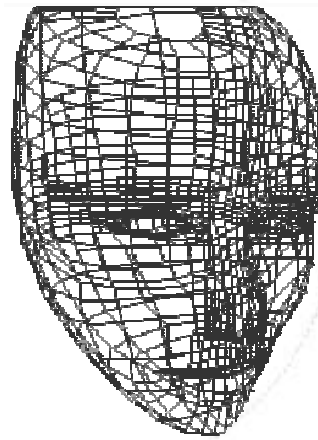


Valeur C et DEA  
*Conception d'applications multimédia*

# Synthèse d'images

## Polyèdres, surfaces d'élévation, facettage



Alexandre Topol

Département Informatique  
Conservatoire National des Arts & Métiers

2001-2002

## 2. Polyèdres, surfaces d'élévation, facettage

---

2.1. Définitions

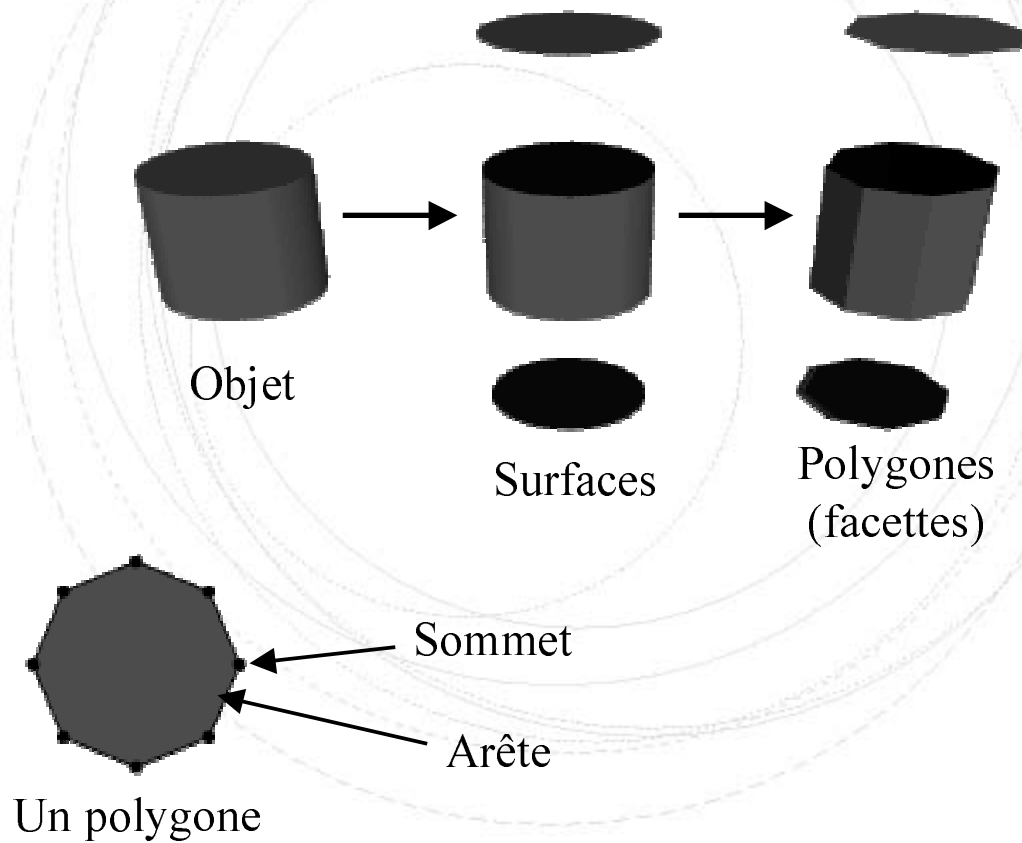
2.2. Représentations

2.3. Surfaces d'élévation

2.4. Facettage d'une forme non plane

2.5. Exercices

## 2 1. Définitions



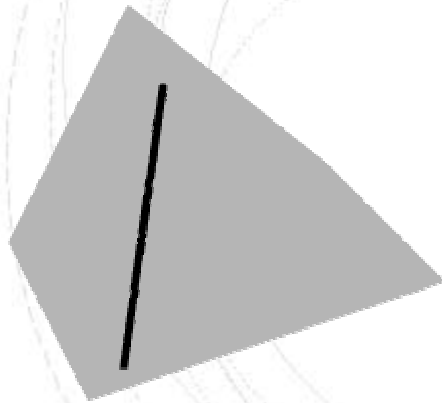
## 2.1. Définitions

---

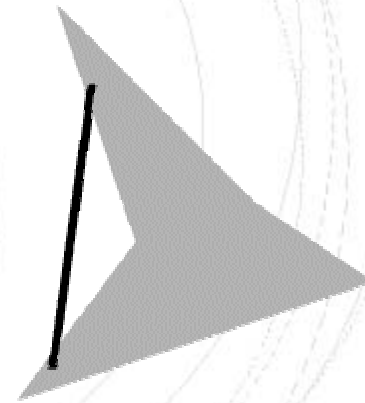
- Polyèdre
  - "solide limité de toutes parts par des polygones plans."  
(Petit Robert)
