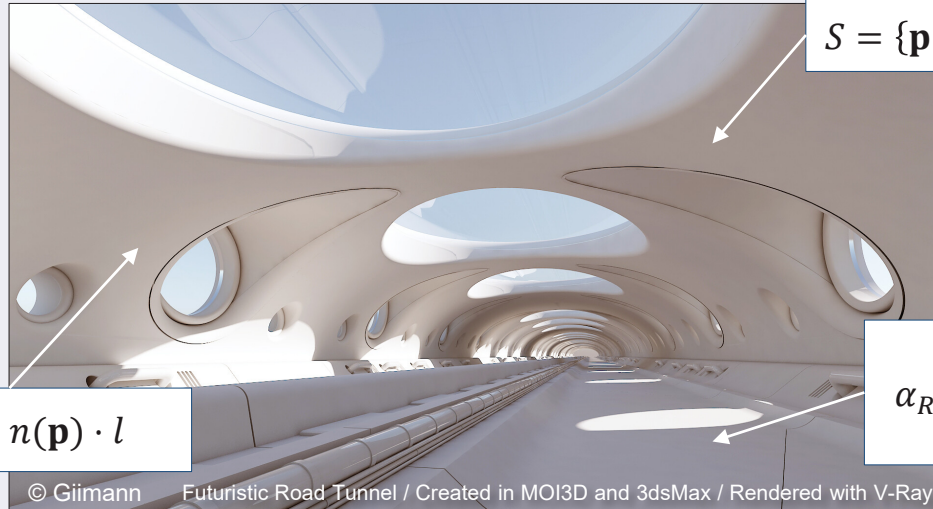


# Computer Graphics

From mathematics ...



$$S = \{\mathbf{p} \in \mathbf{R}^3, f(\mathbf{p}) = 0\}$$

$$d(\mathbf{p}) = n(\mathbf{p}) \cdot l$$

$$\alpha_R(\mathbf{p}) \approx \frac{1}{n} \sum_{i=0}^n \delta_i$$

© Giimann Futuristic Road Tunnel / Created in MOI3D and 3dsMax / Rendered with V-Ray

... to the screen

E. Galin  
Université Lyon 1

# Computer Graphics

Mathematics

**Modeling**

Color and Texturing

Shading

Realistic Rendering

# Introduction

## Classification

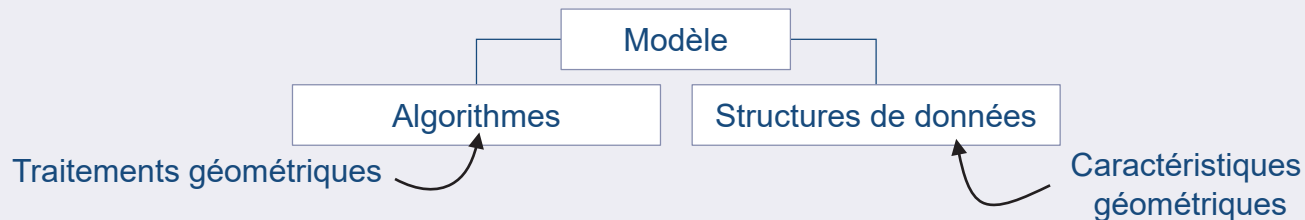
Surfaces

Volumes

Conclusion

## Modélisation géométrique

Description de la forme et des propriétés géométriques et topologiques de l'objet  
Modèles volumiques ou surfaciques



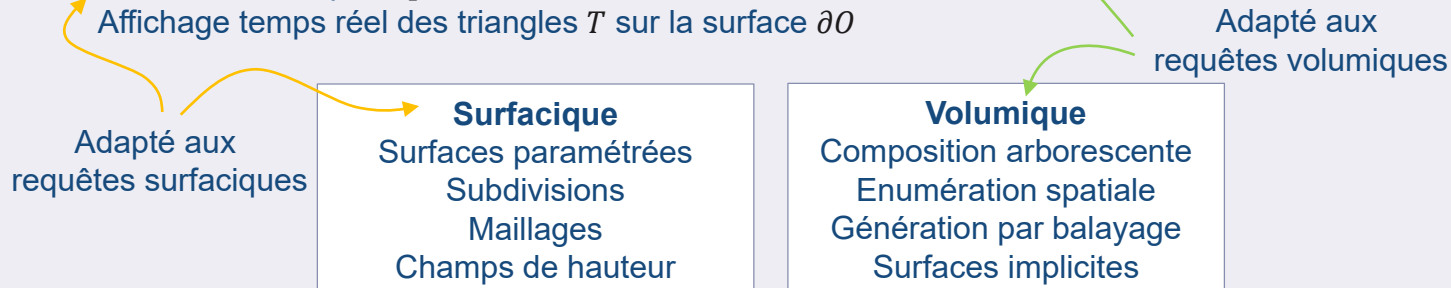
## Traitements géométriques élémentaires

