LOW POLY
FARLAND SKIES

QUICK START

DOC VERSION v2.5.2 - MAY 2022

Contents

Introduction

Thank you for purchasing the **Farland Skies - Low Poly** skybox! We hope you enjoy using the product and that it makes your game development project a success.

If you have questions, please do not hesitate to contact us at *support@borodar.com*, we will be glad to help you out.

Also, when you have a spare moment, please [leave] us a review on the Asset Store.

Integrating and configuring skybox

It's pretty easy to integrate skybox to your scene. All you need to do it's just drag the **SkyboxController** prefab to the hierarchy window.

Skybox will be assigned automatically for this scene and now you can use **SkyboxController** to manage its settings, either from scripts or directly in the editor.

Editor properties

Sky

This section allows you to customize the basic sky gradient with three color values: at the top pole, on the equator, and at the bottom pole. **Top/Bottom Exponent sliders** can be used to adjust linear interpolation between the corresponding pole and equator.

Stars

Here you can configure stars tint, extinction, and twinkling speed. **Stars extinction** is a reduction in stars apparent brightness closer to the horizon. **Stars twinkling** is a variation in stars apparent brightness caused by atmospheric turbulence.
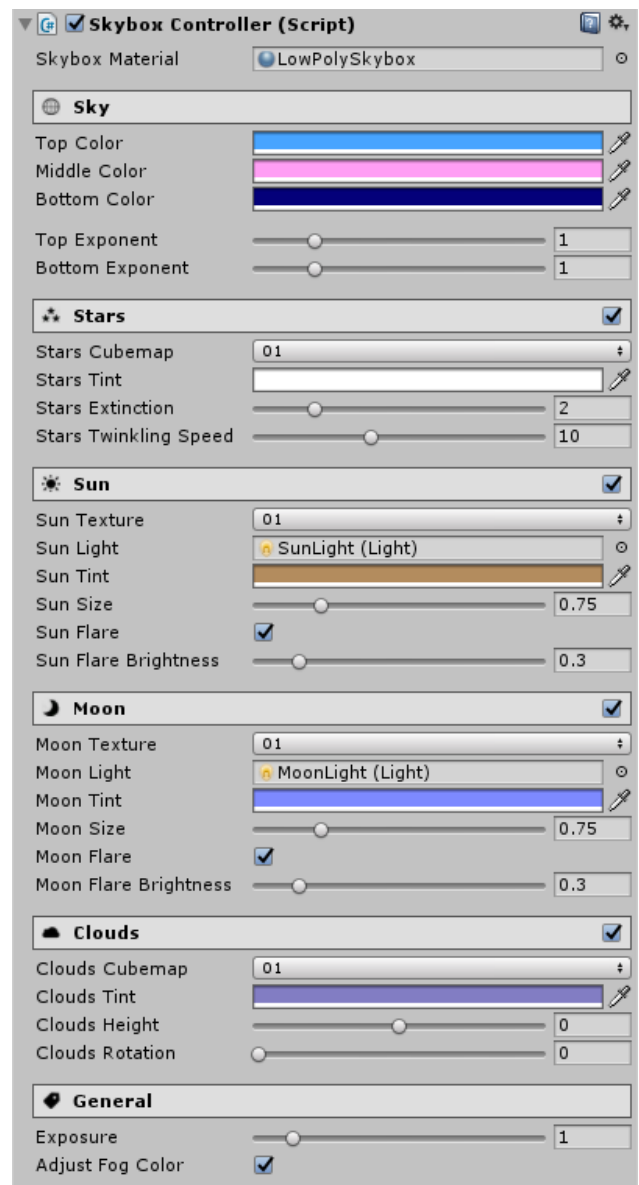
Sun & Moon

In this section, you can customize sun and moon appearance. All settings are straightforward, except *flare brightness*. Actual flare brightness depends on correspondent tint alpha, and this property is just a coefficient for that value.

Clouds

This section allows you to customize clouds colors and positioning.

General

In the general section, you can adjust exposure for the whole sky. Also, there is a checkbox that allows you to keep fog color in sync with the sky middle color automatically.

