



Introduction à Unity3D

Damien Marchal



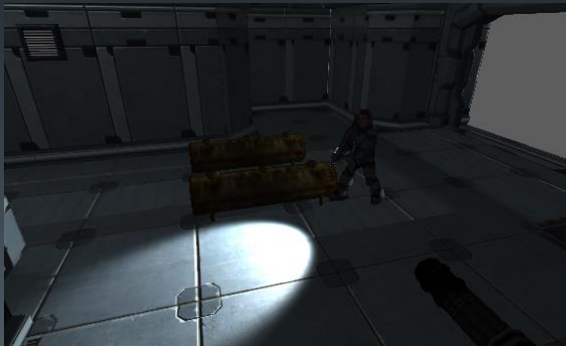
Rappel

- Previously...

Mini projet

« Sur LV-426, *Ripley*, cherche des objets pouvant lui être utile. »

- Faire une ébauche de jeu comprenant:
 - Un niveau avec,
 - Une porte « verrouillée » ,
 - Un personnage se déplaçant au clavier,
 - Des objets à ramasser par un clic souris lorsque le personnage est à proximité,
 - Lorsque tous les objets ont été ramassés la porte est déverrouillée.



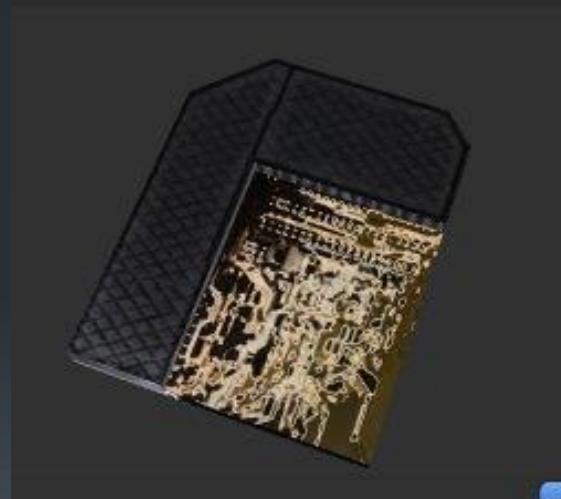


TP 2 et 3

- Créez un nouveau projet dans Unity
- Downloadez les ressources:
<http://www.lifl.fr/~marchal/rvi/2013/tp2.unitypackage>

Partie 1 – Objets à ramasser

- Dans une nouvelle scène ajoutez l'objet à ramasser.
(géométrie dans Models/Other/key)
- Ajoutez lui le script de la semaine dernière pour le faire tourner.
- Ajoutez lui un tag: « Object »



C'est quoi un tag ? =>

<http://docs.unity3d.com/Documentation/Components/Tags.html>

Partie 1 – Objets à ramasser

Les GameObjects d'une scène peuvent être convertis en élément préfabriqués (Prefab). On peut ainsi les réutiliser dans les différentes parties du jeu sans avoir à reconstruire l'assemblage à chaque fois.

Pour créer un prefab à partir d'un GameObject il suffit de glisser celui-ci dans l'onglet **Project**.

- Créez un prefab de cet objet.



