

Representação de Objetos Tridimensionais

Malhas Poligonais

Maria Cristina F. de Oliveira
Rosane Minghim
2007

Modelo

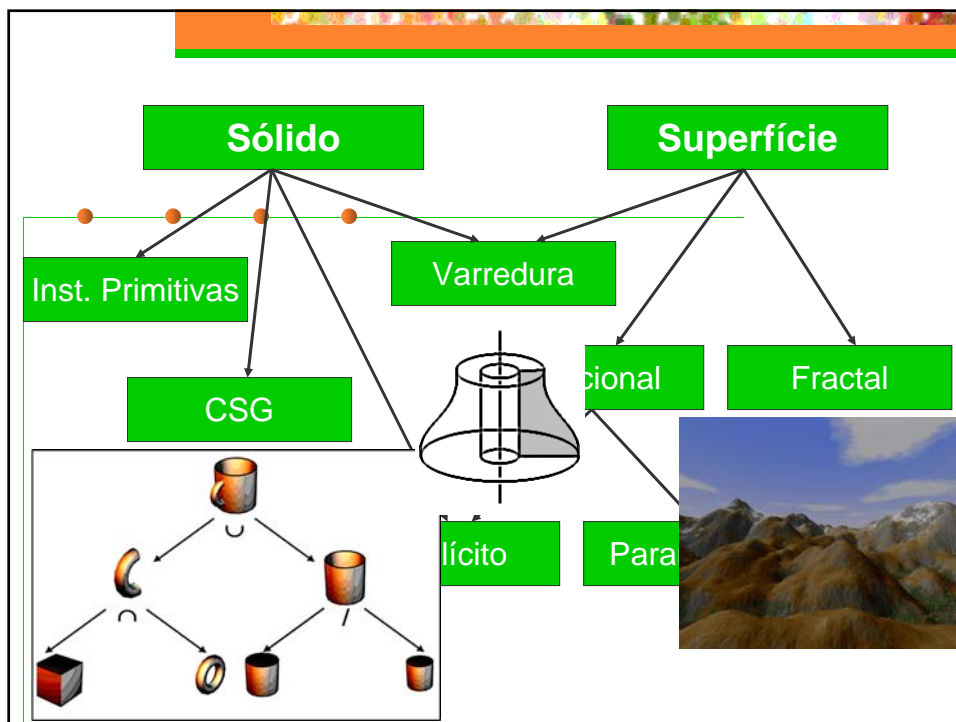
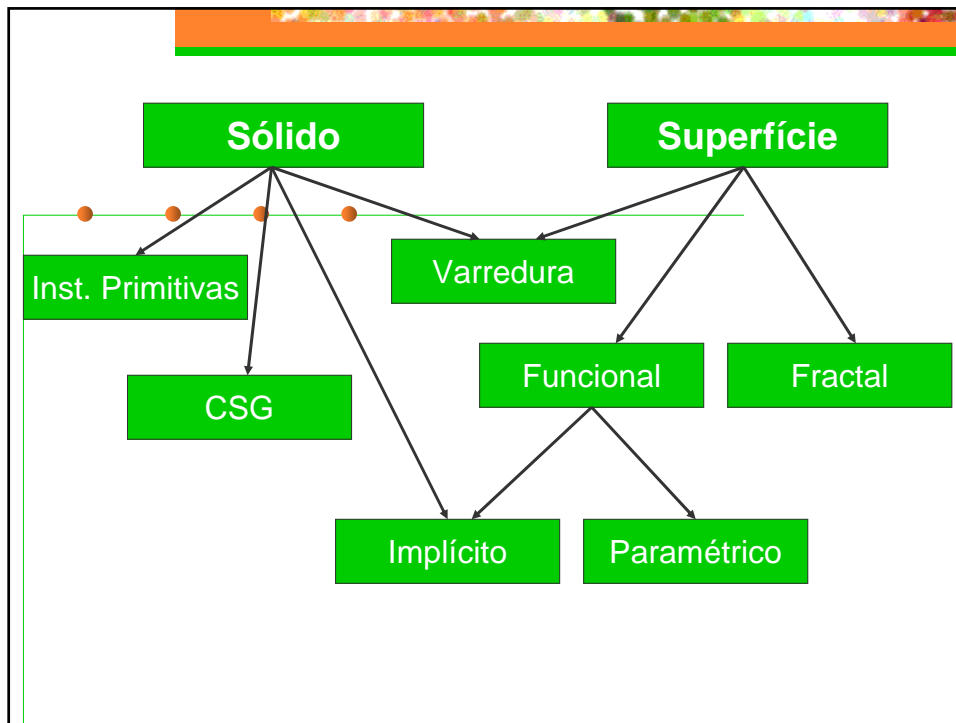
- Representação construída artificialmente para tornar mais fácil a observação/análise de um objeto/fenômeno
 - Nível de detalhe definido pelas aplicações que o utilizam
 - Problemas práticos em CG: modelos geométricos