Intersection avec une droite  $\Delta \cap O$  ou entre objets  $O \cap O$

Position d'un point par rapport au modèle  $p \in O$

Recherche d'un point  $p$  sur la surface  $\partial O$

Affichage temps réel des triangles  $T$  sur la surface  $\partial O$



Il n'existe **pas** de modèle universel



eric.galin@liris.cnrs.fr  
<http://liris.cnrs.fr/~egalain>



Billboards  
Rendering

Billboard & Line  
Rendering & Animation

Textured Mesh  
Rendering

Volume  
Collision

Navigation mesh  
Animation

Textured Mesh  
Rendering

# Computer Graphics

## Surface models



# Surfaces paramétrées

Classification

Surfaces

Volumes

Conclusion

## Définition

On définit  $S = \{\mathbf{p}(u, v), (u, v) \in \Omega\}$

Différents types de surfaces selon les fonctions  $\mathbf{p}(u, v)$

## Construction

$\mathbf{p}(u, v)$  est défini par des polynômes

Construction à partir de points de contrôle  $\mathbf{p}_{ij}$

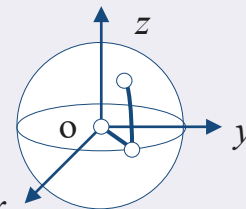
$$\mathbf{p}(u, v) = \sum_{(i,j) \in [0,n]^2} B_i^n(u) B_j^n(v) \mathbf{p}_{ij}$$

$$\mathbf{p}(u, v) = (u^3 \ u^2 \ u \ 1) \mathbf{M} [\mathbf{p}_{ij}] \mathbf{M} (v^3 \ v^2 \ v \ 1)^t$$

## Propriétés

**Régularité** des carreaux

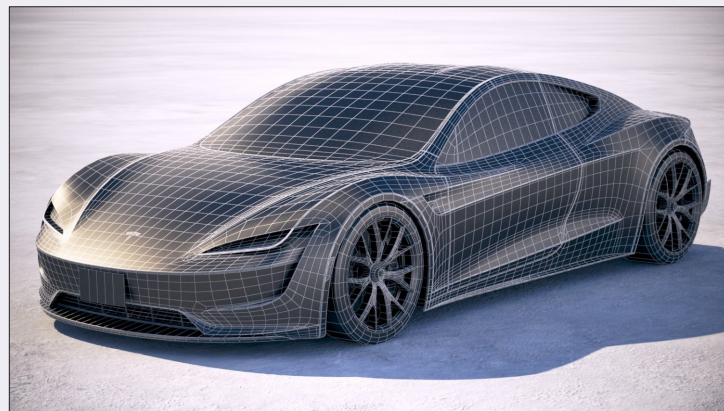
Conversion en maillage trivial



Sphère paramétrée

$$\mathbf{p}(\theta, \phi) = (\cos \theta \cos \phi, \sin \theta \cos \phi, \sin \phi)$$

