Contents

About

Welcome to U-DMX, a robust and versatile yet easy-to-use solution to control DMX lights from your Unity application.

This allows you to turn your light setup completely interactive. Rig.-up your escape room, gamify your theater, and haunt your peculiarly DMX light-ready house with this easy-to-use and easy-to-extend plugin.

We developed this tool to help us realize our own interactive museum exhibition, and combined it with a networked server, to allow our users to walk through over 500 sqm of exhibition space while invoking lighting effects on over 50 DMX light fixtures.
Long story short, we have tried and tested this tool, on a large scale installation, and have optimized it to be usable by everyone, and easily customizable to your needs.

Check out the play through video here:

This plugin uses the **Open DMX USB** protocol to send **DMX** light information to DMX-capable light fixtures.

DMX

DMX is the industry-standard method of controlling stage lights, concert lights, smoke machines, winches and pulleys, and much more. There is a massive ecosystem of devices you can integrate into your project.
Find out more about the DMX protocol here: https://en.wikipedia.org/wiki/DMX512

Open DMX USB

Read more about Open DMX USB, and compatible devices here: https://www.enttec.com/product/lighting-communication-protocols/dmx512/open-dmx-usb/

On the web version of the docs, we will have a list of known-to-be-compatible USB devices, submitted by developers like you. While the Enttec devices are great, you can usually get decent off-brand Open DMX USB devices for $15 - $20 on Amazon, should you be on a tighter budget.

How does it work?

The Unity Side

The goal with U-DMX is to integrate it as seamlessly into your existing workflows as possible. Simply treat your DMX light source like a regular Unity light source, make sure everything looks good in engine, and once you are ready, assign physical light fixtures to your virtual ones.
Your Unity application will then send the DMX-Data to the USB-DMX server, a separate application you can run on any Windows device, including the one running your unity game.

The Open DMX Side

Upon receiving the DMX Data, the standalone server sends the DMX data via an Open-DMX capable USB interface to your light fixtures.

Easy as that.

Caveats

Your USB-DMX device needs to run Windows, while the Unity application can be run on any device able to send signals via web requests.

Due to the way Unity handles web requests, there is a small amount of garbage creation happening when sending the data.

To attempt to protect you and your clients from costly blackouts due to ramping up many high wattage stage lights, we have added a slight fade to the USB-Server when sending the data. This increases latency slightly, but believe me, from experience, a bit of latency is a lot more pleasant than causing a street-wide power outage

(Especially older non-LED DMX lights are notorious for blowing fuses when being turned on from 0 to 512 in one millisecond).

USB-DMX has not the same range as more potent DMX sending gear. After the first 15 lights, and a total of 50 meters of cables, we experienced some flickering during our test runs. This is easily preventable if you opt for a DMX repeater, signal booster, or signal splitter. Just be mindful of the limited initial signal strength that USB can offer.

Setup

Setup in Unity

1. Add a DMX controller to your scene, and pick your USB-DMX server's target URL (the default is localhost:14444)
   Replace localhost with your Server's IP if you want to access it from another device in your network, or outside your local network.
2. Add a 2D or 3D DMX light source to your scene, and adjust the DMX address to match the DMX address of your real-world light fixture
3. Configure your device-specific DMX profile and assign it to the light sources in your scene

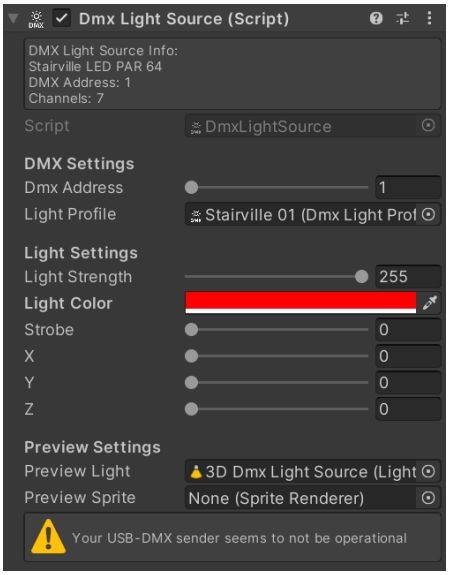
DMX Server Setup

1. Move the "External" folder outside your Unity Assets folder
2. Plug in your Open-DMX capable USB-DMX interface
3. Make sure to install all drivers necessary for your device. For Off brand Open-DMX USB devices, that usually means installing these FTDI drivers: https://ftdichip.com/drivers/d2xx-drivers/
4. Download and unzip the U-DMX Server from: https://hosting.neoludic.games/U-DMX/U-DMX-Server.zip
5. Right-Click on the USB-DMX server EXE, and run it as an administrator

