

Olivier **Saraja**



La 3D libre  
avec  
**Blender**

2<sup>e</sup> édition

© Groupe Eyrolles, 2006, 2007,

ISBN : 978-2-212-12196-4

**EYROLLES**

---

annexe

C



# Les scripts Python

Blender est livré en standard avec un certain nombre de scripts Python, permettant de rendre des services dans des domaines très variés, à l'instar des plugins d'autres applications commerciales, comme 3ds max ou Maya. La différence est que ces scripts sont livrés en standard avec Blender. Ils sont accessibles via le menu de n'importe quelle fenêtre de type *Scripts Window*. En cliquant sur le menu *Scripts*, vous affichez une liste déroulante de catégories, que vous pouvez à leur tour dérouler pour sélectionner un script. Cette annexe n'a pas pour intention d'expliquer l'usage de chacun de ces scripts, mais simplement de vous indiquer leur usage principal. Les catégories actuellement vides ne sont pas abordées. Les scripts livrés en standard avec Blender ne nécessitent pas l'installation de Python sur votre ordinateur. En revanche, certains scripts que l'on peut trouver sur Internet nécessitent que la version complète de Python soit installée (voir les paquetages proposés pour votre système d'exploitation dans le répertoire `/install` du CD-Rom d'accompagnement). Lorsque ce sera le cas, un message d'erreur dans la console ou la fenêtre de commandes MS-DOS vous en informera, et l'installation de Python sera alors indispensable pour faire usage du script en question.

## Image

Ces scripts peuvent être activés dans le menu *Scripts* de la *Scripts Window*, ou dans le menu *Image* de la vue *UV/Image Editor*.

*Fix Broken Paths* : ce script détecte les images dont le chemin ne pointe pas vers une image existante, et transforme les chemins des images en chemins relatifs lorsque cela est possible pour favoriser la migration et les échanges de fichiers entre utilisateurs de Blender.

*Edit externally* : ce script permet de lancer une application tierce pour éditer l'image présente dans l'UV/Image Editor. Au premier lancement, le script propose une application par défaut ; si cela ne fonctionne pas, vous pourrez choisir l'application de votre choix en saisissant son chemin complet, qui sera sauvegardé pour de futurs usages. À noter que pour Windows et Mac OS, les images seront automatiquement ouvertes avec les applications liées à l'extension de l'image.

*Consolidate into one Image* : ce script permet de compiler toutes les images d'un objet au sein d'une seule image, en réassignant bien sûr les coordonnées UV de la nouvelle image aux sommets de l'objet.

*Billboard Render on Active* : ce script permet de générer un rendu d'un objet et de s'en servir comme texture sur un maillage plus simple, comme un quadrangle (*billboard*). Pour s'en servir, il est important de sélectionner l'objet, la ou les lumières l'éclairant et enfin le billboard lui-même, de sorte que ce dernier soit l'objet actif. Cela permet d'afficher simplement des objets dont la modélisation et le rendu sont autrement complexes.

## UV

Ces scripts peuvent être activés dans le menu *Scripts* de la *Scripts Window*, ou dans le menu *UVs* de l'UV/Image Editor.

*Save UV Face Layout* : ce script permet d'enregistrer au format TGA la carte UV dépliée d'un objet, afin de permettre de peindre par-dessus dans un logiciel externe, comme Gimp.

*Seams from Islands* : permet de placer automatiquement des coutures autour des îlots isolés, en vue de leur dépliage ultérieur.

*Lightmap UV Pack* : ce script est accessible en mode *UV/Face Select* lorsque vous utilisez la touche *[U]* pour afficher le menu de calcul des coordonnées UV. Il permet de compacter les coordonnées UV pour limiter les espaces inutiles dans les textures. Optionnellement, il permet de créer une nouvelle image et un nouvel UV Layer par objet sélectionné. Cette

fonction s'annonce pratique lors du *baking* de cartes de luminosité (éclairage précalculé et stocké dans les textures UV).

## System

*Interactive Console* : il s'agit d'une console interactive, similaire à l'interpréteur Python en ligne de commande. Intégrée à Blender, cette console donne accès à tous les modules Python de Blender.

*Scripts Config Editor* : ce script peut être utilisé pour visualiser et éditer les données de configuration stockées par les autres scripts.

## Object

Ces scripts peuvent être activés dans le menu *Scripts* de la *Scripts Window*, ou dans le menu *Object>Scripts* de la vue 3D courante.

*Apply Deformation* : ce script permet de créer des copies d'objets en appliquant de façon définitive l'état de déformation de l'objet au moment où le script est lancé. En effet, Blender peut déformer de façon interactive les objets grâce à des armatures ou des treillis, par exemple. Grâce à ce script, les données descriptives de l'objet sont figées en l'état.

*Axis Orientation Copy* : ce script copie l'orientation des axes (X, Y et Z) de l'objet actif sur tous les objets sélectionnés. Il est utile pour aligner les orientations de tous les maillages structures, comme l'armature d'un personnage, par exemple.