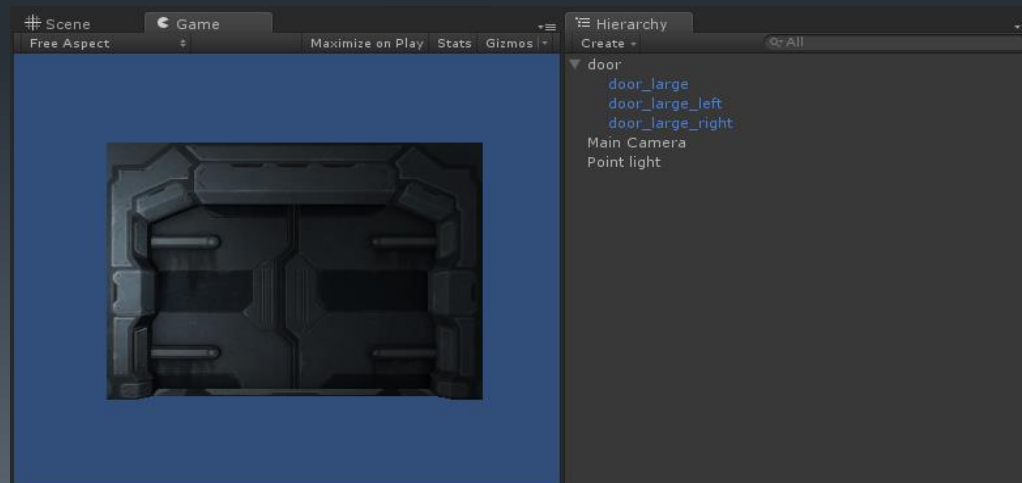


Fin partie 1

Sauvegardez l'état courant de votre projet dans une scène Unity nommée scene3.

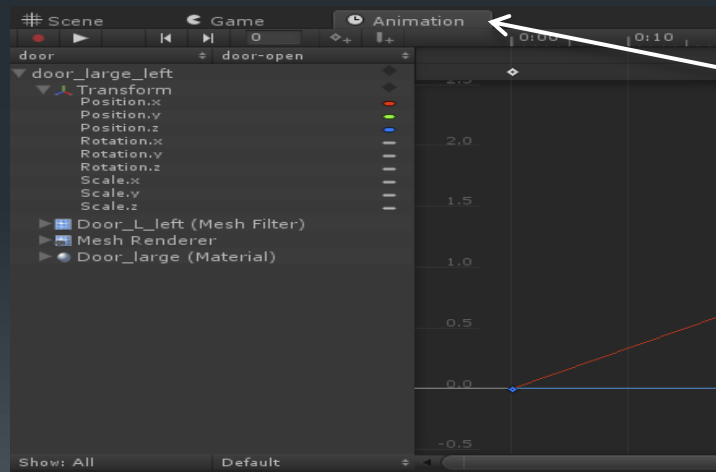
Partie 2 – Animation de la porte

- Dans une nouvelle scène,
- Cherchez les modèles « door_large », « door_large-left » et « door_large_right »,
- Pour chaque modèle, corrigez l'orientation par défaut,
- Assemblez le tout pour faire une porte coulissante.



Partie 2 – Animation de la porte

Il faut maintenant créer les animations par images clefs.