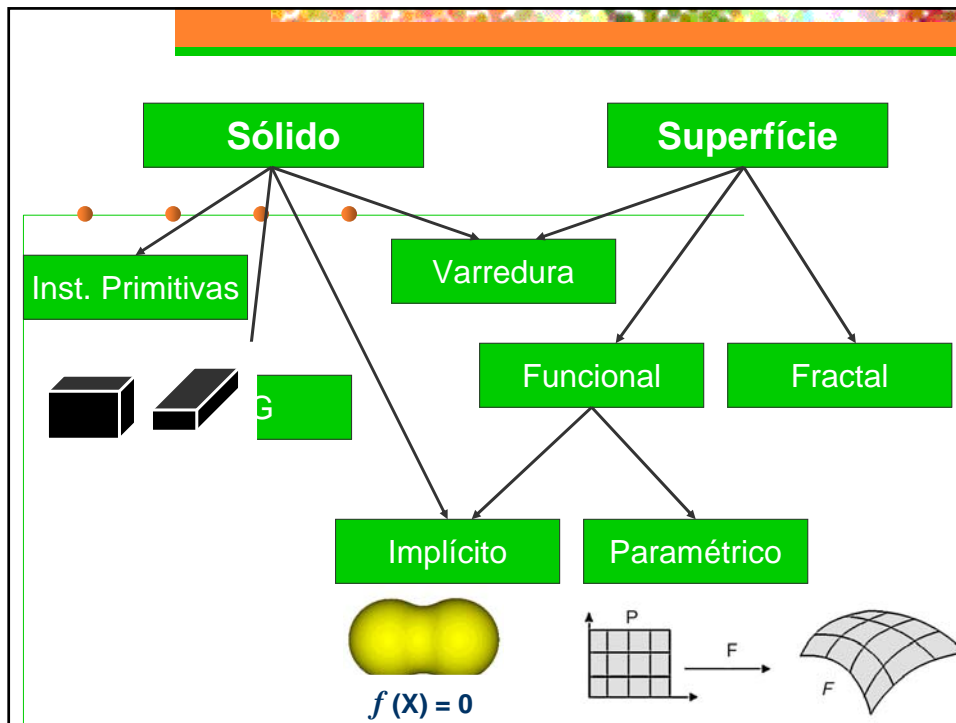
Modelagem Geométrica

- Início dos anos 70
- Coleção de métodos usados para descrever a forma e outras características geométricas de um objeto, bem como para simular processos dinâmicos
- Sistema de modelagem geométrica: sistema computacional que permite a criação, modificação e acesso à representação de objetos por meio de modelos geométricos

Modelos Geométricos

- Cenas gráficas podem conter muitos tipos diferentes de objetos e materiais
 - Não existe uma maneira única capaz de descrever e representar todos os tipos de objetos
- Descrição vs. Representação
 - Descrição do objeto pelo usuário: processo de modelagem
 - Representação do objeto no sistema gráfico: como manter as informações necessárias para renderizar o objeto (e aplicar simulações, por ex.)





Modelagem de Sólidos

- Ramo da M.G. que estuda técnicas para criar, manter e comunicar informação sobre a forma de objetos sólidos
 - envolve a criação e a manutenção de um modelo para futuro acesso e análise
 - permite formular e responder questões sobre propriedades volumétricas (volume, peso, momento de inércia, ...) e topológicas (conectividade, pertinência, ...)