$$(\theta, \phi) \in [0, 2\pi] \times [-\pi/2, \pi/2]$$



eric.galin@liris.cnrs.fr

http://liris.cnrs.fr/~egalin

# Maillages géométriques

Classification

Surfaces

Volumes

Conclusion

## Maillages triangulaires

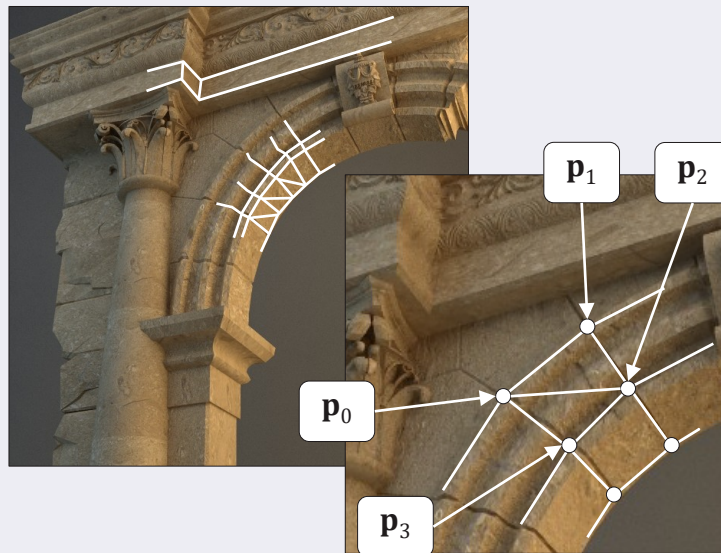
Géométrie G : sommets (normales)

Topologie T : connectivité entre sommets, arêtes, faces

## Structure minimale

Sommets connectés en triangles : T fixe

Triplets  $\{a, b, c\}$  ou double triplet  $\{a, n_a, b, n_b, c, n_c\}$  pour chaque triangle



Géométrie G

$p_0$

$p_1$

$p_2$

$p_3$

Topologie T

0 3 2

1 0 2

```
class Mesh {  
    std::vector<Vector> p ;           // Vertexes  
    std::vector<int> t ;             // Indexes  
};
```



eric.galin@liris.cnrs.fr

<http://liris.cnrs.fr/~egalin>

# Maillages faible résolution

Classification

Surfaces

Volumes

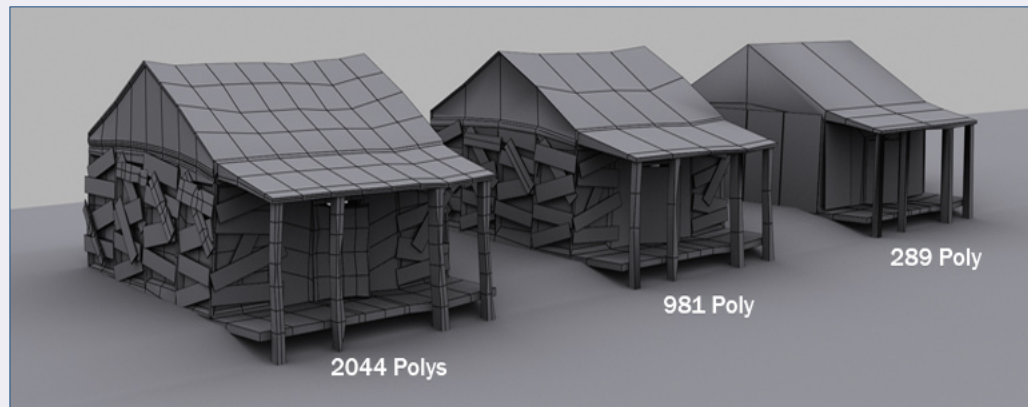
Conclusion

## Fondamentaux

Rupture des frontières entre modélisation et texture

Maillages non conformes, facettes sans raccordements

Utilisation intensive de *billboards*



[eric.galin@liris.cnrs.fr](mailto:eric.galin@liris.cnrs.fr)

<http://liris.cnrs.fr/~egalain>



# Surfaces de subdivision

Classification

Surfaces

Volumes

Conclusion

## Définition

Surface limite  $S$  : subdivisions successives d'un maillage de contrôle  $C$   
Schémas de raffinement de maillage