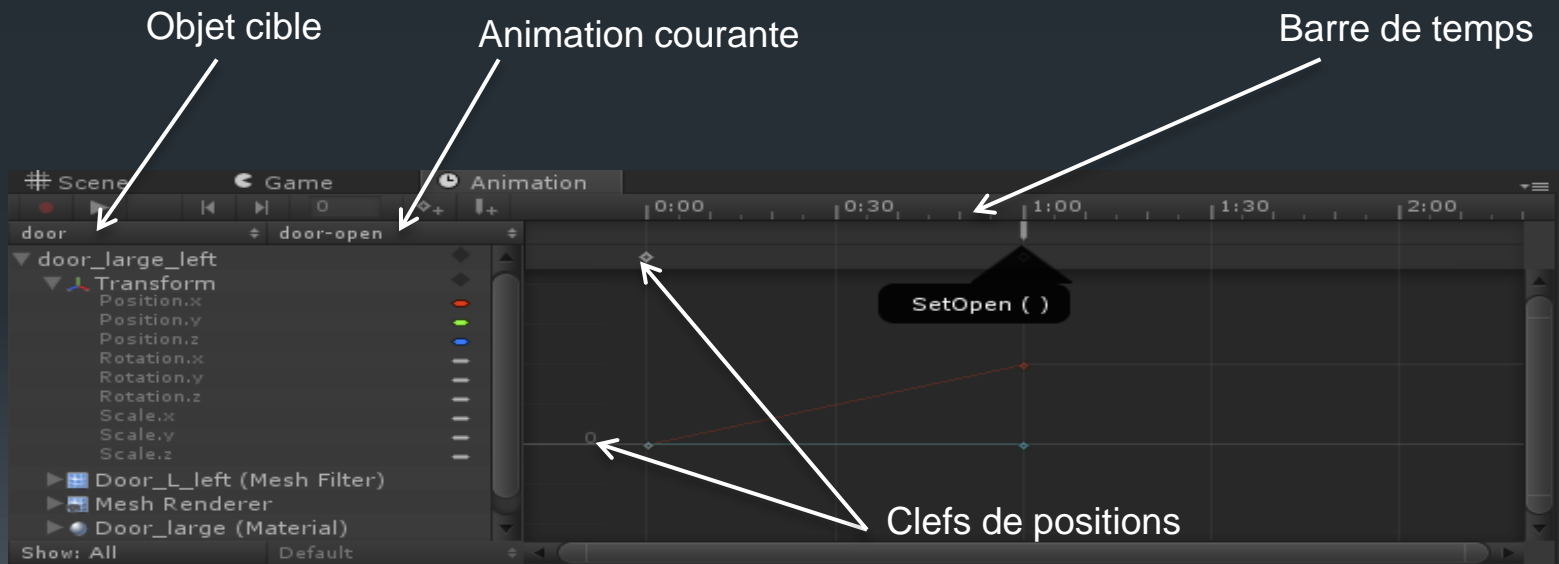


- Ouvrez le panneau Animation (dans les onglets ou le menu Windows).

Plus d'infos:

<http://docs.unity3d.com/Documentation/Components/AnimationEditorGuide.html>

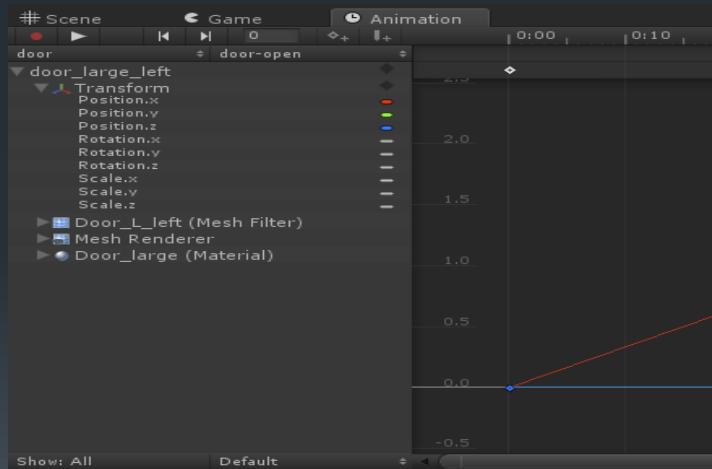
Partie 2 – Animation de la porte



Plus d'infos:

<http://docs.unity3d.com/Documentation/Components/animeditor-UsingAnimationEditor.html>

Partie 2 – Animation de la porte



- Ajouter un composant Animation ([Components/Misc/Animation])
- Créez deux animations:
« door-open » pour son ouverture,
« door-close » pour sa fermeture.
- Ajoutez les clefs de positions.

Plus d'infos:

<http://docs.unity3d.com/Documentation/Components/AnimationEditorGuide.html>



Partie 2 - Animation de la porte

Implémentez le script qui contrôle l'ouverture de la porte en fonction des clics souris.

Vous pourriez avoir besoin des classes ou méthodes suivantes:

`Animation.Play()`,
`animation[« nomdel'anim »].speed`,
`animation[« nomdel'anim »].time`,

Et aussi

`Input.GetButtonDown(0)` ou `OnMouseDown()`

Partie 2 - Animation de la porte



Pour détecter les clics il faut ajouter un Composant Collider à la porte.

