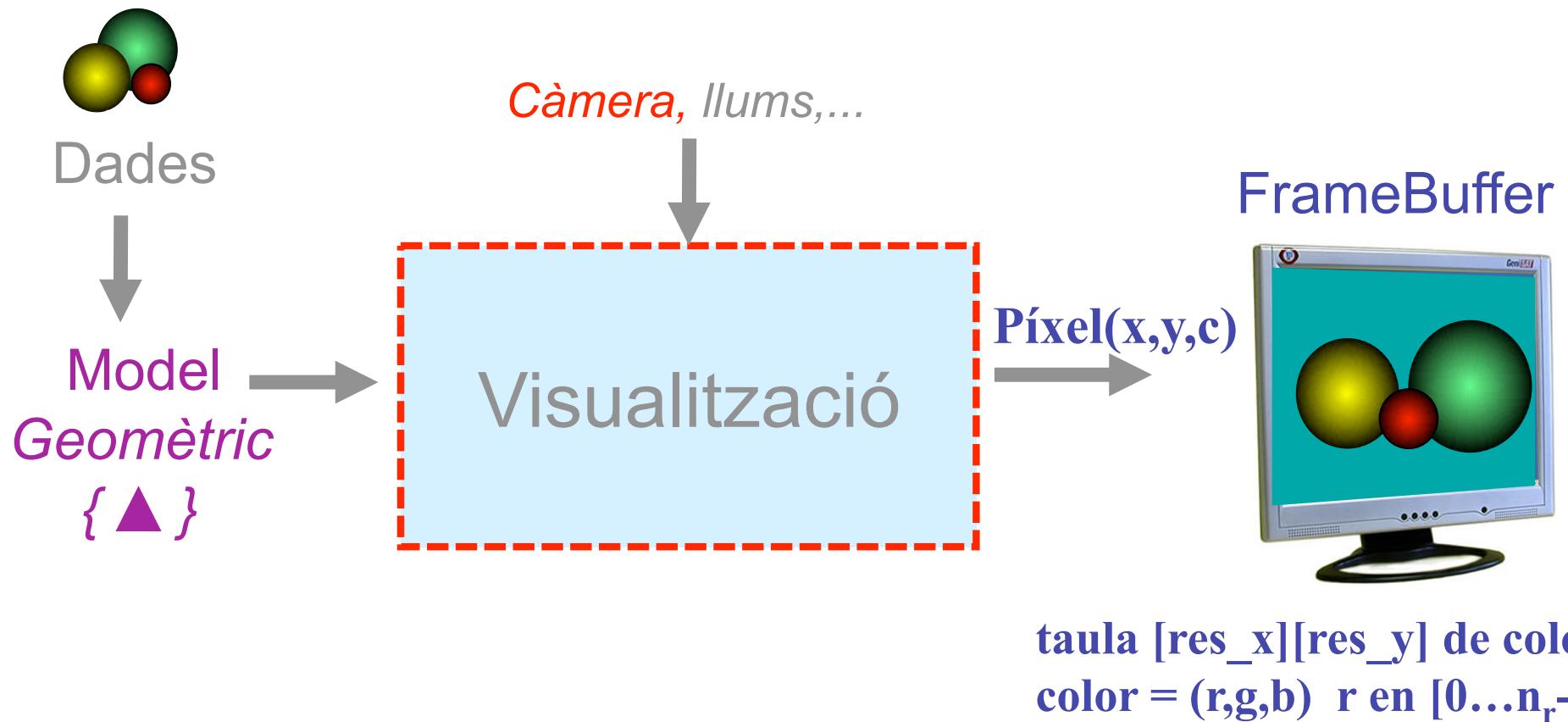
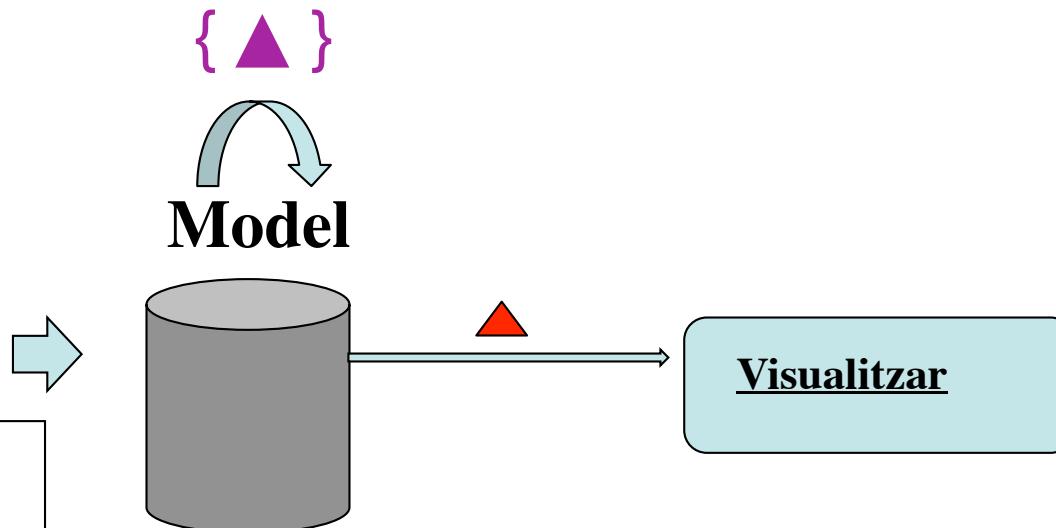