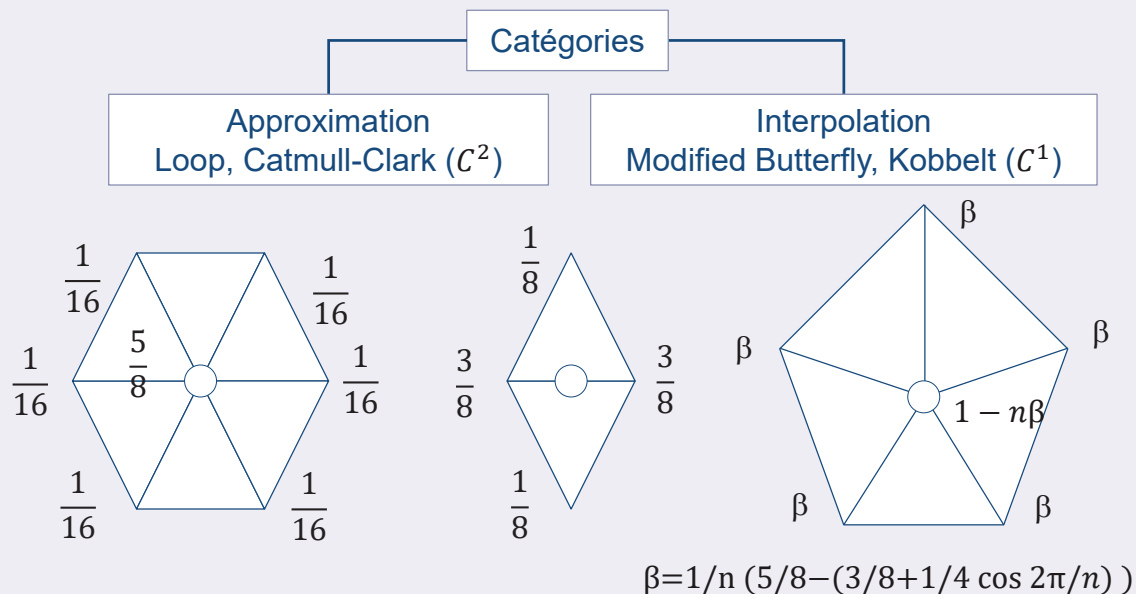
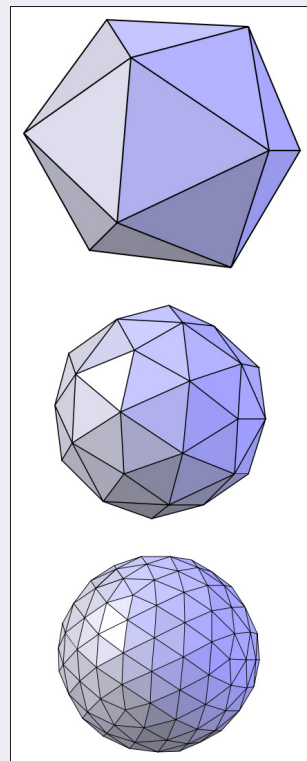


Schéma de Loop



eric.galin@liris.cnrs.fr

http://liris.cnrs.fr/~egalin

E. Catmull and J. Clark. Recursively generated B-Spline surfaces on arbitrary topological meshes. *Computer Aided Design*, **10**(6):350–355, 1978.