Access From Scripts

SkyboxController implements the "singleton" pattern, so you can easily access all its properties from anywhere in your scripts:

```
SkyboxController.Instance.MiddleColor = Color.blue;
SkyboxController.Instance.TopExponent = 0.75f;
SkyboxController.Instance.Rotation += 1f;
```

Please take a look at the *CustomizationDemo* scene to get more details.
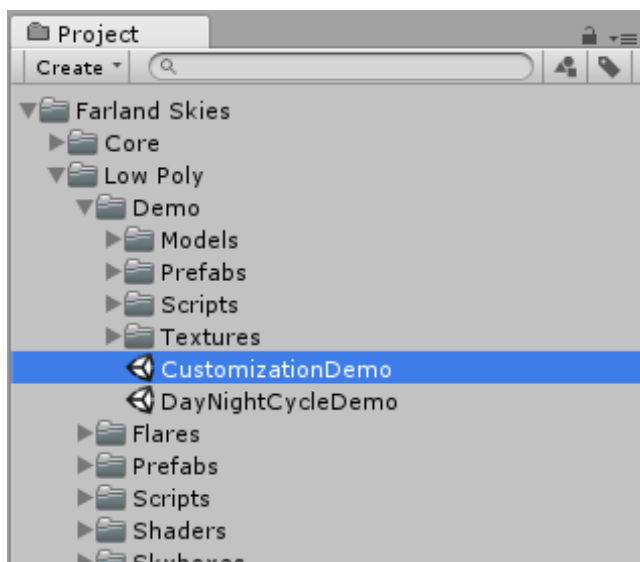
Day-night cycle

This package comes with a handy script to manage skybox day-night cycle.

At first, you need to drag the **SkyboxDayNightCycle** prefab somewhere to your scene.

Then adjust prefab properties to fit your needs:

- **Sky DoT Params** is a list of sky colors, based on time of day. Each list item contains one "time" field that should be specified in percents (0-100). Also, there are three "color" fields, corresponding to the skybox color settings with the same names.

- **Stars DoT Params** list allows you to manage stars tint color over time. It works in the same way - just specify a time in percents and tint color for each element.

- **Sun & Moon DoT Params** are for configuring sun & moon appearance and light params depending on time. Use the same approach as above for setting it up.

- **Clouds Dot Params** list allows you to manage clouds tint color over time. It works in the same way as above.

SkyboxDayNightCycle script will automatically interpolate values between list items and update skybox parameters depending on its **TimeOfDay** property.

However, **SkyboxDayNightCycle** doesn't run through the day-night cycle automatically. It requires a few lines of code by yourself, but in this way, you can fully control your animation cycle duration, make your time flow non-linear, etc.
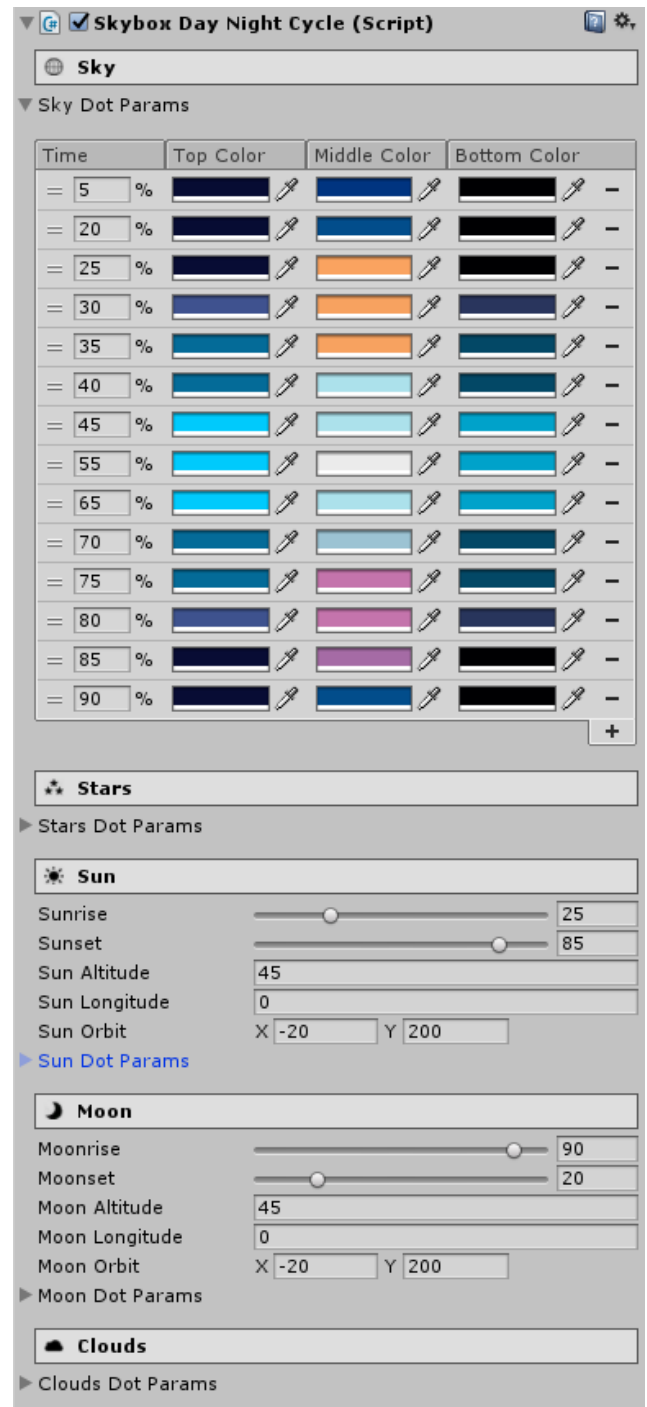
The simplest example to do so:

```
public void Update()
{
    SkyboxDayNightCycle.Instance.TimeOfDay
        += (Time.deltaTime / DayDuration) * 100f;
}
```

