

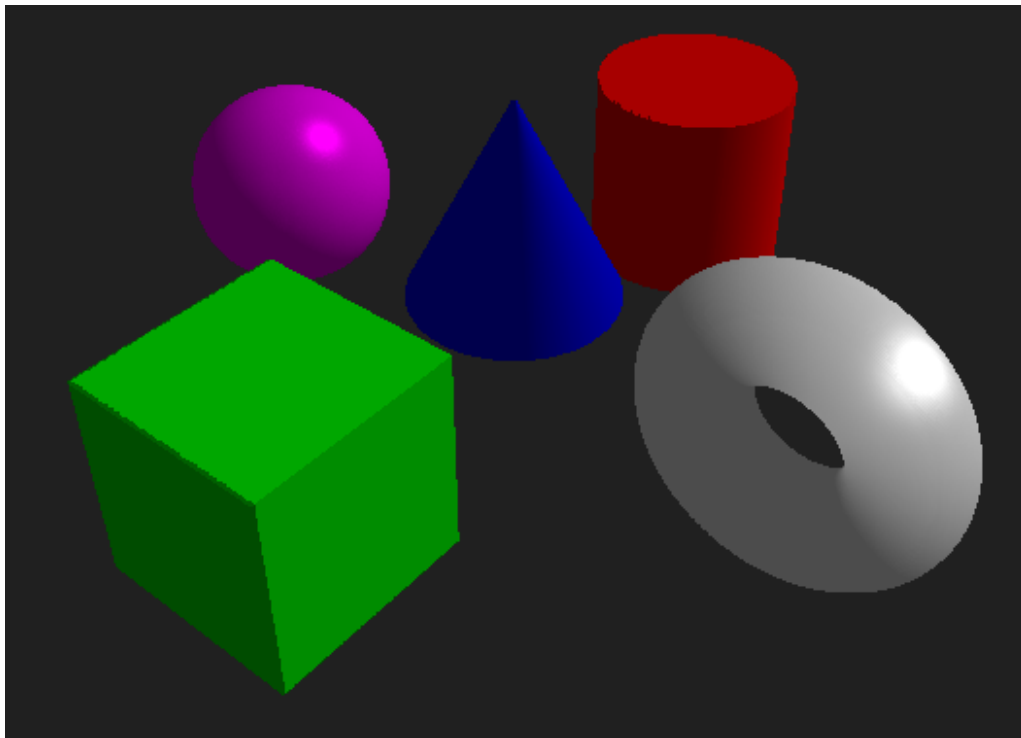
Synthèse d'image

Projet d'infographie : CSG

Table des matières

I. Objets canoniques modélisés.....	3
II. Transformations et opérations.....	3
A. Transformations.....	3
B. Opérations.....	4
III. Arbres CSG.....	4
IV. Parseur.....	4
V. Exemples.....	5
A. Dé à 6 faces.....	5
B. Jeu d'enfant.....	5
C. Tasse.....	5
D. Chaise à 3 pieds.....	6
E. Fleur.....	6
VI. Mode d'emploi.....	6

I. Objets canoniques modélisés



Voici les formes qui ont été réalisées :

- La Sphère de rayon 1,
- Le Cube de côté 2,
- Le Tore de rayon 0,5 et la distance entre le centre du tube et le centre du tore est de 1,
- Le Cylindre de hauteur 2 et de rayon de 1,
- Le Cône de hauteur 2 et de rayon de 1,

Les formes ont été réalisées de manière régulière, en respectant les équations paramétriques. Par contre la densité ne se gère pas dynamiquement, nous l'avons fixée par défaut à 450.

II. Transformations et opérations

A. Transformations

Voici les transformations qui ont été réalisées :

- Translation
- Homothétie,
- Rotation sur l'axe X (resp. Y, resp. Z).

Pour nos transformations, nous tenons à jour des matrices stockées dans chaque nœud de l'arbre.