Volume vs. Superfícies

- Objetos sólidos tridimensionais

- Representados apenas pela sua fronteira
 - representações por fronteira: objeto 3D descrito como um conjunto de superfícies que separa o seu interior do meio externo (geralmente quadriláteros ou triângulos, ou alternativamente superfícies paramétricas)
- Superfície e conteúdo interno representados explicitamente
 - Representações por particionamento espacial descrevem propriedade interiores, particionando a região do espaço que contém o objeto em um conjunto de pequenos sólidos adjacentes não sobrepostos (geralmente cubos ou tetraedros)

Malhas Poligonais

- Atualmente: enorme diversidade de técnicas e modelos em CG
- Vamos estudar inicialmente uma forma de representação por fronteira muito simples, adotada em muitos sistemas gráficos
 - objetos descritos por malhas poligonais que representam a sua superfície (fronteira)
 - Conjunto de vértices, arestas e faces planares (triângulos)
 - Representação adequada para 'rendering' por placas gráficas: objetos gráficos padrão

Malhas Poligonais

■ Poliedros

- Representação poligonal é exata

■ Objetos em geral

- descritos por superfícies curvas
- decompostos (*tesselated*) para produzir uma representação poligonal aproximada

Malhas Poligonais Exemplos

Fonte:

<http://www.bymath.com/studyguide/geo/geo21.htm>

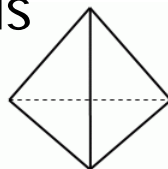


Fig. 99

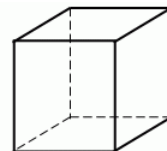


Fig. 100

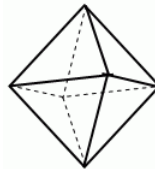


Fig. 101

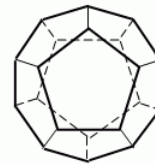


Fig. 102

Poliedros convexos regulares

tetraedro (4 faces, Fig.99)

hexaedro (cubo, 6 faces, Fig.100)

octaedro (8 faces, Fig.101)

dodecaedro (12 faces, Fig.102)

icosaedro (20 faces, Fig.103)

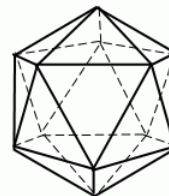


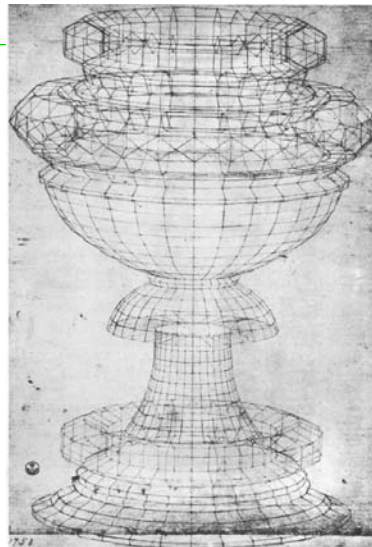
Fig. 103

Malhas Poligonais – Exemplos



Malhas Poligonais – Exemplos

<http://www.siggraph.org/education/materials/HyperGraph/modeling/poly mesh/polymesh.htm>



Malhas Poligonais – Exemplos



Estruturas de Dados

■ Problema

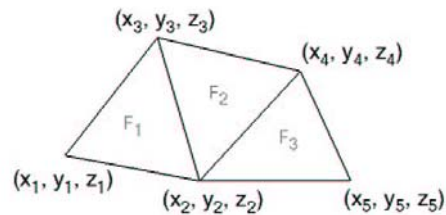
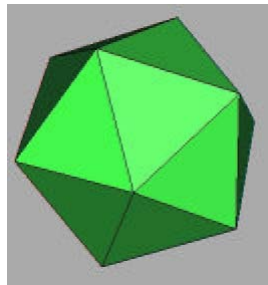
- como armazenar a descrição de um objeto em termos das faces que descrevem sua superfície?
- Diversas soluções possíveis...

■ Sugestões??

Estruturas de Dados

■ Solução simples

- Tabela de faces, cada face informa as coordenadas de seus vértices



FACE TABLE			
F1	(x1, y1, z1)	(x2, y2, z2)	(x3, y3, z3)
F2	(x2, y2, z2)	(x4, y4, z4)	(x3, y3, z3)
F3	(x2, y2, z2)	(x5, y5, z5)	(x4, y4, z4)

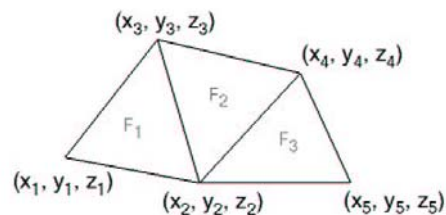
Estruturas de Dados

■ Problemas:

- vértices redundantes, ausência de informação topológica

■ Alternativa

- tabelas de faces e de vértices
- cada face lista referências aos seus vértices
- resolve os problemas?