Model Geomètric (1)



En el marc d'IDI...

Dades



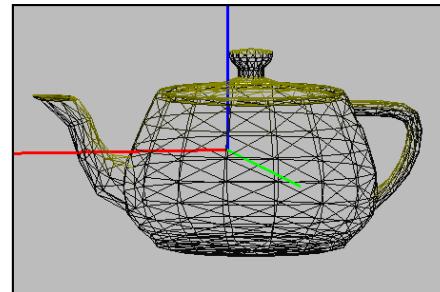
Sòlids

- Limitats per superfície
- Moviments no modifiquen forma

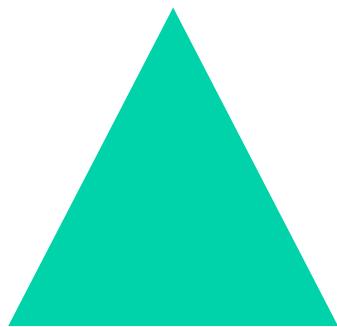


Models Geomètrics:

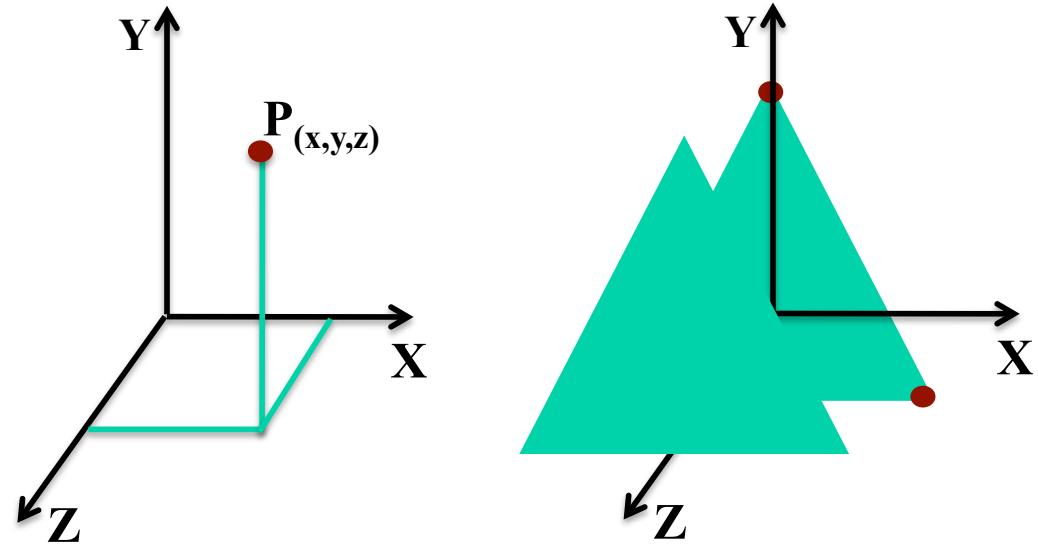
- Model de Fronteres
aproximació per cares planes => triangles
- Procedural, CSG, octrees,...