If you don't want to bother with scripts by yourself, then you can just grab the
SkyboxCycleManager prefab and add it to your scene. It will manage animation cycle
progress automatically.
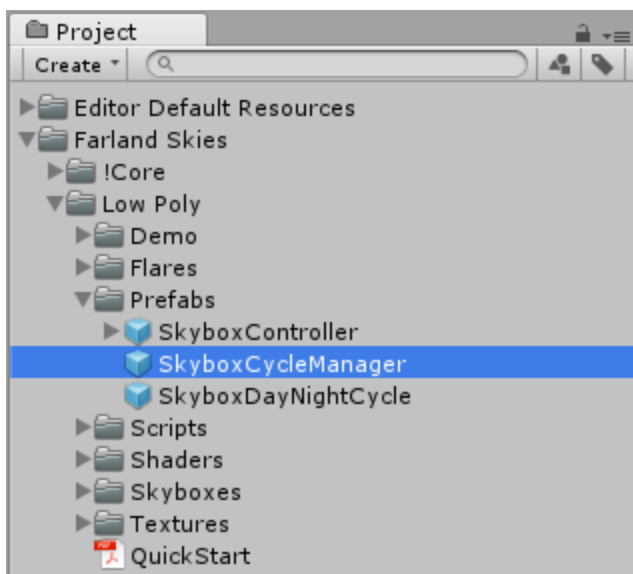
Sun & Moon positioning

Sun & Moon position settings may seem a bit confusing at first look, so let's try to explain it visually:
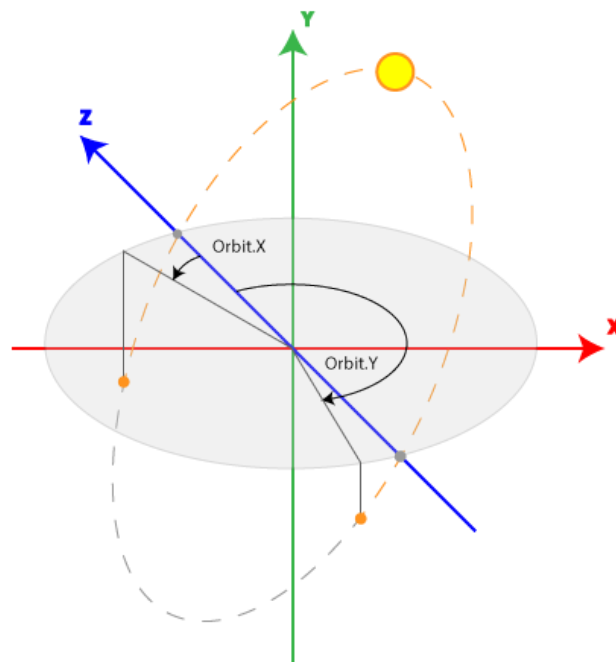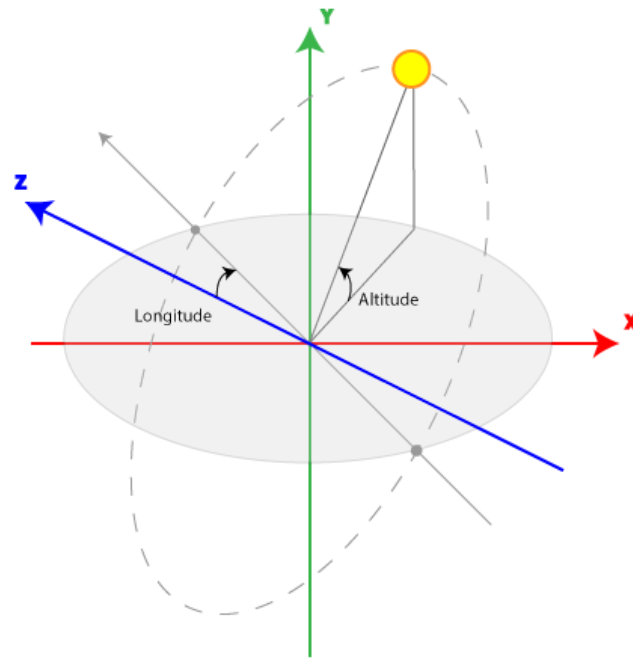

- **Sun Altitude** is the *max* angle between the horizon and the center of sun's disk.
- **Sun Longitude** is the angle between *z-axis* and the center of sun's disk *at sunrise*.
- **Sun Orbit** is a pair of angles that limit visible orbit of the sun.

