



• **Introduction**

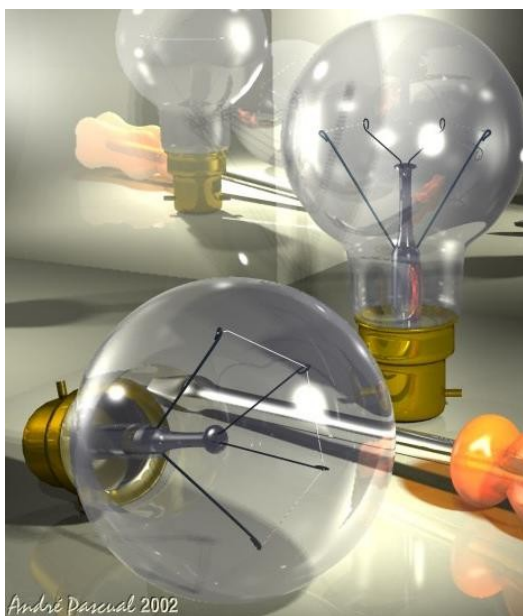
• Introduction

- Partie 1 Messieurs Spielberg, Lucas et Cameron nous ont fort divertis avec des films techniquement virtuoses. Le virtuel y paraît si réel que beaucoup ont senti poindre en eux une vocation de graphiste 3D, imaginant sans complexe qu'il suffirait d'un logiciel spécialisé pour recréer naturellement une scène de Jurassic Park, Stars War ou Titanic. Et c'est ainsi que l'on peut lire sur divers forums des demandes du genre: "Je voudrais m'initier à la 3D; qui peut me fournir une clé pour Studio Max4?". Les bras m'en tombent. C'est un peu comme si, désirant goûter les joies de la patinette, on demandait à autrui de nous fournir un semi-remorque volé afin d'apprendre à garder l'équilibre sur un deux-roues de loisir.
- Partie 2
- Partie 3
- Partie 4
- Partie 5
- Partie 6
- Partie 7
- Conclusion

- Retour
- Index

Attitude puérile autant que crapuleuse. Commençons petitement, avec un petit logiciel coûtant peu, mais facile à maîtriser. Car avant de modéliser Terminator, il importe de se soumettre aux rigueurs de la construction, à l'exigence du détail précis et aux dilemmes du compromis intelligent. Ainsi s'acquièrent les principes –pour ne pas dire les fondamentaux, terme que tout le monde se croit obligé d'employer aujourd'hui– de la modélisation 3D. Pour tâcher d'y atteindre, même avec un petit logiciel, n'importe quel objet usuel conviendra: il suffira qu'on veuille le modéliser –en faire un modèle représentatif– le plus fidèlement possible, sans se laisser rebuter par les difficultés rencontrées.

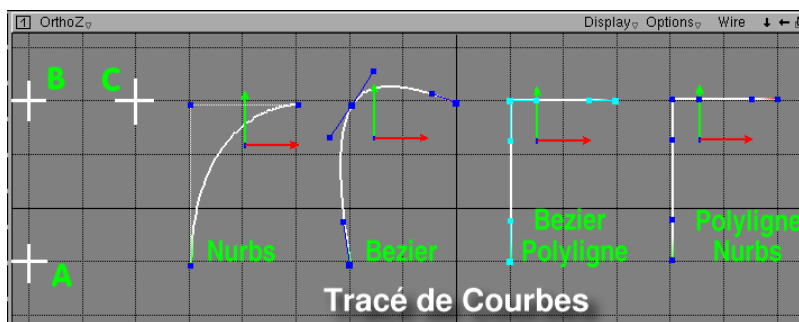
Prenons par exemple une ampoule à baïonnette, banale, quotidienne, sans grandeur: on est loin de Godzilla, certes, mais il y a fort à parier que les auteurs du lézard atomisé savent modéliser une ampoule... sans se faire d'ampoules à force de cliquer au hasard sur les innombrables entrées des menus de Max, à la recherche de la bonne fonctionnalité. Essayons d'approcher la même efficience, avec **Moonlight 3D**.



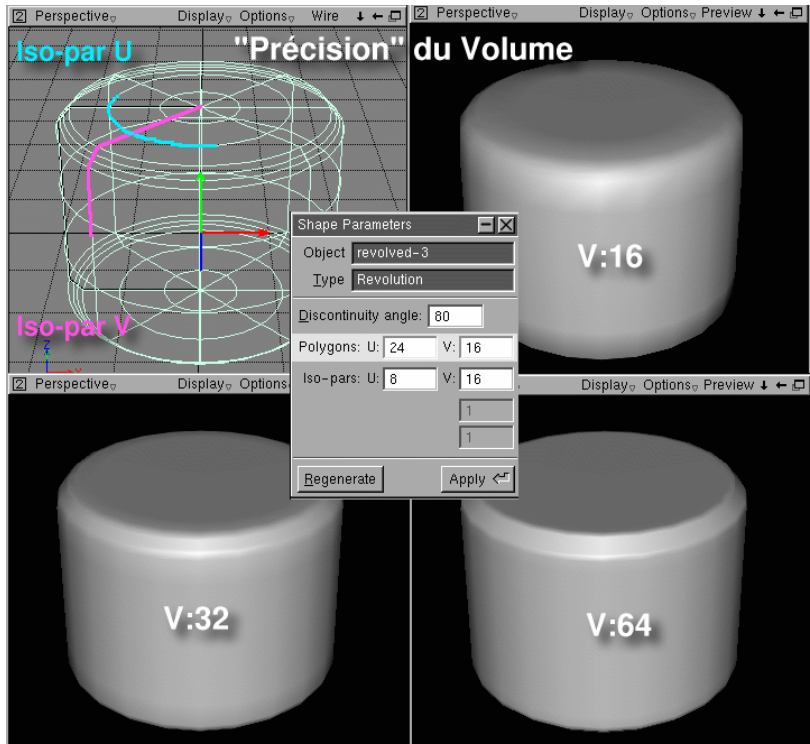
† **Partie 1**⌘ Introduction⌘ Partie 1 **Pour se mettre en jambes...**⌘ Partie 2⌘ Partie 3⌘ Partie 4⌘ Partie 5⌘ Partie 6⌘ Partie 7⌘ Conclusion● Retour● Index

Tout d'abord, précisons que l'interface de **Moonlight**, ainsi que les raccourcis clavier sont supposés connus des utilisateurs. Ensuite, rappelons quelques notions élémentaires.

