

# Modelação de Sólidos

Sistemas Gráficos/

Computação Gráfica e Interfaces

# Modelação de Sólidos

- Em 2D um conjunto de segmentos de recta ou curvas não formam necessariamente uma área fechada.
- Em 3D uma colecção de superfícies não envolve necessariamente um volume fechado.

## Modelação de Sólidos:

Em algumas aplicações é importante: distinguir entre o interior, exterior e superfície de um objecto 3D; e calcular propriedades dos objectos que dependem dessa distinção.

Ex: Simulação de mecanismos, cálculo de volume, centro de massa, aplicação de elementos finitos para determinar a resposta a factores como esforço e temperatura, etc.

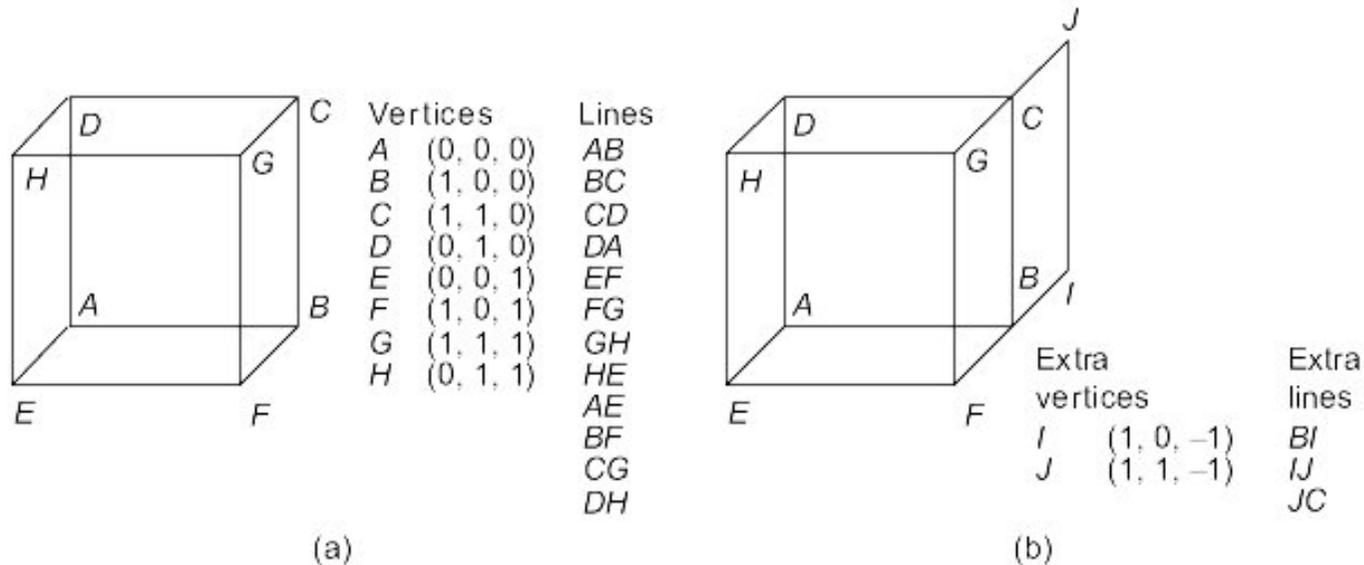
Aplicações: CAD/CAM e criação de imagens *photo-realistic*.

# Características de um modelo de sólido

1. Deve abranger um **domínio de representação** suficientemente abrangente para incorporar todo o tipo de objectos que pretendemos modelar.
2. A representação deve ser **não ambígua** e **única**: uma dada representação deve corresponder a um único sólido; e cada objecto deve ter apenas uma representação possível. A representação única permite comparar dois objectos para determinar a igualdade.
3. **Preciso/Exacto**: uma modelação exacta permite representar o objecto sem aproximações. Sistemas que aceitem apenas representação por segmentos de recta aproximam superfícies curvas.
4. Impossibilidade de criar **objectos inválidos**, i.e. que não correspondem a um sólido.
5. Representação **fechada**: a representação deve manter-se válida depois da aplicação de transformações geométricas. Por exemplo, se houver duplicação de vértices, ao aplicar transformações geométricas podemos obter valores diferentes para o mesmo ponto
6. Representação **compacta** para otimizar a utilização de memória.

# Características de um modelo de sólido

Exemplo de objectos não válidos como sólido.

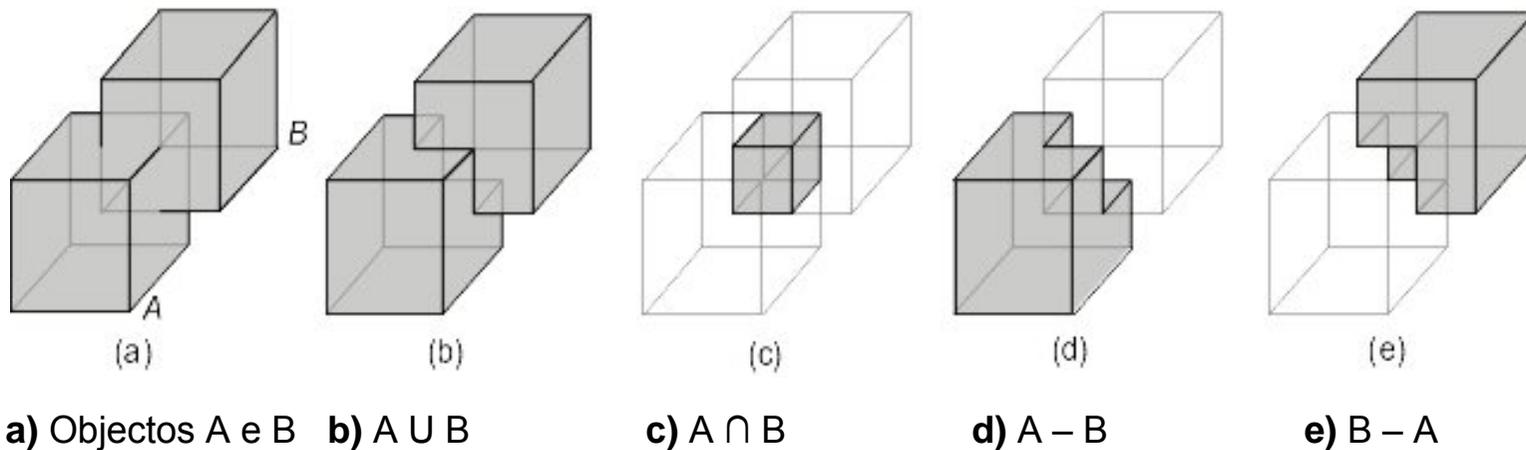


- A representação de **a)** não identifica claramente as faces do cubo, apenas indica arestas.
- Podemos considerar que uma sequência de 4 segmentos formam uma face?  
Mas o sólido **b)** seria erradamente considerado sólido.

Em geral as representações usadas não possuem todas as características apresentadas, sendo escolhido o modelo de acordo com as características do objecto a modelar.

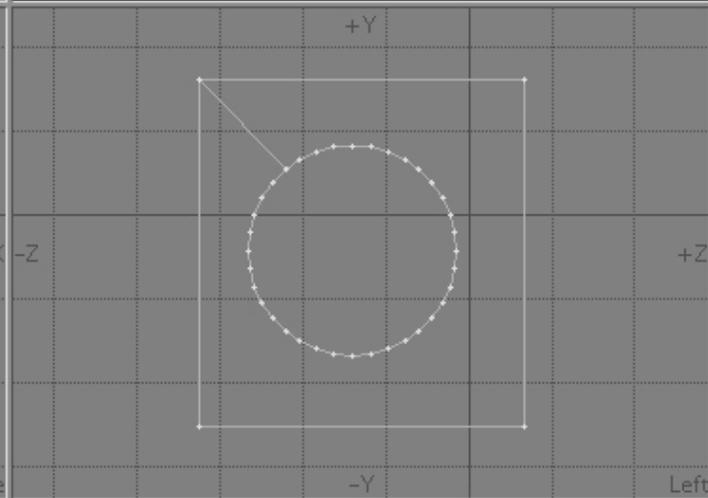
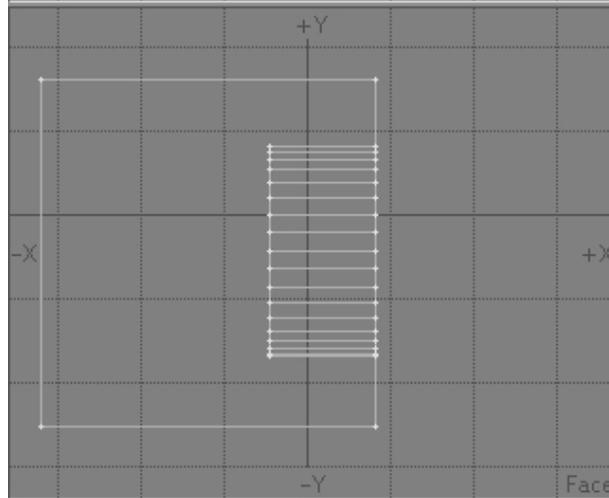
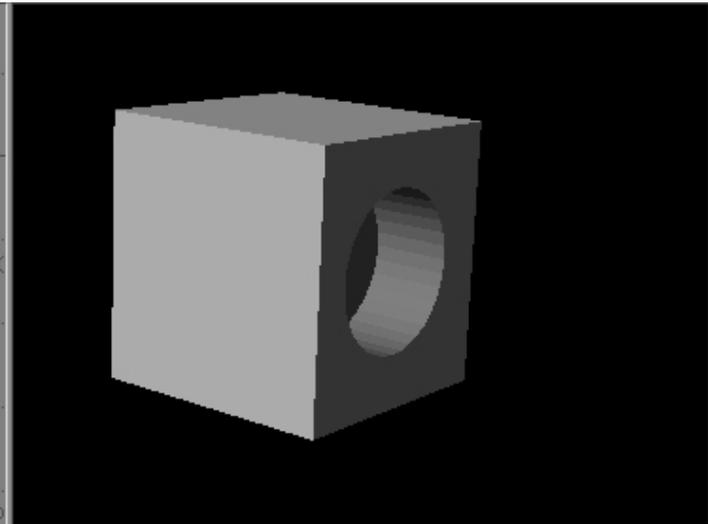
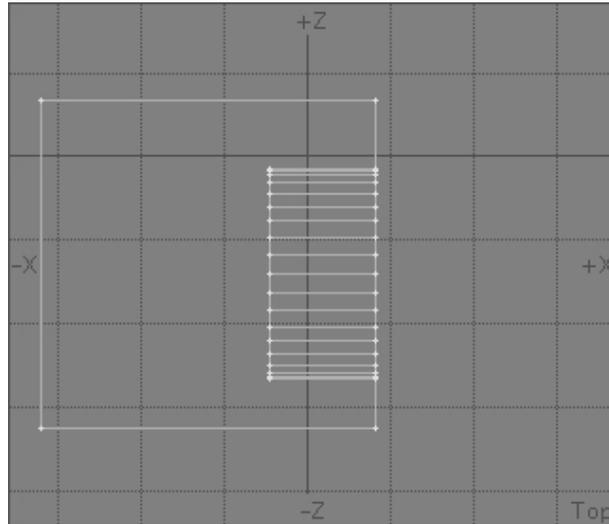
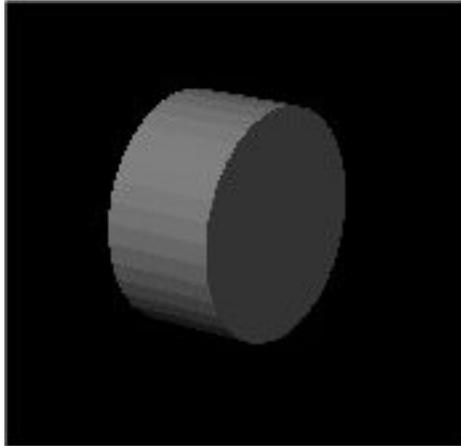
# Operações Booleanas