Usage

DMX Light Sources

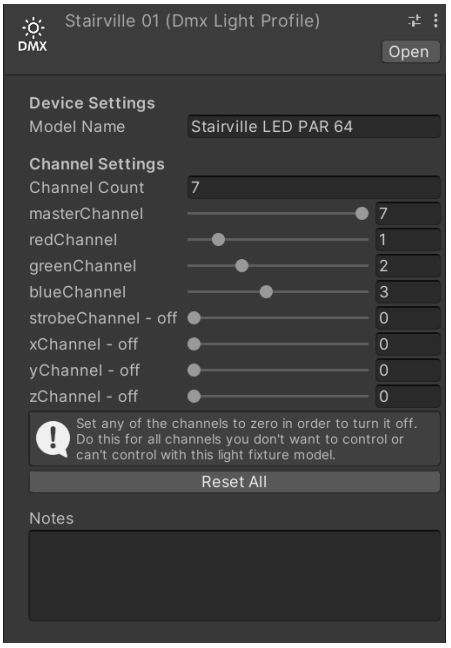
Your DMX light sources in Unity are designed to behave exactly like the normal light sources. Simply set the strength, color, and other variables through animations, scripts, or events, and they will be synced accordingly. Importantly, the DMX address corresponds to the DMX address you have specified on your real-world light fixture. Make sure no two virtual light sources share the same DMX address.



DMX Light Profiles

The DMX Light Profile is responsible for translating each virtual DMX light's information into specific DMX values relevant to your real-world DMX lights. Take a look at your devices' manuals to find out which channels are in use, and which channel corresponds to which local addresses. If a channel is not in use, simply set its address to 0.

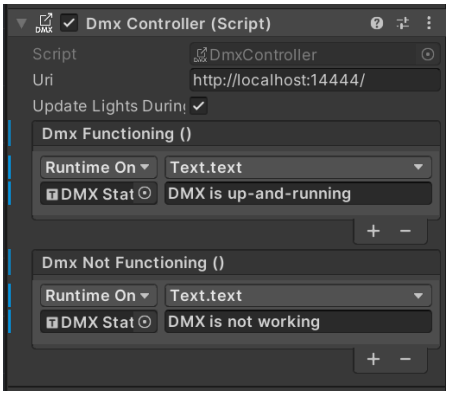
Make sure to assign a light profile to every DMX light source in your scene.



DMX Controller

The DMX controller takes care of storing and sending all DMX information to the U-DMX-Server application. The most important setting here is the URI. If you want to send this data to another device running the server, simply insert it's local or global IP address and the port of your choice.

By default, the server will run on port **14444.**



neoludic games

DMX Server

The U-DMX Server takes care of sending the DMX data via an Open DMX USB interface to your DMX light sources. Sadly, at present, because it relies on Windows internal libraries, the server can only run on a Windows machine.

Make sure all relevant drivers for your USB interface are installed, the device is plugged in, and you run the server as an administrator.

Configuration:
The server includes a .config file. Here you will find 3 values:

Name	Default	Description
Port	14444	The network port of the server
MaxLightStepPer SendingInterval	80	This is the maximum change of any value per update. Note that the value range for DMX is 512.[1]
MilliSecondsDelay BetweenSendingIntervals	1	The server will wait X millisecond after sending a DMX signal and before sending another. Adjust this value as you see fit.[2]

You can adjust these as you see fit. To reset these values, simply delete the file and run the server.

Contact

If you encounter bugs, have questions or feedback, or just want to share your cool DMX installation, send us a mail at: **u-dmx@neoludic.games**

[1] We have added this feature because experience has shown, that if many high wattage (especially older stage lights) are changed too abruptly, your venue's fuses will blow. I don't need to tell you why that is terrible, during an exhibition, escape room play through or theater performance. Be mindful of your gear as you adjust this value. You might even find that you need to lower the step further to avoid outages. Obviously, we can take no responsibility for outages or gear failure whatsoever. Employ U-DMX carefully at your own risk.
[2] On shorter DMX chains, you can safely leave this value at 1. If you find yourself experiencing some glitches etc. try increasing this value. The DMX guidelines mention support for 50 to 100 Hz (DMX signals per second) on most devices. Note slowing down the sending frequency will also slow down the fading speed.