

Éclairage et lumières

Éclairage par défaut dans Unity

Toute nouvelle scène créée dans Unity est déjà éclairée de deux façons :

- 1- Une lumière directionnelle présente dans l'onglet *hierarchy*
- 2- Un éclairage ambiant de base (fenêtre *lighting*→*settings*).

Modification de l'éclairage de base

Même en désactivant toutes les sources de lumière dans votre environnement, ce dernier reste légèrement éclairé par la lumière ambiante qui éclaire tous les objets de façon égale. Pour contrôler et modifier cet éclairage ambiant, choisir le menu **Windows**→**Rendering**→**LightingSettings**, puis l'onglet **Scene**.

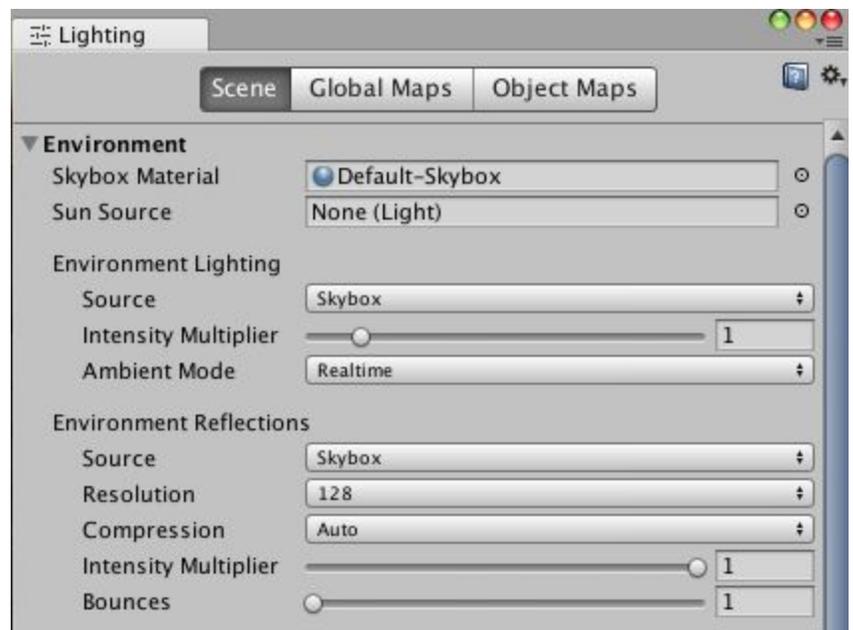
Environment

Skybox material: L'image de "ciel" dans votre environnement. Un skybox est généralement composé de 6 images (haut, bas, gauche, droite, avant, arrière) et permet de simuler un ciel. Si le skybox est supprimé, une option supplémentaire apparaîtra dans la fenêtre (*Ambient Color*) permettant de choisir la couleur de la lumière ambiante

(option non visible dans l'image de droite).

Sun Source : La lumière directionnelle utilisée par le ciel (skybox) pour émettre de la lumière. Si aucune n'est spécifiée, c'est la lumière directionnelle avec la plus haute intensité qui sera utilisée.

Note : en modifiant l'angle de la lumière qui est spécifiée comme "soleil", l'éclairage du ciel sera modifié et pourra simuler la nuit, le début de journée, la fin de journée, etc.



Environment Lighting

Source : La source lumineuse utilisée pour l'éclairage ambiant. Par défaut, il s'agit de la lumière du *skybox*, mais il est possible de choisir aussi un dégradé de couleur (*gradient*) ou une couleur unie (*color*). Notez que si *skybox* est choisi, c'est la couleur du ciel qui aura un impact sur les teintes de l'éclairage ambiant.

Intensity multiplier : L'intensité lumineuse de la lumière ambiante (entre 0 et 8, valeur par défaut de 1)

Ambient mode : Permet de déterminer si l'éclairage ambiant est intégré sur des textures (*baked*) ou calculé en

temps réel (*realtime*). Voir la section "optimisation et éclairage" de ce document pour plus de détails.

Environment Lighting : Les paramètres de cette section permettent de contrôler les réflexions de lumière.