  - "solide limité par des polygones plans dont chaque côté (arête) est commun à exactement deux polygones."  
(Quid )
- Polyèdre convexe :
  - "Polyèdre situé tout entier du même côté du plan de chacune de ses faces" (Quid)
  - "Une droite reliant n'importe quels points de l'intérieur du polyèdre reste à l'intérieur" (Cubaud)

## 2.1. Définitions

---



Polygone convexe



Polygone concave

## 2.1. Définitions

---

- Formule d'Euler pour tout polyèdre convexe

S = nombre de sommets

A = nombre d'arêtes

F = nombre de faces

On a :  $S + F = A + 2$

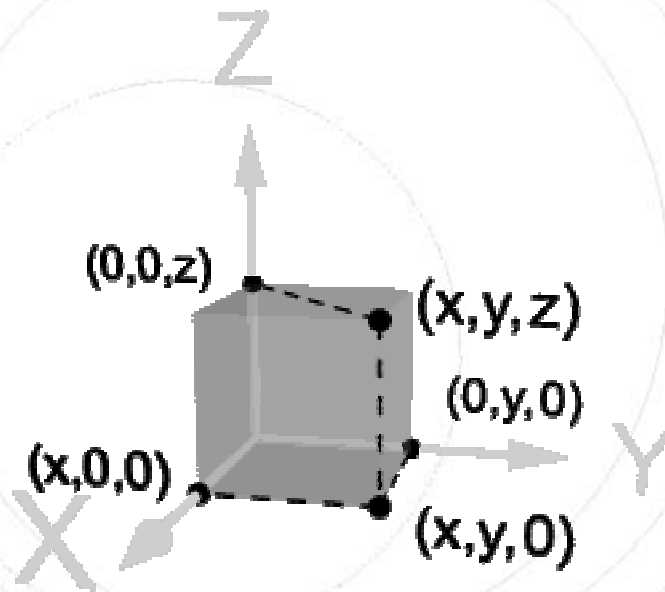
exemple : cube :  $S=8, F=6, A=12$

⇒ Utile pour détecter un oubli dans la description.

- Condition nécessaire mais pas suffisante de convexité !!!
- En plus : au plus 3 arêtes ont un sommet donné en commun

