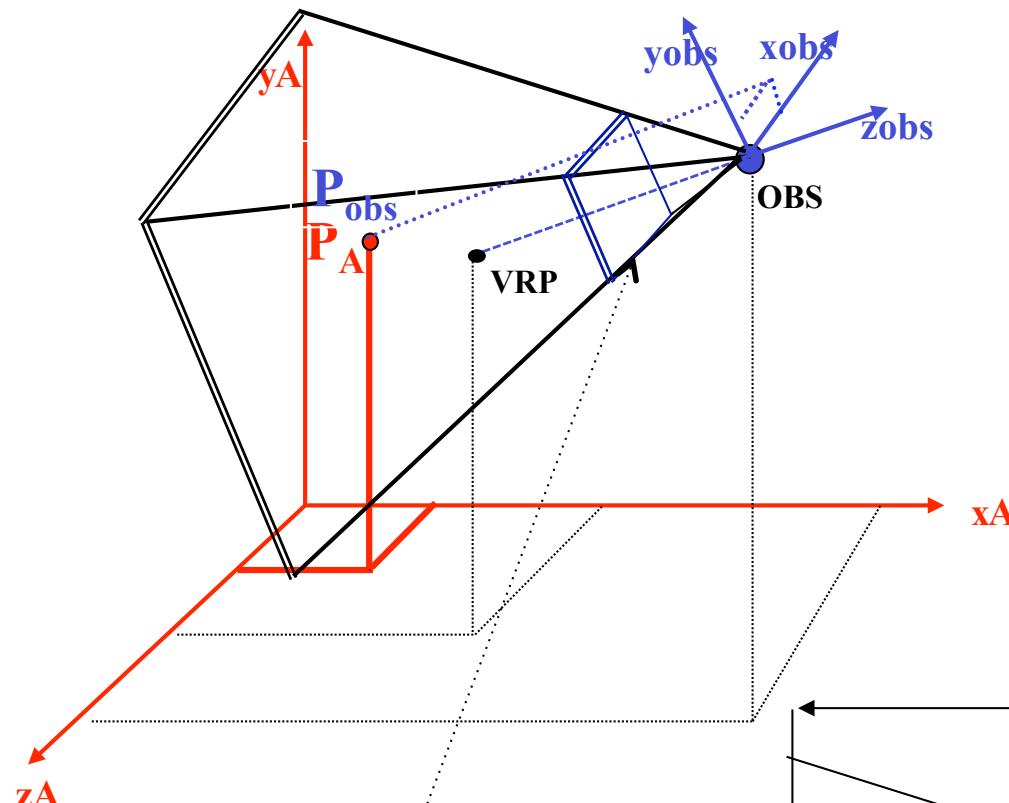


Classe 5: contingut

- Òptica ortogonal: definició
- Zoom
- Exercicis

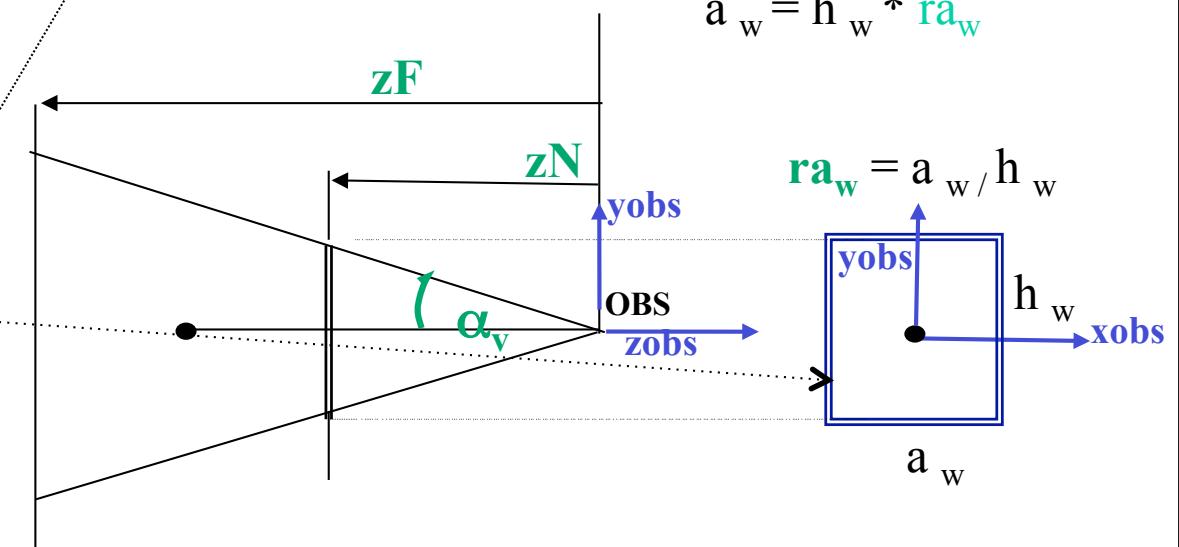


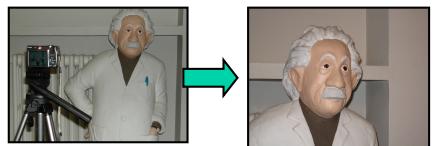
Tipus de càmera: Perspectiva

α_v (FOV = $2\alpha_v$), **zNear**, **zFar**, ra_w

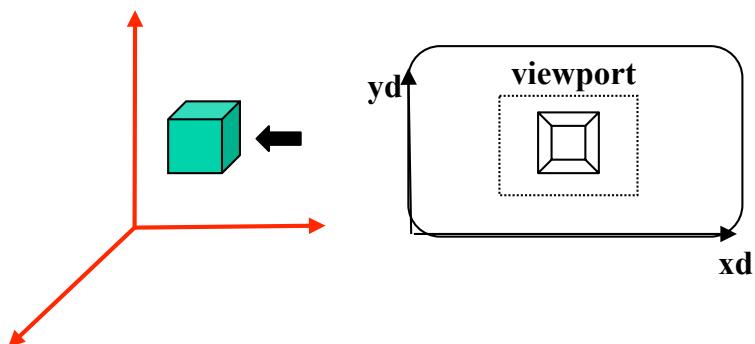
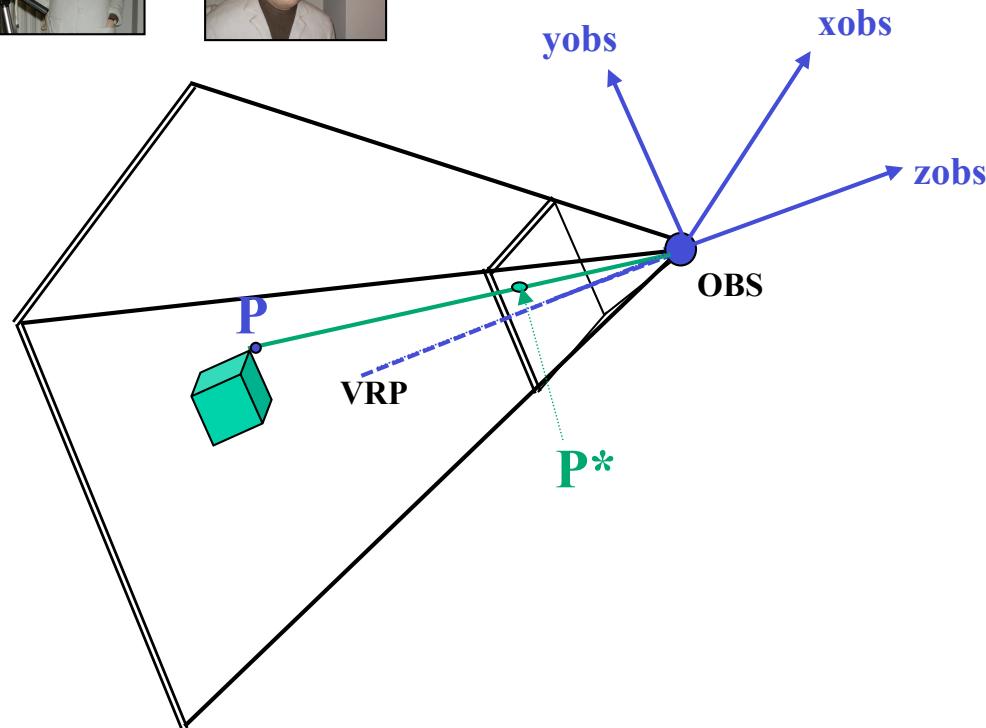
$$h_w = 2 zN \operatorname{tg}(\alpha_v)$$