Also, You can change sun and moon positions just by rotating corresponding direction lights, under the *SkyboxController* object.
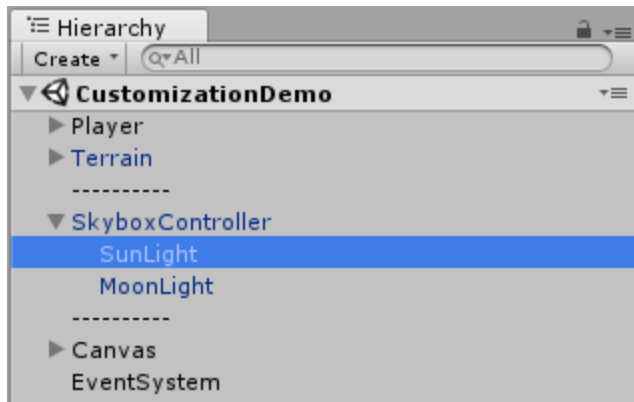
Please take a look at the *DayNightCycleDemo* scene to get more details.
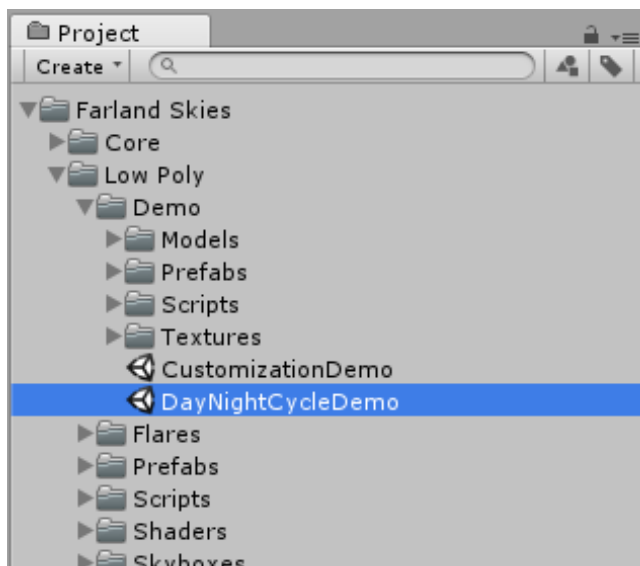
Upgrading

Please always do a clean import of the **Farland Skies - Low Poly** (delete the old version before importing the new one), otherwise you may receive a number of difficult-to-diagnose issues.

- Delete the **Assets/.../Farland Skies** folder
- Delete the **Assets/Editor Default Resources/Farland Skies** folder, if exists
- Import the new version from the Asset Store

Folder location

The **Farland Skies** folder is movable and doesn't require to be in the root of your assets folder.

You can freely move it wherever you want (to some "3rd Party" folder, for example). Then just go to **Edit → Project Settings → Borodar → Farland Skies** and update the folder location:

Changelog

v 2.5.2

- Better compatibility with Unity 2020.3 or greater
- Various optimizations and fixes

v 2.5.1

- Added support for Single-Pass Instanced rendering
- Minor performance improvements

v 2.5

- Better compatibility with Unity 2019.1
- Added support for LWRP
- Replaced obsolete PreferenceItem with new SettingProvider
- Fixed bug with performance degradation when continuously rotating clouds on iOS
- Minor bugfixes

v 2.4.3

- Better compatibility with Unity 2018.2
- A number of small optimizations and fixes

v 2.4.2

- Fixed bug with SkyboxController initialization for large scenes
- Better compatibility with Unity 2017.2

v 2.4.1

- Added option to specify frames interval between skybox day-night cycle updates
- Better compatibility with Unity 2017.1
- Minor bugfixes

v 2.4

- Added option to move the "Farland Skies" folder wherever you want in your project
- Got rid of the "Editor Default Resources" folder
- A number of small optimizations and fixes

v 2.3

- Implemented the possibility to enable/disable clouds, stars, sun, or moon as a shader feature
- Added three different types of clouds
- Added three different types of stars
- Added five different types of sun
- Added five different types of moon
- Various optimizations and fixes

v 2.1

- Custom editor for skybox controller
- Custom editor for skybox day-night cycle script
- A number of small optimizations and fixes

v 2.0

- Completely new single-pass shader
- Added procedural sun and moon
- Various optimizations and fixes

v 1.1

- Added stars
- Various performance improvements

v 1.0

- Initial release