A combinação de objectos por operações booleanas permite definir novos objectos, independentemente da representação usada. As operações são **união**, **diferença** e **intercepção**.



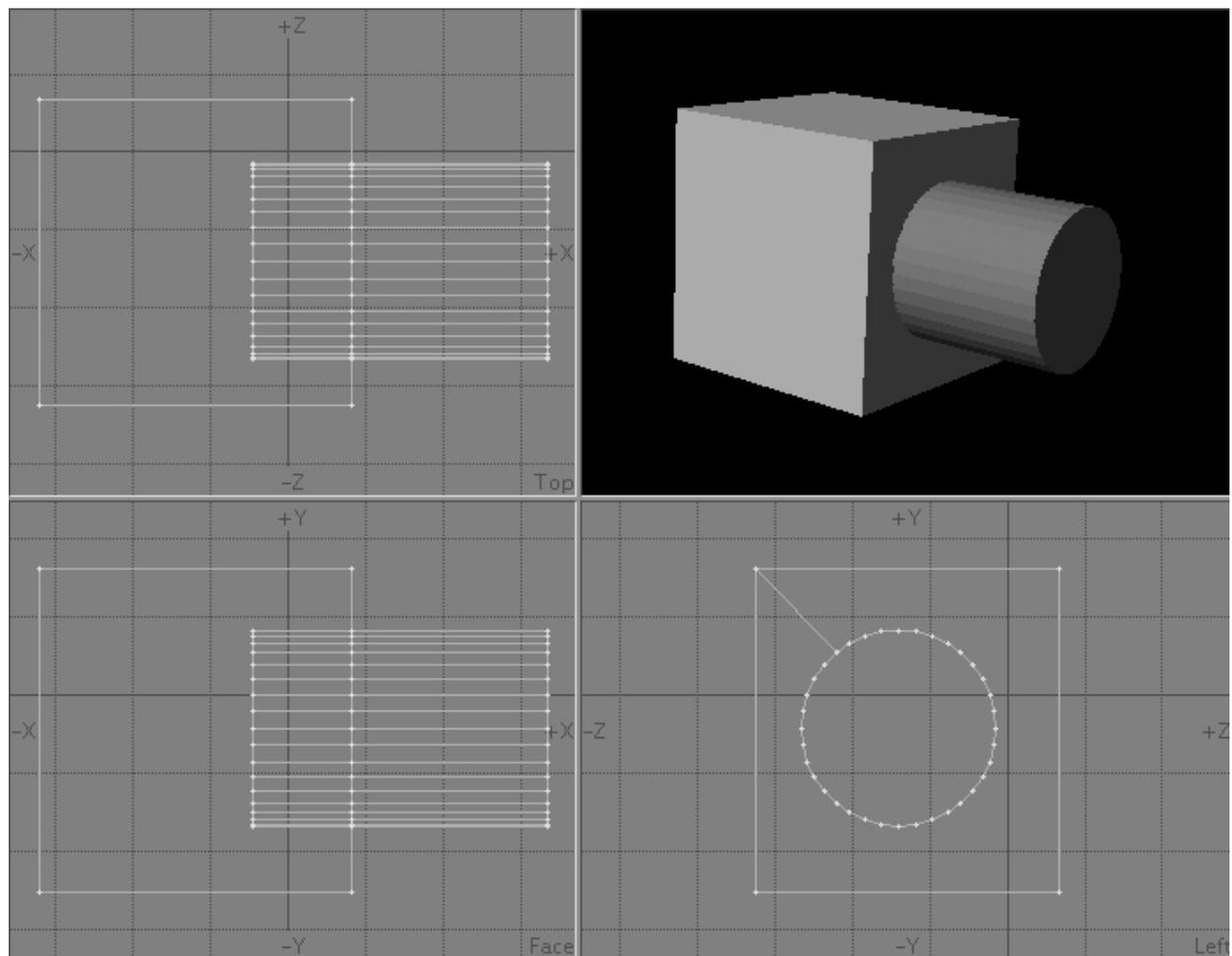
# Subtracção

Subtrai o volume de um objecto a outro.



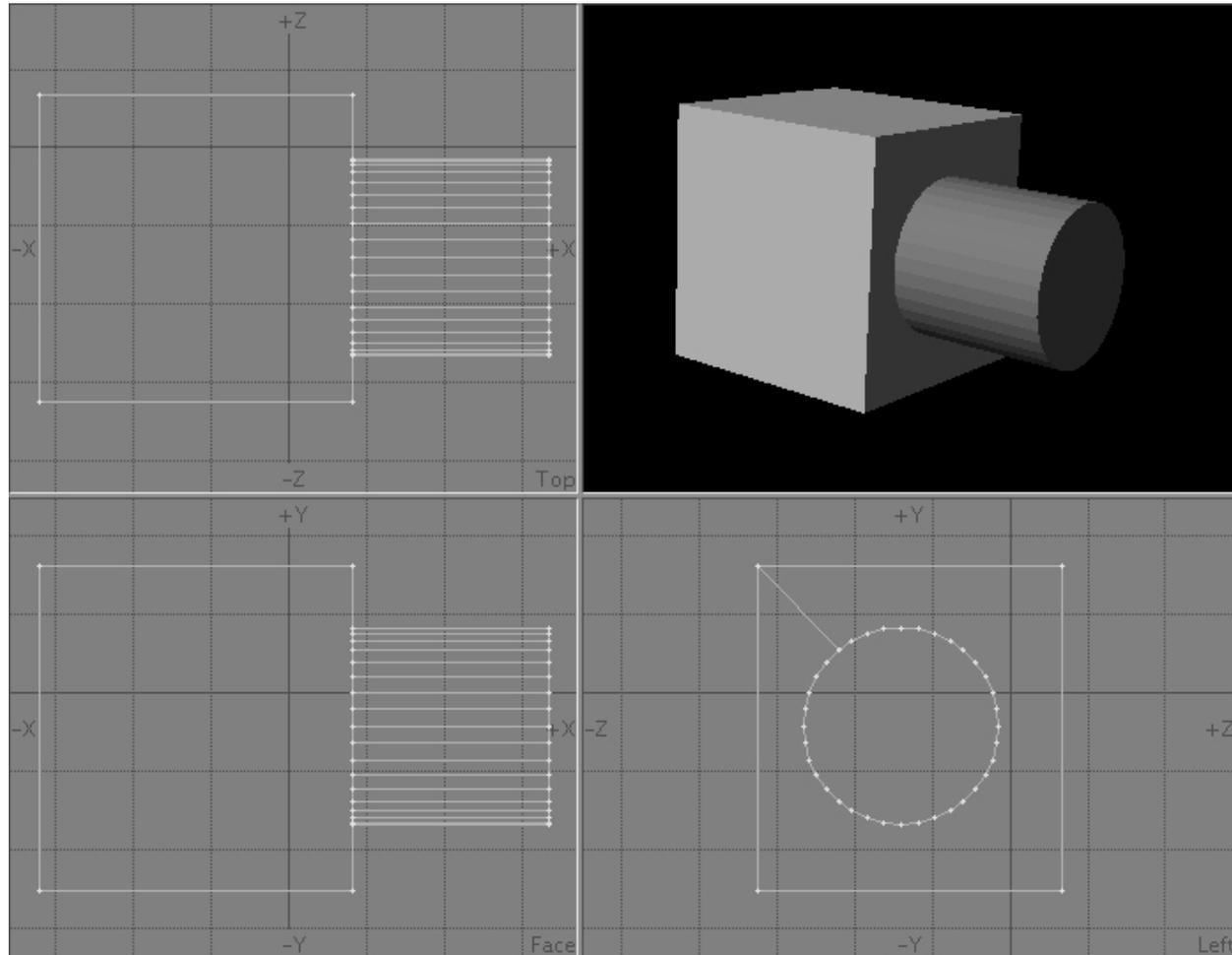
# Agrupamento de objectos

Adiciona os dois volumes. Não é retirada a parte da malha sobreposta.