A chaque nouvelle transformation demandée nous créons les matrices correspondantes (directe, inverse et normale) et nous les multiplions avec les matrices de transformations déjà stockées dans le nœud.

Une fois que toutes les opérations ont été ajoutées et juste avant de dessiner la racine de notre arbre, nous appliquons pour chaque nœud sa matrice directe sur tous les points de son objet, pour appliquer les transformations.

(Toutes nos transformations ont été entièrement ré-écrites, nous ne faisons qu'utiliser des types de la librairie G3X)

B. Opérations

Voici les opérations qui ont été réalisées :

- Union (*),
- Intersection (+),
- Soustraction (-).

Ces opérations ont été réalisées en utilisant les `pt_in` des objets.

Au moment de dessiner la racine de l'arbre, nous vérifions pour toutes les feuilles si leurs points appartiennent à la racine en l'arbre, des appels récursifs et des opérateurs booléens s'occupent alors de représenter les différentes opérations. (voir fonction **`bool pt_in_node(Csg_node * n, G3Xpoint * p)`** de `csg.c`)

III. Arbres CSG

L'arbre csg a été réalisé avec la structure de nœud donné en cours. Un nœud contient un objet, un opérateur, ses fils et les matrices directes, inverses et normales permettant de lui appliquer des transformations.

Un nœud interne ne contient pas d'objet, seulement des opérateurs (union *, intersection +, soustraction -). Les objets sont eux stockés dans les feuilles.

Une transformation ajoutée sur une feuille modifiera directement les matrices de la feuille alors qu'une transformation ajoutée sur un nœud interne ira récursivement l'ajouter sur toute les feuilles de ce nœud.

Nous ne stockons aucun objet entre les feuilles et la racine.

La création de l'objet de la racine se fait au cours d'un parcours de l'arbre juste avant de dessiner, l'objet de la racine est alors seulement composé des points des feuilles ayant favorablement répondu au `pt_in` sur la racine de l'arbre. Nous avons fait le choix de sauvegarder l'objet final pour éviter le re-calcul de tous les points à chaque changement de position de la caméra et donc pour gagner en efficacité.

IV. Parseur

Le format du fichier est le suivant : **`forme,couleur,R:axe:angle,H:x:y:z,T:x:y:z`**

Avec :

- pour **forme** les trois premières lettres de la forme,
- pour **couleur** les valeur de rgba séparé par des :,
- pour effectuer une rotation mettre **R** suivie de l'axe et de l'angle le rotation séparer par des :,
- pour effectuer une homothétie mettre **H** suivie des valeurs du vecteur séparé par des :,
- pour effectuer une translation mettre **T** suivie des valeurs du vecteur séparé par des :.

Dont chaque champs est séparé par des virgules.

Pour crée une forme il est obligatoire d'avoir au moins sa couleur, et respecter l'ordre des transformations.

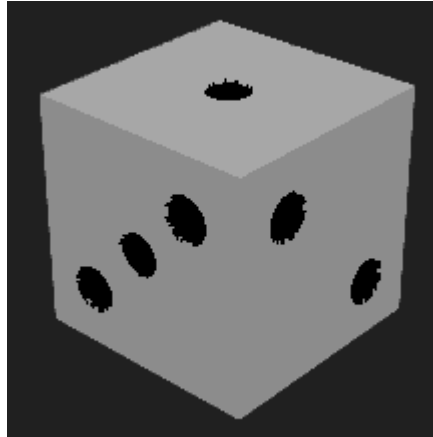
Par exemple pour dessiner un losange de couleur blanche translaté grâce au vecteur (0,0,2) et faire pivoter sur l'axe z de 45 degré: **`cub,1.00:1.00:1.00:0.00,R:z:45,T:0:0:2`**

Par défaut les différentes parties de l'objet final garderont les couleurs qui leur auront été attribué à la créations des différents objets, mais si un résultat d'une unique couleur est souhaité, il est possible de mettre une couleur en dernière ligne du fichier a parser, et l'affichage final sera de cette couleur.