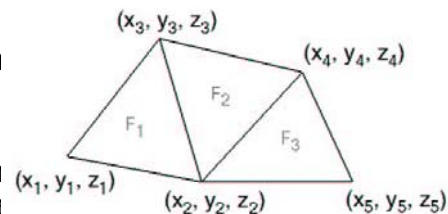
VERTEX TABLE			
v1	x1	y1	z1
v2	x2	y2	z2
v3	x3	y3	z3
v4	x4	y4	z4
v5	x5	y5	z5

FACE TABLE			
F1	v1	v2	v3
F2	v2	v4	v3
F3	v2	v5	v4

Estruturas de Dados

■ Outra alternativa

- tabelas de faces, de aresta e de vértices
- cada face lista referências às arestas que a compõem
- cada aresta lista referência aos vértices
- Todos os 'elementos topológicos' (faces, arestas e vértices) armazenados explicitamente...
- Informação topológica (adjacência) implícita

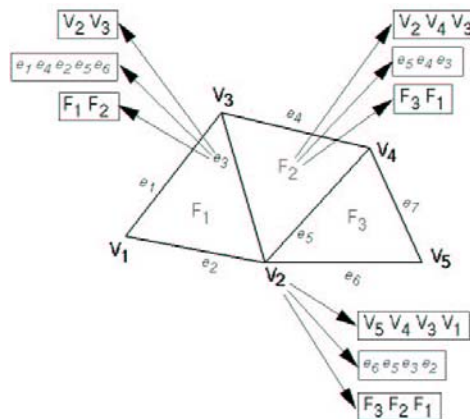


VERTEX TABLE			
v_1	x_1	y_1	z_1
v_2	x_2	y_2	z_2
v_3	x_3	y_3	z_3
v_4	x_4	y_4	z_4
v_5	x_5	y_5	z_5

Estruturas de Dados

■ Listas de adjacências

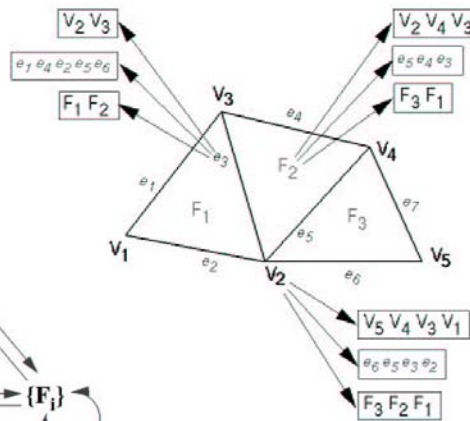
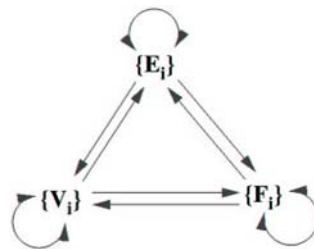
- Armazena explicitamente todas as adjacências entre vértices, faces e arestas
- Toda a informação topológica explícita
- Custo extra de armazenagem



Estruturas de Dados

■ Listas parciais de adjacências

- Possível armazenar algumas relações de adjacência, e inferir outras?



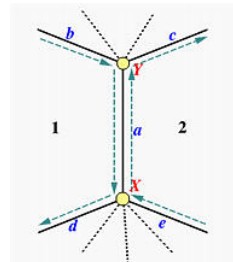
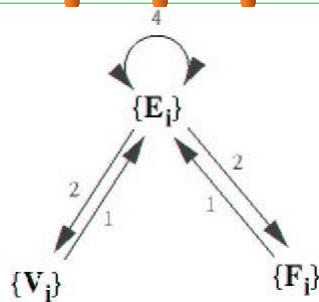
Estruturas de Dados

■ Winged-edge

- Associa informações de adjacência às arestas
 - Todas as adjacências entre elementos topológicos recuperadas em tempo $O(1)$
 - Custo extra de armazenagem pequeno (registros de tamanho fixo)
 - Consegue representar polígonos arbitrários

Estruturas de Dados

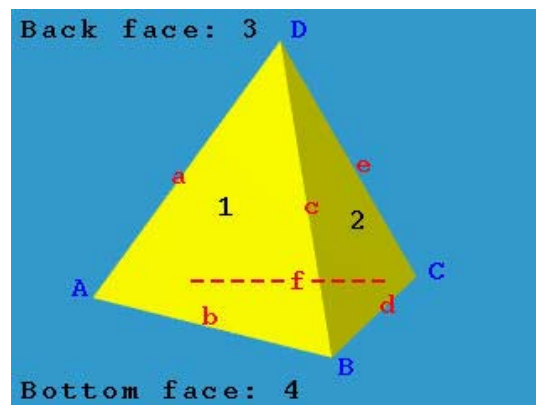
Winged-edge



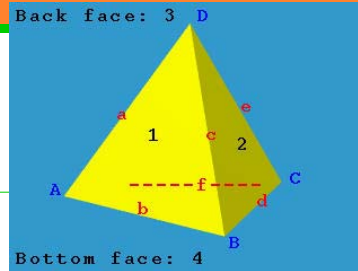
Aresta	Vert.		Fases		Esquerda		Direita	
Id	Inic	Fim	esq	dir	Pred	Suc	Pred	Suc
a	X	Y	1	2	b	d	e	c