# Surfaces de subdivision

Classification

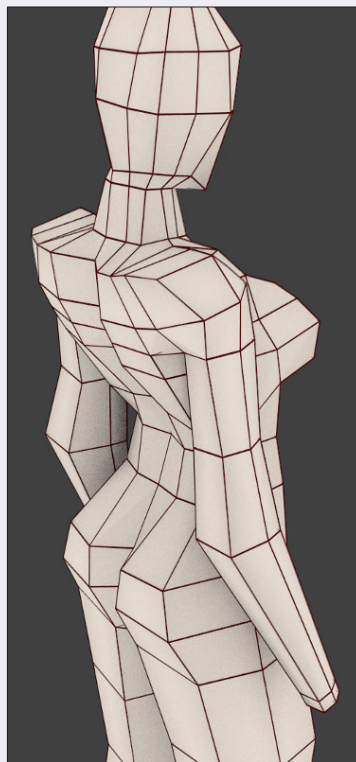
Surfaces

Volumes

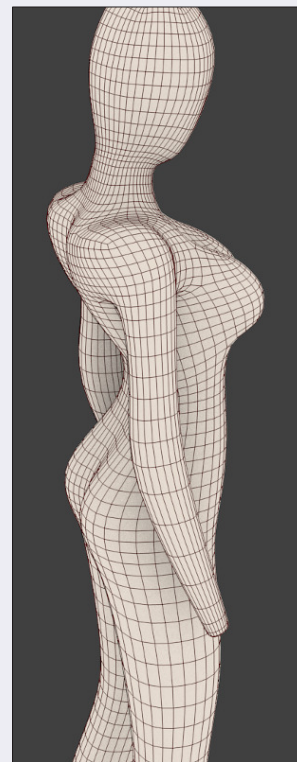
Conclusion

## Modélisation

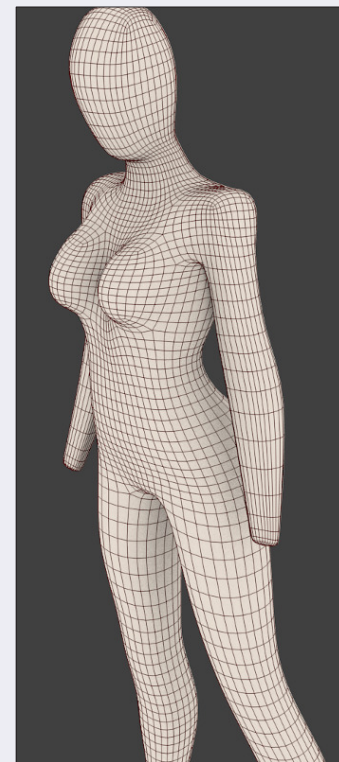
Création de formes lisses à partir d'un squelette de contrôle



Maillage grossier



Subdivision de niveau 2



# Computer Graphics

## Volumetric models

# Enumération spatiale

Classification

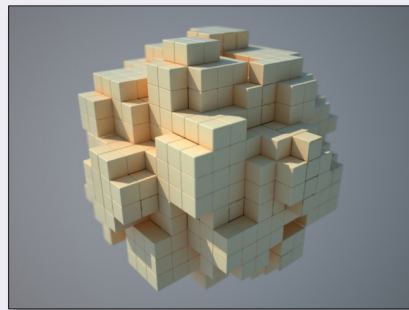
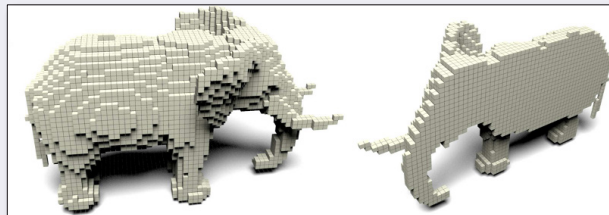
Surfaces

Volumes

Conclusion

## Structure régulière

Subdivision de l'espace en grille régulière,  
Stockage du type de matériau dans les voxels



Structure coûteuse en mémoire :  $O(n^3)$  cellules et accès efficace en  $O(1)$   
Traitements locaux efficaces, coûteux sur l'ensemble du modèle

```
class Voxel
{
protected:
    int n ;
    unsigned char* array ;
    Vector a, b ; // End vertices of cubic domain
public:
    // ...
    int Index (int, int, int) const ;
    unsigned char At(const Vector&) const ;
};
```

```
int Voxel::Index (int x,int y, int z) const {
    return x*n*n + y*n + z*n ;
}
unsigned char Voxel::At (const Vector& p) const {
    // Real coordinates inside the cubic region [0,1]
    double x=(p[0]-a[0])/(b[0]-a[0]); // Same for y , z
    int ix=int (x*n); // Same for y , z

    // Should check range here
    return array[ Index (ix,iy,iz) ];
}
```