## 2.2. Représentations Système de coordonnées

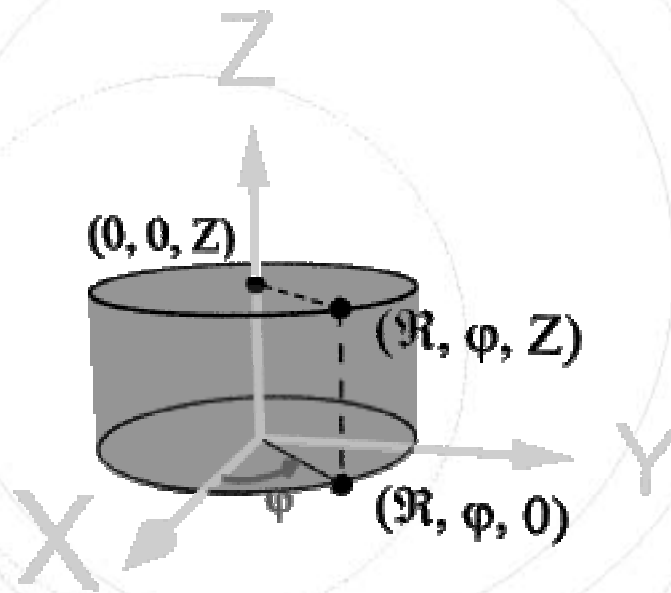
---



Système cartésien

## 2.2. Représentations Système de coordonnées

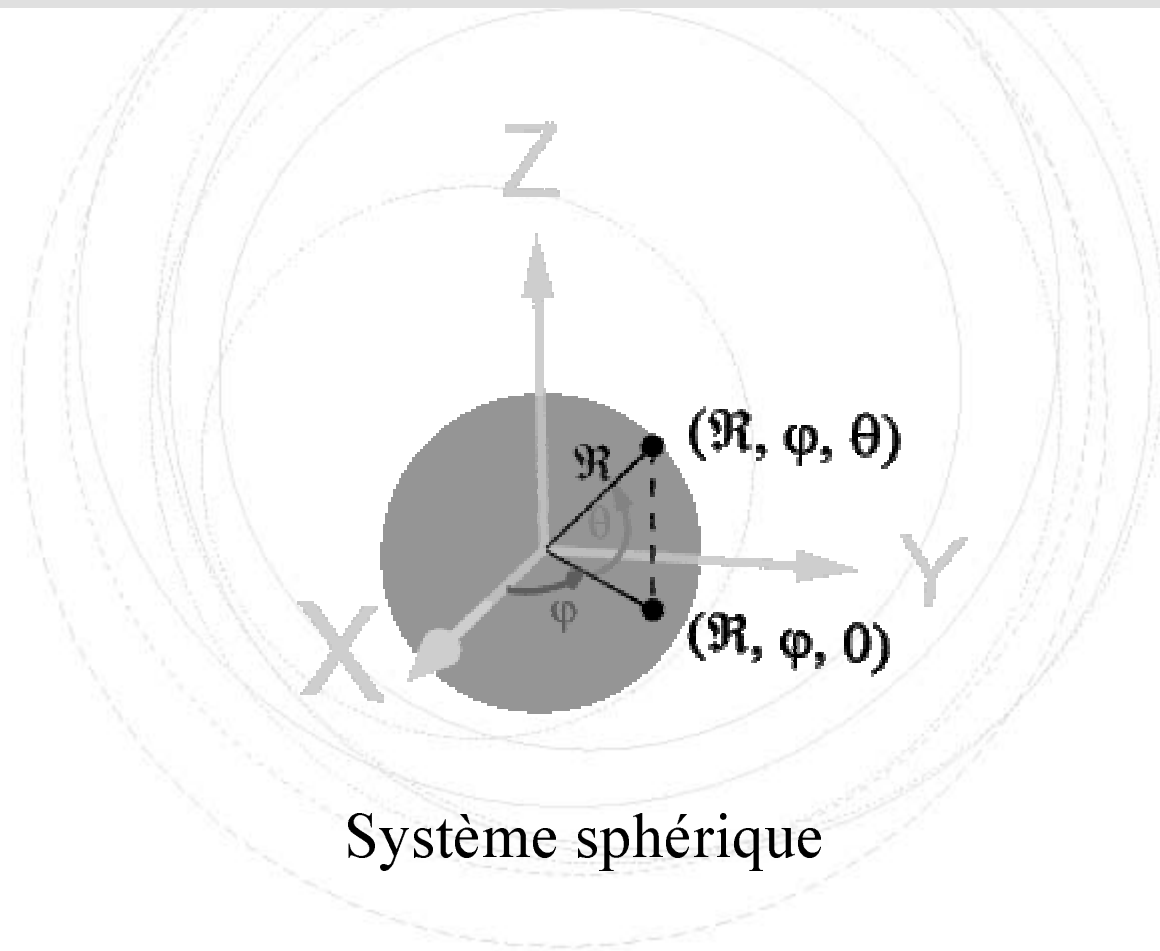
---



Système cylindrique ou polaire

## 2.2. Représentations Système de coordonnées

---

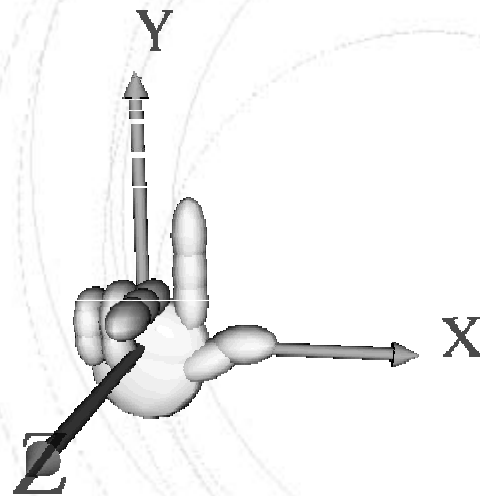


Système sphérique

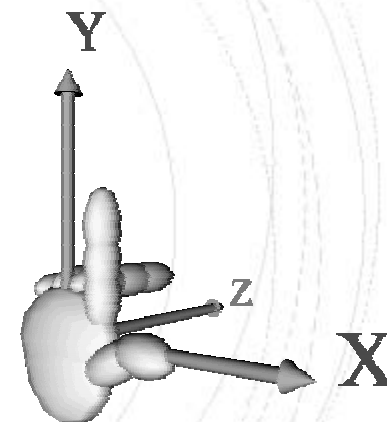
## 2.2. Représentations

### Système de coordonnées cartésiens

---

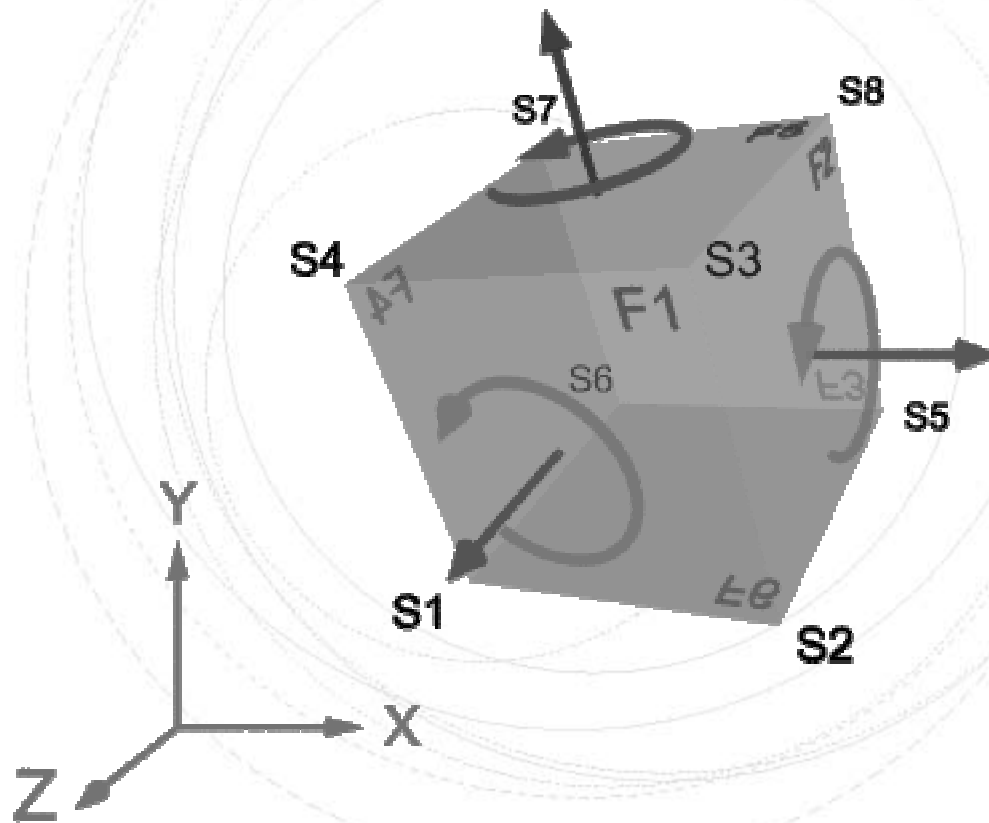


Système direct  
Repère main droite



Système indirect  
Repère main gauche

## 2.2. Représentations



## 2.2. Représentations

### Représentation explicite

---

- Toute face  $F$  est décrite par une liste de sommets :  
$$F = ((x_1, y_1, z_1), \dots, (x_n, y_n, z_n))$$
- Les sommets sont donnés dans l'ordre pour décrire la frontière de la face, et on suppose qu'il y a une arête entre  $(x_n, y_n, z_n)$  et  $(x_1, y_1, z_1)$
- Problèmes :
  - Gaspillage, les arêtes communes sont décrites deux fois
  - Comment "prendre" le polyèdre par un de ses sommets sans connaître les faces concernées ?

## 2.2. Représentations

### Représentation par liste de sommets

---

- Méthode la plus commune (Direct3D, VRML, PovRay, ...)
- On gère deux tables :

