if present in the scene) to the relevant builder which will acquire the image to present for the message. The message also contains a function called BuildMessageForTextmesh to get the formatted message for it to work in the relevant TextMeshPro field. These builders are covered in more details in the document Additional Provided Class Information.

From here you have access to what the IRC can provide, any messages that Twitch sends can be processed for you to read and respond to. For greater depth of control of what you would like to receive from Twitch I recommend you look into the EventSub, as with the IRC it sends everything to you vs the EventSub which you only receive what you subscribe to.