eric.galin@liris.cnrs.fr  
http://liris.cnrs.fr/~egalain

# Enumération spatiale

Classification

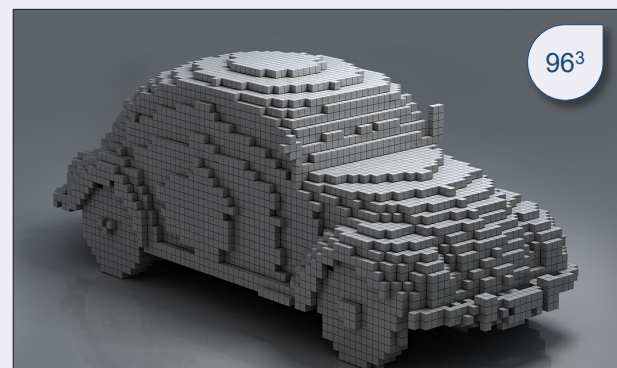
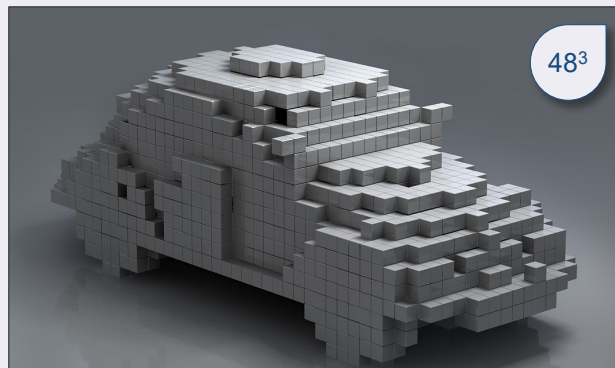
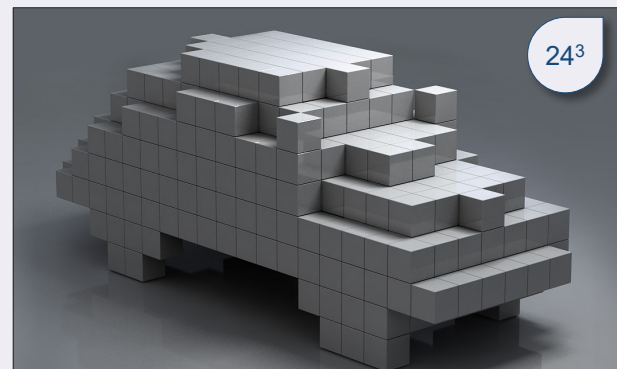
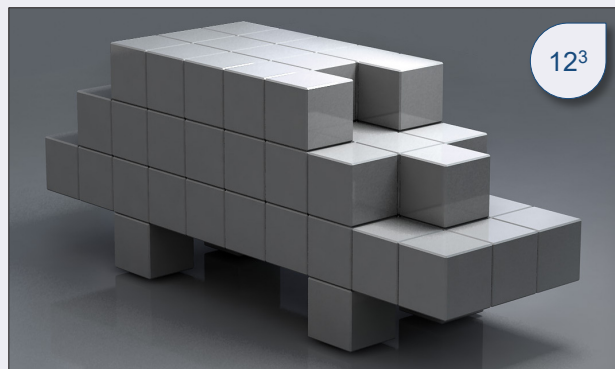
Surfaces

Volumes

Conclusion

## Précision

Même objet à différentes résolutions



eric.galin@liris.cnrs.fr  
<http://liris.cnrs.fr/~egalain>



# Enumération spatiale

Classification

Surfaces

Volumes

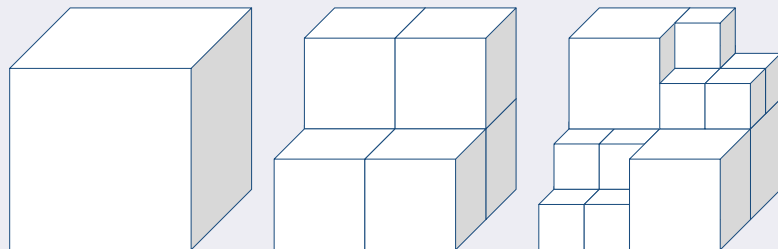
Conclusion

## Améliorations

Compression des données type run length encoding

Décomposition adaptative : octree

Sparse voxel octrees



Redondances et motifs fréquents

Directed acyclic graphs permettent de gérer des résolutions jusqu'à  $[32 - 128k]^3$



[eric.galin@liris.cnrs.fr](mailto:eric.galin@liris.cnrs.fr)

<http://liris.cnrs.fr/~egalin>

- S. Laine, T. Karras. Efficient sparse voxel octrees. *Symposium on Interactive 3D Graphics and Games*, 55–63, 2010.
- S. Laine, T. Karras. Efficient sparse voxel octrees. *IEEE Transactions on Visualization and Computer Graphics* **17**(8), 2011.
- V. Kampe, E. Sintorn, U. Assarsson. High Resolution Sparse Voxel DAGs. *Proceeding of Siggraph*, 2013.

# Construction arborescente

Classification

Surfaces

Volumes

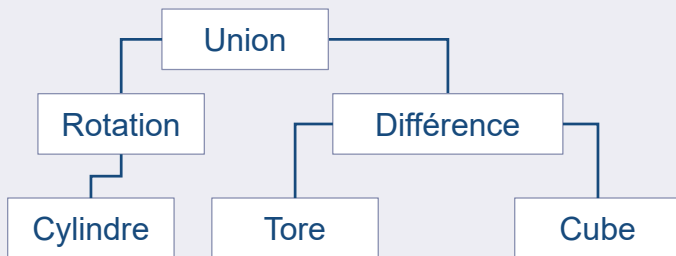
Conclusion

## Structure

Arbre de construction combinant des primitives simples

Opérateur booléens

Transformation affines



## Requêtes

Position d'un point par rapport au modèle  $\mathbf{p} \in O$  par descente récursive de l'arbre jusqu'aux feuilles