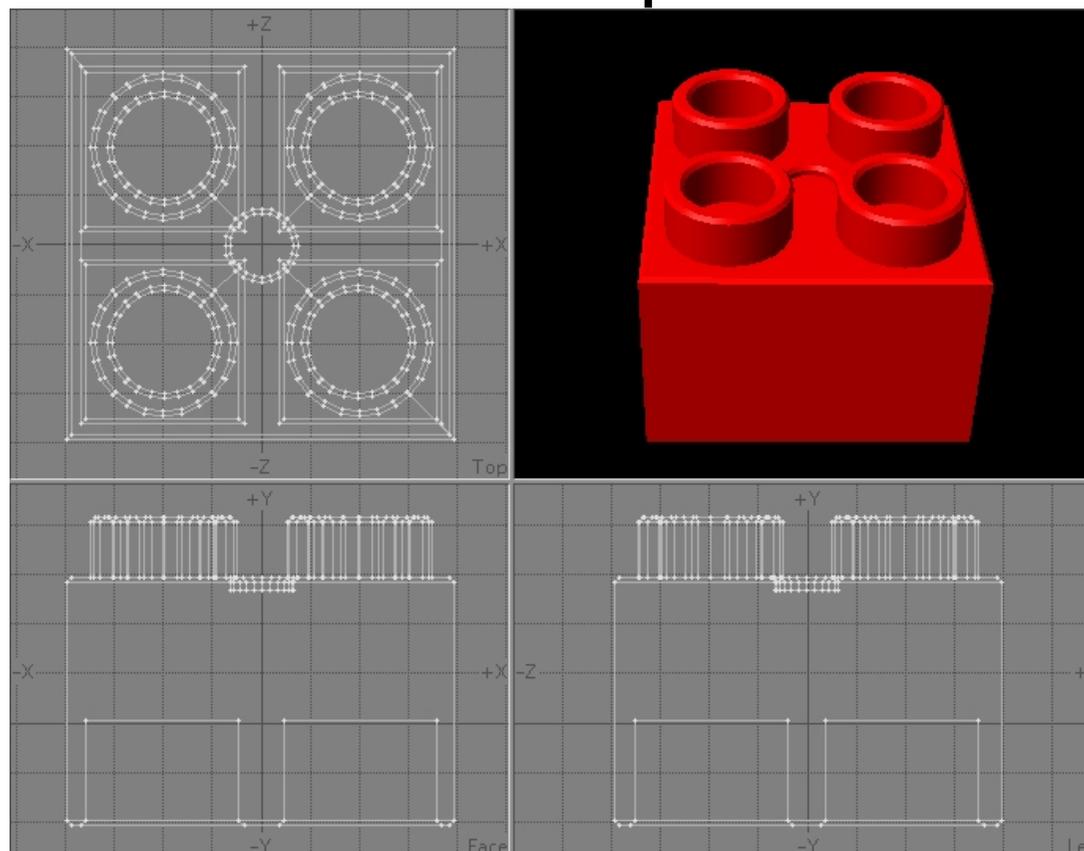


# União

Adiciona os dois volumes e remove a parte da malha sobreposta.



# Exemplo



- Peça de Lego realizada com operações booleanas. Os sólidos usados são cubo e cilindro.

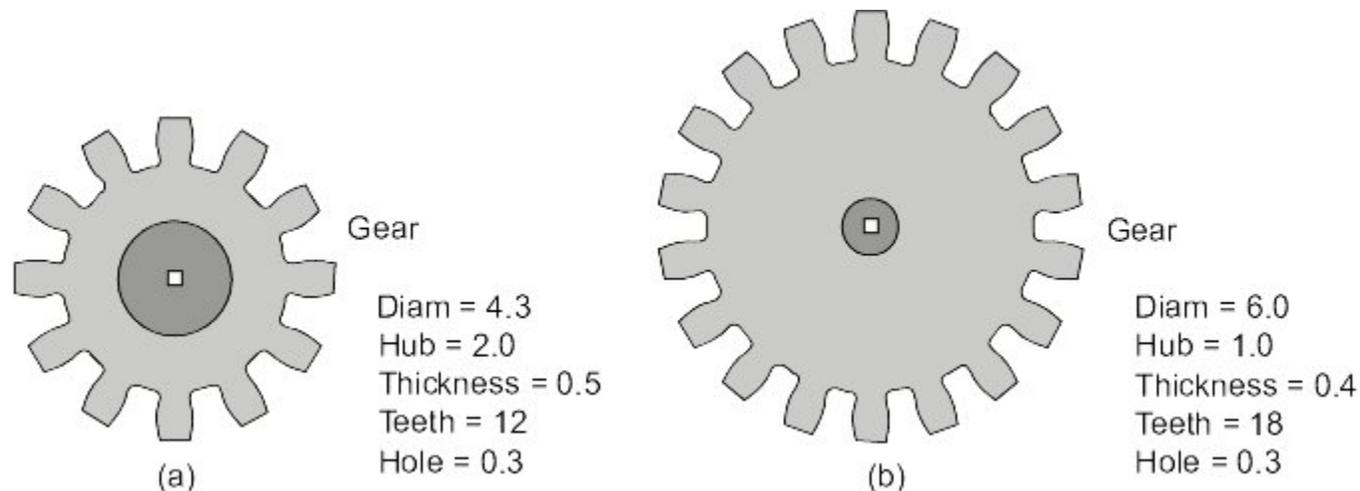
Este tipo de modelação é essencialmente usado para objectos regulares como o exemplificado.

# Tipos de Representação

1. Representação por Instanciação de Primitivas
2. Representação por Varrimento
3. Representação pela Fronteira (Boundary Representation)
4. Representação por Decomposição Espacial
5. Representação Construtiva (CSG)

# Representação por Instanciação de Primitivas

O sistema de modelação tem pré-definido um conjunto de sólidos 3D úteis para a modelação pretendida.



O utilizador pode controlar a forma do objecto definindo os parâmetros que o caracterizam.

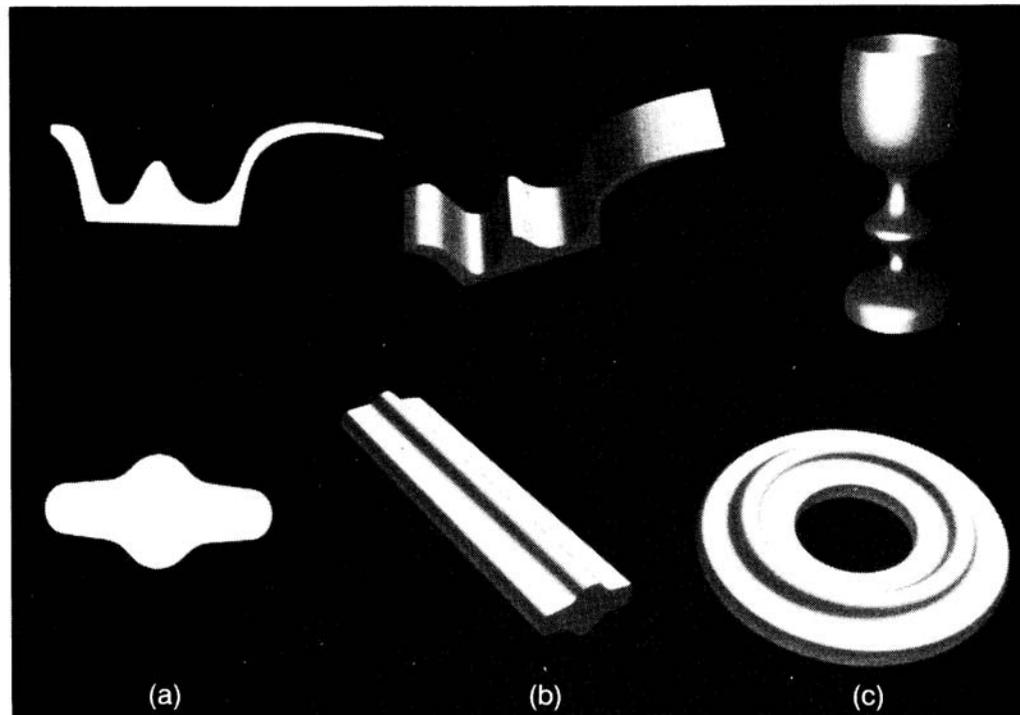
Não prevê a combinação de objectos como por exemplo por operações booleanas. Aplicado para peças complexas.

# Representação por Varrimento

O deslocamento de um objecto segundo uma trajectória define um outro objecto:

- **Translação** (extrusão)
- **Rotação**

Ex: A **translação** de um rectângulo 2D ao longo de uma trajectória perpendicular ao plano do mesmo cria um objecto paralelepípedo. Uma extensão simples consiste em variar a dimensão do objecto 2D ao longo da trajectória.

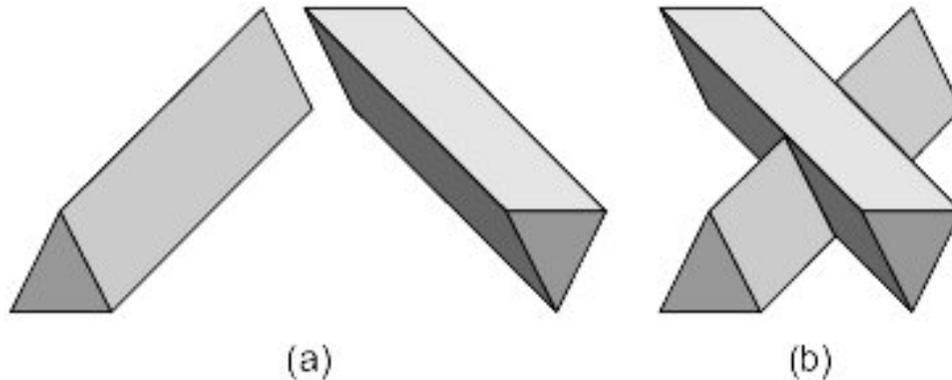


# Representação por Varrimento

- A utilização deste método sem restrições de trajectória pode resultar numa **modelação ineficiente** do objecto.

Ex: Se o objecto se intersecta a si mesmo dificulta o cálculo de volume.

- Pode não gerar um sólido válido se o movimento for no plano.
- Em geral as aplicações convertem os objectos criados por varrimento para uma outra das representações de objectos.