*Batch Object Name Edit* : ce script permet de changer en une seule fois le nom de plusieurs objets de Blender en même temps. Il fournit des options pour remplacer le texte dans les noms courants, tronquer le début ou la fin du nom, ou ajouter des préfixes ou des suffixes à ceux-ci.

*Bone Weight Copy* : ce script permet de copier le poids (*Weight*) des os (*Bones*) d'une armature depuis un maillage source vers un ou plusieurs maillages distincts, en fonction de la distance séparant ces maillages du maillage source. Cela permet, par exemple, de copier l'influence de l'armature d'un personnage nu, et de « distribuer » celle-ci aux vêtements modélisés par-dessus. Ainsi, lorsque l'armature est animée, le personnage et ses vêtements le sont aussi.

*Cookie Cut from View* : ce script est une sorte d'outil couteau avancé. Pour fonctionner, il nécessite deux maillages. Le premier est le maillage à découper ; le second est en fait une boucle d'arêtes, sans facettes !

---

Placez-vous dans la vue de votre choix, et la forme de la boucle d'arêtes sera découpée dans le premier maillage.

*Data Copier* : ce script permet de copier les attributs de l'objet actif vers les objets sélectionnés du même type. Bien qu'à un stade peu avancé de développement, ce script est déjà très utile pour la duplication des attributs d'objets de type *Lamp* ou *Camera*. Plus d'informations sur le site de l'auteur : [http://jmsoler.free.fr/didacticiel/blender/tutor/cpl\\_lampdatacopier.htm](http://jmsoler.free.fr/didacticiel/blender/tutor/cpl_lampdatacopier.htm).

*Drop Onto Ground* : cet outil sympathique permet de faire chuter des objets sur un sol prédéfini, ce qui le rend utile pour planter des arbres ou disposer des cailloux sur un paysage, par exemple. Le sol doit être placé sous les autres objets (du point de vue de la coordonnée *Z*) et être le dernier objet actif de la sélection.

*Find by Data Use* : ce script permet de trouver/sélectionner un objet de la scène en fonction du nom connu d'un bloc de données auquel il serait lié (par exemple, le nom de son *Material*).

*Object Name Editor* : ce script permet de renommer un groupe d'objets sélectionnés selon des règles définissables.

*Randomize Loc Size Rot* : ce script permet de rendre aléatoires la position, la taille et la rotation des objets sélectionnés. Il est possible de spécifier le taux de changement aléatoire, ainsi que les axes (*X*, *Y*, *Z*) affectés, avec possibilité d'imposer un taux identique sur les trois axes.

*Selection to Dupli Group* : transforme la sélection en un groupe, sauf si ce groupe est lié ; les opérations de transformation, en particulier les déplacements, rotations et redimensionnements, s'appliquent au groupe entier. En fait, les plus curieux remarqueront que le script crée une scène supplémentaire pour y coller les données géométriques des objets d'origine.

*UVCopy from Active* : ce script permet simplement de copier les coordonnées UV d'un maillage sur un maillage identique.

## Mesh

Ces scripts peuvent être activés dans le menu *Scripts* de la *Scripts Window*, ou dans le menu *Mesh>Scripts* de la vue 3D courante.

*Bevel Center* : ce script permet de chanfreiner ou de rayonner des arêtes. Il est plus puissant que la fonction *Bevel* classique (accessible grâce à la touche *[W]* en mode *Edit*), en ce sens où il fonctionne uniquement sur les arêtes sélectionnées, au lieu d'affecter le maillage entier.

*Clean meshes* : ce script permet de nettoyer les maillages, en éliminant par exemple les sommets ou les arêtes isolés, ou encore les facettes n'atteignant pas une surface suffisante.

*Discombobulator* : ce script permet d'habiller et de détailler la surface d'un objet par création de protubérances ou d'ajout de sous-géométrie simple. Ce script est idéal pour l'habillage de murs de science-fiction ou de coques de vaisseaux spatiaux, entre autres choses. Des informations concernant son usage sont disponibles sur le site de l'auteur : [http://www.nccn.net/~w\\_rosky/evan/evan/programs/discombobulator/tutorial.html](http://www.nccn.net/~w_rosky/evan/evan/programs/discombobulator/tutorial.html).

*Edges to Curve* : ce script permet de transformer des boucles d'arêtes (*edge-loops*) en polygones (courbes de Bézier), qu'elles soient fermées ou ouvertes.

*Mirror Vertex Locations & Weight* : ce script permet de rendre symétrique la distribution des sommets et des poids (*weight*) de ceux-ci, depuis un maillage initialement symétrique, mais qui a été retouché manuellement.

*Poly Reducer* : ce script permet de simplifier le maillage d'un objet ; il permet de l'alléger, tout en préservant à la fois la forme de base de l'objet initial, ses textures UV et ses poids (*weight*).