Pour ne pas avoir de ciel (skybox)

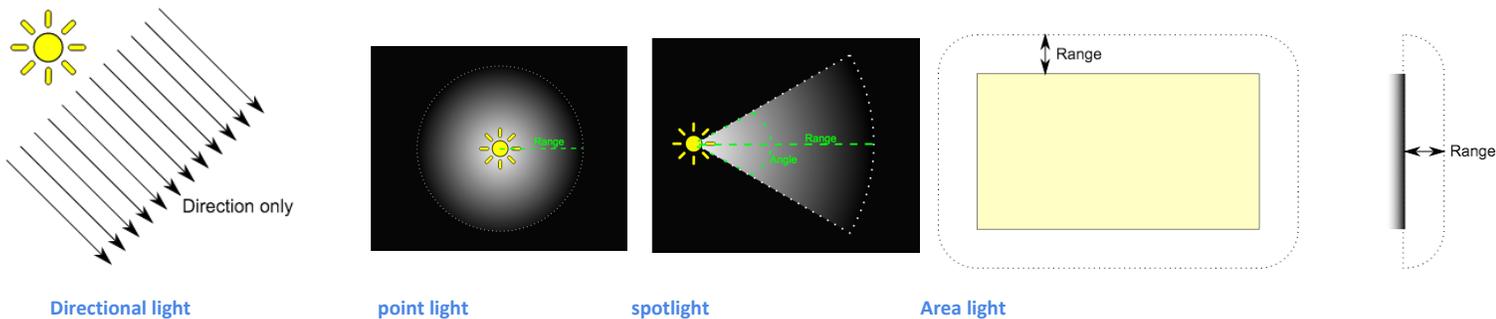
Si votre jeu ne nécessite pas de skybox, vous pouvez le supprimer dans la fenêtre lighting. Pour contrôler la couleur unie d'arrière-plan, il faudra alors modifier la propriété "background" de vos caméras.

Vous pouvez également conserver le *skybox* (ex.: pour une caméra d'extérieur) et utiliser une couleur unie pour une autre caméra (ex.: pour une caméra intérieure). La propriété de caméra "Clear flags" permet de choisir "skybox" ou "solid color".



Les différents types de lumières

Plusieurs types de lumières peuvent être créées dans Unity : lumière directionnelle (*directional light*), point de lumière (*point light*), lumière "spot" (*spotlight*) et région de lumière (*area light*)



Lumière directionnelle (*Directional light*)

Semblable au soleil. La position ou la mise à l'échelle n'ont pas d'influence sur une lumière directionnelle, uniquement l'orientation.

Point de lumière (*point light*)

Semblable à une ampoule. La rotation ou mise à l'échelle n'ont pas d'influence sur un point de lumière, uniquement la position.

Lumière spot (*spotlight*)

Projecteur de lumière. Projette la lumière dans une direction (Z+). La mise à l'échelle n'a pas d'influence sur une lumière directionnelle.

Région de lumière (*area light*) - Fonctionne seulement en mode "*baked*", donc sur les objets statiques

Région d'éclairage rectangulaire qui émet de la lumière seulement d'un côté du rectangle.

Différentes propriétés des lumières

Type	Le type de lumière (<i>Directional, Point, Spot</i> et <i>Area</i>)
Baking	Permet de choisir si les effets de la lumière seront intégrés à des textures (baked), modifiés en temps réel (realtime) ou mixés (mixed). Si l'option "mixed" est choisie, les effets de lumières seront intégrés sur des textures pour les objets statiques et calculés en temps réel pour les objets non-static. *Voir la section optimisation et l'éclairage pour plus de détails.
Range	La portée de la lumière (<i>Point</i> et <i>Spot</i> seulement).
Spot Angle	Détermine l'angle (en degrés) d'ouverture du <i>spotlight</i>
Color	La couleur de la lumière émise
Intensity	La luminosité de la lumière.
Bounce Intensity	Permet de faire varier l'intensité de la lumière indirecte (<i>baked</i> seulement)
Shadow Type	Détermine le type d'ombres générées par la lumière (<i>Hard Shadows, Soft Shadows</i> ou <i>no shadows</i>) <i>Hard shadows</i> assure un rendu plus rapide, mais moins réaliste.
Baked Shadow Radius	Permet de contrôler l'adoucissement du bord des ombres (<i>baked</i> seulement)

Baked Shadow Angle	Permet de contrôler l'adoucissement du bord des ombres (<i>baked</i> seulement)
Draw Halo	Génère un halo de lumière sphérique autour de la lumière. Le rayon du halo est égal à la portée de la lumière (<i>range</i>).
Flare	Permet de spécifier l'image de reflet (<i>flare</i>) qui sera rendu à la position de la lumière. 
Render Mode	L'importance de la lumière dans le rendu de la scène.
Culling Mask	Permet d'inclure ou d'exclure les effets de la lumière sur les objets qui se trouvent sur certains <i>layers</i> .