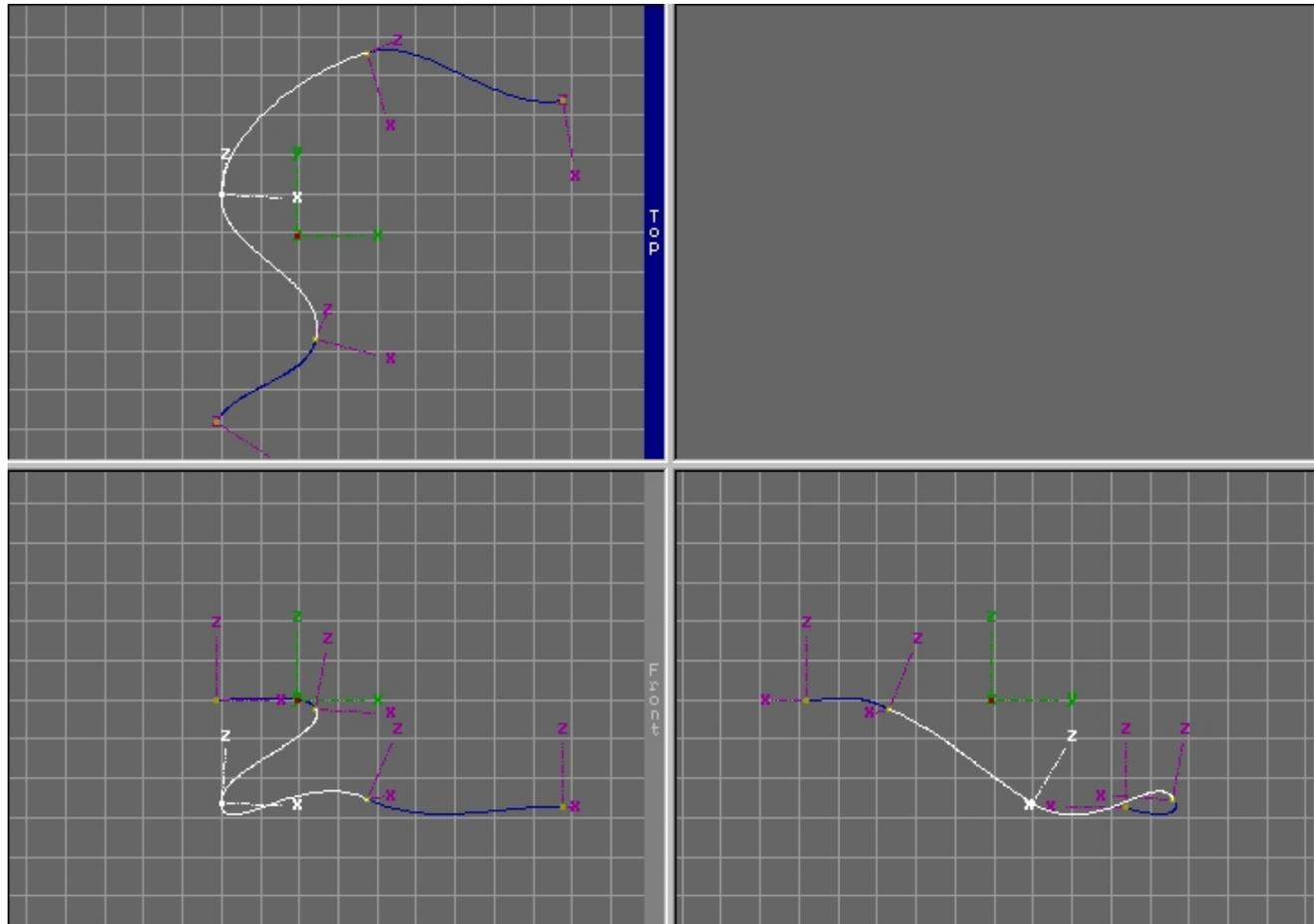


A combinação por **operações booleanas** de objectos criados por varrimento permite obter outros objectos que não seriam possíveis por varrimento.

# Representação por Varrimento – Exemplo

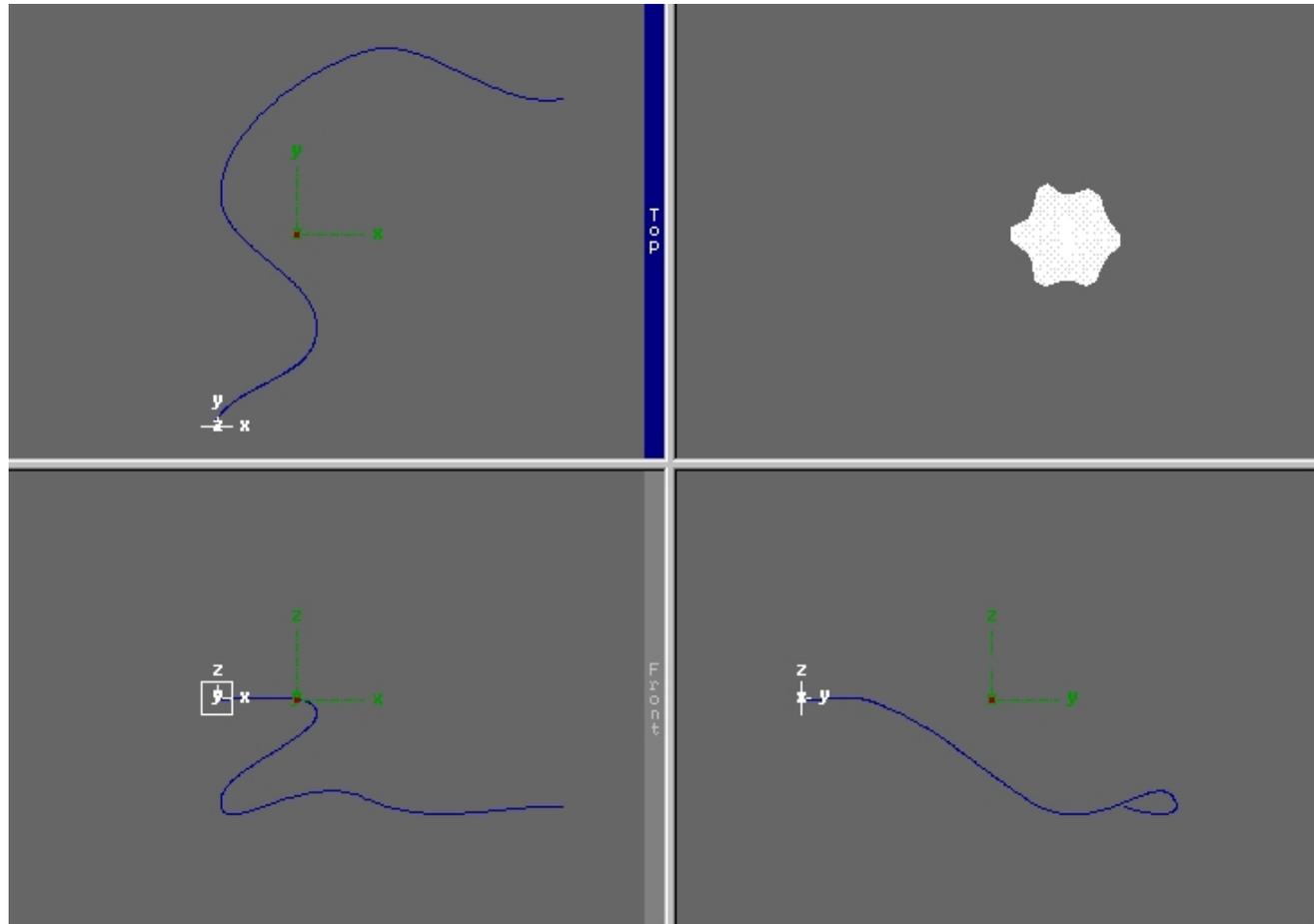


# Representação por Varrimento – Exemplo



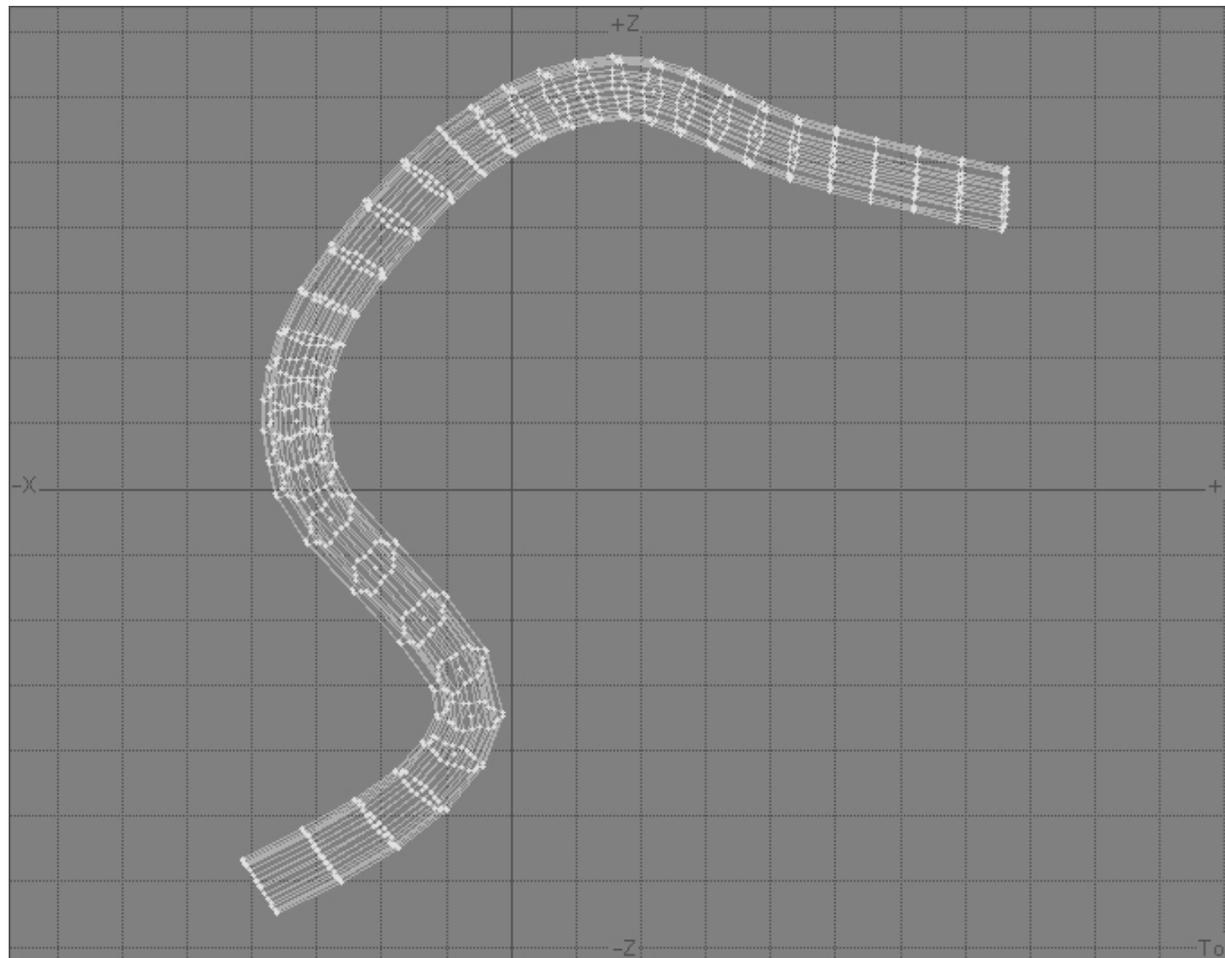
1. Definição de um caminho para efectuar varrimento por translação.

# Representação por Varrimento – Exemplo



2. Definição da forma da secção do objecto final.

# Representação por Varrimento – Exemplo



Objecto obtido por translação.