TS = table des sommets et TF = table des faces

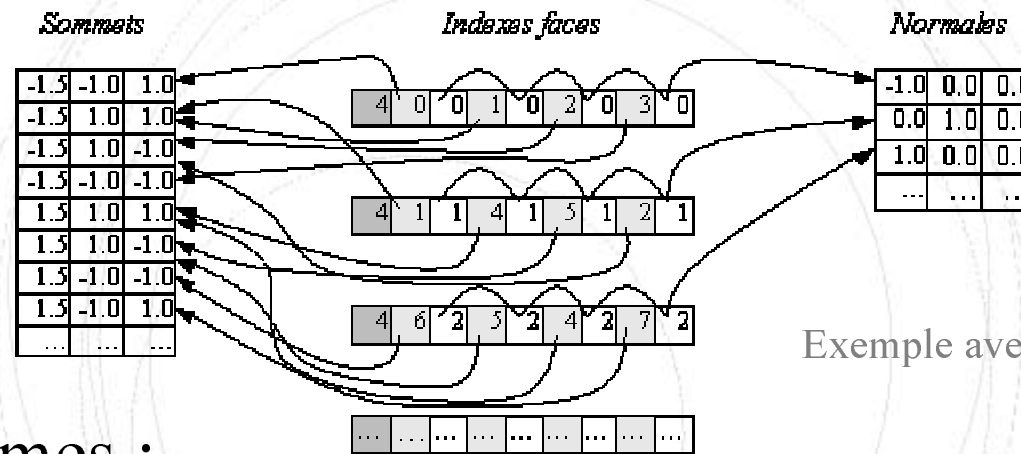
Exemple du cube précédent :

TS	X	Y	Z
S1	-0.5	-0.5	0.5
S2	0.5	-0.5	0.5
S3	0.5	0.5	0.5
S4	-0.5	0.5	0.5
S5	0.5	-0.5	-0.5
S6	-0.5	-0.5	-0.5
S7	-0.5	0.5	-0.5
S7	0.5	0.5	-0.5

TF	Liste des sommets
F1	S1, S2, S3, S4
F2	S2, S5, S8, S3
F3	S5, S6, S7, S8
F4	S6, S1, S4, S7
F5	S4, S3, S8, S7
F6	S6, S5, S2, S1

## 2.2. Représentations

### Représentation par liste de sommets



Exemple avec Direct3D

- Problèmes :
  - Spécifier les sommets des faces **toujours** dans le même sens de parcours (trigonométrique / anti horlogique)
  - Les arêtes communes sont toujours dessinées deux fois
  - Comment retrouver facilement quelles faces ont telle arête en commun ?

## 2.2. Représentations

### Représentation par liste d'arêtes

---

- On gère 3 tables :
  - En plus de TS et TF, TA = table des arêtes
- Exemple du cube précédent :

TA	Sommets & Faces
A1	S1, S2, F1, F6
A2	S2, S3, F1, F2
A3	S3, S4, F1, F5
A4	S4, S1, F1, F4
A5	S2, S5, F2, F6
...	...

- Résolution des problèmes précédents

## 2.3. Surfaces d'élévation

---

- Une grille où chaque point représente une altitude
- Très utilisé en topographie (cartographie)
  - Ex du format DEM (*Digital Elevation Model*)
  - US Geological Survey  
<http://www.usgs.gov>
- Nombreux logiciels de visualisation :
  - MICRODEM : pour Windows 95 et NT
  - DEM Reader : pour macintos  
[http://home.san.rr.com/bwagner/DEM\\_Reader.html](http://home.san.rr.com/bwagner/DEM_Reader.html)

