jps2ps : version 0.16

par Jean-Paul Vignault
Groupe des Utilisateurs de Linux Poitevins (GULP)
(**jpv@melusine.eu.org**)
17 Janvier 2007

Ce document présente les différences entre la version 0.16 et la précédente.

1. Divers

$array_1$ **doublebubblesort** $array_2$ $array_3$ $\longrightarrow$ $array_3$ est obtenu en triant $array_1$ par ordre croissant et $array_2$ correspond à la position des indices de départ dans le tableau d'arrivé, ie si $array_1 = [13, 12, 14, 11]$, alors $array_2 = [3, 1, 0, 2]$

u **normalize** $v \longrightarrow v$ est le vecteur de norme 1 obtenu à partir de u, ie. $\vec{v} = \frac{1}{\|u\|}\vec{u}$

2. Solides

2.1 - Solides non convexes

Pour le tracé des solides non convexes, on utilise *l'algorithme du peintre* qui consiste à représenter les faces du solide en commaneçant par les plus éloignées de l'observateur. Ce tracé est obtenu par la commande **drawsolid****. Dans ce cas, il n'y a pas de tracé des arêtes cachées.

solid **drawsolid**** $- \longrightarrow$ affiche le solide *solid* en utilisant l'algorithme du peintre

2.2 - Modes de résolution

Pour la plupart des solides, un *mode de résolution* est proposé, afin de gérer la finesse du maillage. Ce procédé permet de préparer ses dessins en basse résolution, pour ne passer en haute résolution (plus onéreuse en termes de temps de calculs) qu'au moment final de la production. Pour chacun des solides concernés, 5 modes sont proposés, numérotés de 0 à 4. Le niveau 0 correspond à la résolution la plus faible. Si aucun mode n'est demandé, on utilise le mode de résolution par défaut.

Le numéro de mode est transmis, optionnellement, par un exécutable contenant l'indice choisi. Par exemple

```
0 3 2 {0} newcylindre
```

génére un cylindre avec le mode de résolution 0.

On peut également créer son propre mode. Pour cela on passe en paramètre une chaîne de caractères contenant les valeurs souhaitées. Ainsi

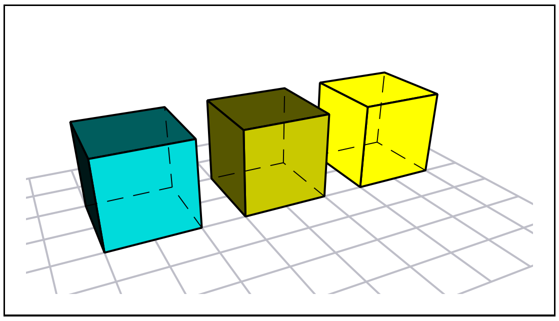
```
0 3 2 (10 36) newcylindre
```

génére un cylindre avec un maillage défini par 36 points sur la circonférence, et 10 étages (soit 11 points) sur la hauteur.

2.3 - Éclairage par une source lumineuse ponctuelle

La source lumineuse est définie par sa position *lightsrc*, sa couleur *lightcolor* et son intensité *lightintensity*. On modifie ces paramètres en utilisant les commandes **setlightsrc**, **setlight** et **setlightintensity**. Seule *lightintensity* est définie par défaut. Si *lightsrc* n'est pas définie, alors le solide est colorié avec des teintes uniformes. Si *lightsrc* est défini mais pas *lightcolor*, alors on considère que la lumière est blanche, et l'intensité de la teinte de chaque facette dépend de l'orientation par rapport à *lightsrc*. Enfin, si en plus la variable *lightcolor* est définie, alors les solides sont considérés blanc et la teinte de chaque facette dépend de l'orientation de la source lumineuse et de sa couleur.