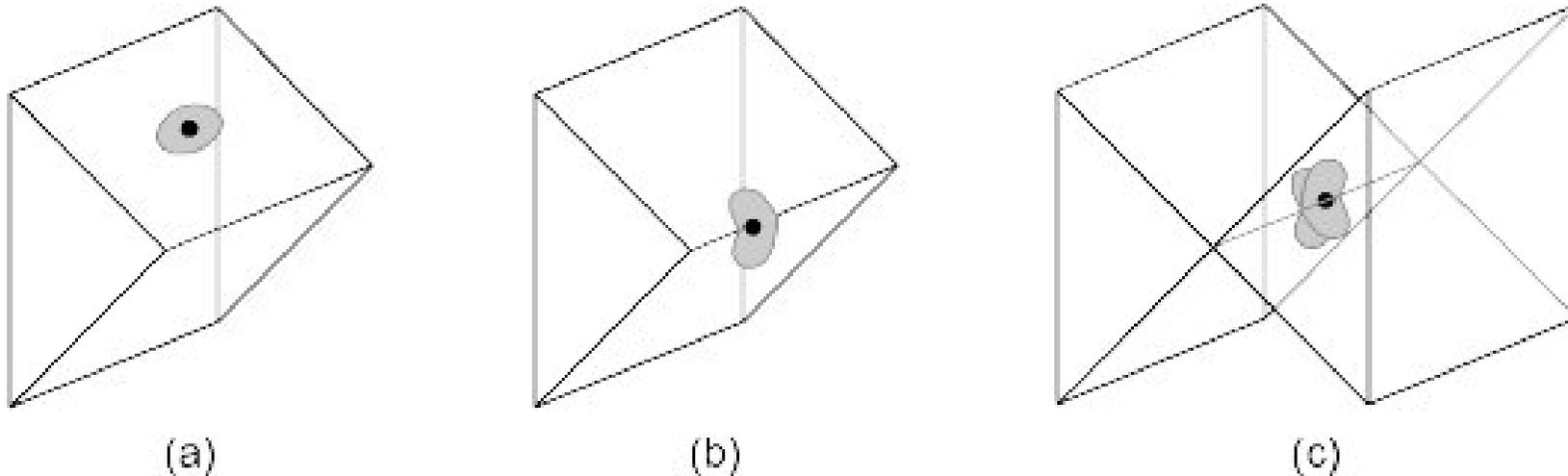
# Representação por Varrimento – Exemplo



Objecto obtido por translação, rotação em torno do eixo de deslocamento e escalamento ao longo do percurso.

# Representação pela Fronteira (b-rep)

- Os sólidos são descritos pela sua superfície de fronteira. Utiliza a descrição por vértices, arestas e faces.
- A representação mais comum é a fronteira por polígonos planos.
- Vão ser considerados apenas os sólidos com fronteira **2-manifolds**, i.e. os pontos vizinhos de um qualquer ponto da fronteira estão num disco (o mesmo é dizer que cada aresta é partilhada por 2 faces)



(a) e (b) são **2-manifold**, (c) não é **2-manifold**

# Representação pela Fronteira (b-rep)

## Poliedro

Sólido delimitado por um conjunto de polígonos cujas arestas pertencem a dois polígonos (para sólidos *2-manifolds*).

## Fórmula de Euler

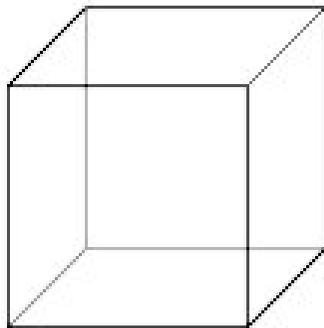
Um poliedro simples, sem buracos, obedece à **fórmula de Euler**:

$$V - E + F = 2$$

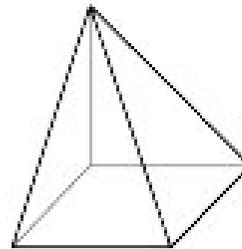
V – Vértices

E – Arestas (edges)

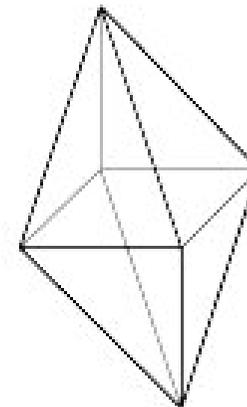
F – Faces



$$\begin{aligned} V &= 8 \\ E &= 12 \\ F &= 6 \end{aligned}$$



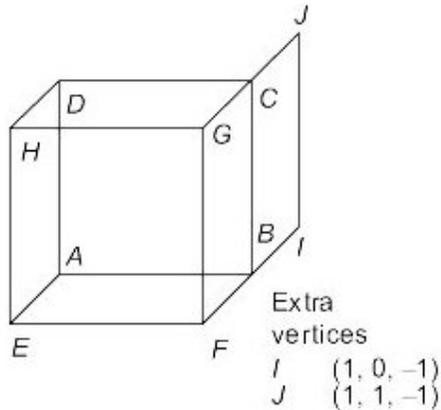
$$\begin{aligned} V &= 4 \\ E &= 6 \\ F &= 4 \end{aligned}$$



$$\begin{aligned} V &= 6 \\ E &= 12 \\ F &= 8 \end{aligned}$$

# Representação pela Fronteira (b-rep)

A **fórmula de Euler** é necessária mas não suficiente para garantir que um objecto seja um poliedro simples.

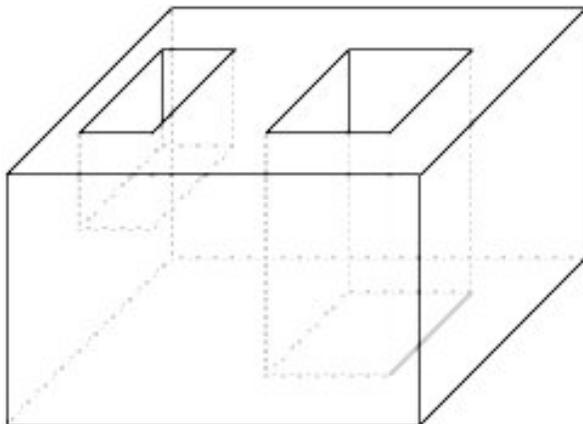


## Condições adicionais:

1. Cada aresta liga 2 vértices e é partilhada por 2 faces
2. Pelo menos 3 arestas encontram-se no mesmo vértice

## Generalização da Fórmula de Euler para poliedros com buracos:

$$V - E + F - H = 2(C - G)$$



$$V - E + F - H = 2(C - G)$$

24 36 15 3 1 1

- V – Vértices
- E – Arestas (edges)
- F – Faces
- H – Número de buracos nas faces
- G – Número de buracos que atravessam o objecto
- C – número de partes do objecto

# Exercício

5. A estrutura de dados junta representa uma malha poligonal.

a)- I: Quais as faces que partilham o vértice  $V_3$ ?

II: Quais as faces vizinhas de  $F_2$ ?

b)- Verifique se, em termos de modelação sólida, aquela malha poligonal pode ou não corresponder à fronteira de um poliedro válido.

*NVert=9*

	<i>x</i>	<i>y</i>	<i>z</i>
1	0	0	0
2	1	0	0
3	1	0	1
4	0	0	1
5	0	1	0
6	1	1	0
7	1	1	1
8	0	1	1
9	1	0	0

*NEdges=12*

	<i>V1</i>	<i>V2</i>	<i>F1</i>	<i>F2</i>
1	1	2	1	5
2	2	3	2	5
3	3	4	3	5
4	4	1	4	5
5	5	6	-	1
6	6	7	-	2
7	7	8	-	3
8	8	5	-	4
9	1	5	4	1
10	2	6	1	2
11	3	7	2	3
12	4	8	3	4

*NFaces=5*

	<i>E1</i>	<i>E2</i>	<i>E3</i>	<i>E4</i>
1	5	10	1	9
2	6	10	2	11
3	7	11	3	12
4	12	4	9	8
5	2	1	4	3

# Representação por Decomposição Espacial

- Um sólido é decomposto em:
  - Num conjunto de sólidos mais primitivos que o original
  - Os sólidos primitivos são adjacentes e não se intersectam
- **Tipos** de Representação por Decomposição Espacial
  - Decomposição Celular
  - Enumeração da Ocupação Espacial
  - Octrees
  - Árvores binárias de partição do espaço

# Representação por Decomposição Espacial

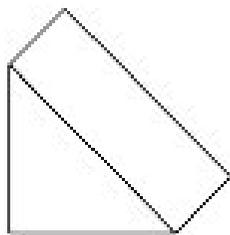
## Decomposição Celular

### Na Decomposição Celular:

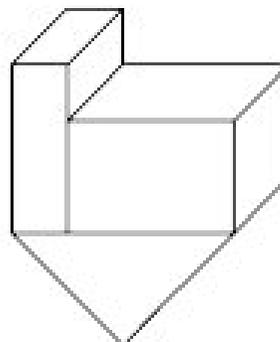
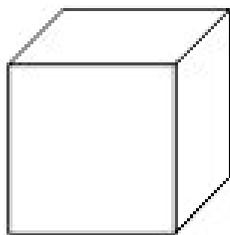
- Existe um conjunto de células primitivas, parametrizáveis
- Podem ser curvas
- Difere da Instância de Primitivas, por admitir a composição de objectos mais complexos, a partir de outros já criados

### Operação de colagem

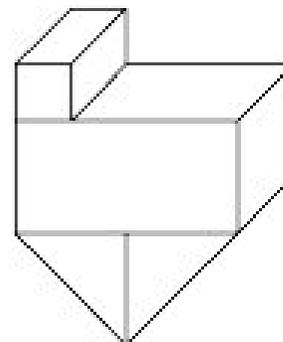
Trata-se de uma união de células que não se intersectam



(a)



(b)



(c)