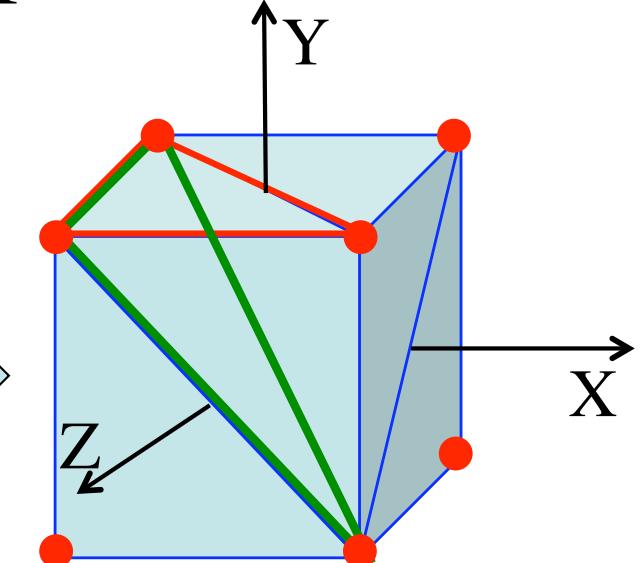
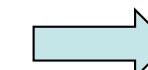
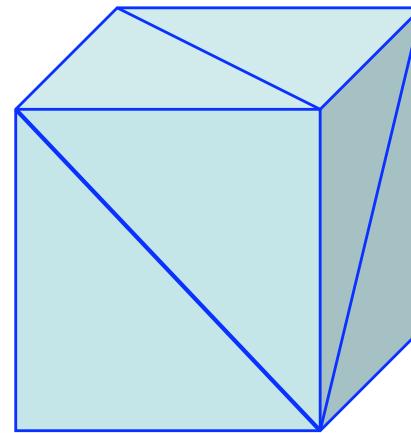
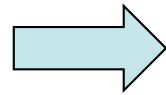
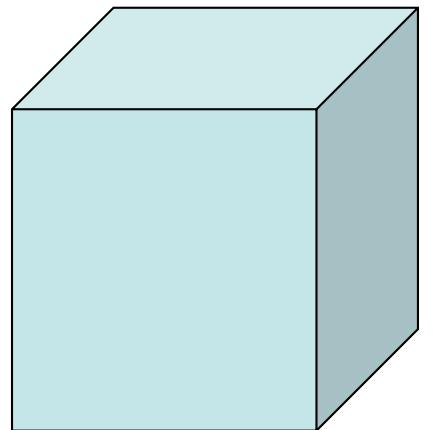
Models Geomètrics (intro)



• *Com guardar?*



Model Fronteres: Exemple Cub

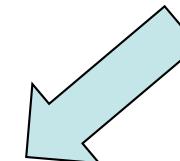


Per cada triangle

- Geometria
- Topologia
(implícitament)