Par exemple, voici un cube tricolore

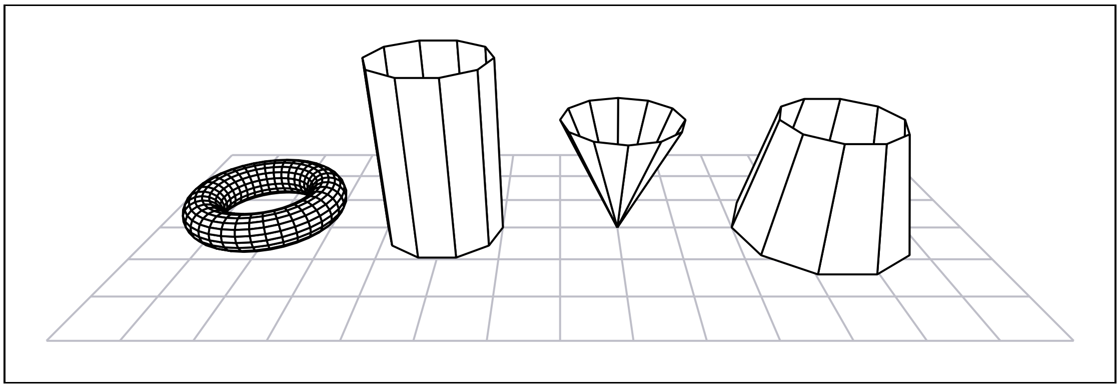


source jps

```
-4.5 5 setxrange
-2 3 setyrange
20 setxunit
5 10 6 SetCamPos 0 0 0 SetCamView
qplanxy
2 newcube             %% un nouveau cube
dup (jaune) outputcolors
dupsolid
{-3 0 1 translatepoint3d} solidtransform
drawsolid*
5 10 6 setlightsrc   %% avec lumiere blanche
dupsolid
{0 0 1 translatepoint3d} solidtransform
drawsolid*
{cyan} setlight      %% avec lumiere cyan
{3 0 1 translatepoint3d} solidtransform
drawsolid*
```

2.4 - Solides précalculés

Quelques nouveaux solides font leur apparition : le tronc de cône creux, le cylindre creux, le cône creux et le tore.



r R option **newtore** *solid* $\longrightarrow$ crée un nouveau tore, de type *solid*, de centre O, de rayon principal R, et de rayon du tube r

z_0 z_1 r_1 *option* **newcylindrecreux** *solid* $\longrightarrow$ crée un nouveau cylindre creux, de type *solid*, d'axe Oz, de rayon r, allant du plan $z = z_0$ jusqu'au plan $z = z_1$

z_0 r z_1 **newcone** *solid* $\longrightarrow$ crée un nouveau cône, de type *solid*, d'axe Oz, de rayon r, allant du plan $z = z_0$ jusqu'au plan $z = z_1$

z_0 r_0 z_1 r_1 *option* **newtroncconecreux** *solid* $\longrightarrow$ crée un nouveau tronc de cône creux, de type *solid*, d'axe Oz, de rayon de base r_0 (sur le plan $z = z_0$) et de rayon au sommet r_1 (sur le plan $z = z_1$)

2.5 - Opérations sur les solides

Quelques nouvelles opérations pour les solides :

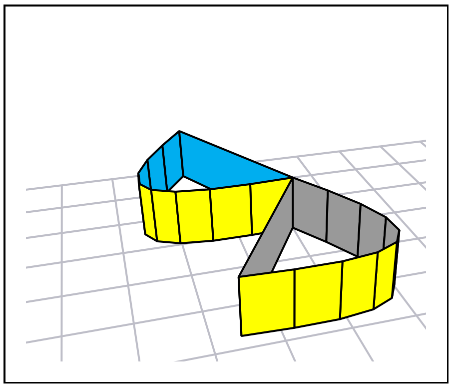
solid **videsolid** $- \longrightarrow$ ajoute au solide *solid* ses faces internes

solid **creusesolid** $- \longrightarrow$ enlève les faces d'indice 0 et 1 du solide *solid*, puis ajoute toutes les faces internes

solid i **solidrmface** $- \longrightarrow$ enlève la face d'indice i du solide *solid*

solid $string_0$ $string_1$ **inoutputcolors** $- \longrightarrow$ affecte la couleur définie par $string_0$ aux faces internes du solide *solid*, et la couleur définie par $string_1$ aux faces externes