Plus d'infos:

<http://docs.unity3d.com/Documentation/Manual/Scripting.html>



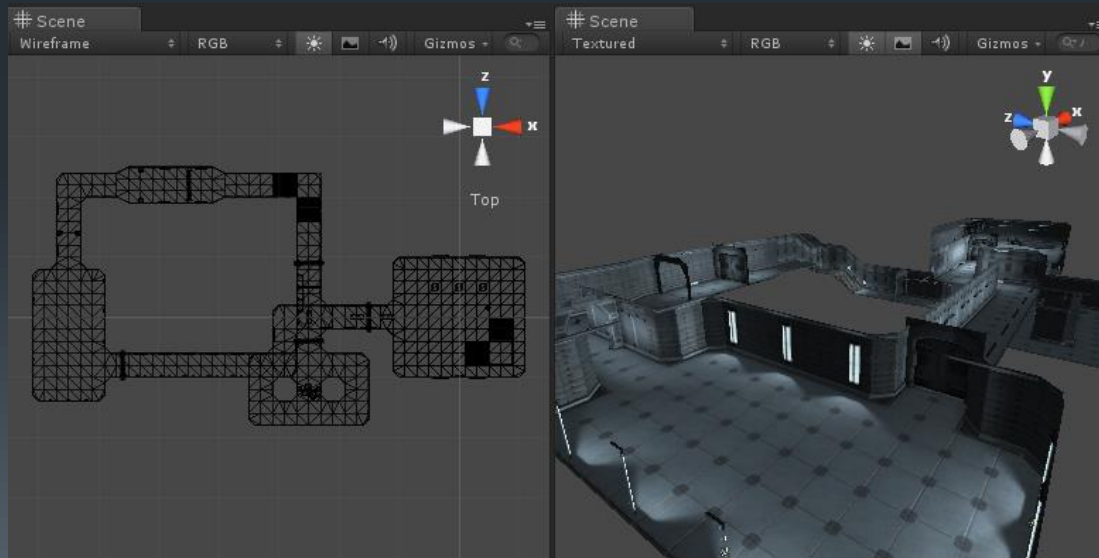
Fin partie 2

Faites un prefab de la porte.

Sauvegardez l'état courant de votre projet dans une scène Unity nommée scene2.

Partie 3 – Modélisation de la scène

- Faire un niveau composé d'éléments préfabriqués: murs, couloir, éléments de décors, lumières et caméra et la porte.