Sphère  $\mathbf{p} \in S$  si  $\|\mathbf{p} - \mathbf{c}\| < r$

Test terminal

Union  $\mathbf{p} \in A \cup B$  si  $\mathbf{p} \in A$  et  $\mathbf{p} \in B$

Récursion



eric.galin@liris.cnrs.fr

<http://liris.cnrs.fr/~egalin>

# Construction arborescente

Classification

Surfaces

Volumes

Conclusion

## Implémentation

Hiérarchie de classes avec polymorphisme

Chaque classe implémente ses méthodes spécifiques

```
class Node
{
public:
    virtual int Inside(const Vector&) const = 0;
};
```

```
class Sphere : public Node
{
protected:
    Vector c; // Center.
    double r; // Radius.
public:
    // ...
    int Inside(const Vector&) const;
};
```

```
class Union : public Node
{
protected:
    Node* a; // Left child
    Node *b; // Right child
public:
    // ...
};
```

```
int Sphere::Inside(const Vector& p) const
{
    // Center
    Vector n=c-ray[0];

    // Distance to center
    double pc=Norm (p-c);

    if ( pc<r) return 1;
    else return 0;
}
```

```
int Union::Inside(const Vector& p) const
{
    // Check children nodes
    if (a->Inside(p)) return 1;
    if (b->Inside(p)) return 1;

    return 0;
}
```



[eric.galin@liris.cnrs.fr](mailto:eric.galin@liris.cnrs.fr)

<http://liris.cnrs.fr/~egalin>

# Surfaces Implicites

Classification

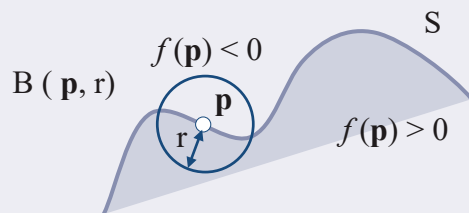
Surfaces

Volumes

Conclusion

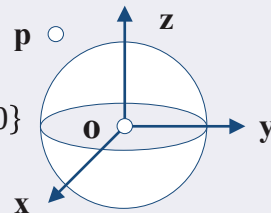
## Définition

On définit  $S = \{\mathbf{p} \in \mathbf{R}^3 \mid f(\mathbf{p}) = 0\}$



## Sphère implicite

$$S = \{(x, y, z) \in \mathbf{R}^3 \mid 1 - x^2 - y^2 - z^2 = 0\}$$



## Propriétés

Modélisation de formes lisses (raccordement)

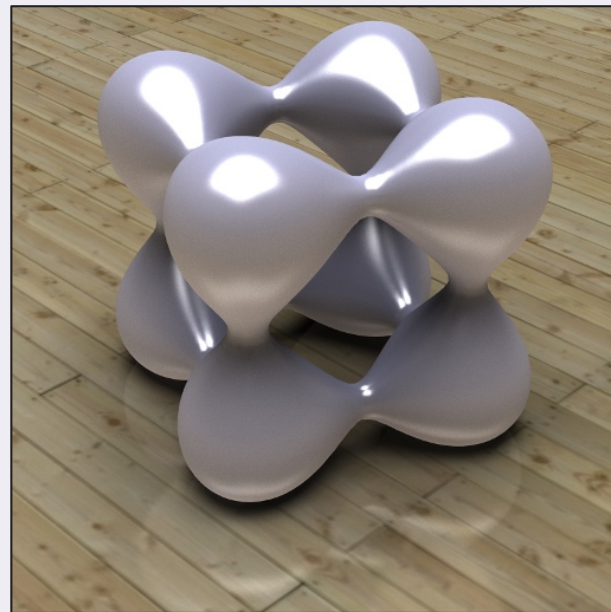
Objets de topologie variable

Visualisation complexe

## Catégories

Surfaces implicites à squelettes

Surfaces variationnelles (nuages de points)



B. Wyvill, A. Guy and E. Galin. Extending the CSG Tree (Warping, Blending and Boolean Operations in an Implicit Surface Modeling System). *Computer Graphics Forum*, 18 (2), 149-158, 1999.



eric.galin@liris.cnrs.fr  
<http://liris.cnrs.fr/~egalin>

# Computer Graphics

## Supplementary material