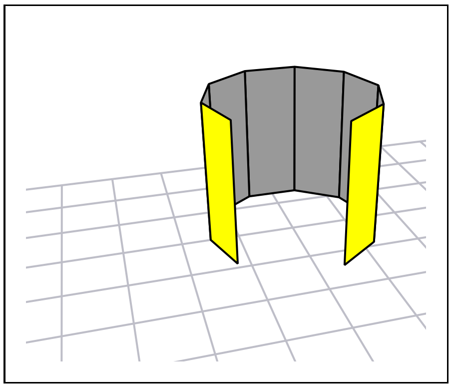
solid string **outputcolors** $- \longrightarrow$ affecte la couleur définie par *string* à toutes les faces du solide *solid*



source jps

```
-4 2 setxrange
-2 3 setyrange
25 setxunit
3 10 6 SetCamPos
0 0 0 SetCamView
qplanxy
2 setlinejoin
10 setresolution
newpath
   pi 0 smoveto
   pi 0 {Sin 2 mul} Courbe_
currentcpathpointstable 0 1 genereprismedroit
dup creusesolid
dup (cyan) (jaune) inoutputcolors
dupsolid {0 0 -60 rotateOpoint3d} solidtransform
drawsolid**
dup (.6 setgray) (jaune) inoutputcolors
dupsolid {0 0 60 rotateOpoint3d} solidtransform
drawsolid**
```

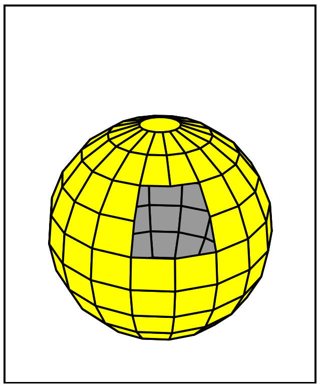
Un cylindre ouvert :



source jps

```
-4 2 setxrange
-2 3 setyrange
25 setxunit
3 10 6 SetCamPos
0 0 0 SetCamView
qplanxy
2 setlinejoin
0 1.5 2.5 newcylindre
dup 3 solidrmface
dup 3 solidrmface
dup creusesolid
dup (.6 setgray) (jaune) inoutputcolors
drawsolid**
```

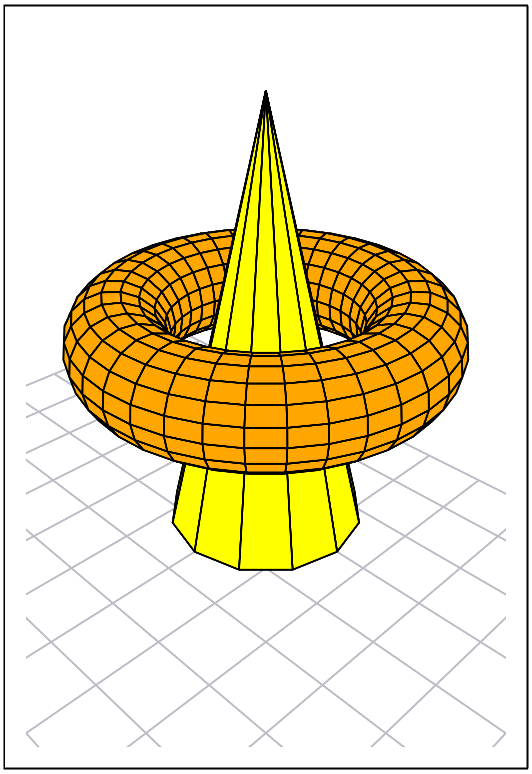
Une ouverture dans une sphère :



source jps

```
-2 2 setxrange
-2 3 setyrange
25 setxunit
3 10 6 SetCamPos
0 0 0 SetCamView
2 setlinejoin
2 (10 18) newsphere
dup 40 solidrmface
dup 40 solidrmface
dup 56 solidrmface
dup 56 solidrmface
dup videsolid
dup (.6 setgray) (jaune) inoutputcolors
drawsolid**
```