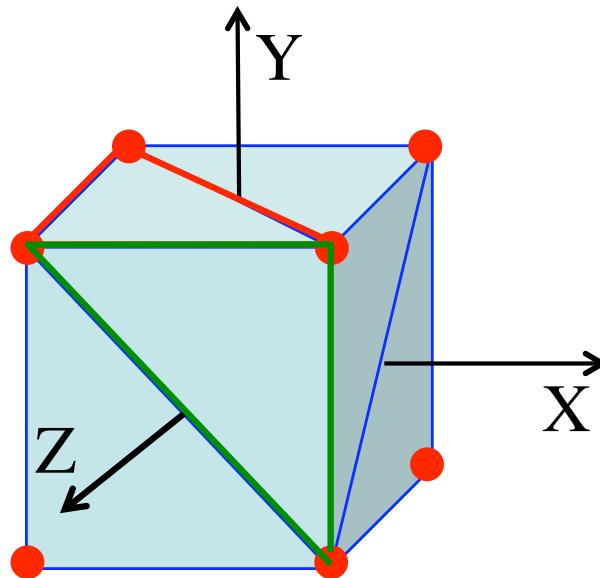
Vèrtexs repetits ☺

x	y	z
-1	+1	-1
-1	+1	+1
+1	+1	+1
...



x	y	z
-1	+1	-1
-1	+1	+1
+1	-1	+1
...

Exemple: Model Fronteres



Cares

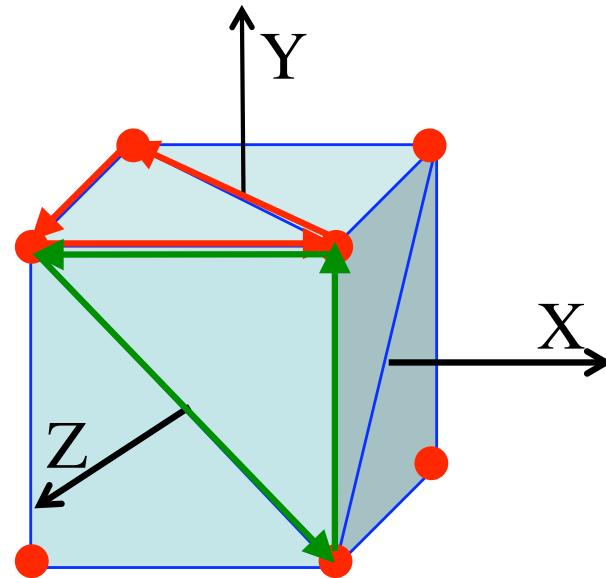
normal	<i>Id Vèrtexs</i>
a_1, b_1, c_1	1,2,3
a_2, b_2, c_2	2,4,3
...	...