$$a_w = h_w * ra_w$$

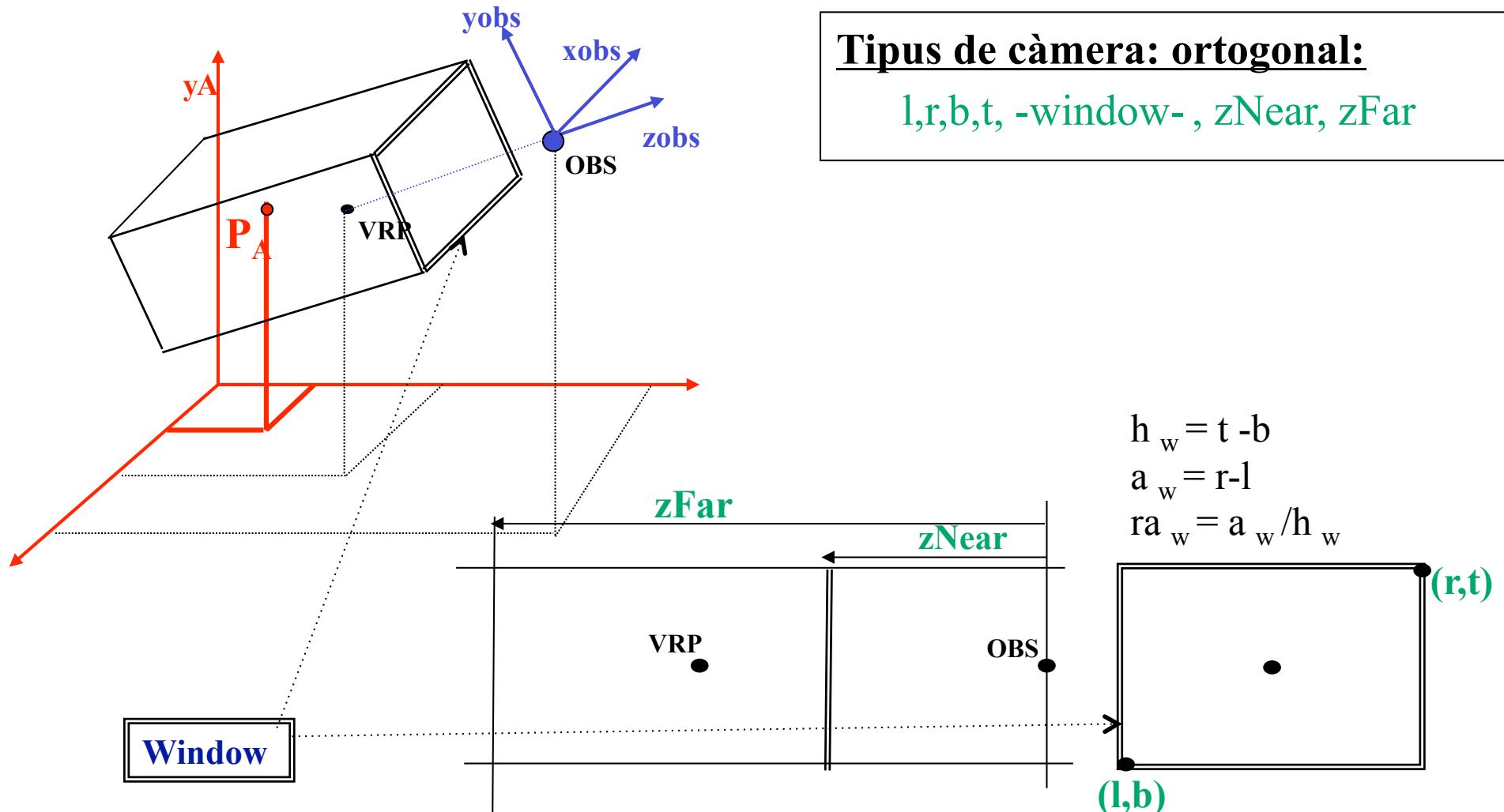




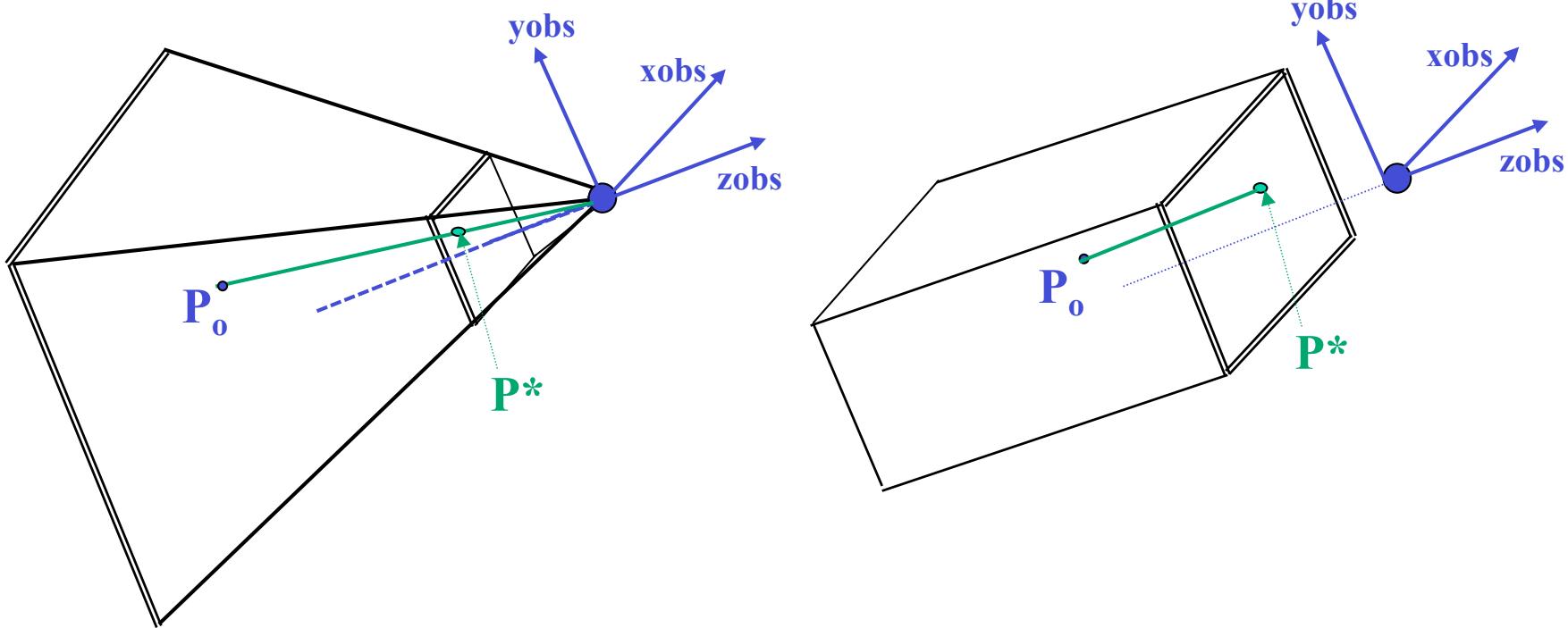
Project transformation