Un volume, ou une surface, s'obtient à partir d'un profil que l'on fait pivoter autour d'un axe (Revolution) ou que l'on translate suivant un axe ou un chemin (Extrusion et Sweep). Laissons de côté pour l'instant les volumes prédéfinis que sont les Primitives, et les surfaces complexes: Skin, Loft, Birail, Tube, Composite... Un profil sera donc tracé, et il pourra l'être de plusieurs manières (voir figure [Tracé de courbes](#)). Supposons que le profil soit défini par trois points A, B et C. Nous disposons pour le dessiner de la courbe Nurbs, Bezier et Bezier Polyligne, cette dernière pouvant être par la suite convertie en Polyligne Nurbs. La figure parle d'elle-même: avec les deux premières méthodes, dans ce cas particulier, si l'on désigne les points A,B et C, le résultat obtenu est sans doute loin de ce que l'on était en droit d'attendre. Aussi est-ce pour cette raison que dans la suite de l'exercice, il sera toujours fait appel à la Bezier Polyligne. Celle-ci s'affine en déplaçant (m+drag) les points de contrôle et les poignées des tangentes.



Les surfaces quant à elles, sont définies par une succession de facettes. Si l'on génère un cylindre chanfreiné par Révolution, à partir d'un profil Bezier Polyligne, la surface obtenue est caractérisée par un nombre de polygones suivant U et V (voir figure [Précision du Volume](#)) attribués par défaut. Une pré visualisation OpenGL de ce "volume", montre que le chanfrein se présente comme un raccordement. En augmentant la valeur V, la forme du "volume" devient plus précise, et le chanfrein ressemble enfin à un chanfrein. Evidemment, dans d'autres cas de figure, il eût fallu augmenter la valeur U, ou U et V en même temps. Il n'est ici question que du principe. Les Iso-par U (couleur cyan) et V (couleur magenta) définissent le nombre de tracés du "volume" en mode filaire, mais n'influent pas sur la précision de celui-ci.



↑ **Partie 2**⌕ Introduction⌕ Partie 1 **La Hiérarchie de construction**⌕ Partie 2⌕ Partie 3⌕ Partie 4⌕ Partie 5⌕ Partie 6⌕ Partie 7⌕ Conclusion● Retour● Index

Une société sans hiérarchie est aussi déstructurée qu'une modélisation à la va-comme-je-te-pousse.

Avant de commencer quoi que ce soit dans **Moonlight 3D**, il convient de déposer sur l'Établi la boîte de dialogue Hierarchy, puis de faire l'analyse de l'objet à modéliser. On notera en premier lieu que l'on ne va modéliser qu'un objet, et qu'il s'agit d'une ampoule. La boîte Hierarchy va de ce fait contenir une entrée: Ampoule. Mais cette entrée doit correspondre à un objet modélisé. Ce sera un Null object (une croix), qui n'aura pas de "réalité", c'est-à-dire qu'il n'apparaîtra pas lors du rendu. Considérons-le comme un point d'ancrage de tous les autres objets à modéliser. Il sera l'objet **Parent**, celui dont tous les autres dépendront.

Ces autres, quels sont-ils? Analysons:

L'ampoule est constituée d'un bulbe, d'une douille et d'un intérieur, chacune de ces parties étant composée elle-même de différents éléments. Les objets Bulbe, Douille et Intérieur\_Ampoule sont donc des sous-ensembles, **Parents** des éléments qui les composent, mais **Enfants** de l'objet **Parent principal Ampoule**.

On verra en fin d'exercice quel avantage apporte cette structure pyramidale.

Pour l'instant, définissons dans la fenêtre OrthoZ un Null object, que l'on renomme Ampoule. Une croix, blanche lorsqu'elle est sélectionnée, est apparue à l'intersection des axes XY. Par commodité pour la suite, il convient de la déplacer (v + drag) vers le haut de la fenêtre. A noter qu'elle peut se trouver n'importe où dans l'écran. (voir figure [Hiérarchiser: Null Object](#))