## 2.3. Surfaces d'élévation

### Exemple avec POV

```
#include "colors.inc"

camera {
    location <-25,5,-30>
    angle 37
    look_at <-4,-3,4>
}

light_source { < -10,20,-25> color White }

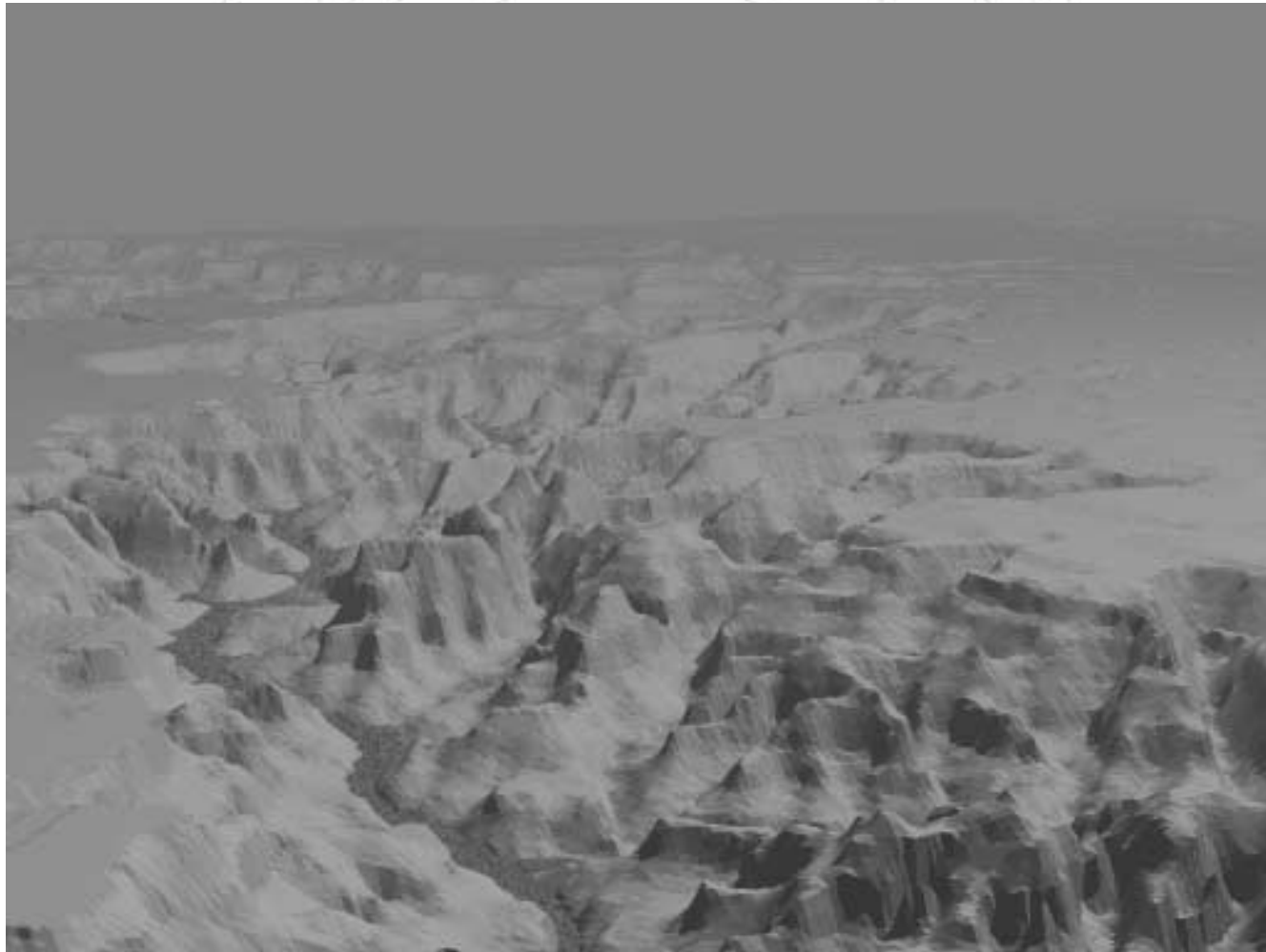
object {
    height_field {
        gif "grandcan.gif"
        scale <50,2,50>
        translate<-25,0,-25>
    }
    pigment { White }
}
```



## 2.3. Surfaces d'élévation

### Exemple avec POV

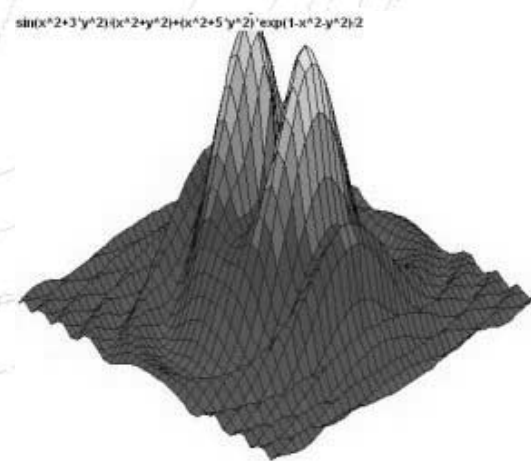
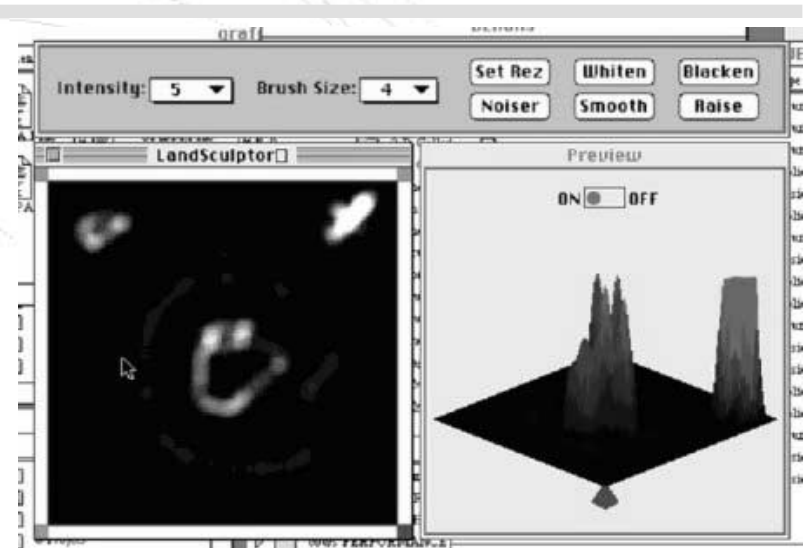
---