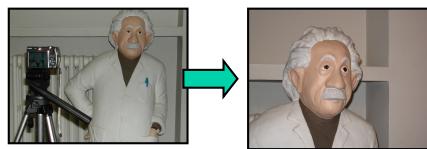


Òptica ortogonal: paràmetres

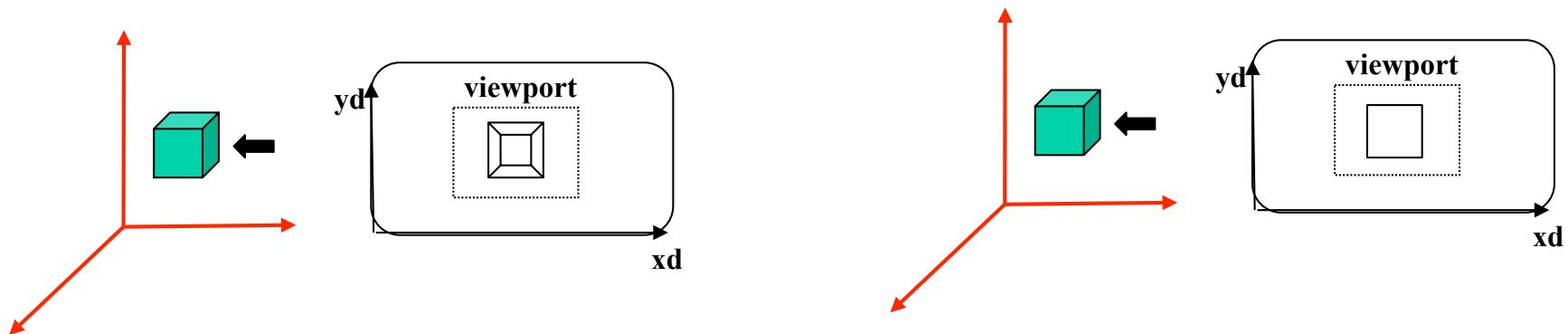
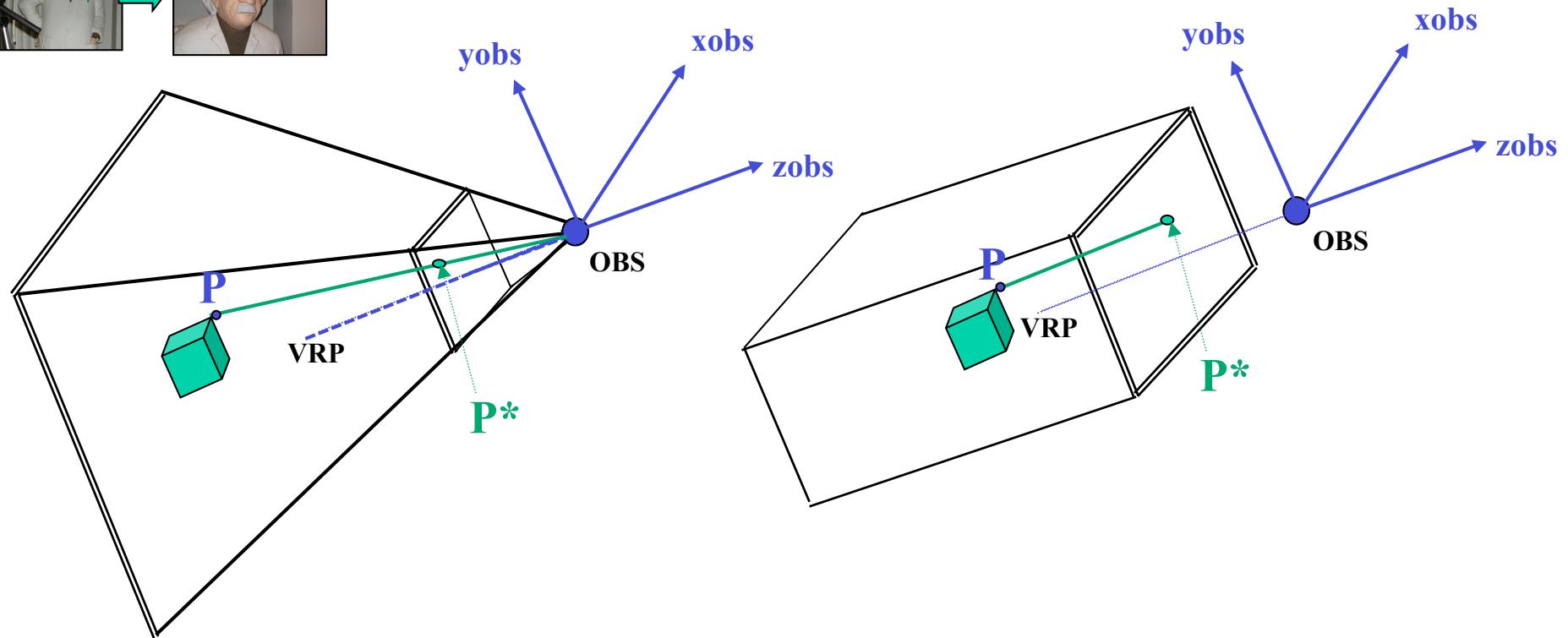