- ↑ **Partie 3**

- ↳ Introduction

- ↳ Partie 1 **Une petite Révolution, pour commencer: le Bulbe**

- ↳ Partie 2

- ↳ Partie 3

- ↳ Partie 4

- ↳ Partie 5

- ↳ Partie 6

- ↳ Partie 7

- ↳ Conclusion

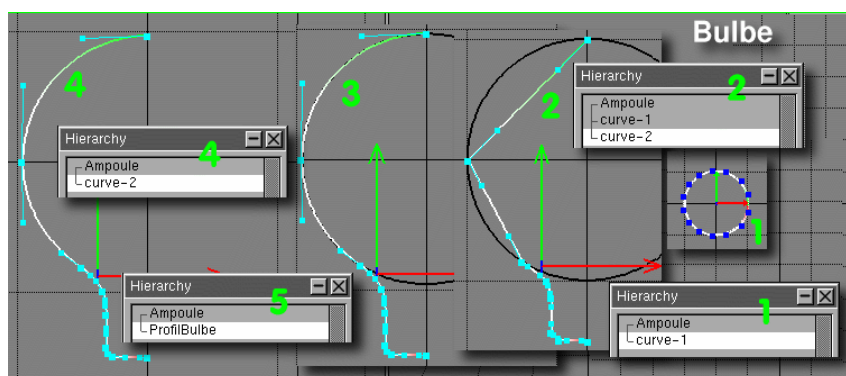
- ↳ Retour

- ↳ Index

Le bulbe en verre est un objet de révolution. On va donc le construire en respectant sa réalité géométrique. Il se compose d'une sphère et d'un cylindre raccordé par un rayon important. Le tout se génère par une rotation de 360° –une Révolution– autour d'un axe d'une courbe semblable à la courbe 4 (voir figure: Bulbe). La difficulté dans le tracé de cette courbe réside dans le dessin du 3/4 de cercle de sa partie supérieure. Il convient donc de ruser comme suit:

- 1– Toujours dans la fenêtre OrthoZ, tracer un Cercle. En changer l'échelle (x + drag) à sa convenance.
- 2– Tracer une Bezier Polyligne en s'appuyant sur le cercle
- 3– Faire épouser à courbe de Bezier la forme du cercle à l'aide des points de contrôle (m + drag)
- 4– Supprimer le cercle ( Backspace + clic)

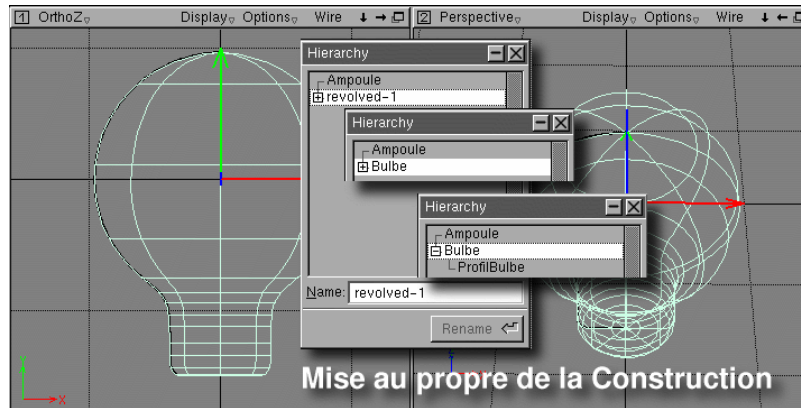
On remarque dans la Hierarchy, que le cercle s'appelle curve-1(Hier 1) et la Bezier, curve-2 (Hier 2). Pour l'instant, ces courbes sont indépendantes entre elles ainsi que du Null Object Ampoule. Lorsque le cercle est supprimé, Hierarchy ne contient plus que curve-2 (Hier 4) , que l'on s'empresse de renommer ProfilBulbe (Hier 5)



Par défaut, le cercle de base se trouve centré à l'intersection XY et si l'on a construit correctement le profil, celui commence et se termine exactement sur l'axe Y. C'est une bonne chose. En effet, pour générer la surface Bulbe, il faut faire pivoter le profil autour de Y –Relations>Revolution>Revolution around Y–. On comprend donc que si la courbe n'avait pas coïncidé avec l'axe, on aurait obtenu une surface trouée aux extrémités, ou avec des surfaces entrecroisées.

La Hierarchy contient un nouvel objet Revolved-1, que l'on nomme Bulbe. En cliquant sur le signe + précédant l'entrée Bulbe, on découvre que l'objet ProfilBulbe est **Enfant** de Bulbe: ils sont dépendants. En sélectionnant ProfilBulbe dans la Hierarchy, il est possible de modifier la courbe et donc la surface résultante, en temps réel, ce qui met en évidence l'inter-dépendance de la surface et du profil.

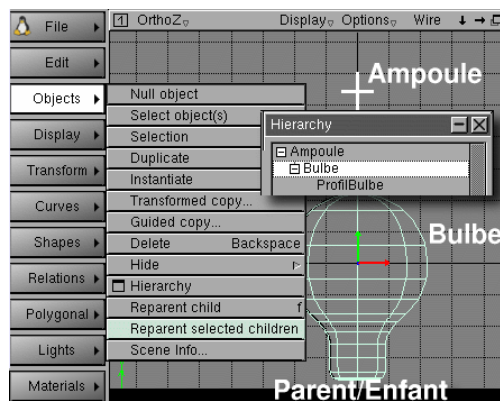
Renommer ainsi les objets d'une manière explicite est une Mise au propre de la construction (voir figure: Mise au propre )



Mais le Bulbe, comme il a été dit plus avant, est un sous-ensemble –terme impropre, puisque constitué d'un seul élément– de l'objet Ampoule. Il faut donc en faire un **Enfant** de l'objet **Parent principal** Ampoule. Pour ce faire:

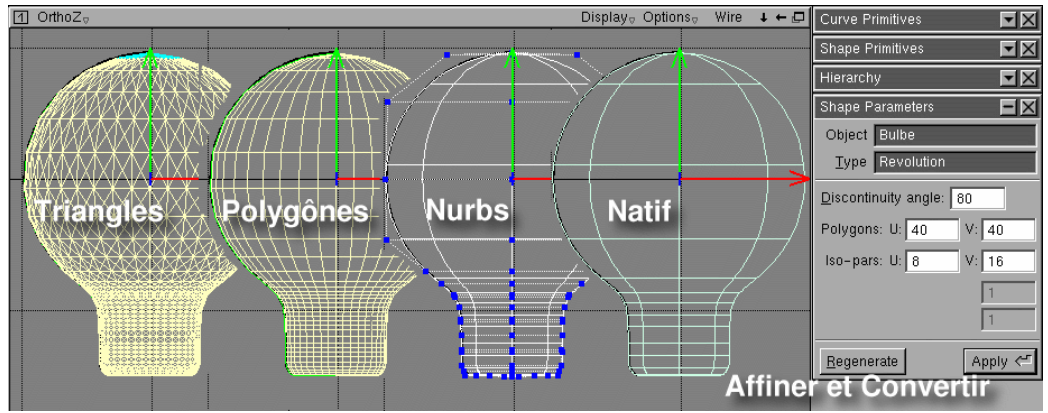
- 1– Sélectionner la surface Bulbe (Espace + clic) dans la fenêtre, ou désigner l'objet Bulbe dans la Hierarchy
- 2– Sélectionner Objects> Reparent selected children
- 3– Sélectionner dans la fenêtre le NullObject Ampoule (la croix)

Aussitôt, la hiérarchie se reconstruit (voir figure : Parent/Enfant)



Cette démarche sera utilisée pour toute la suite de l'exercice. Elle ne sera plus explicitée.

A ce stade du travail, il serait judicieux de sauvegarder. Ouvrons une parenthèse: **Moonlight 3D** est bogue. Pour des raisons que la raison ignore, il se peut qu'à la ré-ouverture on ne retrouve pas son travail dans l'état où il était lors de l'enregistrement. Une astuce consiste à convertir les surfaces natives de **Moonlight** soit en Nurbs (dès cet instant le profil générateur devient inutile et peut être supprimé), soit en Polygones, soit en Triangles. Par expérience, je peux dire que la transformation en polygones donne les meilleures garanties d'intégrité, mais le fichier grossit considérablement, surtout si l'on affine la surface en augmentant les valeurs Polygons U et V. (voir figure : **Affiner et Convertir**). Quelle que soit la représentation choisie, on sauvegardera à la fin de chaque étape de construction, sans qu'il soit besoin de le spécifier par la suite.





- † **Partie 4**
- φ **Introduction**
- φ **Partie 1**    **Un sous-ensemble: la douille**
- φ **Partie 2**
- φ **Partie 3**
- φ **Partie 4**    Celle-ci est composée du Culot, des Ergots, de l'Isolant et du Contact\_1 et du Contact\_2. Il va falloir hiérarchiser.
- φ **Partie 5**
- φ **Partie 6**
- φ **Partie 7**    Un Null Object Douille, **Enfant** de l'objet **Parent principal** Ampoule doit être créé.
- φ **Conclusion**    Lui-même sera **Parent** des objets modélisés Culot, Ergot et Contact\_1 et Contact\_2. Plusieurs étapes sont nécessaires à la construction de la Douille.
- **Retour**
- **Index**

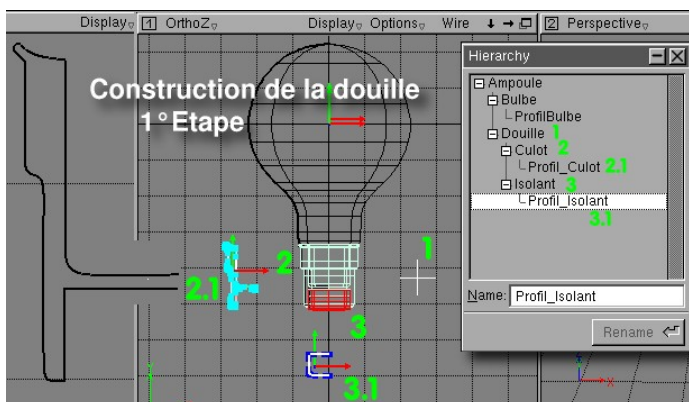
### 1° Etape: le Culot.

Créer un Null Object (1)

–A l'aide d'une Bezier Polyligne, tracer le profil du culot (2.1) et en faire une Revolution autour de Y (2)

–A l'aide d'une autre Bezier Polyligne, tracer le profil de l'isolant (3.1) et en faire une Revolution autour de Y (3)

–Renommer surfaces et profils comme indiqué dans Hierarchy et hiérarchiser la construction tel que les objets modélisés Culot et Isolant dépendent de l'objet Douille, et que celui-ci dépende du **Parent principal** Ampoule. (Voir figure **Construction de la Douille, 1° Etape**)

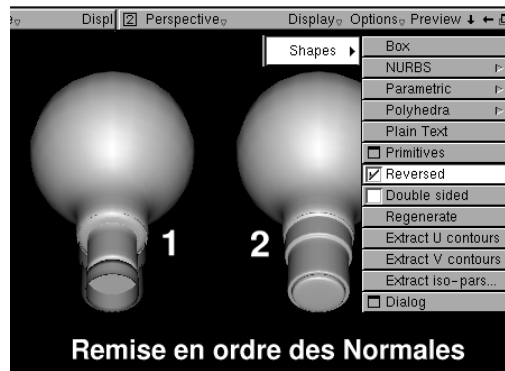


### Et si l'on vérifiait le résultat?

Avant de sauvegarder, il convient de vérifier ce qui a été construit. En effet, la vision filaire est trompeuse. Telle surface semble être correcte alors qu'en réalité, elle est retournée comme une chaussette. Ceci est visible si l'on demande un affichage Preview en lieu et place de Wire. Si tel est le cas, cela signifie que les Normales aux surfaces ne sont pas dirigées dans le bon sens. Cependant, un remède a été prévu: Shapes>Reversed. La visualisation OpenGL est immédiatement rafraîchie. (Voir figure **Remise en ordre des Normales** ). En 1, les Normales sont inversées, en 2, elles sont

