

This guide is for **Windows** only. There is a script that sets everything up, including a small dataset for testing. For that, skip to **Step 3**. Follow **Step 1** and **2** for preparing other datasets.

## 1. Dataset

Download a point cloud dataset with at least **color** and/or **intensity** attributes (**las**, **laz**, **binary ply**, **xyz**, and **ptx** file formats are supported).

## 2. Convert to Potree File Format

UnityPIC adopts the file format used by Potree, so first convert the point cloud data into the Potree format using PotreeConverter:

Download [PotreeConverter\\_1.7\\_windows\\_x64.zip](#)

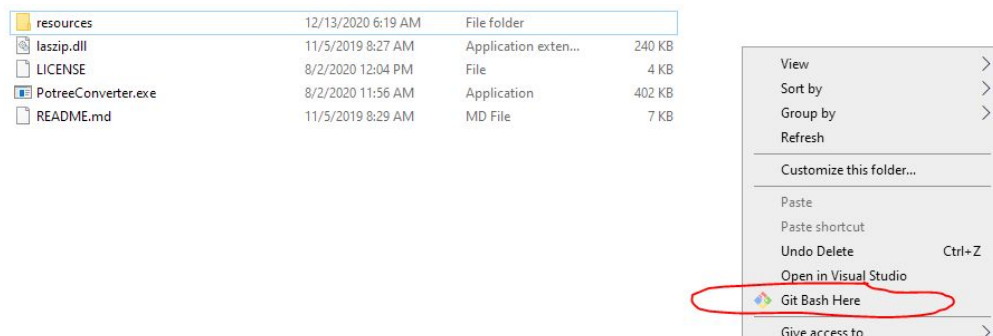
**Note that UnityPIC only supports PotreeConverter 1.6 and 1.7. PotreeConverter 2.0 is currently available, but not yet documented.**

Unzip the downloaded file to get the following:

Name	Date modified	Type	Size
resources	12/13/2020 6:19 AM	File folder	
laszip.dll	11/5/2019 8:27 AM	Application exten...	240 KB
LICENSE	8/2/2020 12:04 PM	File	4 KB
PotreeConverter.exe	8/2/2020 11:56 AM	Application	402 KB
README.md	11/5/2019 8:29 AM	MD File	7 KB

PotreeConverter is a command line tool, so it takes parameters via command line arguments.

[Cmder](#) is a popular terminal for windows (Mini version will suffice). Or if you already have [git](#) installed, git-bash is available and it can be opened on the current folder through the right-click menu:



Otherwise, navigate to the folder where PotreeConverter is extracted:

```
Yao@DESKTOP-PSR0UN0 MINGW64 /f/PotreeConverter_1.7_windows_x64
$
```

Execute the following command:

```
./PotreeConverter.exe <data.las> -a RGB INTENSITY
```

where **<data.las>** is replaced by the directory of the downloaded point cloud data file. **RGB** and **INTENSITY** specifies the extracted point attributes to be color and intensity. It is also fine to omit the **-a RGB INTENSITY** clause. However, only color and intensity can be shown by UnityPIC.

```
Yao@DESKTOP-PSR0UN0 MINGW64 /f/PotreeConverter_1.7_windows_x64
$ ./PotreeConverter.exe /f/datasets/NEONDSampleLiDARPointCloud/NEONDSampleLiDARPointCloud.las -a INTENSITY
== params ==
source[0]:          F:/datasets/NEONDSampleLiDARPointCloud/NEONDSampleLiDARPointCloud.las
outdir:             NEONDSampleLiDARPointCloud.las_converted
spacing:            0
diagonal-fraction:  200
levels:             -1
format:             0
scale:              0
pageName:
projection:

processing following attributes:
POSITION_CARTESIAN
intensity

AABB: {
  "min": [256000.000000, 4111000.000000, 384.660000],
  "max": [256999.990000, 4111999.990000, 510.250000],
  "size": [999.990000, 999.990000, 125.590000]
}

cubicAABB: {
  "min": [256000.000000, 4111000.000000, 384.660000],
  "max": [256999.990000, 4111999.990000, 1384.650000],
  "size": [999.990000, 999.990000, 999.990000]
}

total number of points: 6,609,829
spacing calculated from diagonal: 8.66017
READING: F:/datasets/NEONDSampleLiDARPointCloud/NEONDSampleLiDARPointCloud.las
INDEXING: 1,000,000 of 6,609,829 processed (15%); 1,000,000 written; 0.746 seconds passed
INDEXING: 2,000,000 of 6,609,829 processed (30%); 2,000,000 written; 1.339 seconds passed
INDEXING: 3,000,000 of 6,609,829 processed (45%); 3,000,000 written; 1.884 seconds passed
INDEXING: 4,000,000 of 6,609,829 processed (60%); 4,000,000 written; 2.481 seconds passed
INDEXING: 5,000,000 of 6,609,829 processed (75%); 5,000,000 written; 3.057 seconds passed
INDEXING: 6,000,000 of 6,609,829 processed (90%); 6,000,000 written; 3.623 seconds passed
closing writer

conversion finished
6,609,829 points were processed and 6,609,829 points ( 100% ) were written to the output.
duration: 8.428s

Yao@DESKTOP-PSR0UN0 MINGW64 /f/PotreeConverter_1.7_windows_x64
$
```

After the conversion is completed, a new folder can be found in the current directory, named **<dataset\_name>.las\_converted**. Inside the folder is the converted point cloud.

### 3. UnityPIC

First, clone the repository with the demo standalone:

```
git clone https://github.com/yaoc1996/unitypic-demo.git  
cd unitypic-demo
```

For the automatic setup script (should take up about 400MB, the data download can take up to a few minutes), run:

```
chmod +x quick_test.sh  
./quick_test.sh
```

The following folders should be generated:

converted	12/13/2020 8:51 AM	File folder
PotreeConverter_1.7_windows_x64	12/13/2020 8:51 AM	File folder
raw	12/13/2020 8:41 AM	File folder
UnityPIC	12/13/2020 8:51 AM	File folder

**converted** contains the converted potree format of the test point cloud.

**PotreeConverted\_1.7\_windows\_x64** contains the PotreeConverter version 1.7.

**raw** contains the test point cloud in las format.

**UnityPIC** contains the standalone executable.

Otherwise, extract **UnityPIC.zip** for the standalone executable:

```
unzip UnityPIC.zip
```

To start the demo, run

```
./UnityPIC/UnityPIC.exe ./converted 1000000
```

Click anywhere on the screen to grab the mouse cursor. Use **WASD** to move the camera, **Q** and **E** ascends and descends the camera, and hold **Left SHIFT** to speed. Press **ESC** to release the mouse cursor.

For other datasets, replace **./converted** with the directory of the converted dataset (the directory where cloud.js is found). **1000000** is the point budget (maximum number

of points rendered). The recommended range is 1 million to 10 million. If the FPS is low (laggy), try lowering the point budget.