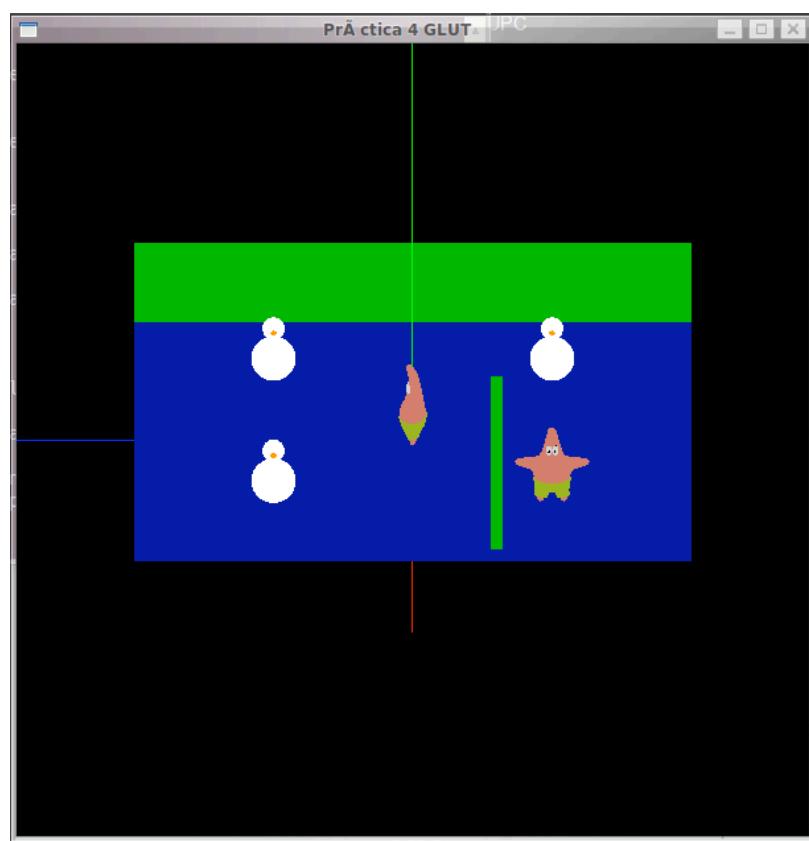
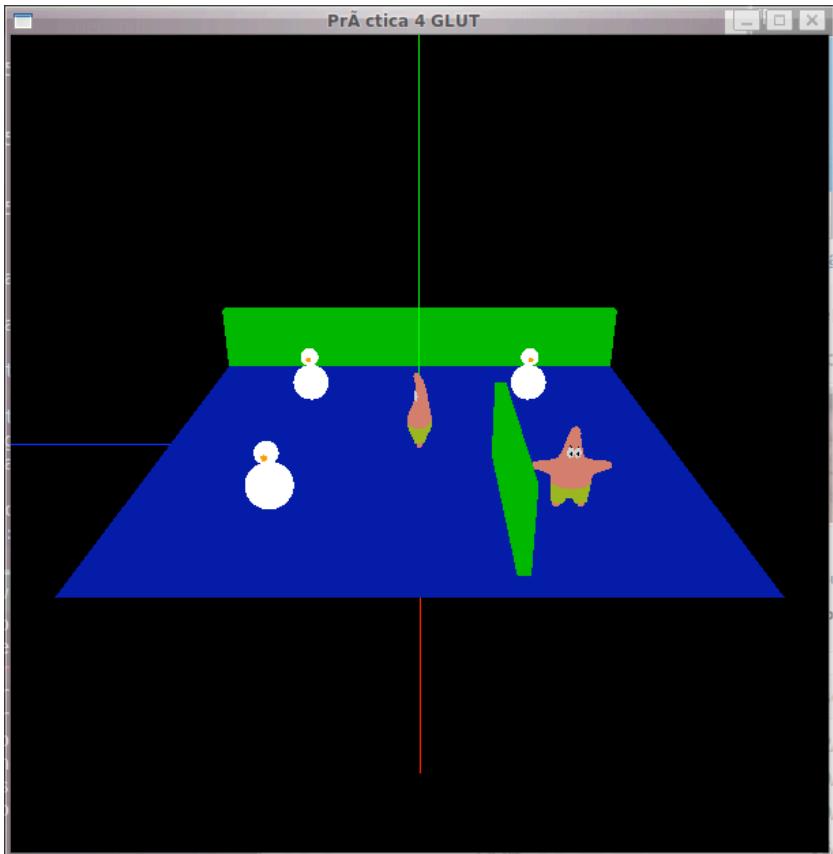
Òptica ortogonal versus perspectiva