Il est également possible de « fusionner » deux solides pour n'en obtenir qu'un. Ceci permet d'appliquer l'algorithme du peintre sur l'ensemble. Par exemple :



source jps

```
-3 3 setxrange
-3 6 setyrange
30 setxunit
-7 7 8 SetCamPos
0 0 0 SetCamView
2 setlinejoin
qplanxy

0 1.5 6 newcone
dup (jaune) outputcolors

.7 2 newtore
{0 0 3 translatepoint3d} solidtransform
dup (orange) outputcolors

solidfuz
drawsolid**
```

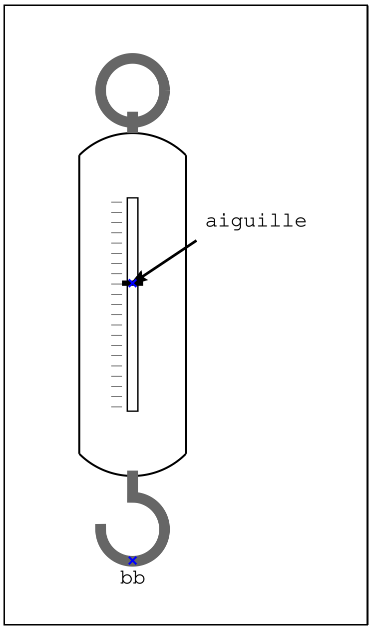
solid$_1$ *solid*$_2$ **`solidfuz`** *solid* $\longrightarrow$ dépose sur la pile le solide obtenu à partir de la fusion des solides *solid*$_1$ et *solid*$_2$

3. Divers

3.1 - Environnement picture

3.1.1 - Le dynamomètre

Un nouvel objet fait son apparition : le dynamomètre, offert par Luís A. V. Ferreira.



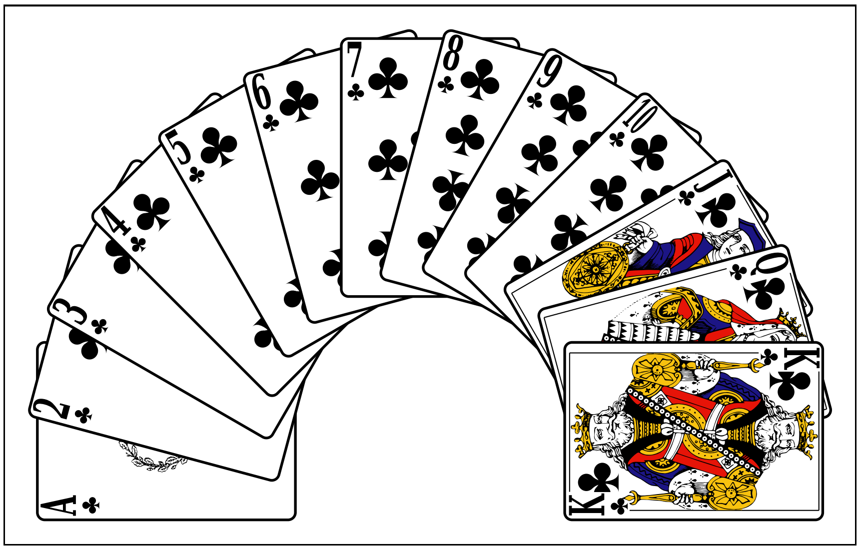
Nom : **`dinamometro`**

Points spéciaux :