Partie 3 – Modélisation de la scène





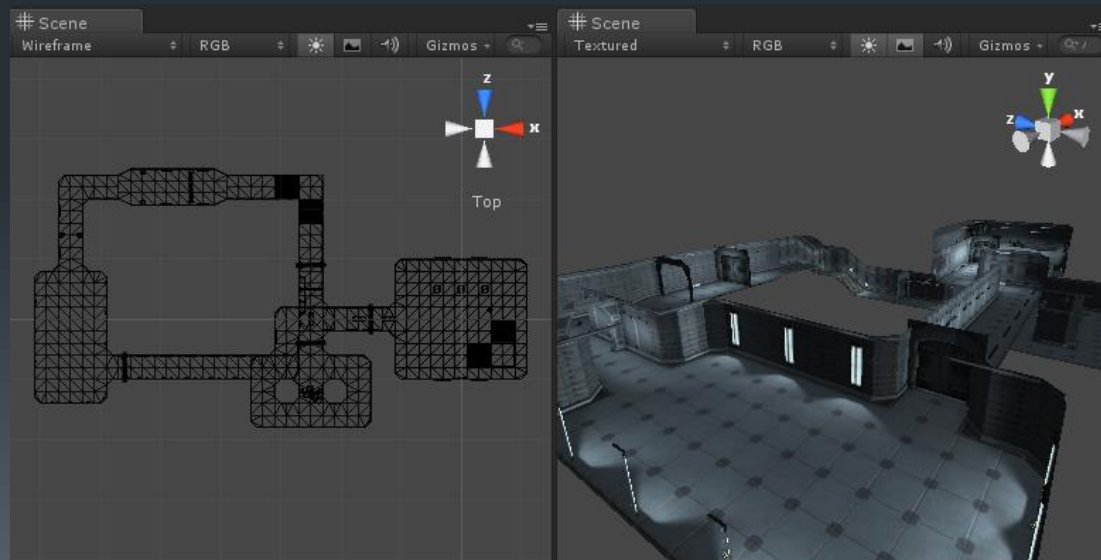
Partie 3 – Modélisation de la scène

- Pour l'assemblage fin des objets vous pourriez utiliser:
 - F: focus sur l'objet courant,
 - Snapping sur une grille: [CTRL+LMB],
 - Snapping par vertex:
 - Passer en mode translation,
 - Appuyez sur V pour sélectionner le point de l'objet à snapper,
 - Puis déplacer par [LMB].

Tips: dans l'éditeur, Unity affiche uniquement un seul coté des triangles.

Partie 3 – Modélisation de la scène

- Ajoutez dans ce décor 5 des objets à ramasser.





Fin partie 3

Sauvegardez l'état courant de votre projet dans une scène Unity nommée scene3.

Partie 4 - Le héros

- Dans votre scène ajoutez un soldat:
(depuis Models/Soldier/3D/soldier)
- Ajoutez lui un composant Animation [Component/Misc/Animation].
- Ajoutez au composant animation les animations: soldierIdleRelaxed, soldierWalk, soldierSprint
(depuis Models/Soldier/Animations)
- Ajoutez au héros un Tag: « Hero »





Partie 4 - Le héros

Implémentez le script qui contrôle le personnage.

Le personnage se déplace lors des appuis sur les touches

UpArrow, DownArrow, LeftArrow, RightArrow

Vous pourriez avoir besoin des classes ou méthodes suivantes:

- `Input.GetKeyDown(KeyCode.*), Input.GetKeyUp(KeyCode.*),...`
- `Transform.Translate(...), Transform.Rotate(...)`
- `Animation.CrossFade(),`
- `animation[« nomdel'anim »].speed, animation[« nomdel'anim »].time,`



Partie 4 - Le héros

Pour l'instant votre héros passe à travers les murs. Il faut rajouter la gestion des collisions.

Pour ajouter la collision aux objets il faut leur rajouter des composants « Collider ».

Pour ajouter la collision à un personnage il faut rajouter un composant `CharacterController`.

- Modifiez votre script pour utiliser le composant `CharacterController` lors des déplacements de votre personnage.

Vous pourriez avoir besoin des classes ou méthodes suivantes:

- `GetComponent<CharacterController>()`,
- `Move(...)`



Partie 4 - Le héros

Lorsque le joueur clique sur un objet, le personnage le ramasse si celui-ci est à proximité.

Un objet ramassé est supprimé de la scène.

Vous pourriez avoir besoin de:

- Gestion des clics: `Ray`, `RayHit`, `Physics.Raycast()`, `Camera.ScreenPointToRay(...)`, `Vector3.magnitude`, `Destroy(...)`, `Input.mousePosition`,
- Pour supprimer un objet: `Destroy()` et `DestroyImmediate()`



Fin partie 4

Sauvegardez l'état courant de votre projet dans une scène Unity nommée scene4.

Partie 5 – Déverrouillage de la porte

- Ajoutez un script qui déverrouille la porte lorsque tous les objets ont été trouvés.