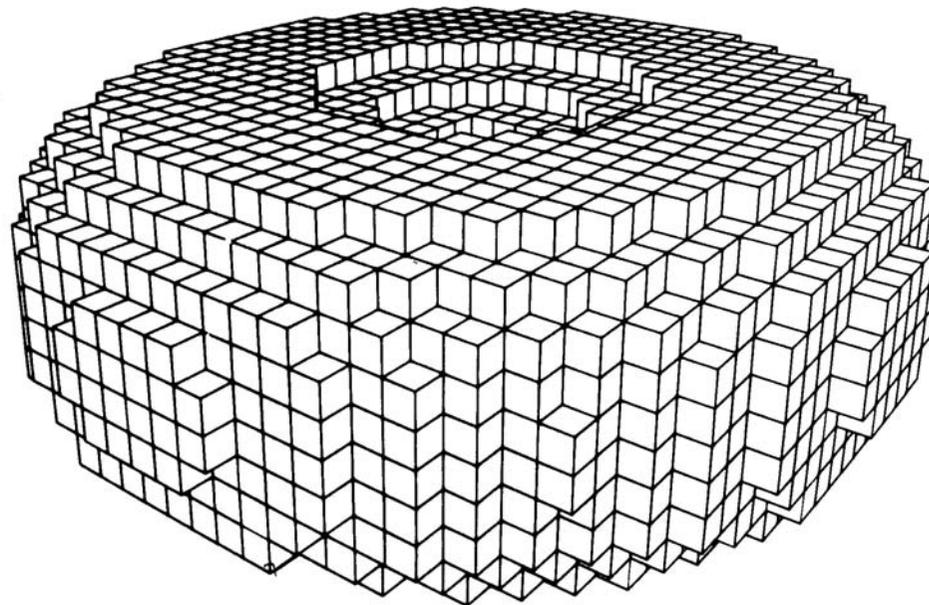
- a) Células primitivas a transformar
- b) e c) são o mesmo objecto final criado com diferentes combinações

# Representação por Decomposição Espacial

## Enumeração da Ocupação Espacial

A **Enumeração da Ocupação Espacial** é um caso particular da **Decomposição Celular**:

- Sólido formado por células idênticas de igual dimensão colocadas numa grelha regular.
- As células são designadas por **Voxels** (volume elements) por analogia com pixels
- Controla-se apenas a presença ou ausência da célula em cada posição da grelha
- A forma mais usual para a célula é o **cu**bo
- O objecto é codificado por uma lista única de células ocupadas

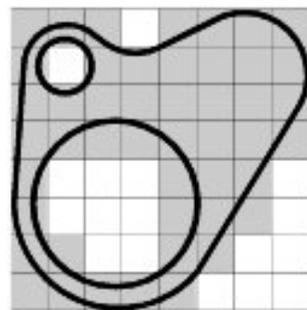


# Representação por Decomposição Espacial Octrees

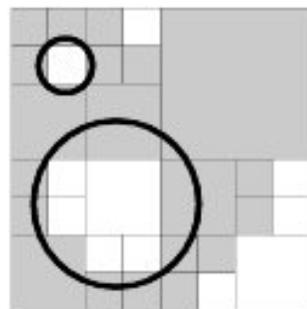
As **Octrees** são um caso especial da **Enumeração da Ocupação Espacial** :

- As células são de diferentes dimensões. Células vizinhas com a mesma informação são aglomeradas numa única.

Em 2D designam-se de **Quadrees**: a imagem é sucessivamente subdividida em 4 quadrantes. A subdivisão por quadrante pára quando este é homogéneo.

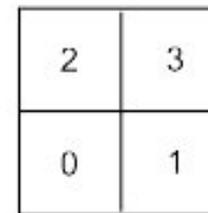


(a)

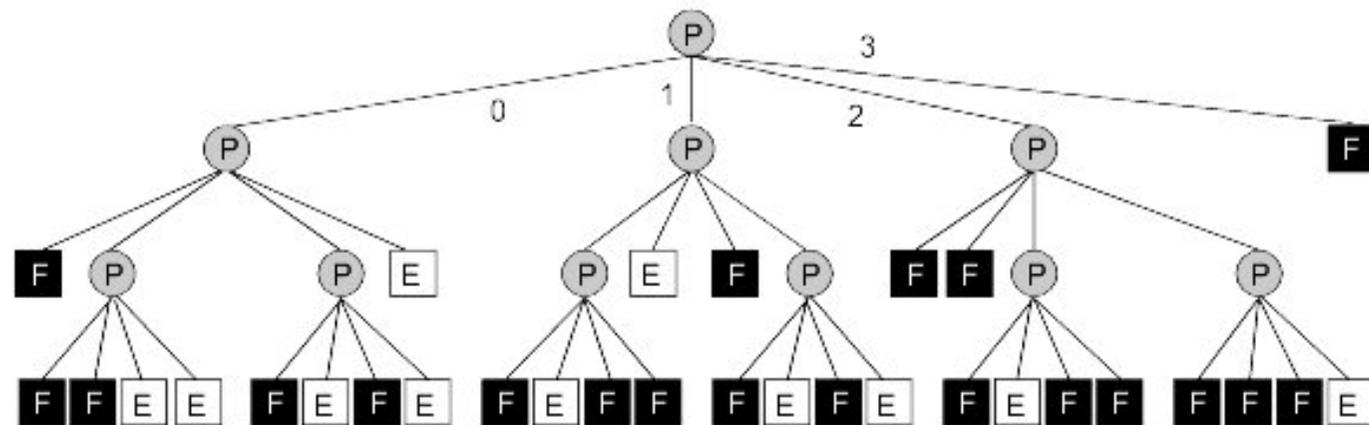


(b)

IHAR  
PORT



Quadrant numbering



F – Completo, P – Parcial, E - Vazio

# Representação por Decomposição Espacial

## Octrees

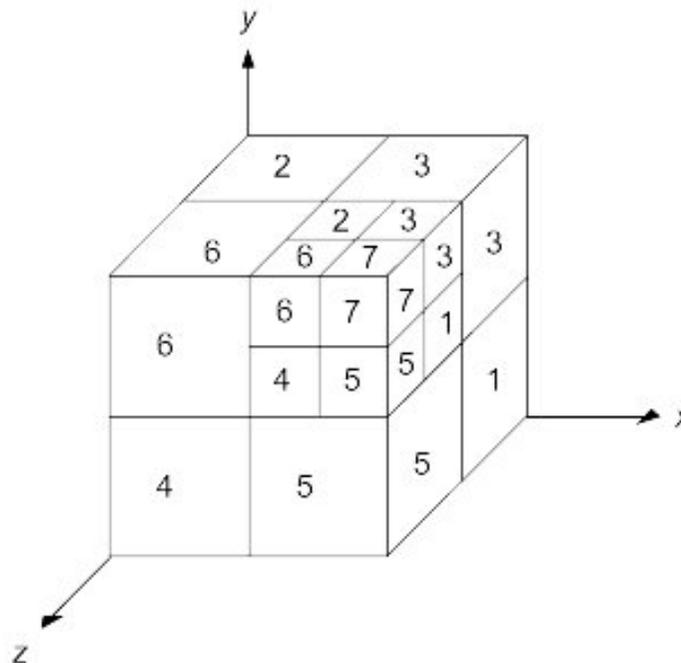
### A Octree é semelhante à quadtree

- A **octree** é 3D e a divisão do espaço é em **octantes**

### Número de nós de uma octree

- É proporcional à superfície do objecto porque a necessidade de divisão do espaço só ocorre na superfície.

Enumeração da **octree**



# Representação por Decomposição Espacial

## Octrees

### Operações Booleanas

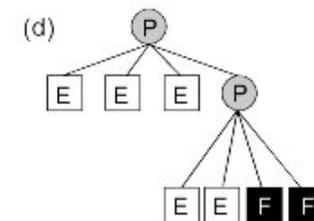
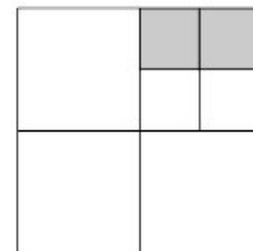
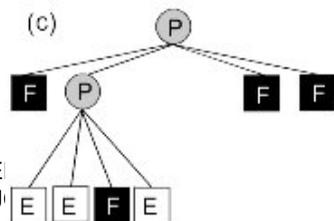
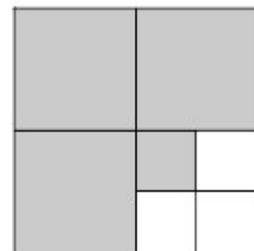
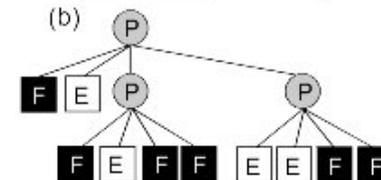
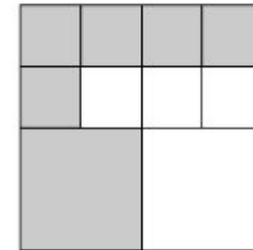
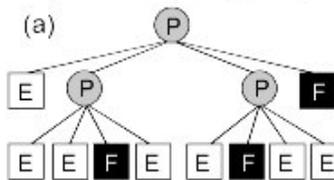
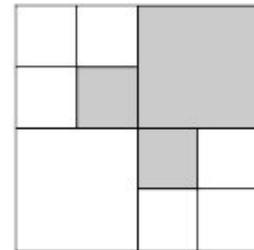
- As árvores dos objectos são atravessadas, top-down.
- O conteúdo dos nós são comparados e junta-se à árvore final um nó de acordo com a operação executada.

(a) Quadtree S

(b) Quadtree T

(c) Quadtree  $S \cup T$

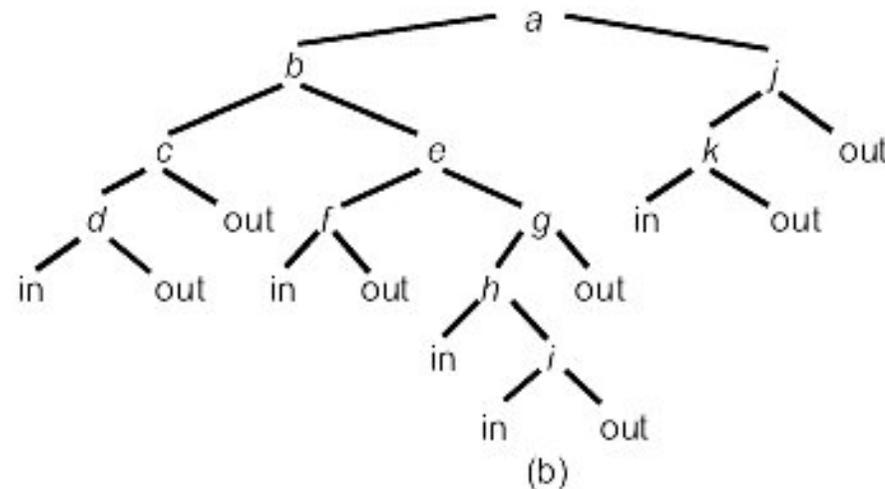
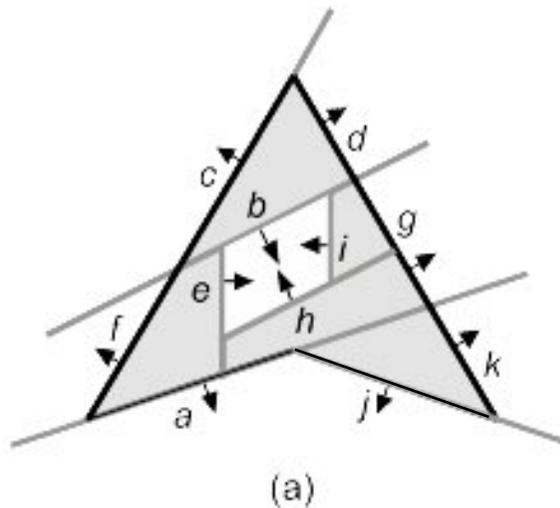
(d) Quadtree  $S \cap T$