Vèrtexs

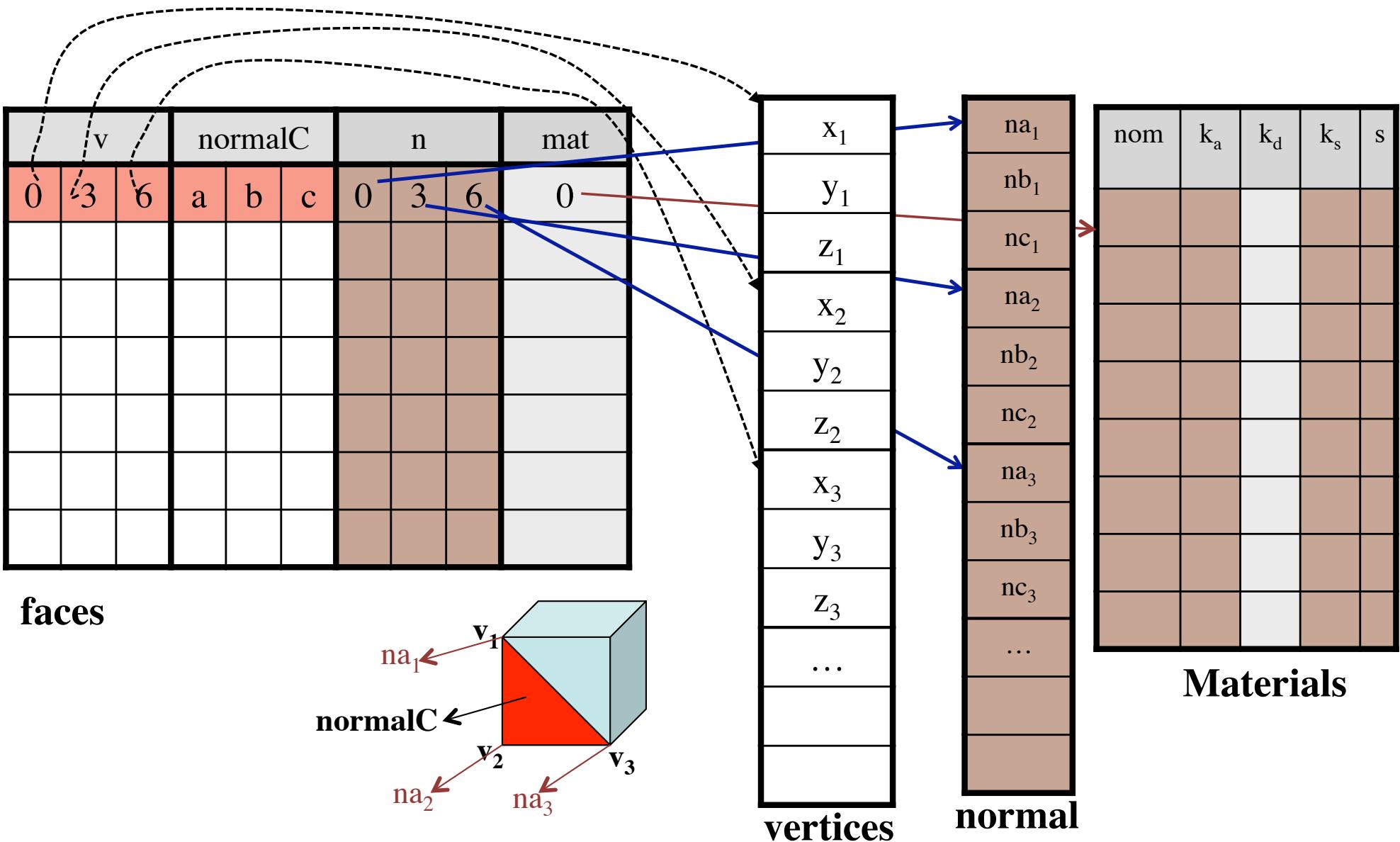
x	y	z
-1	1	-1
-1	1	1
1	1	1
1	-1	1
1	-1	-1
1	1	-1
-1	-1	-1
-1	-1	1

Model Fronteres Vàlid

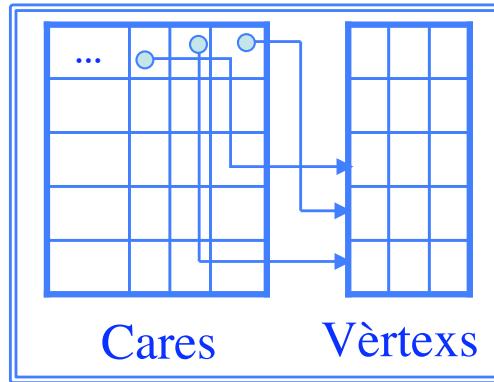
- Cares “orientades”.
- Ordenació vèrtexs coherent amb l’orientació de les cares.
- Cada aresta separa 2 cares.