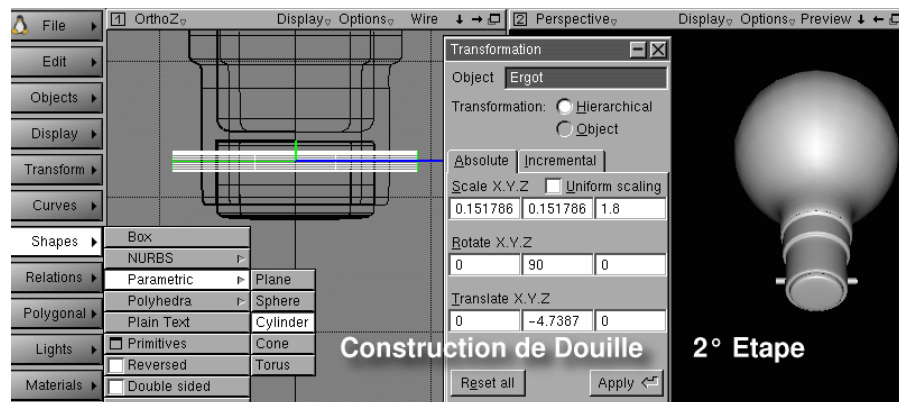


remises en ordre. Profitons de cette parenthèse pour signaler que le Bulbe en verre précédemment modélisé devra réfléchir la lumière sur sa face extérieure ainsi que sur sa face intérieure. Il y a donc lieu d'indiquer qu'il y a deux directions de Normales pour cette surface en cochant l'option Double Sided.



## 2° Etape: l'Ergot

Celui-ci est constitué d'un cylindre que l'on pourrait obtenir soit en extrudant un cercle, mais il faudrait alors refermer les extrémités, soit en faisant pivoter un profil autour d'un axe. Nous allons plus simplement utiliser la Primitive Shapes > Parametric > Cylinder (ou Nurbs > Cylinder). Le cylindre qui se met en place suite à cette commande est pourvu de caractéristiques par défaut (diamètre, longueur, orientation). Il faut donc le redimensionner et le mettre en place à la demande, soit en changeant d'échelle (x + drag) à la volée, soit plus rigoureusement en utilisant des valeurs de modification, ce que l'on fait dans la boîte de dialogue Transformation après qu'on l'a déposée sur l'Etabli, c'est-à-dire la partie droite de l'écran. Cf le didacticiel relatif à ce sujet sur < <http://www.linuxgraphic.org/section3d/moonlight/didacticiels/index.html> > . Les valeurs utilisées, qui sont à adapter aux dimensions des autres objets déjà modélisés, apparaissent dans la boîte Transformation (voir figure Construction de la Douille 2° Etape). Ne pas oublier de hiérarchiser, c'est à dire de renommer le cylindre Ergot et de le faire dépendre de l'objet Douille (qui est une croix Null Object)



## 3° Etape: les contacts

Dans la réalité, ces contacts en plomb ont une forme ovoïde. On se contentera d'une forme sphérique obtenue à l'aide d'une primitive Sphère, Nurbs ou Parametric, étant

donné le peu de relief de cet élément. C'est en ce sens qu'en début d'article il était question de compromis raisonné: quand se satisfaire pour un élément donné d'une forme approchante qui ne dénaturera pas l'ensemble du travail?

Dans le cas présent, sélectionner Shapes>Nurbs>Sphere.

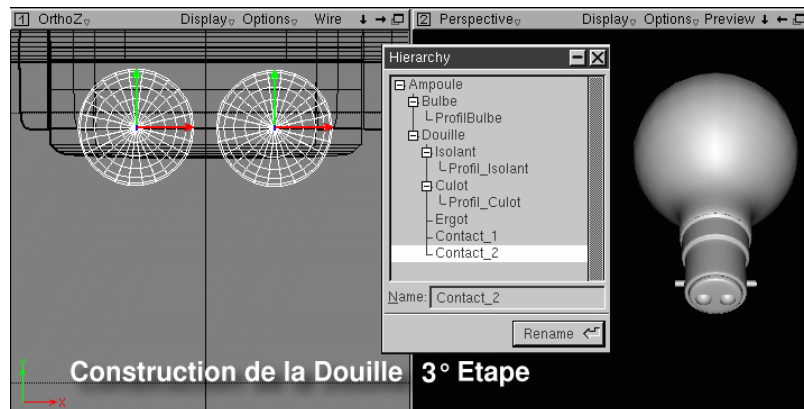
-Dimensionner (x + drag) la sphère apparue

-La mettre en place (v + drag)

-La copier Objet>Duplicate

-Mettre l'objet copié en place (pour l'instant, il est sélectionné et juxtaposé à l'original)

-Hiérarchiser : nommer les objets Contact\_1 et Contact\_2, les placer dans Hierarchy, qui devrait ressembler à celle-ci (voir figure Construction de la Douille 3° Etape)



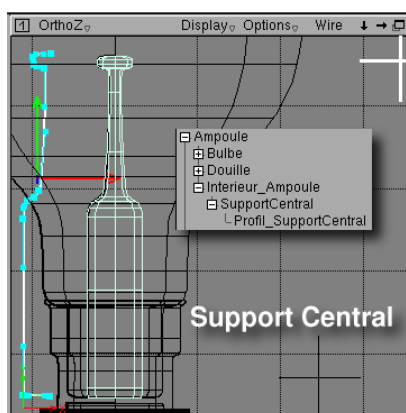
**Partie 5****Introduction****Partie 1 C'est encore plus beau à l'intérieur****Partie 2****Partie 3****Partie 4****Partie 5****Partie 6****Partie 7****Conclusion****Retour****Index**