Vous pourriez avoir besoin:
`GameObject.Find()` ou
`GameObject.FindWithTag()`

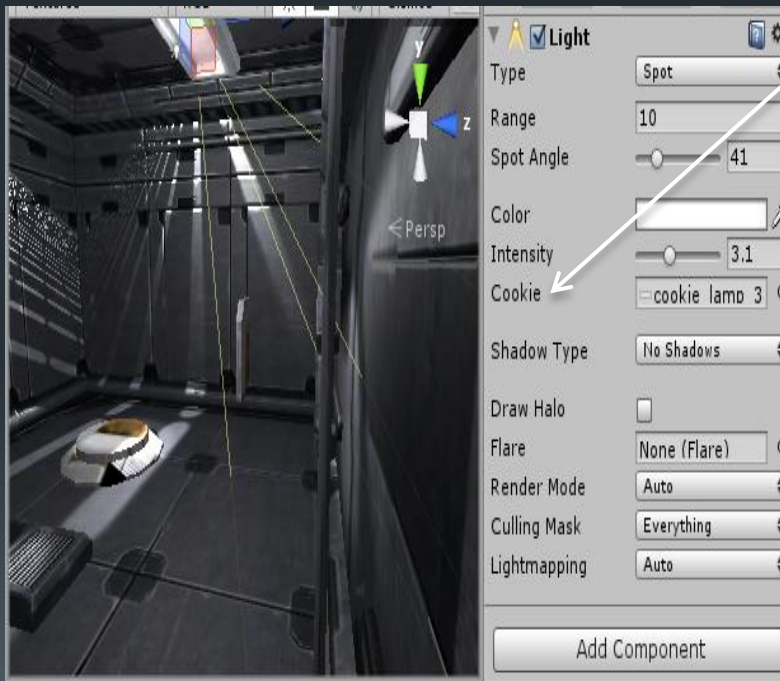




Fin partie 5

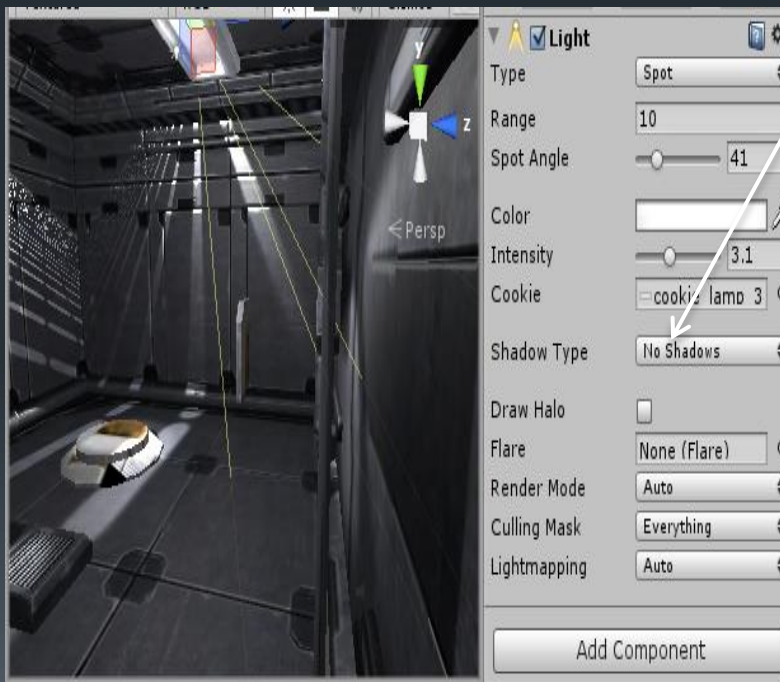
Sauvegardez l'état courant de votre projet dans une scène Unity nommée scene5.

Extra - Les lumières



- Pour améliorer l'éclairage on peut utiliser des masques de lumières.
- Pour les SpotLights ce sont des textures en niveau de gris.
- Pour les PointLights la texture doit être de type cubemap.

Extra - Les lumières



- On peut aussi utiliser des ombres.
- Pour avoir des ombres avec tous les types de lampes il faut activer le mode de rendu Deferred Lighting voir le composant Camera.

Extra - Systèmes de particules



- Vous pouvez aussi rajouter un système de particule [Component/Effect/Particle System]
- Voir aussi les exemples dans le répertoire standard assets.

