



SMART SHADOWS

v1.1.0

for Unity 5.5 to 2019.3

©Arnaud Emilien

November 12, 2019

Contents

1	Overview	4
1.1	Introduction	4
1.2	Requirements	4
2	Quick Start Guide	5
3	Understanding The Light Sorting Algorithm	6
3.1	Light Priority	6
3.2	Configuring your Scene	6
4	Configuration Guide	9
4.1	Shadow Manager	9
4.1.1	Maximum Simultaneous Shadows.	9
4.1.2	Default Maximum Distance.	9
4.1.3	Light Range and Intensity Reduction.	9
4.1.4	Animations.	9
4.1.5	Automatic Importance.	10
4.1.6	Shadow Resolution Reduction.	10
4.2	Shadow Controller	10
4.2.1	Importance and Priority.	10
4.2.2	Maximum Distance.	10
4.2.3	Light Range and Intensity Reduction.	11
4.3	Shadow Manager Auto Quality Settings	11
5	FAQ	12
5.1	Enable NGSS support	12
5.2	Adding Lights at Runtime	12
6	Support	12

CONTENTS

Thank you for purchasing *Smart Shadows*! I wish this package will meet your needs and expectations. Please do not hesitate to contact me if you have a feature request, or any question, issue or suggestion.

Arnaud,
support@wildmagegames.com

1 OVERVIEW

1.1 Introduction

Smart Shadows is a unique tool that helps you dramatically improve your game performance when using real time shadows. Contrary to the default Unity behavior that renders each shadow map, even for far and not visible lights, Smart Shadows automatically and smoothly enable pertinent shadows and disable others, improving a lot your game rendering performance.

With this tool, you will get the best of dynamic shadows, without the performance issues.

1.2 Requirements

- Unity 5.5 and later, pro or personal edition.

2 QUICK START GUIDE

- Add the *Shadow Manager* component to the gameobject from which your shadow distance will be computed. Usually it is your player or main camera. We will call this object the *ShadowOrigin*.

Tip: Note that you can slightly offset the *ShadowOrigin* in the player or camera forward direction, to favor lights in front of your player.

- Configure the *Shadow Manager* *maximum shadow distance* and *maximum simultaneous shadows*. If a light distance to the *ShadowOrigin* is greater than the *maximum shadow distance*, its shadow will be disabled. All lights are sorted by priority and only the *maximum simultaneous shadows* first ones are enabled.
- If you start your game, you will see that some shadows are automatically disabled. If you move the *ShadowOrigin*, the priority list will be updated and some shadows will be enabled and other disabled.
- To better control which shadows are enabled, please read the next sections.

Demo Scene Setup. The demo scene is best rendered using the new Unity Post Processing Stack. Set the camera as *Deferred Rendering*. Add a *Post Process Behavior*. Set the `PostProcessProfile.asset` as the behavior profile.

3 UNDERSTANDING THE LIGHT SORTING ALGORITHM

3.1 Light Priority

In order to be as predictable and configurable as possible, we use a deterministic priority sorting algorithm.

Here is how we determine if a light has priority over an other light:

1. All inactive lights are ignored.
2. All lights out of the shadow range have their shadow disabled.
3. All lights with an importance level higher than other lights have priority. See Section [4.2.1](#) for more details.
4. All lights with the same importance level but with a higher priority level value than other lights have priority.
5. All lights nearest to the *Shadow Origin* than other lights have priority.

Debug View You can visualize the current active shadows by enabling the debug view of the *Shadow Manager* (Figure 1).