Prenons la peine d'examiner l'intérieur de l'ampoule. On y trouve un filament soutenu par deux supports longs conducteurs, deux supports courts, le tout maintenu par une implantation dans une sorte de pylône central en verre, soit au total six objets. Il s'agit donc d'un sous-ensemble de l'ampoule, que l'on va baptiser Intérieur\_Ampoule et matérialiser par un Null Object. Les autres objets, SupportCentral, Support\_1, Support\_2, Support\_3, Support\_4 et Filament seront **Enfants** de Intérieur\_Ampoule, lui-même **Enfant** du **Parent principal** Ampoule. (voir en 5 figure Utiliser la Hiérarchie)

**5.1 Le Support Central en verre**

Allons-y:

- Créer un Null Object et le renommer Intérieur\_Ampoule
- Tracer le profil du SupportCentral à l'aide d'une Bezier Polyligne
- Effectuer une Revolution autour de Y
- Renommer les Curve et Revolved en Profil\_SupportCentral et SupportCentral
- Hiérarchiser comme indiqué (voir figure Support Central )

**5.2 Les supports longs conducteurs (d'électricité)**

Et maintenant, un peu plus sophistiqué: la Relation Sweep, qui permet de créer une surface par déplacement d'une courbe fermée ou non le long d'un chemin. Dans ce cas précis, il s'agit de modéliser un support de filament, en fait un fil métallique comportant une boucle à l'une de ses extrémités pour maintenir ledit filament. Le chemin de balayage est constitué par l'axe du support et la courbe déplacée est un cercle équivalent à la section du support (son « diamètre » si l'on préfère).

Procéder comme suit: (voir figure Support\_1 ).

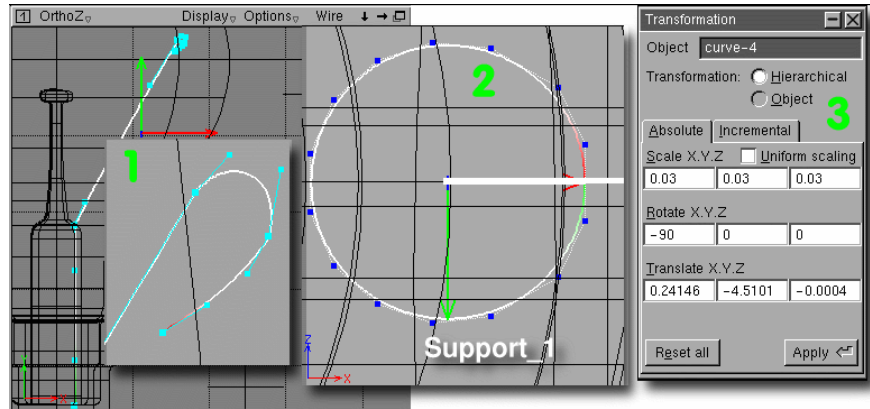
–Tracer une Bezier Polyligne figurant l'axe du support (1).

–Tracer un cercle.

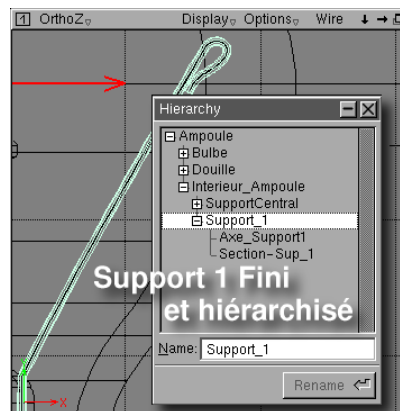
–Le redimensionner (x + drag) jusqu'à obtenir un diamètre satisfaisant (3).

–Le mettre en place à l'extrémité inférieure de l'axe du support (2).

–Créer la surface par: Relation>Sweep, en désignant d'abord le chemin de balayage puis la courbe à déplacer.



–Renommer les éléments et les hiérarchiser en les faisant dépendre de l'objet Parent Interieur\_Ampoule (voir figure Support 1 Fini et hiérarchisé). On remarquera dans la Hierarchy, le souci de nommer explicitement les objets: Support\_1, réalisé à l'aide d'un chemin de balayage, Axe\_Support1, et d'une courbe à déplacer, Section-Sup\_1.



Si le résultat est convaincant, on duplique ce Support par: Objet>Duplicate, que l'on met en place symétriquement au premier. On rappelle qu'un objet dupliqué est sélectionné par défaut dès sa création, et qu'il recouvre l'objet original. Inutile donc de le chercher ailleurs. De plus, si la boîte de dialogue Transformation a été déposée sur l'Etabli, la symétrie de l'objet dupliqué s'obtient par l'introduction de la valeur 180°, ligne Rotate, champ Y (celui du milieu)... à condition d'avoir dessiné dans la fenêtre OrthoZ. Ce nouveau support, qu'on appellera Support\_2, est automatiquement disposé dans la Hierarchy, par le fait que le modèle était lui-même hiérarchisé. D'où l'on déduit l'importance de la chronologie des étapes. A ce sujet: si un matériau avait été affecté au modèle, il aurait également été affecté à la copie. Comme cela n'a pas été le cas ici, dans un but pédagogique, il faudra attribuer les matériaux objet par objet.

