Walkthrough of Custom SVL Asset Creation

Author: Curtis Brinker

Overview

This document acts as a compilation of various guides provided by the SVL developers and gives an end to end walkthrough for the steps required to set up your computer for custom asset creation in the SVL simulator.

The major steps include:

- Installing Unity Hub
- Configuring Unity Modules
- Configuring Git
- Building SVL from source
- Creating a custom sensor
- Building and uploading assets

Requirements

This guide assumes that you already meet the following requirements:

- Using a 64-bit Windows 10 system
- Git is installed on the system
- Access to a valid Unity license

Installing Unity Hub

Unity Hub is required to manage various dependencies for while creating a Unity project. Download and install Unity Hub from the following link:

https://unity3d.com/get-unity/download

Then, download Unity 2019.4.18f1 from this link, preferably using Unity Hub.

https://unity3d.com/get-unity/download/archive

Unity Hub may ask you to sign in during this step. If it does, sign in (or create an account) and add your Unity license.

Configuring Unity Modules

Once Unity 2019.4.18f1 has been added to Unity Hub, go to 'Installs' and click the three dots on the 2019.4.18f1 entry, then select 'Add modules'

Add the following modules:

- Linux Build Support (IL2CPP)
- Linux Build Support (Mono)
- Windows Build Support (IL2CPP)
- Documentation

The following modules are optional:

- Microsoft Visual Studio Community 2019

After selecting the modules, let them install. It may take a while for the installation to complete.

Configuring Git

The SVL simulator GitHub repo uses git-lfs (Git Large File Storage). Assuming git is already installed on your system, go to the following link:

https://git-lfs.github.com/

Download the latest version on the website and install it. Once it has been installed, open a terminal and run the following command:

```
git lfs install
```

If it was successfully installed the following message should appear:

```
Git LFS initialized.
```

Downloading the SVL source code

Before completing this step, ensure that git-lfs has successfully been installed on the system. Errors will occur if that it was not previously installed for the user.

In the terminal, navigate to the folder that SVL simulator source code should be saved at and run the following command:

```
git clone
https://github.com/lgsvl/simulator.
git
```

Building SVL Simulator

Open Unity Hub and open the 'Projects' menu. Click 'Add' and navigate to the 'simulator' folder that was downloaded in the previous step.

Click on the project to open it, Unity should launch. In the menu at the top of Unity, select 'Simulator' then 'Build'. A new 'Build Maps and Vehicles' panel should open.

In the 'Build Maps and Vehicles' panel, check the 'Build Simulator box. In the textbox to the right, select the desired location to build SVL at. Finally, click the 'Build' button at the bottom of the panel.

It may take a long time for the build to complete.

Creating Custom Assets in Unity for SVL

Custom assets can be created for SVL using Unity. Steps will be given for creating custom sensors.

Creating Custom Sensors

The general steps to build a sensor for SVL include:

1. Creating a GameObject
2. Structuring sensor files
3. Writing sensor logic
4. Building and exporting the sensor

First we must create a new game object for the custom sensor. To do this, right click in the 'Hierarchy' panel (by default on the left side of Unity), then click 'Create Empty'.

Next we must structure files for the sensor to be detected by SVL. In the 'Project' panel at the bottom of the screen navigate to 'Assets/External/Sensors' and right click to make a new folder. Name the folder the name of the custom sensor.

Drag the GameObject from the 'Hierarchy' panel into the new folder. Then, rename the GameObject to the same name as the folder.

Next, we must add a C# script to attach to the GameObject. While inside the folder for the custom sensor, right click and select 'Create -> C# Script'. Name it the same as your custom sensor.

Finally, we must attach C# script to the game object. In the 'Assets' panel, click on the custom sensor's GameObject. Then, in the 'Inspector' panel on the right side of the screen, click 'Add Component'. A search window will pop up. Type in the name of the C# script and select the first result. The sensor's GameObject can now be controlled by the C# script.

Writing Sensor Logic

The amount of programming knowledge required for this step depends on the complexity of the sensor you wish to build.

You can modify the C# program in any way to specify the behavior of the sensor.

For examples of how you may want to write your script, you can see the example code for a custom sensor provided by the SVL development team. The code for the sensor can be found at:

https://github.com/lgsvl/ComfortSensor

The code of the built in sensors are also available on the LGSVL GitHub account, though they do not have documentation.

Building and Exporting Custom Assets

Once you are done creating a custom asset for SVL, it can be exported and uploaded to the SVL simulator website.

To do this, go to the 'Build Maps & Vehicles' panel. Then check the box next to the custom asset you want to build. If your custom asset does not appear, save all of your work and restart Unity.

After selecting the asset, go to the bottom of the panel and ensure 'Build Simulator' is unchecked. Then, click 'Build'.

Once building is complete, log in to your SVL account at:

https://wise.svlsimulator.com/

Finally, upload your sensor to your account. An example will be given for uploading a sensor. On the SVL website, use the left panel to navigate to 'Library -> Plugins'. Then click the 'Add New' button on the top right of the window.

You will be prompted to upload an asset bundle. The asset bundle for your custom asset can be found within your directory containing the source code for the SVL simulator. The custom assets are found within the 'AssetBundles' folder of the source code.

Upload the desired AssetBundle and fill in the descriptions as necessary. Afterwards you can use your new asset in SVL just like any other SVL asset.

References

The information in this walkthrough is composed of various segments originally by the SVL team. The purpose of this document was not to create new information, but rather to compile it. Please read the following links for more details on these processes.

1. https://www.svlsimulator.com/docs/installation-guide/build-instructions/
2. https://docs.github.com/en/github/managing-large-files/installing-git-large-file-storage
3. https://www.svlsimulator.com/docs/user-interface/web/library/
4. https://www.svlsimulator.com/docs/plugins/sensor-plugins/
5. https://github.com/lgsvl/ComfortSensor