adap. De <http://www.cs.mtu.edu/~shene/COURSES/cs3621/NOTES/model/winged-e.html>

Exemplo: winged-edge do tetraedro



Estruturas de Dados



Winged-edge (exemplo)

<i>Aresta</i>	<i>Vert.</i>		<i>Faces</i>		<i>Esquerda</i>		<i>Direita</i>	
<i>ID</i>	<i>Inic.</i>	<i>Fim</i>	<i>Esq.</i>	<i>Dir.</i>	<i>Pred</i>	<i>Suc</i>	<i>Pred</i>	<i>Suc</i>
a	A	D	3	1	e	f	b	c
b	A	B	1	4	c	a	f	d
c	B	D	1	2	a	b	d	e
d	B	C	2	4	e	c	b	f
e	C	D	2	3	c	d	f	a
f	A	C	4	3	d	b	a	e

Estruturas de Dados

Observação

- Em muitas E.D. inconsistências podem ocorrer se o processo de geração da malha não for cuidadoso para evitá-las
- a malha pode descrever objetos não 'factiveis'
 - Ex. vértices e arestas isolados, polígonos não fechados, polígonos isolados, etc.

- Certas estruturas, como a *winged-edge*, não admitem tais ocorrências: modelo descreve objeto consistente

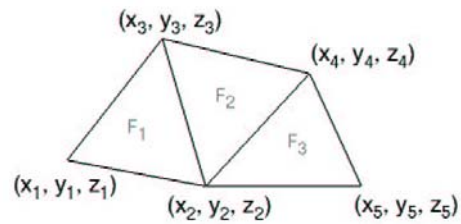
Malhas de Triângulos

■ Propriedades

- Cada face tem exatamente 3 vértices
- Cada vértice compartilha um número arbitrário de faces

■ Estrutura de Adjacência simples

- Faces armazenam referências a vértices e faces vizinhas
- Maioria das relações de adjacência recuperada em tempo constante



Malhas de Triângulos

■ Um problema bastante atual é o da simplificação de malhas (*decimação*)

- Redução do número de polígonos/triângulos necessários para descrever um modelo
- Veja, por exemplo
<http://amber.rc.arizona.edu/lw/decimate.html>
- Porquê?
 - *Rendering* mais rápido
 - Menor custo de armazenagem
 - Manipulação mais simples

Informações Geométricas

- As coordenadas dos vértices contém a informação geométrica necessária para o *rendering* do modelo
 - A partir das coord's dos vértices, pode-se computar a inclinação das arestas, o retângulo envoltório (*bounding box*) de cada face, a equação do plano que contém cada face, etc.
 - Informações necessárias para algoritmos de recorte, remoção de superfícies ocultas e de *rendering*...

Informações Geométricas

- Equação do plano:

$$Ax + By + Cz + D = 0$$

em que (x,y,z) é um ponto qqr no plano, A, B, C e D são ctes. que descrevem as propriedades espaciais do plano.

Como obter a equação do plano, dados 3 pontos que estão no plano?

v. Hearn & Baker, Seção 3.15

Informações Geométricas

- Equação do plano
- Coefs. A, B, C e D podem ser armazenados na tabela de faces
- Orientação do plano no espaço dada pela sua normal: $\mathbf{N} = (A, B, C)$
- Importante distinguir os 2 lados: 'dentro' e 'fora': especificar os vértices sempre na direção anti-horária de alguém observando o plano do lado 'de fora'

Exercício

- Desenvolver um programa para preencher uma winged-edge a partir da descrição simples do modelo, lida de um arquivo, contendo vértices, arestas, e faces
- => Qual é a menor quantidade de informação necessária para fazer isso?

Bibliografia

- Seção 3.15 – Hearn & Baker Computer Graphics in OpenGL
- Lecture Notes by Thomas Funkhouser at Princeton University:
<http://www.cs.princeton.edu/courses/archive/fall02/cs526/lectures/meshes.pdf>