3.2 Configuring your Scene

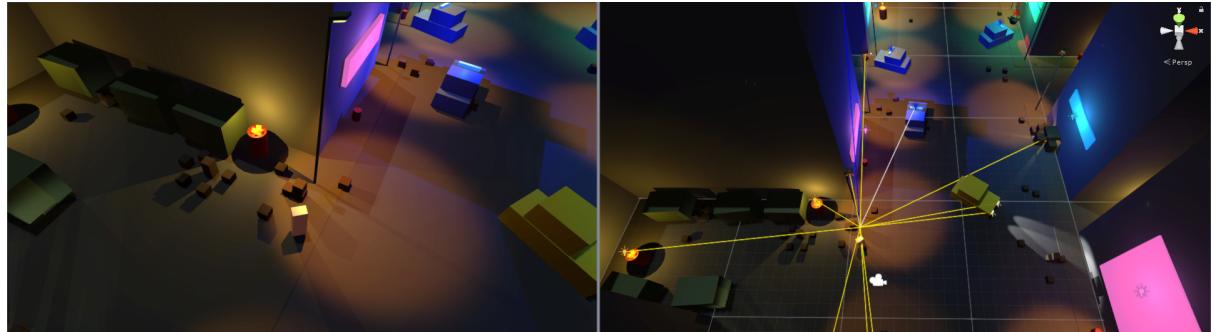
By default, the shadow manager will attribute an importance to lights based on their range and intensity. To better control the result, you can add *ShadowController* components to manually set the light importance.

Start by setting the importance level (Section [4.2.1](#)) of important lights to *HIGH*, like lights with a huge range that covers a large part of the scene. Set the importance level to *LOW* for all minor lights, like small candles and other decorative elements. Configure the *ShadowManager* to attribute the *MEDIUM* importance to all other lights.

For a finer control, change the priority level value of some lights, for instance two lights in the same room but with one being more important. For extremely important lights, like the sun, it is advised to set their importance to *UNRESTRICTED*.

If some lights should have their shadow enabled when they are far from the *Shadow Origin*, you can set a *CUSTOM Shadow Distance* (Section [4.2.2](#)) to a higher value than the default one.

To better control the scene ambiance when lights are disabled, you can enable the intensity and range light reduction, for all the lights or just some of them (Section [4.2.3](#)).



(a) Left: Game View. Right: Scene View.



(b) Left: Game View. Right: Scene View.

Figure 1: Shadows activation example. The rays point to lights with active dynamic shadows. The ray color represents the light importance: blue *UNRESTRICTED*, white *HIGH*, yellow *MEDIUM*, orange *LOW*. (a) The player is on the bottom left, only the most pertinent shadows are enabled. (b) The player moved to the center of the scene, the algorithm disabled and enabled the shadows depending on their pertinence. For instance, note how the fires shadows at the bottom left of the scene are disabled in (b) right.