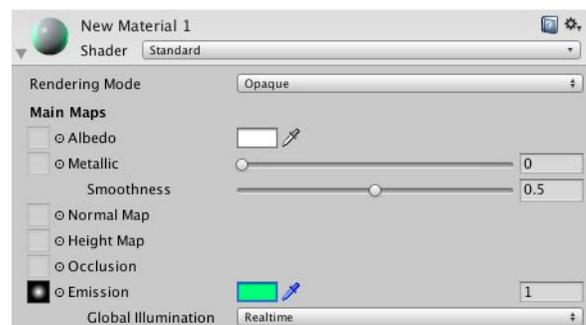
Éclairage provenant d'un objet

En plus des différentes lumières mentionnées précédemment, il est possible d'ajuster les propriétés du matériel d'un objet pour qu'il émette de la lumière :

Noir = aucune émission

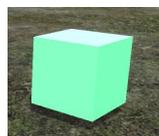
Blanc = émission maximale

Autre couleur : lumière émise teintée de cette couleur. Le niveau de gris dans la couleur déterminera son intensité.



Il est également possible de glisser une texture dans le carré à gauche de la propriété "Emission". Les parties noires de cette texture bloqueront la lumière émise et les parties blanches laisseront passer la lumière. Les zones grises bloqueront partiellement la lumière émise.

Exemple : Dans cet exemple, le cube émet de la lumière verte. L'image de gauche nous montre le résultat sans avoir spécifié de texture (masque) alors que l'image de droite nous montre l'effet avec un masque.



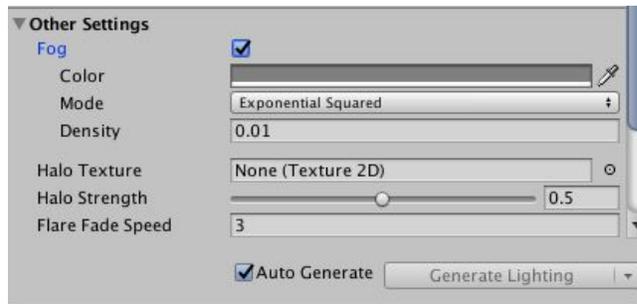
sans masque



avec masque

Effet de brouillard (fog)

Depuis Unity 5, l'utilisation de brouillard (fog) fait maintenant partie des paramètres d'éclairage. Pour activer le brouillard, vous devez ouvrir la fenêtre *lighting-->settings* et cochez la case *Fog*. Vous pourrez alors choisir la couleur et le mode de brouillard souhaité.



Mode linéaire : Vous pouvez contrôler le moment où commence la zone de brouillard et où elle se termine. Au point de départ (*start*), le brouillard sera de très faible intensité et au point de fin (*end*) il sera à l'intensité maximale. Conséquemment, plus l'écart entre ces deux valeurs sera petit, plus la zone de brouillard sera dense et évidente. **Notez que les distances spécifiées sont par rapport à la caméra active.**

Mode exponentiel et exponential squared : Vous pouvez contrôler la densité du brouillard

Optimisation et éclairage

Voici quelques trucs qui vous permettront d'optimiser vos jeux :

Objets statiques et non-statiques

Par défaut, les objets créés et importés dans Unity (sauf les terrains) sont non-statiques, c'est-à-dire que Unity considère qu'ils peuvent bouger pendant l'exécution du jeu. Cela empêche le moteur de jeu de procéder à plusieurs techniques d'optimisation, notamment au niveau de l'éclairage (*lightmapping*) et du rendu des objets (*Draw Call Batching*).



Prendre l'habitude d'identifier les objets statiques

Pour optimiser vos projets, prenez l'habitude de définir les objets statiques qui sont généralement les objets de décors qui ne bougeront pas.

Choisissez judicieusement la bonne valeur de "baking" pour toutes vos lumières :

Realtime : Pour les lumières qui doivent avoir un impact sur les objets animés de votre jeu (personnage, ennemis, etc.). L'effet de ces lumières est calculé en temps réel. C'est moins optimal, mais nécessaire pour le réalisme du jeu.



Baked : Pour les lumières qui éclairent le décor. Seuls les objets statiques seront affectés par ces lumières. Avant la lecture du jeu, Unity intégrera l'effet de ces lumières sur des textures (*lightmap*), ce qui veut dire qu'aucun calcul ne sera nécessaire pendant la lecture du jeu. C'est plus optimal, mais souvenez-vous que les objets non-statiques ne seront pas affectés par ces lumières.

Mixed : C'est le compromis entre les deux premières options. Les effets de lumières seront intégrés sur des textures pour les objets statiques et calculés en temps réels pour les objets non-statiques.