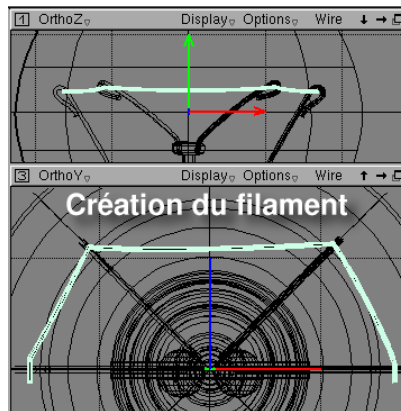
### 5.3 Les supports courts

De la même manière que précédemment, et sans qu'il soit nécessaire de répéter le déroulement des opérations, nous allons créer les deux supports courts, appelés Support\_3 et Support\_4 (voir figure Support 3 & 4). Le Support\_3 est à placer à 45° du Support\_1, et le Support\_4 à 90° du Support\_1. Mise en figure Mise en place du Support 4)



#### 5.4 Le filament

Si l'on est parvenu jusqu'ici sans encombres, il n'y a aucune raison de chuter maintenant. Le filament se modélise comme les supports, à partir d'un chemin de balayage et d'une section à déplacer sur ce chemin pour générer une surface Sweep. On notera simplement que la section du filament devra être inférieure à celle des supports et que le chemin de balayage devra passer à l'intérieur des boucles des supports comme le fil dans le chas de l'aiguille. (voir figure Création du filament ). Bien entendu, ce nouvel élément doit être hiérarchisé.



↑ **Partie 6**φ **Introduction**φ **Partie 1** **Vous prendrez bien une dernière goutte de Hierarchy?**φ **Partie 2**φ **Partie 3**φ **Partie 4**φ **Partie 5**φ **Partie 6**φ **Partie 7**φ **Conclusion**● **Retour**● **Index**

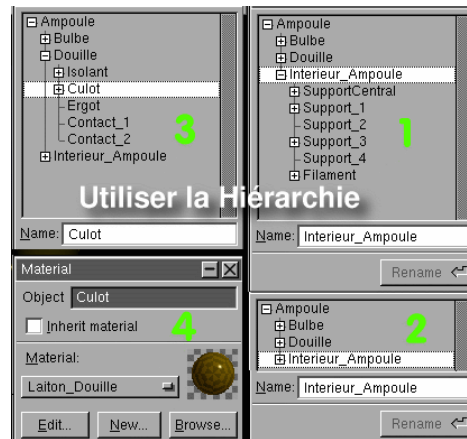
La modélisation de l'ampoule électrique à culot baïonnette est terminée. Tout au long de cet exercice il a été pesamment question de la hiérarchisation de la construction. A quoi cela peut-il bien servir? Tout d'abord à disposer d'un plan de travail structuré que l'on peut lire, interpréter et manipuler sans aucune difficulté. Imaginons un instant que les objets n'aient été ni renommés ni mis en relation les uns par rapport aux autres: nous disposerions d'une liste de curve-x, de revolved-x et de sweep-x, sans savoir à quel objet dessiné attribuer telle ou telle courbe génératrice que l'on désirerait modifier pour changer la forme de la surface résultante.

Ensuite, la structure en ensemble et sous-ensembles, permet de sélectionner la totalité d'un objet complexe en ne sélectionnant dans la Hierarchy ailleurs cela s'appelle un arbre de construction, features manager etc,- que le Null Object qui le représente. Il peut alors être déplacé (v + drag) , pivoté (c + drag) et redimensionné (x + drag) sans que l'on ait à se préoccuper de ses composants. Si l'objet complexe n'était pas structurellement construit, il conviendrait de sélectionner la totalité de ses composants, et, s'il était alors possible de translater le tout, le pivotement serait quant à lui impossible: chaque objet tournerait autour de son propre système d'axe (repère local) et non autour du repère global. L'objet serait complètement disloqué.

La structure en sous-ensembles permet aussi d'agir sur un certain nombre d'éléments à l'intérieur d'un ensemble plus complexe; elle permet également de représenter d'une manière littérale la composition physique de l'objet. L'ampoule est bien constituée réellement d'un culot en laiton et d'un bulbe en verre contenant le système à incandescence: donc trois sous-ensembles qui ont été appelés: Bulbe, Douille, Interieur\_Ampoule. Cette décomposition aide à la planification du travail. Je parle d'expérience. (voir figure Utiliser la Hiérarchie) En 1, le sous-ensemble Interieur\_Ampoule est déployé pour en montrer ses composants; en 2, la structure est partiellement regroupée, puisque les sous-ensembles sont visibles. Il faudrait cliquer sur le signe moins (-) précédant l'objet Ampoule pour refermer totalement l'arbre de construction. A un seul objet global à l'écran, une ampoule électrique, correspondrait logiquement un seul objet Ampoule dans la Hierarchy.

Un objet modélisé, ou un élément nécessaire à sa modélisation (courbe génératrice, surface...) se sélectionne soit dans les fenêtres de visualisation, ce qui est très difficile dans une scène en filaire comportant un grand nombre d'objets, soit en le désignant dans l'arborescence déployée. C'est la meilleure méthode, surtout si l'on a nommé explicitement les objets. Ainsi, l'attribution de matériaux se fait-elle naturellement: par exemple, sélectionner Culot (3) et lui affecter un matériau Laiton (4). Ou créer le matériau: il n'y a pas de shaders dans **Moonlight** . Heureusement, tous les matériaux créés pour d'autres scènes restent disponibles. Pour ce faire, il convient de

sauvegarder l'objet –en fait la scène– par: >File >Save in database as.. Ce que l'on sauvegarde ainsi est alors prêt à être intégré dans d'autres scènes que l'on est train d'élaborer. Il faudra aller le chercher par: >File> Get> Scene, pour l'insérer dans un nouveau travail en cours.