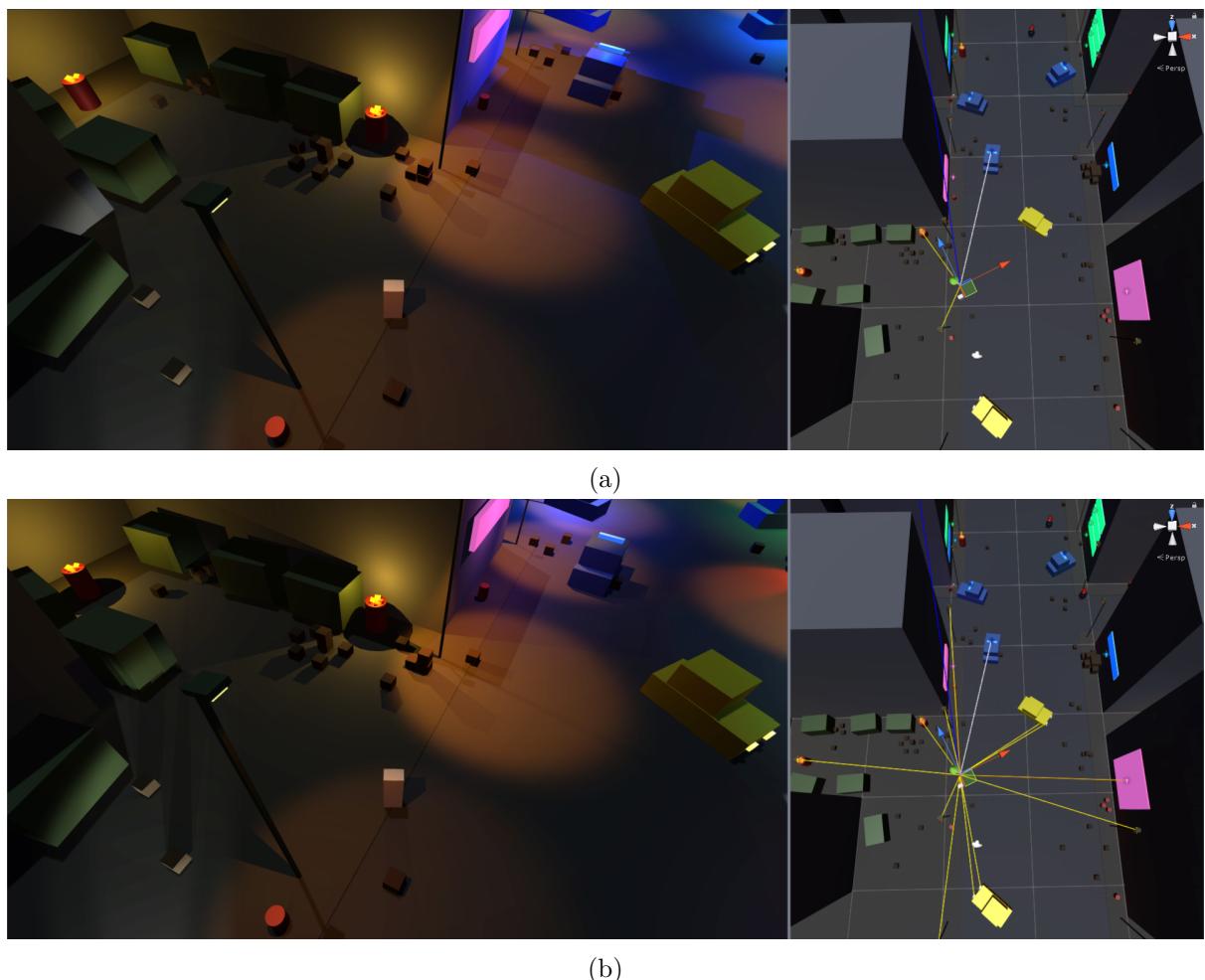


Figure 2: Shadow activation example. The maximum simultaneous shadows and shadow distance are higher in (b).

4 CONFIGURATION GUIDE

4.1 Shadow Manager

Configure the shadow manager to define the default light settings. Add a *Shadow Controller* component to a light to configure its custom settings.

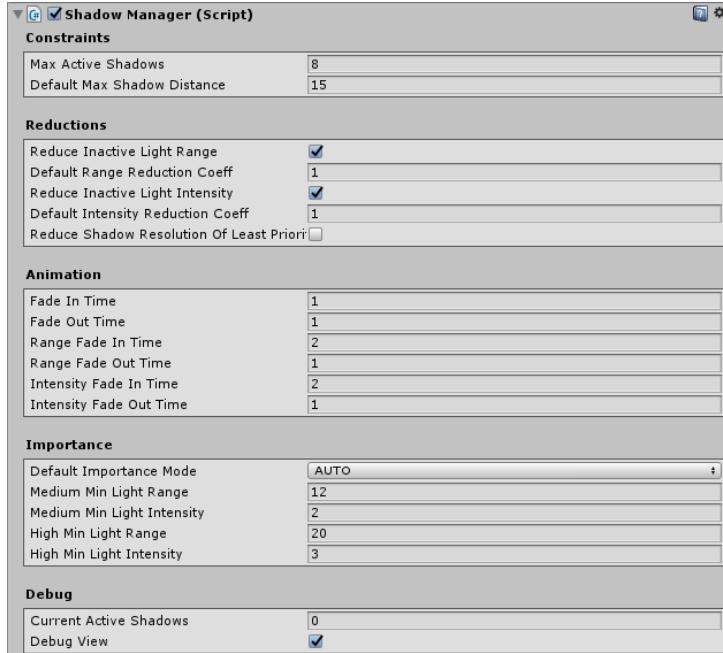


Figure 3: The Shadow Manager.

4.1.1 Maximum Simultaneous Shadows.

Defines the maximum of simultaneous active shadows. The shadows are sorted by priority, the first ones are enabled, all other are disabled.

Note that during transitions this number can temporarily be higher than this limit, while the fading of the shadows to be disabled is performed.

4.1.2 Default Maximum Distance.

Defines the default maximum distance from the *Shadow Origin* for which a shadow can be activated. See Section 4.2.2 for more details.