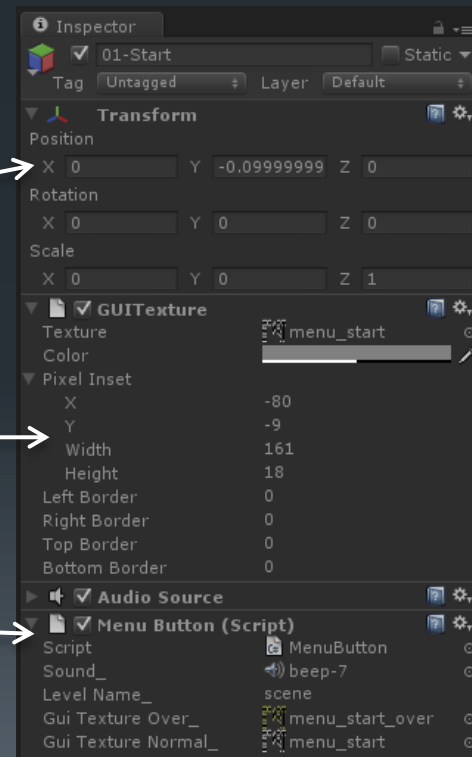
Extra : Menu

- Créez une nouvelle scène et faites un menu qui ressemble à:



Extra : Menu

- Utilisez les objets de type `GUITextures` pour afficher une image 2D au dessus de la scène.
- Les coordonnées des `GUITexture` sont comprises entre 0.0 et 1.0 correspondant à la gauche et la droite (resp. bas/haut) du viewport.
- Pour centrer la `Texture2D` dans le `GUITexture`. Il faut utiliser les paramètres de `Pixel Inset`.
- Il faut un script pour l'interaction.
Voir: `Application.LoadLevel(...)`





Extra – Camera à la Resident Evil

- Utiliser plusieurs caméra et changer de caméra en fonction de la position du personnage,

Vous pourriez avoir besoin de:

`Camera.main`, `Camera.current`, `Camera.enabled`



Extra - Scripts

- Afficher le nom des objets lorsque le personnage s'en approche.

Pour afficher du texte vous pouvez utiliser un objet `GUIText`.

Vous pourriez avoir besoin de:

`Instantiate`, `Destroy`,

`GameObject.name`,

`Camera.WorldToViewportPoint(...)`



Extra - Scripts

- Tant que la mission n'est pas finie on affiche le temps passé,

Vous pourriez avoir besoin de:

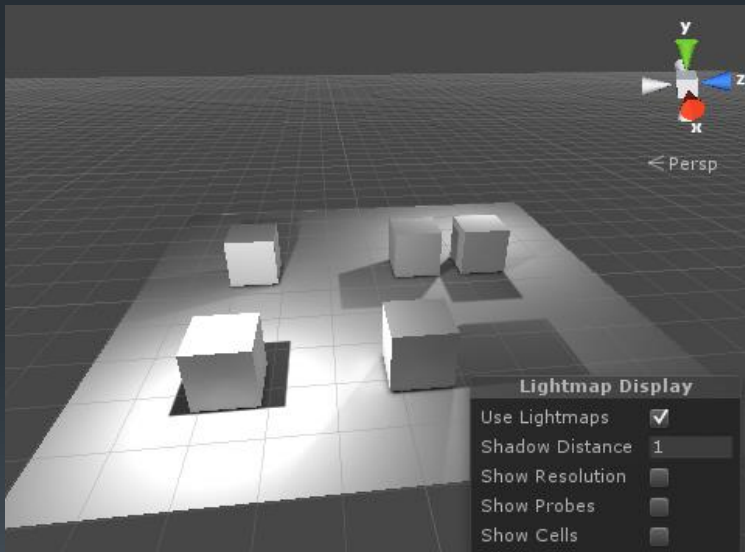
`Timer.time, GuiText.text, Application.LoadLevel(...)`

Extra - L'ambiance: Lumières



- Dans la version Pro on peut aussi rajouter des filtres de post-process qui s'appliquent à toute l'image.
- Voir les exemples dans le répertoire standard assets (il faut importer le package Image Effects).
[Assets/Import...]

Extra - L'ambiance: Lumières



- Avec la version Pro d'Unity il est possible de précalculer les carte d'éclairements.
- Pour ça il faut rajouter l'onglet [Lightmapping].

Plus d'infos:

<http://docs.unity3d.com/Documentation/Manual/Lightmapping.html>