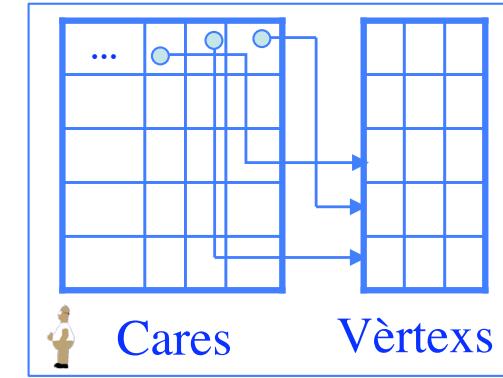
Exemple Laboratori



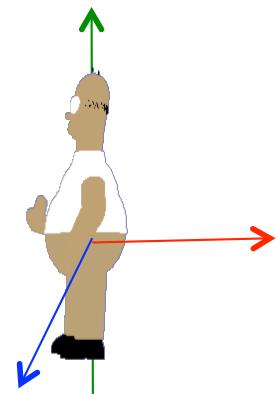
En el marc de les classes de teoria de IDI...



Objectes Coord. de Model



homer.OBJ

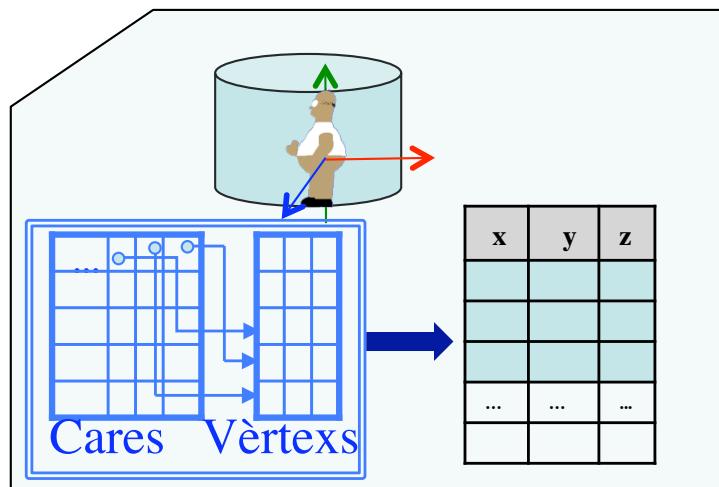


*I com ho fem per
visualitzar el Model?*

Pintar en OpenGL 3.3: “core” mode

Un sol cop cal enviar/passar el *model/geometria* a la GPU/
OpenGL com una llista amb les coordenades dels vèrtexs de
cada triangle o com *triangle strips*.

Per tant, caldrà crear una estructura auxiliar/temporal amb
aquesta informació a partir de la nostra estructura de dades
per poder enviar-la a la GPU.