4.1.3 Light Range and Intensity Reduction.

When a shadow is disabled, its light range and intensity can be reduced. You can set the default reduction coefficients of the lights. See Section 4.2.3 for more details.

4.1.4 Animations.

You can set the fading time in seconds, when the shadow is enabled and disabled, for the shadow strength, and the light range and intensity.

4.1.5 Automatic Importance.

There are two default importance modes.

AUTO will attribute an default importance to the lights which does not have a custom *ShadowController* based on their range and intensity. All directional lights are set to *UNRESTRICTED*. You can configure the minimum value to be considered as *HIGH* or *MEDIUM*. All other lights will be set as *LOW*.

DEFAULT_VALUE will attribute the same default value to all lights without a custom *ShadowController*.

Add a *ShadowController* component to all lights for which you want to manually set their importance, and configure other custom settings.

4.1.6 Shadow Resolution Reduction.

If you want to have a high shadow number but still want good performances, a good way to improve the computational time is to reduce the shadow resolution of lowest priority lights.

If n shadows are enabled and you set this value to $m < n$, all shadows from m to n will have their resolution reduced by one quality level.

4.2 Shadow Controller

When a level is loaded, the manager automatically adds a *Shadow Controller* component to all scene lights. You can add that component to a light to set its custom settings.

4.2.1 Importance and Priority.

There are two priority levels:

Importance defines the coarse shadow priority. All shadows of a given importance will have priority over shadows of a lower importance.

UNRESTRICTED will be counted as an active shadow but not affected by the manager. Their shadow will never be disabled by the manager. This is recommended for extremely important scene lights, like the sun.

IGNORE will not be counted as an active shadow and not affected by the manager. This is recommended for all lights not part of the scene rendering, for instance lights with a very specific culling mask or lights used for UI rendering.

Priority defines the fine priority of the shadow. Within the same importance level, all shadow of a given priority value will have priority over shadows of a lower priority value.

4.2.2 Maximum Distance.

Distance Mode defines up to what distance to the *Shadow Origin* this shadow can be enabled. *DEFAULT* uses the manager default value. You can set the distance to be *PROPORTIONAL TO THE LIGHT RANGE* or *CUSTOM*.

4.2.3 Light Range and Intensity Reduction.

Sometimes, disabling a light's shadow can be too visible, particularly in cases where the light should cast a lot of shadows and have therefore a low contribution to the scene illumination.

For instance, the light in Figure 4b casts a lot of shadows, and have therefore have a low contribution to the scene illumination. However, when its shadow is disabled, the light illuminates a lot around itself (Figure 4a), creating a lighting inconsistency. To reduce this effect, you can activate an automatic range and/or intensity reduction on the light when its shadow is disabled. On Figure 4c, the range and intensity of the light are reduced when the shadow is disabled, making the light mean illumination of the scene closer to when its shadow is enabled.

This is especially useful when a lot of far but visible lights are out of the shadow range, and illuminate a lot the scene with their shadow disabled, while they don't when there are activated.



Figure 4: Intensity and Range reduction example.

Intensity Reduction Mode and *Range Reduction Mode* define how the light intensity and range of this shadow's light is reduced when the shadow is disabled.

IGNORE will not modify the light range and/or intensity, even if activated by the *ShadowManager*.

DEFAULT uses the manager default value.

You can set the intensity reduction to a *CUSTOM* coefficient.

4.3 Shadow Manager Auto Quality Settings

If you want quality-dependent settings, you can add the *Shadows Manager Auto Quality Settings* component to your object. All settings of the *Shadow Manager* will be overwritten by these settings, depending on the current Unity quality level.

5 FAQ

5.1 Enable NGSS support

If you are using the NGSS asset, you need to enable the support on each of your target platforms. Select the *ShadowManager* and click on *Enable* in the NGSS section.

5.2 Adding Lights at Runtime

If you add lights at runtime, for instance by instancing objects, the light will not be taken into account by the light manager. To take them into account, simply add a *Game Light* or a *Shadow Controller* component to the light when created. This can be done automatically by adding one of these component to the object prefab.

Alternatively you can call the *LightManager RegisterLight(Light light)* or *RegisterAllScene-Lights()*.

6 SUPPORT

Please do not hesitate to contact me if you have a feature request, or any question, issue or suggestion: support@wildmagegames.com.