# Representação por Decomposição Espacial

## Árvores binárias de Partição do Espaço (BSP)

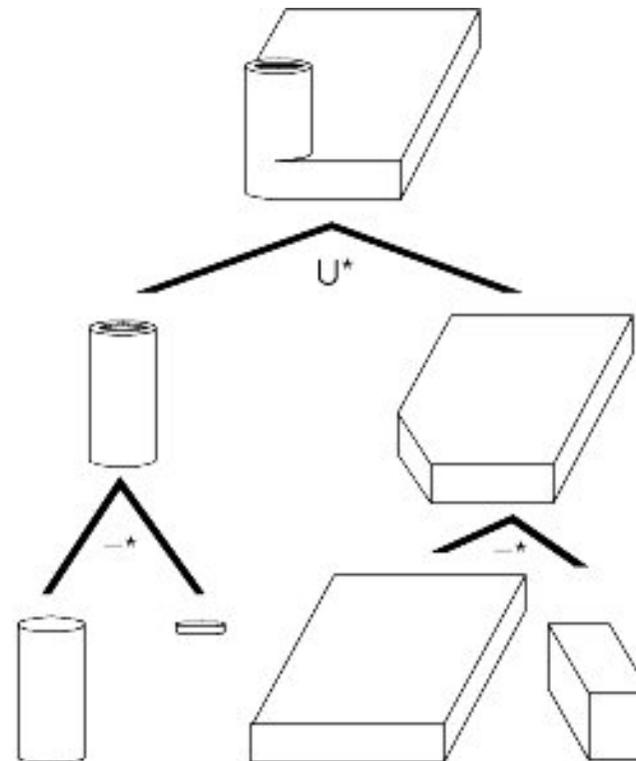
- Em cada passo, o espaço é dividido por um plano de posição e orientação arbitrários
- A cada nó interno da árvore está associado um plano e 2 apontadores (um para o lado de dentro do polígono e outro para o lado de fora).
- Se um sub-espço é homogêneo (totalmente interior ou exterior), deixa de ser dividido.



- (a) Polígono côncavo, com a fronteira definida por linhas pretas. Linhas de divisão a cinzento escuro. Interior a cinzento claro.
- (b) Árvore BSP.

# Representação Construtiva (CSG – Constructive Solid Geometry)

- O objecto é obtido pela combinação de primitivas simples através de operadores booleanos.
- O objecto é guardado como uma árvore, em que os nós interiores são operadores e as folhas são primitivas simples
- Alguns nós representam operações booleanas, enquanto outros efectuem translações, rotações e escalamentos.



# Exercício

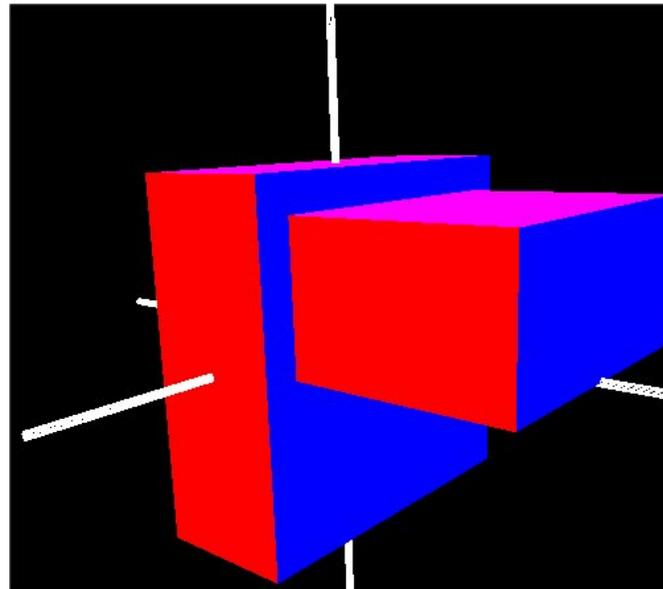
4. Sejam, num sistema de modelação sólida baseada em CSG, dois sólidos  $B_1$  e  $B_2$ , correspondentes a instanciações de um cubo de aresta unitária e centrado na origem, acompanhadas da aplicação, respectivamente, das transformações geométricas:

$$M_1 = S(4, 10, 10)$$

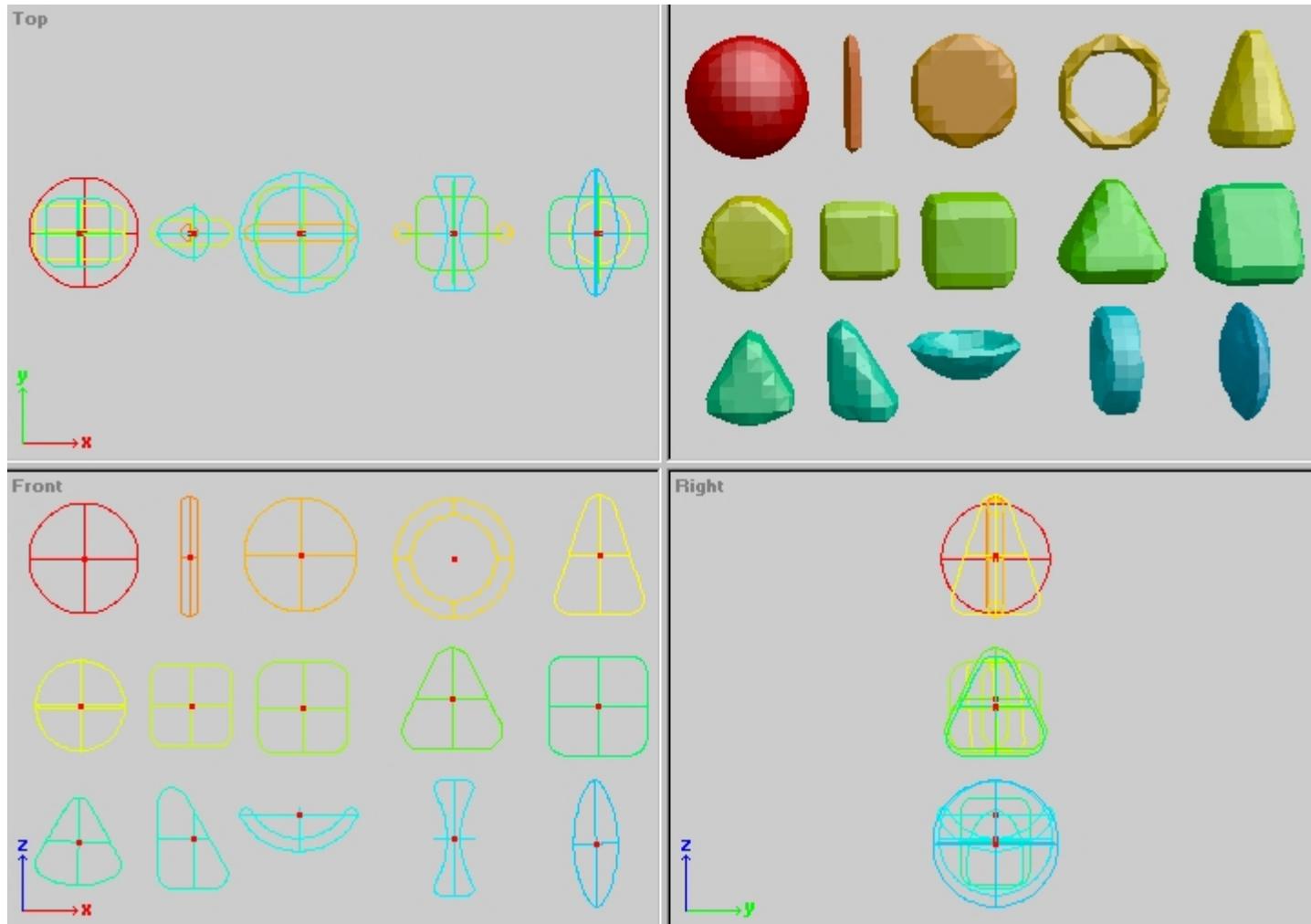
$$M_2 = T(4, 2, 0) \cdot S(8, 4, 8)$$

Nota:  $S$  e  $T$  são respectivamente, escalamento e translação.