*Rvk1 to Rvk2* : ce script permet de transférer les clés relatives de sommet (*Relative Vertex Keys*, ou *Shape Keys*) d'un objet à l'autre, pour peu que les deux objets possèdent le même nombre de sommets. Plus d'informations sur ce script sur la page de l'auteur : [http://jmsoler.free.fr/didacticiel/blender/tutor/cpl\\_rvk1versrvk2.htm](http://jmsoler.free.fr/didacticiel/blender/tutor/cpl_rvk1versrvk2.htm).

*Solid Wireframe* : ce script fait une copie du maillage de l'objet, et lui donne une certaine épaisseur.

*Solidify Selection* : ce script permet de dupliquer le maillage et de le décaler dans une direction ou l'autre, selon les normales, afin de donner de l'épaisseur à un maillage. Facilite la modélisation de verres, de boîtes en carton ou de récipients en général.

*Unfold* : ce script permet de déplier un maillage, comme s'il était en papier, et de le mettre totalement à plat. Il permet également d'animer le dépliage grâce à un objet de type Empty. Optionnellement, il permet également d'enregistrer les coordonnées UV du maillage déplié. Non seulement le script tiendra compte des coutures (*Seams*) présentes, mais il est également possible de guider le dépliage grâce au *Vertex painting*. Plus d'informations sur ce script très intéressant à cette adresse : <http://celeriace.net/unfolder/>.

*Unweld vertex/ices* : permet de disjoindre des facettes au sommet de sélection, permettant ainsi de créer des déchirures ou des ouvertures dans un maillage sans avoir à supprimer des facettes et à reconstruire la

---

topologie. Offre plusieurs variantes intéressantes. Des exemples d'usage sont proposés sur le site de l'auteur : [http://jmsoler.free.fr/didacticiel/blender/tutor/cpl\\_faces2vertex.htm#exemple](http://jmsoler.free.fr/didacticiel/blender/tutor/cpl_faces2vertex.htm#exemple).

## Import

Grâce à ses différents filtres d'import, Blender est capable de lire les formats de fichiers suivants, issus d'autres applications graphiques, 2D ou 3D :

- 3D Studio (.3ds)
- AC3D (.ac)
- COLLADA 1.3.1 (.dae)
- COLLADA 1.4 (.dae)
- DEC Object File Format (.off)
- DirectX (.x)
- Drawing eXchange Format (.dxf)
- Lightwave (.lwo)
- Load MDD to Mesh RVKs
- MD2 (.md2)
- Motion Capture (.bvh)
- OpenFlight (.flt)
- Paths (.svg, .ps, .eps, .ai, Gimp)
- Pro Engineer (.slp)
- Raw Faces (.raw)
- Stanford PLY (.ply)
- Wavefront (.obj)

## Export

Grâce à ses différents filtres d'export, Blender est capable d'écrire dans les formats de fichiers suivants, issus d'autres applications graphiques, 2D ou 3D :

- 3D Studio (.3ds)
- AC3D (.ac)
- Autodesk FBX (.fbx)
- COLLADA 1.3.1 (.dae)
- COLLADA 1.4 (.dae)



- Cal3D (.cfg, .xaf, .xsf, .xmf, .xrf)
- DEC Object File Formal (.off)
- DirectX (.x)
- LightWave (.lwo)
- LightWave Motion (.mot)
- MD2 (.md2)
- OpenFlight (.flt)
- OpenInventor (.iv)
- Quake 3 (.map)
- Raw Faces (.raw)
- Save Current Theme
- Save Mesh as MDD
- SoftImage XSI (.xsi)
- Stanford PLY (.ply)
- VRML 97 (.wrl)
- Wavefront (.obj)
- X3D Extensible 3D (.x3d)
- xfig export (.fig)

## Animation

*BlenderLip Synchro* : ce script importe des fichiers *Voice Export* produits par des logiciels comme *Papagayo* ou *JLipSync* et synchronise ceux-ci avec les *Shape Keys* de phonèmes correspondants. Plus d'informations sur <http://blenderlipsynchro.blogspot.com/>.

*Camera Changer* : ce script permet de déterminer la caméra courante en fonction de la *frame* d'animation, ce qui est très utile pour tous les changements de points de vue au cours d'une animation sans sauter d'une scène à l'autre.

*Envelop Symmetry* : ce script permet de créer des enveloppes parfaitement symétriques, avec les mêmes conventions que le script *Armature Symmetry*. Il fait partie d'un ensemble de scripts présentés sur <http://www.mindfloaters.de/blender/>.

*Shape Widget Wizard* : ce script permet de créer des objets qui vont piloter les *Shape Keys*. En effet, il est parfois utile de réaliser des animations faciales, par exemple, avec une interface graphique interactive. Ce script vous permet de définir différents types de contrôleurs permettant de piloter chacun entre un et quatre *Shape key*. Des informations, des

---

exemples et des explications sur ce script à l'adresse : <http://blenderartists.org/forum/showthread.php?t=52435>.

*Trajectory* : ce script permet d'afficher en temps réel la trajectoire d'un objet, et de modifier celle-ci soit en déplaçant l'objet (courbe rouge), soit en insérant de nouvelles clés de positions (*Insert Key*, touche *I*) (courbe bleue).