## Articles

### ↑ **Partie 7**

#### ↳ Introduction

#### ↳ Partie 1 Tenue de soirée exigée

#### ↳ Partie 2

#### ↳ Partie 3

#### ↳ Partie 4

#### ↳ Partie 5

#### ↳ Partie 6

#### ↳ Partie 7

#### ↳ Conclusion

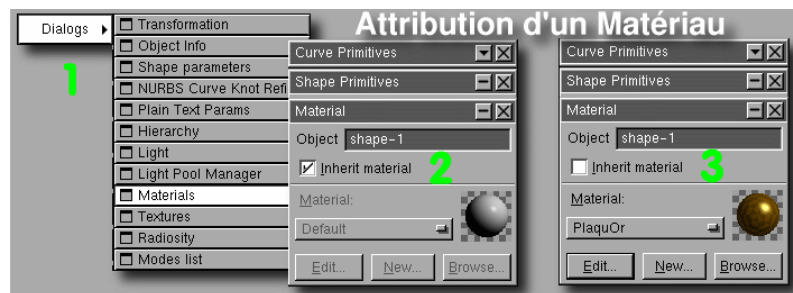
#### ↳ Retour

#### ↳ Index

Un objet, si exact soit-il dans sa représentation géométrique, n'acquiert son aspect réaliste qu'après habillage. On entend par ce terme l'attribution de paramètres (couleur, brillance, transparence, réflexion, réfraction, relief...) à l'objet afin de lui conférer l'apparence voulue. Nous supposons, encore une fois, familières les manipulations relatives au paramétrage des matériaux; et si la mémoire défaille, il est toujours possible de consulter: <

[http://www.linuxgraphic.org/section3d/articles/moon\\_materiaux/index.html](http://www.linuxgraphic.org/section3d/articles/moon_materiaux/index.html) >. Rappelons toutefois les principes de base (voir figure Attribution d'un Matériau).

Pour créer, charger ou éditer un matériau, il faut en premier lieu avoir déposé sur l'Etabli la boîte de dialogue Material, que l'on ouvre par: >Dialogs>Materials (1). Dès lors qu'une surface est générée, elle se voit affectée d'un matériau par défaut appelé Inherit material, de couleur grise, utilisé pour la pré-visualisation OpenGL (2). A ce moment de la manipulation, les fonctions Edit, New et Browse de la boîte Material sont inaccessibles, car le matériau affecté à l'objet est celui par défaut dont les caractéristiques sont pré-définies, et ne peuvent donc être modifiées. En décochant l'option Inherit material, on accède à ces fonctions. Il est alors possible de charger un matériau sauvegardé en base de données –le database de **Moonlight**– en activant Browse, de le modifier en activant Edit, ou d'en créer un nouveau en activant New (3)



Rappelons également qu'un matériau est défini par un algorithme de rendu (BRDF) à choisir parmi la douzaine que propose **Moonlight**, par cinq paramètres intervenant sur l'aspect coloré (Emittance, Diffuse, Specular, Reflection, Transmission), par l'application d'un facteur de correction (Factor) et par la modification des valeurs de spéculaires propres à l'algorithme de rendu choisi. Les autres paramètres (Fog, Direct/Global, Shadows) concernent la scène et surtout, dans celle-ci, le comportement des objets auxquels le matériau en cours d'édition sera affecté. On notera de ce qui vient d'être dit, que chaque objet d'une scène **Moonlight** pourra être rendu avec un BRDF différent en fonction de sa « nature » et donc de l'aspect qu'il doit présenter: Oren-Nayar, Anisotropique, Cook-Torrance-Sparrow, Stauss-Phong, Minnaert, Poulin-Fournier, Beard-Maxwell, He-Sillion-Greenberg, Lommel-Seeliger, Lafortune cosine lobe,



Hapke–Lommel et Default. On notera également que chaque objet peut ou non contribuer à l'illumination globale, autrement appelée Radiosité, qu'il peut ou non être ombré par les éclairages directs, qu'il peut ou non générer des ombres, ou en recevoir, ou non... Telle est la finesse du moteur de rendu de **Moonlight** (voir figure **Edition/Création d'un Matériau**), qui donne aux objets leur parure de gala.





## Conclusion



### Introduction



### Partie 1



### Partie 2



### Partie 3



### Partie 4



### Partie 5



### Partie 6



### Partie 7



### Conclusion



### Retour



### Index

L'apprentissage de la rigueur est souvent rébarbatif; il me semble cependant que dans cette discipline de précision qu'est la modélisation 3D, on ne peut faire l'économie de l'apprentissage. On oublie souvent que les grands peintres aux palettes flamboyantes dont la vision du monde tangible bouscule le réel tel que nous l'appréhendons naturellement, se sont soumis d'abord aux contraintes du dessin précis, fidèle au modèle, exigeant à l'extrême. Quant on sait dessiner, c'est-à-dire modéliser en 2D, on peut s'attaquer humblement au grand oeuvre pictural. Quant on accepte de modéliser fidèlement les objets usuels, avec pour objectif d'apprendre encore et toujours, on peut songer sereinement aux grandes scènes époustouflantes de réalisme. Lequel réalisme réside également, pour grande partie, dans l'éclairage de la scène, le choix des textures et le paramétrage des matériaux, choses dont il n'a pas été complètement question ici. Il y a donc encore beaucoup à apprendre. C'est ce que j'essaie de faire chaque fois qu'il est possible.



NOTA: **Moonlight** a été abandonné par ses auteurs, Stéphane Rehel et Alexandre Belloste. Mais une petite équipe de développeurs a décidé de reprendre les sources à partir de la version 0.5.3 de **Moonlight**, qui était encore en GPL à ce moment-là. Si vous vous sentez une âme de programmeur en quête de projet, vous pouvez les rejoindre sur: <http://studentenbude.ath.cx/projects/moonlight.html> .

André PASCUAL  
<[andre@linuxgraphic.org](mailto:andre@linuxgraphic.org)>