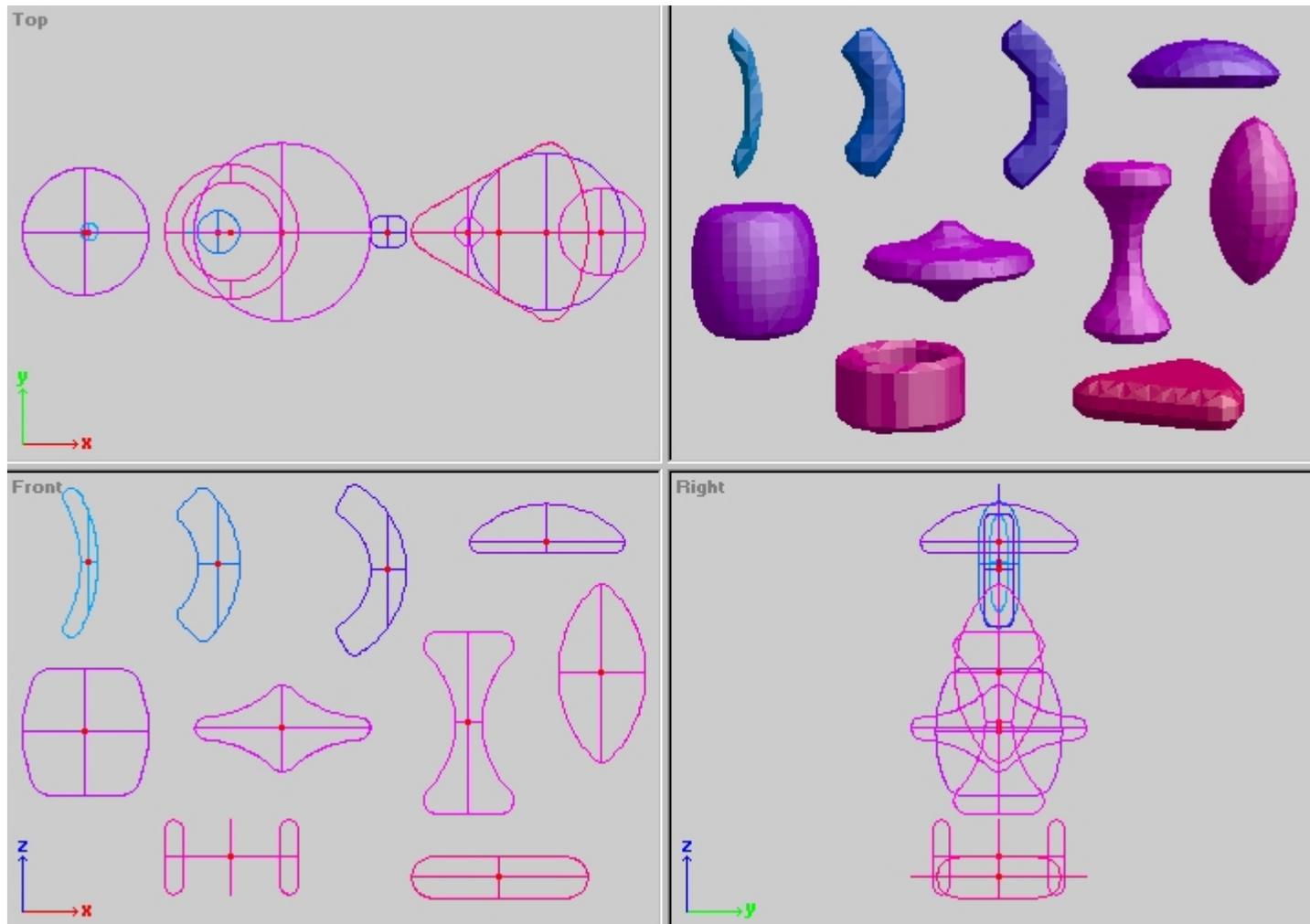
- a)- Esboce o sólido resultante da árvore  $A = B_1 \cup B_2$  no referencial  $xyz$ .  
b)- Verifique a validade do sólido obtido, à luz da fórmula de *Euler* Generalizada.



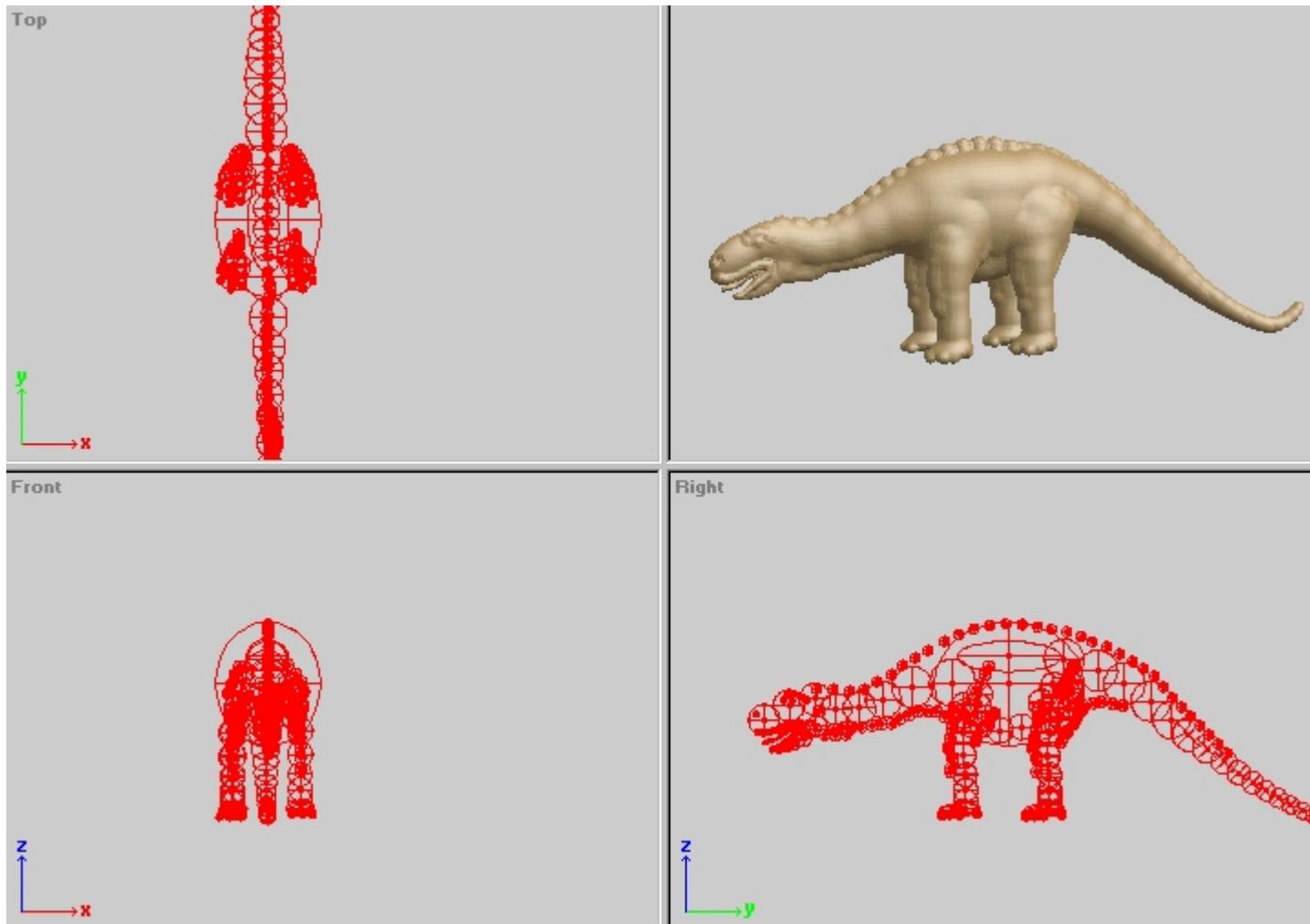
# Modelação interactiva com *metaballs/metashapes*



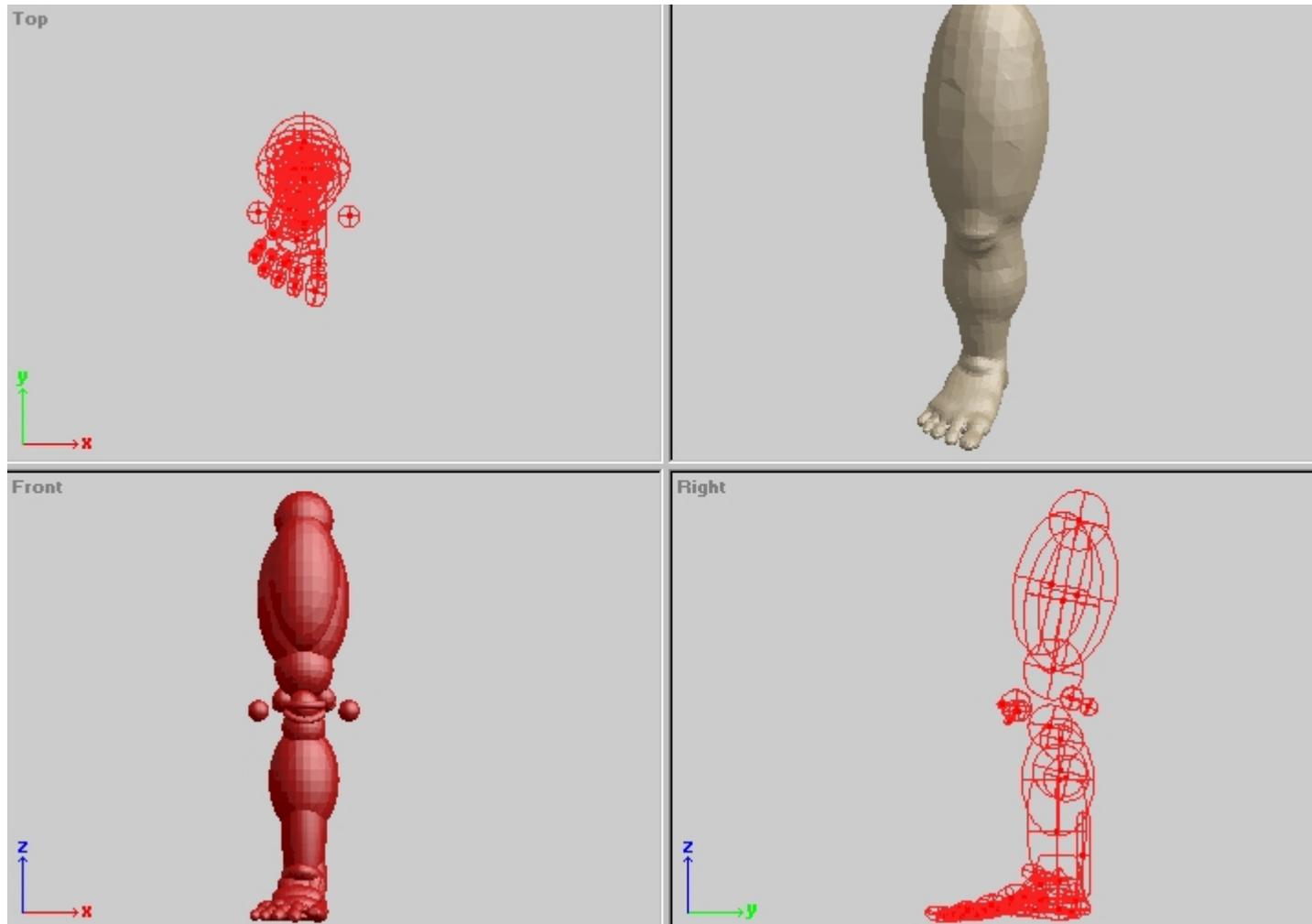
# Exemplo de modelação com *metaballs/metashapes*



## Exemplo de modelação com *metaballs/metashapes*



## Exemplo de modelação com *metaballs/metashapes*



# Referências

- 3D Modeling & Surfacing  
Bill Fleming  
Morgan Kaufmann, Academic Press, 1999
- Introduction to Computer Graphics  
James Foley, A. van Dam, S. Feiner, J. Hughes,  
R. Phillips  
Addison-Wesley Publishing Company, 1996