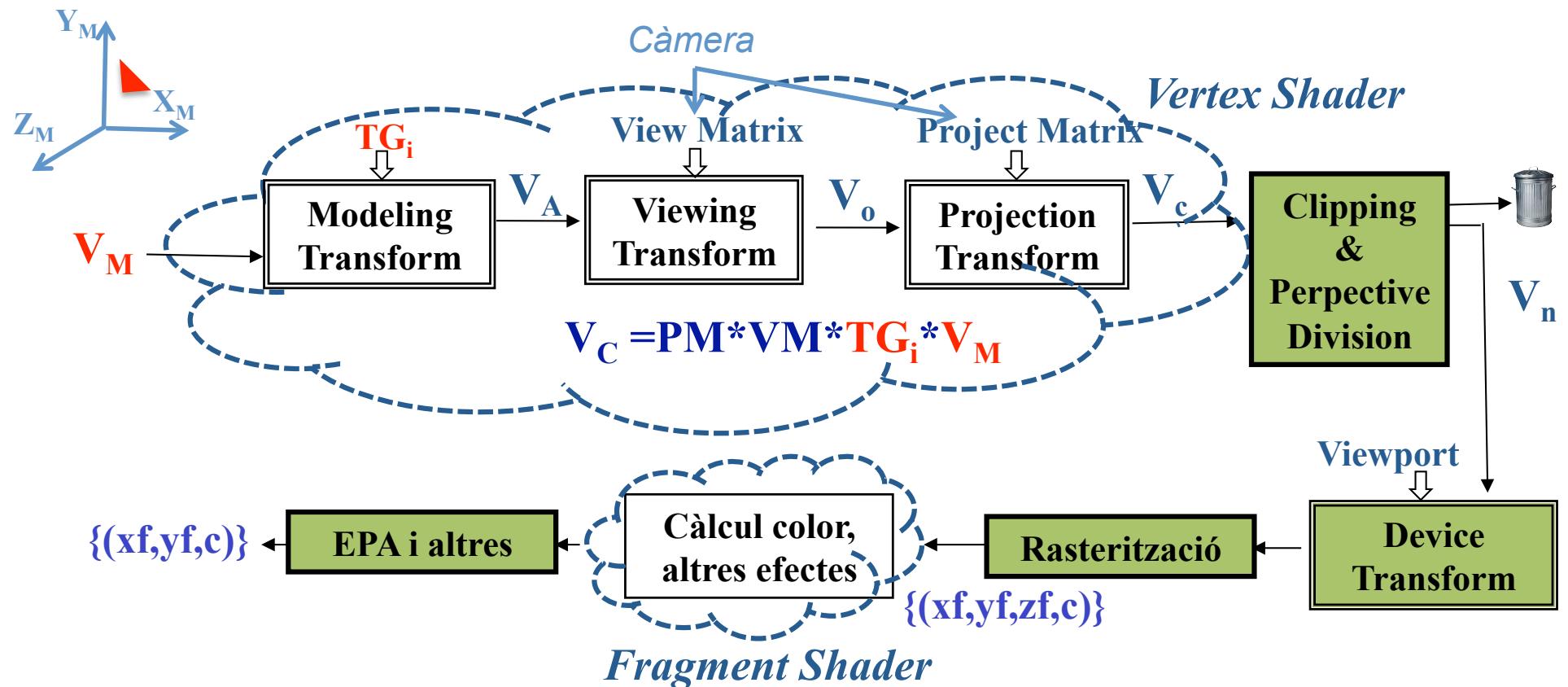
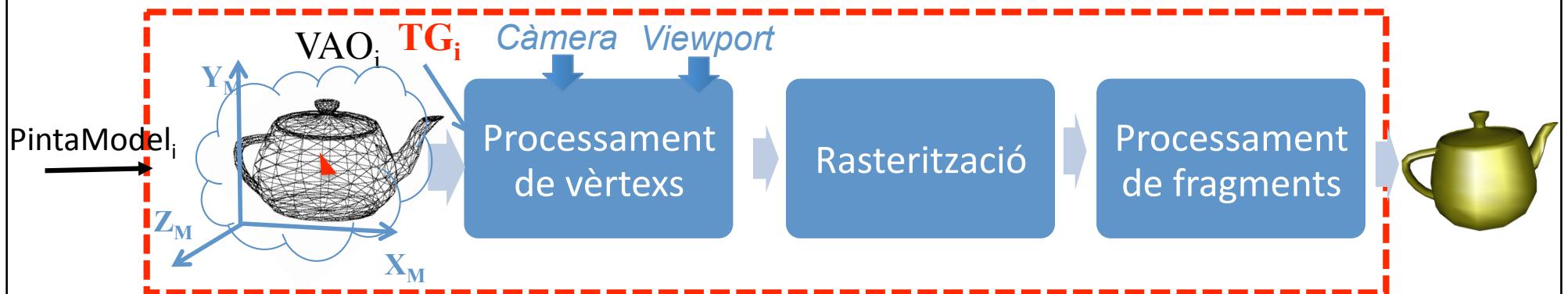