## 2.3. Surfaces d'élévation

### Autres méthodes

- Logiciels de génération manuelle
  - Ex : "LandSculptor3D" pour Macintosh
- Par équations de surfaces :  $z=f(x,y)$ 
  - Ex : "3D Surfaces" pour Macintosh



## 2.4. Facettage d'une forme non plane

---

- Les algorithmes de rendu sont souvent limités aux faces planes (polyèdres)
  - Cas de OpenGL pour les sphères et les surfaces paramétrées (Bezier)
  - ⇒ les surfaces plus complexes doivent être approximées ("tessalated")

## 2.4. Facettage d'une forme non plane

### Algorithme naïf pour la sphère

---

```

procedure drawsphere(xc,yc,zc,r:real);
var r1,r2,h:real;      i,j:integer;      p1,p2,p3,p4:vector;
begin
  r1:=r;h:=0;
  for j:=1 to 4 do begin
    h:=h+r/4;
    r2:=sqrt(r*r-h*h);
    for i:=0 to 31 do begin
      p1[1]:=r1*cos32[i]+xc;p1[2]:=r1*sin32[i]+yc;p1[3]:=zc+h-r/4;
      p2[1]:=r1*cos32[i+1]+xc;p2[2]:=r1*sin32[i+1]+yc;p2[3]:=zc+h-r/4;
      p3[1]:=r2*cos32[i]+xc;p3[2]:=r2*sin32[i]+yc;p3[3]:=zc+h;
      p4[1]:=r2*cos32[i+1]+xc;p4[2]:=r2*sin32[i+1]+yc;p4[3]:=zc+h;
      line3d(p1,p2);line3d(p2,p4);line3d(p4,p3);line3d(p3,p1);
      p1[3]:=zc-h+r/4;p2[3]:=zc-h+r/4;p3[3]:=zc-h;p4[3]:=zc-h;
      line3d(p1,p2);line3d(p2,p4);line3d(p4,p3);line3d(p3,p1);
    end;
    r1:=r2
  end
end;
end;

```



## 2.4. Facettage d'une forme non plane Un algorithme plus efficace

---

- On peut le source pour générer un fichier POV ...

```
camera {
  location <0.0 , 0.0 ,-5.0>
  look_at  <0.0 , 0.0 , 0.0>
}

union {
  polygon { 4, <1, 0, 0>, <0, 0, 1>, <0, 1, 0>, <1, 0, 0> }
  polygon { 4, <0, 1, 0>, <0, 0, 1>, <-1, 0, 0>, <0, 1, 0> }
  polygon { 4, <-1, 0, 0>, <0, 0, 1>, <0, -1, 0>, <-1, 0, 0> }
  polygon { 4, <0, -1, 0>, <0, 0, 1>, <1, 0, 0>, <0, -1, 0> }
  polygon { 4, <1, 0, 0>, <0, 1, 0>, <0, 0, -1>, <1, 0, 0> }
  polygon { 4, <0, 1, 0>, <-1, 0, 0>, <0, 0, -1>, <0, 1, 0> }
  polygon { 4, <-1, 0, 0>, <0, -1, 0>, <0, 0, -1>, <-1, 0, 0> }
  polygon { 4, <0, -1, 0>, <1, 0, 0>, <0, 0, -1>, <0, -1, 0> }
  pigment { rgb <1, 0, 0> }
}
```