V. Exemples

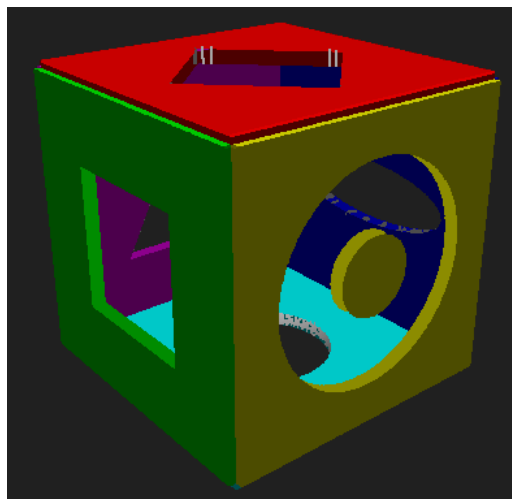
A. Dé à 6 faces

Cet exemple est composée de 22 formes (sphères et cube), de plusieurs soustractions.



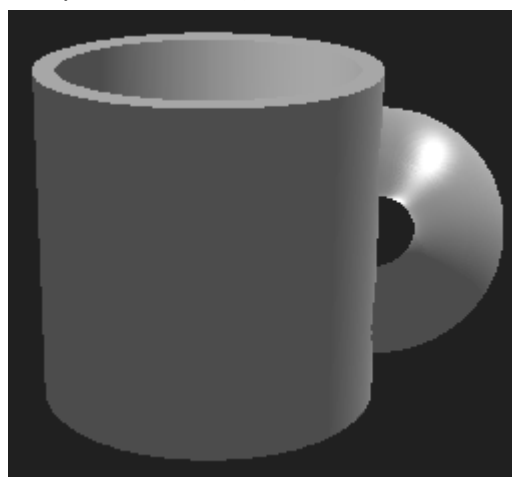
B. Jeu d'enfant

Cet exemple est composée de 12 formes (sphères, cubes, cône et tore), de plusieurs unions et une soustraction.



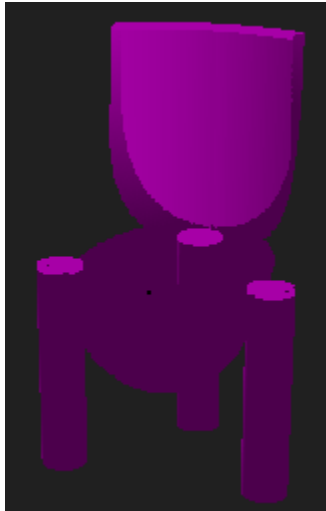
C. Tasse

Cet exemple est composée de 3 formes (cylindres et tore), d'une unions et une soustraction.



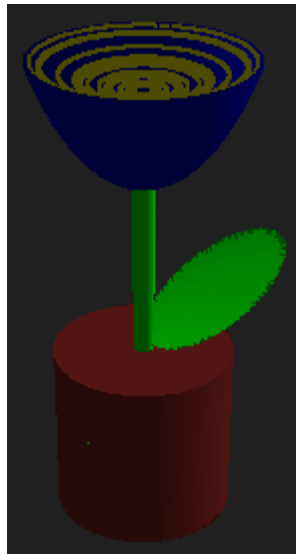
D. Chaise à 3 pieds

Cet exemple est composée de 7 formes (cylindres, sphère et tore), de plusieurs unions et une soustraction.



E. Fleur

Cet exemple est composée de 6 formes (cylindres et sphères), de plusieurs unions, une soustraction et une intersection.



VI. Mode d'emploi

Lancer le makefile : **make**

Pour dessiner un des exemples il suffit de taper : **./main exemples/fleur.txt**