Project transformation





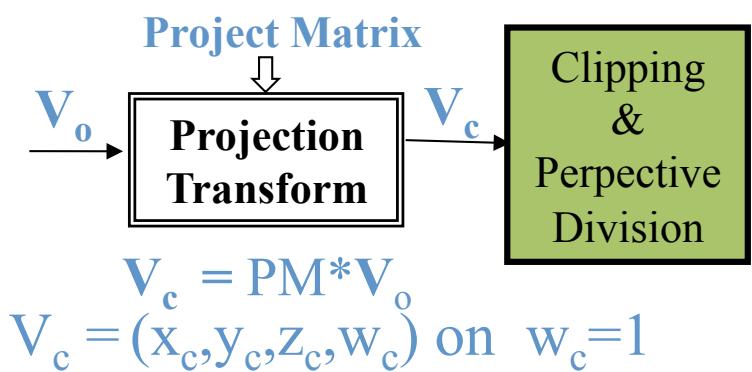
Paradigma projectiu bàsic amb OpenGL 3.3



$l, r, b, t, zNear, zFar \rightarrow$ Càcul de matriu de projecció (PM)

$$PM_o = \begin{pmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & d \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

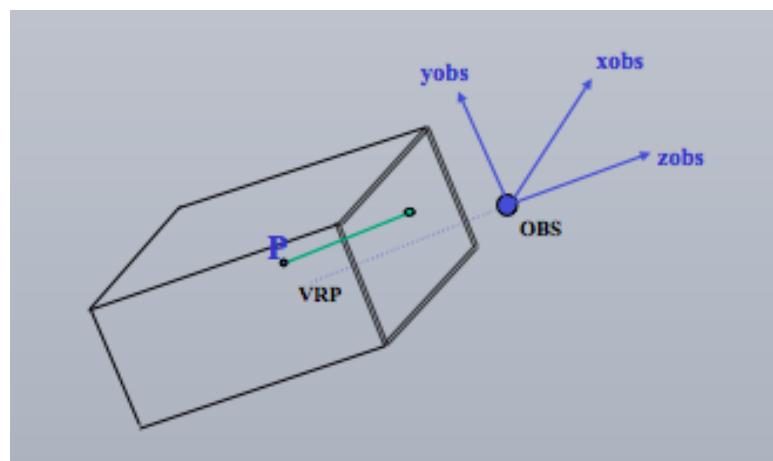
$$\begin{aligned} a &= 2/(r-l) & b &= 2/(t-b) \\ c &= 2/(f-n) & d &= (n+f)/(f-n) \end{aligned}$$



```
PM=ortho(l,r,b,t,zN,zF);
projectMatrix(PM);
```

```
PM= glm::ortho (l, r, b, t, zn, zf);
```

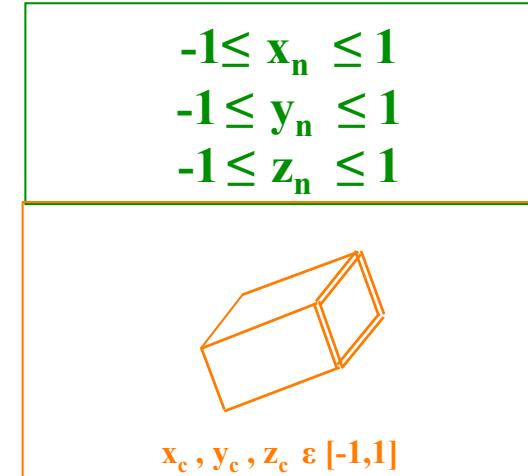
```
glUniformMatrix4fv(....,&PM[0][0]);
```



$$\begin{aligned} -w_c &\leq x_c \leq w_c \\ -w_c &\leq y_c \leq w_c \\ -w_c &\leq z_c \leq w_c \end{aligned}$$