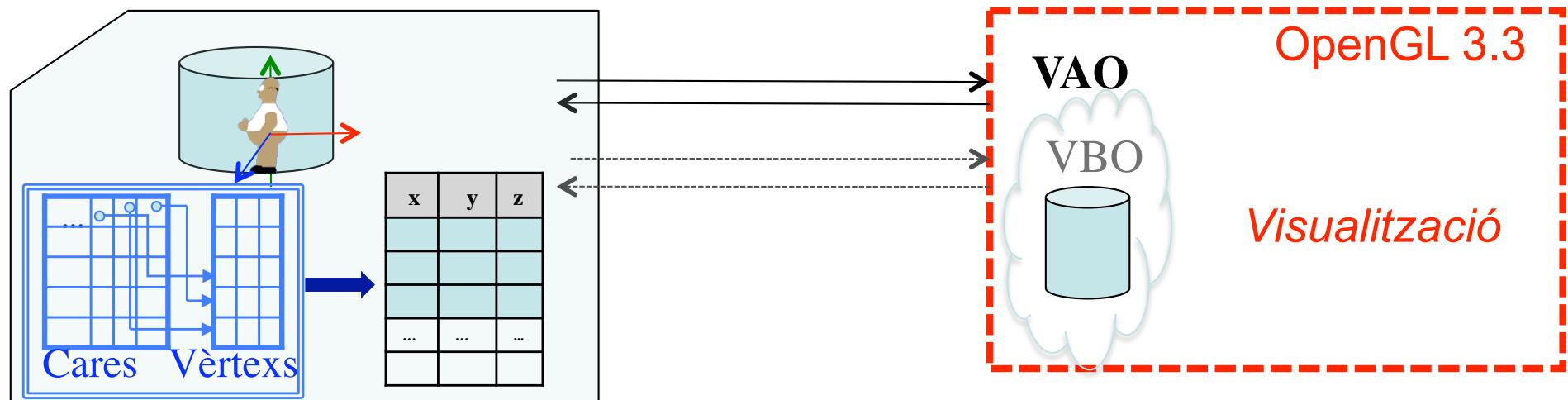


Aplicació. Model Geomètric



Pintar en OpenGL 3.3: “core” mode

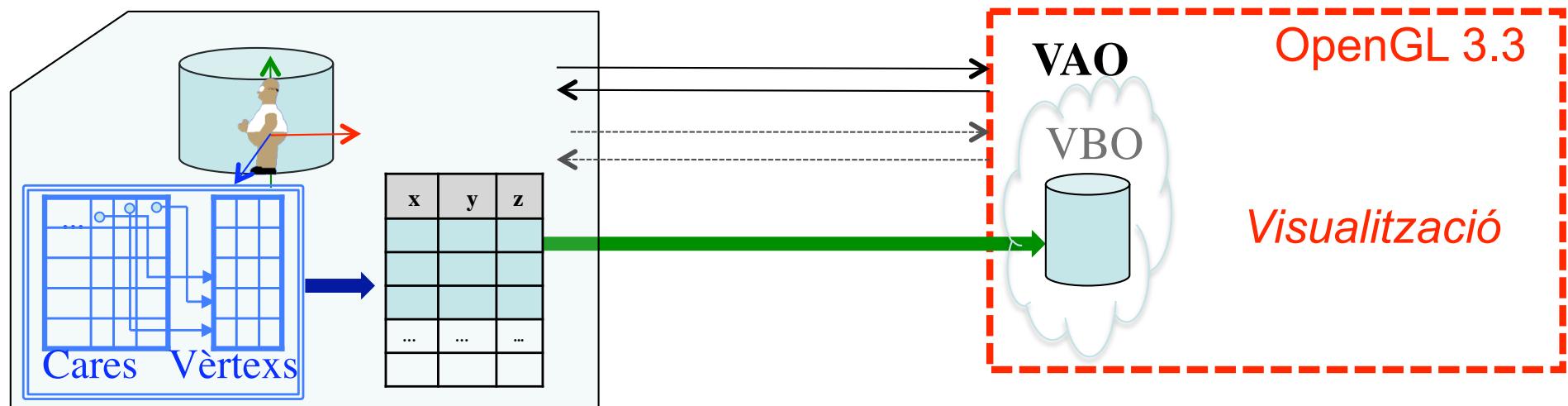
1. Crear en GPU/OpenGL un *VAO* que encapsularà dades del model. Crear *VBO* que guardarà les coordenades dels vèrtexs (potser cal altres per normal, color,...)



Aplicació. Model Geomètric

Pintar en OpenGL 3.3: “core” mode

1. Crear en GPU/OpenGL un *VAO* que encapsularà dades del model.
Crear *VBO* que guardarà les coordenades dels vèrtexs (potser cal altres per normal, color,...)
2. Guardar llista de vèrtexs (amb repetició) o *strips* en el *VBO*
(i si cal, color i normal en els seus *VBO*)



Aplicació. Model Geomètric

Pintar en OpenGL 3.3: “core” mode

1. Crear en GPU/OpenGL un *VAO* que encapsularà dades del model. Crear *VBO* que guardarà les coordenades dels vèrtexs (potser cal altres per normal, color,...)
2. Guardar llista de vèrtexs (amb repetició) o *strips* en el seu *VBO* –i si cal, color i normal en els seus *VBO*-
3. Cada cop que es requereix pintar, indicar el *VAO* a pintar i dir que es pinta: *glDrawArrays(...)*. Acció **pinta_model()** a teoria.