- **`aiguille`** : centre de l'aiguille du dynamomètre
- **`bb`** : comme son nom l'indique

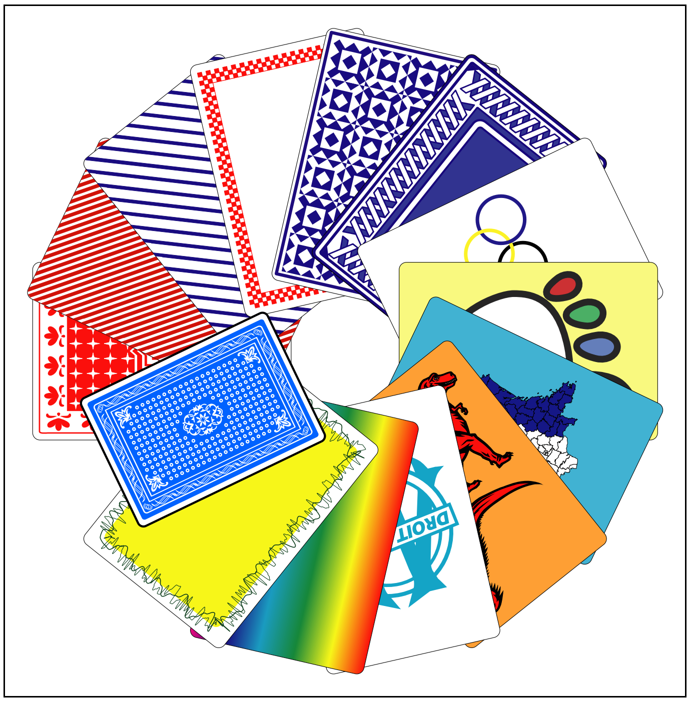
3.1.2 - Cartes à jouer

Grâce aux fichier SVG de David Bellot (**`http://david.bellot.free.fr/svg-cards/`**), l'environnement picture s'enrichit de 54 cartes et 14 dos de cartes. Voici par exemple les 13 cartes de trèfle



Le nom des objets : **01-coeur**, **02-coeur**, ..., **V-coeur**, **D-coeur**, **R-coeur** pour la couleur *cœur*, et des noms analogues pour trèfle, carreau et pique.

On dispose également de 14 dos de cartes



dont les noms sont : **carte_dos_01**, **carte_dos_02**, **carte_dos_03**, **carte_dos_04**, **carte_dos_05**, **carte_dos_06**, **carte_dos_07**, **carte_dos_08**, **carte_dos_09**, **carte_dos_10**, **carte_dos_11**, **carte_dos_13**, **carte_dos_14**, **carte_dos_bleu**.

3.2 - Commandes 3d

Cinq macros pour dessiner des triangles sphériques :

$r\ \theta\ \phi$ **rtp2xyz** $x\ y\ z \longrightarrow$ Passage des coordonnées sphériques vers les coordonnées cartésiennes

$r\ \theta_1\ \phi_1\ r\ \theta_2\ \phi_2$ **arcspherique** $- \longrightarrow$ trace l'arc de cercle entre les points A et B de coordonnées respectives (r, θ_1, ϕ_1) et (r, θ_2, ϕ_2)

$r\ \theta_1\ \phi_1\ r\ \theta_2\ \phi_2$ **geodesique_sphere** $- \longrightarrow$ trace le cercle passant par les points A et B de coordonnées respectives (r, θ_1, ϕ_1) et (r, θ_2, ϕ_2)

$A\ B\ C$ **trianglespherique** $- \longrightarrow$ trace le triangle sphérique ABC, où les points A, B et et C sont donnés par leurs coordonnées sphériques respectives (r, θ_1, ϕ_1), (r, θ_2, ϕ_2) et (r, θ_3, ϕ_3)

$A\ B\ C$ **trianglespherique*** $- \longrightarrow$ version étoilée de **trianglespherique**