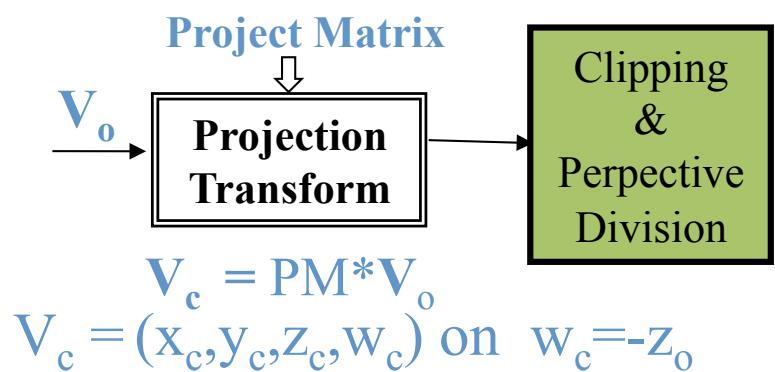
$$\begin{aligned} -1 &\leq x_n \leq 1 \\ -1 &\leq y_n \leq 1 \\ -1 &\leq z_n \leq 1 \end{aligned}$$

$$V_n = V_c / w_c$$

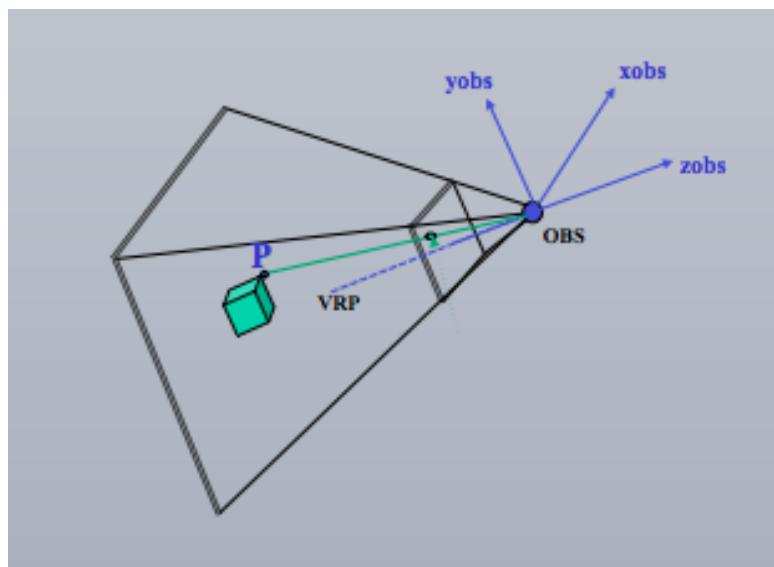


FOV, zNear, zFar, ra → Càcul de matriu de projecció (PM)

$$PM = \begin{pmatrix} 1/ra*a & 0 & 0 & 0 \\ 0 & 1/a & 0 & 0 \\ 0 & 0 & c & d \\ 0 & 0 & -1 & 0 \end{pmatrix} \quad \begin{aligned} a &= \tan(FOV/2) \\ c &= (f+n)/(n-f) \\ d &= 2nf/(n-f) \end{aligned}$$



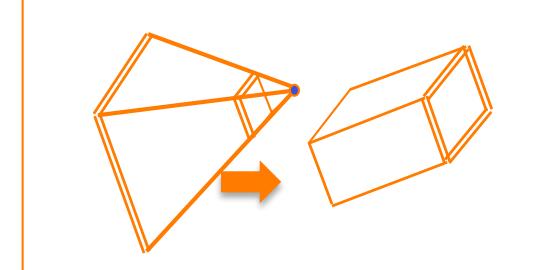
```
PM=perspective (FOV,ra,zN,zF);
projectMatrix(PM);
```



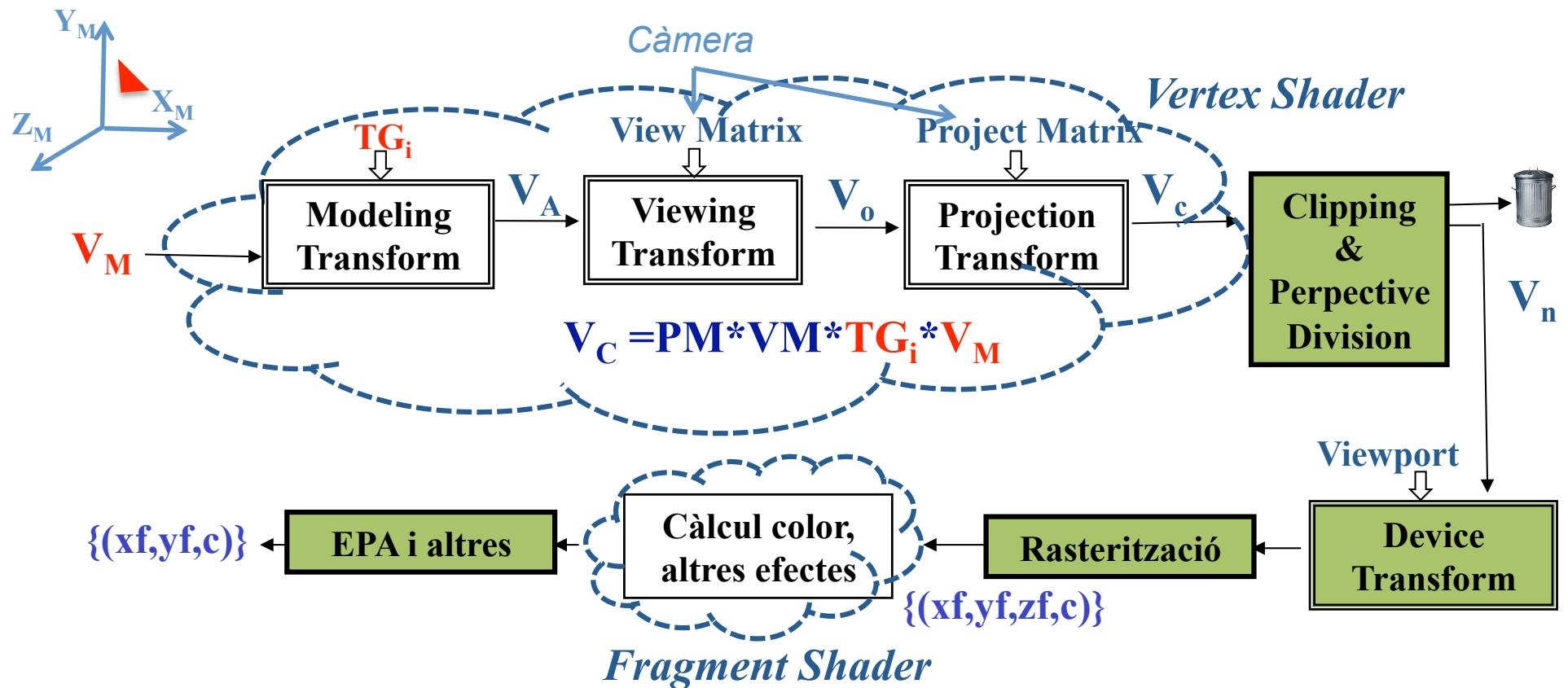
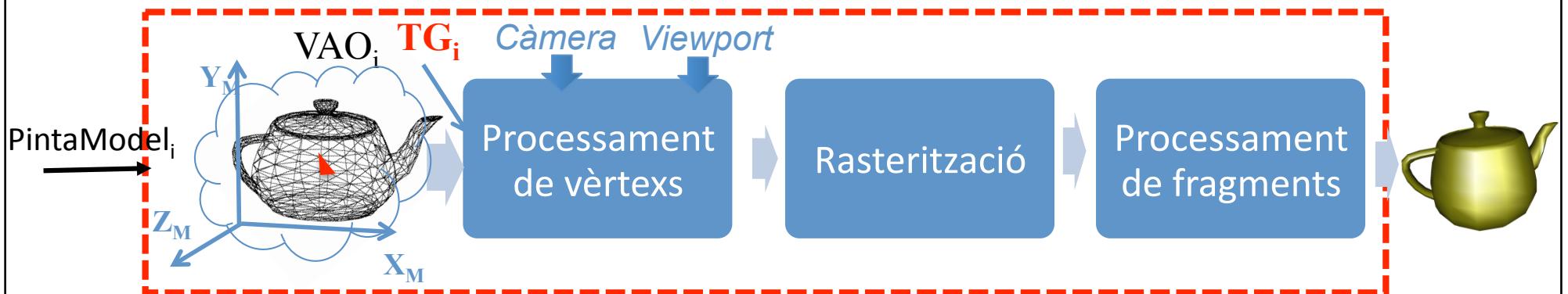
$$\begin{aligned} -w_c &\leq x_c \leq w_c \\ -w_c &\leq y_c \leq w_c \\ -w_c &\leq z_c \leq w_c \end{aligned}$$