## 2.4. Facettage d'une forme non plane

### Un algorithme plus efficace

---

- ou un fichier VRML

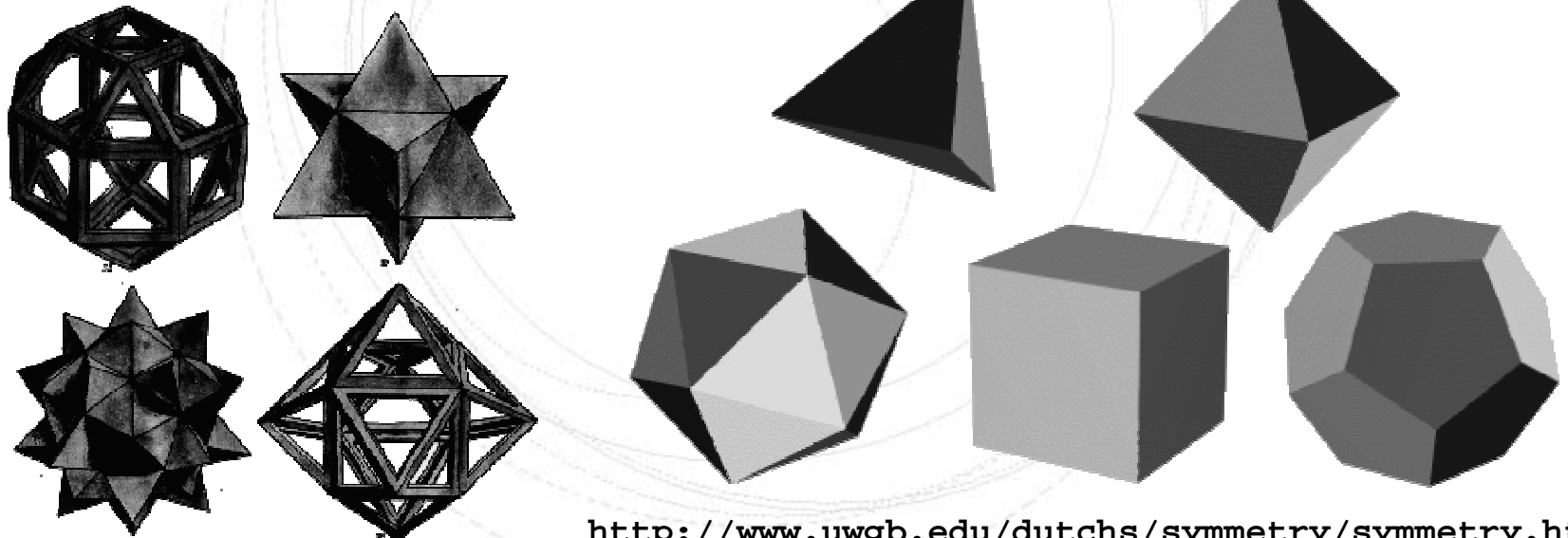
```
#VRML V2.0 utf8

DirectionalLight { direction 0 0 -10}

Shape {
  geometry IndexedFaceSet {
    coord Coordinate {
      point [1 0 0, 0 0 1, 0 1 0, 0 1 0, 0 0 1, -1 0 0, -1 0 0 ,
            0 0 1, 0 -1 0, 0 -1 0, 0 0 1, 1 0 0, 1 0 0, 0 1 0 ,
            0 0 -1, 0 1 0,-1 0 0, 0 0 -1, -1 0 0, 0 -1 0, 0 0 -1,
            0 -1 0, 1 0 0, 0 0 -1 ]
    }
    coordIndex [ 0,1,2,-1, 3,4,5,-1, 6,7,8,-1, 9,10,11,-1,
                12,13,14,-1,15,16,17,-1,18,19,20,-1, 21,22,23,-1 ]
  }
  appearance Appearance {
    material Material { diffuseColor 1 0 0 }
  }
}
```

## 2.5. Exercices

- Calculer les coordonnées des 5 polyèdres réguliers (platoniques), des polyèdres archimédiens, des solides de Johnson...

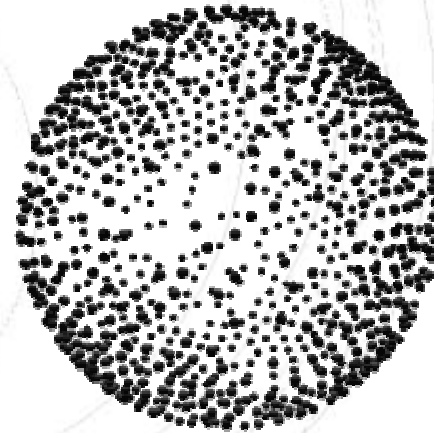


<http://www.uwgb.edu/dutchs/symmetry/symmetry.htm>

attribués à Léonard de Vinci in C.A. Reichen  
"Histoire de la physique" Ed. Rencontre, 1963.

## 2.5. Exercices

- Appliquer une autre technique de triangulation de la sphère :
  - on pose  $N$  points au hasard sur la sphère
  - ceux-ci se "répulsent" selon une loi en  $1/d^2$
  - on arrête quand le critère de convergence est vérifié



sources :

comp.graphics.algorithms

"Sphere FAQ" du groupe sci.math

<http://www1.ics.uci.edu/~epstein/junkyard/software.html>

## 2.5. Exercices

- Modèles de cristaux

source : LAPPARENT "Précis de minéralogie", Blanchard, 1965 (Nouveau tirage)