$$\begin{aligned} -1 &\leq x_n \leq 1 \\ -1 &\leq y_n \leq 1 \\ -1 &\leq z_n \leq 1 \end{aligned}$$

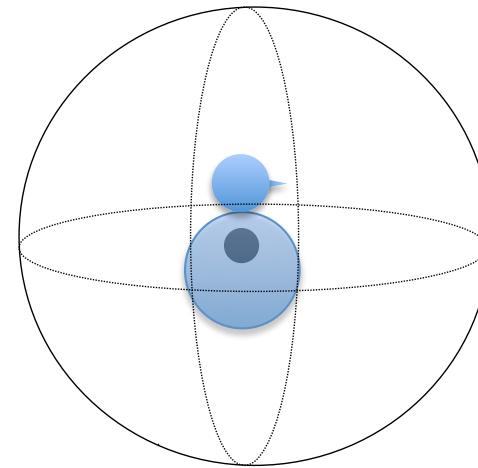
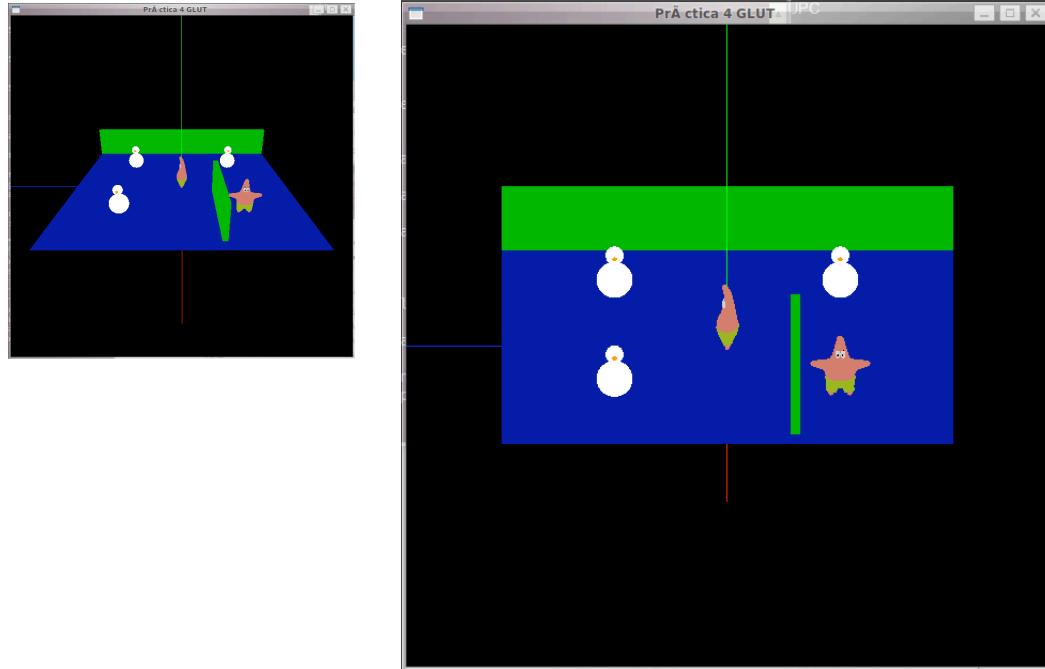
$$V_n = V_c / w_c$$



Paradigma projectiu bàsic amb OpenGL 3.3



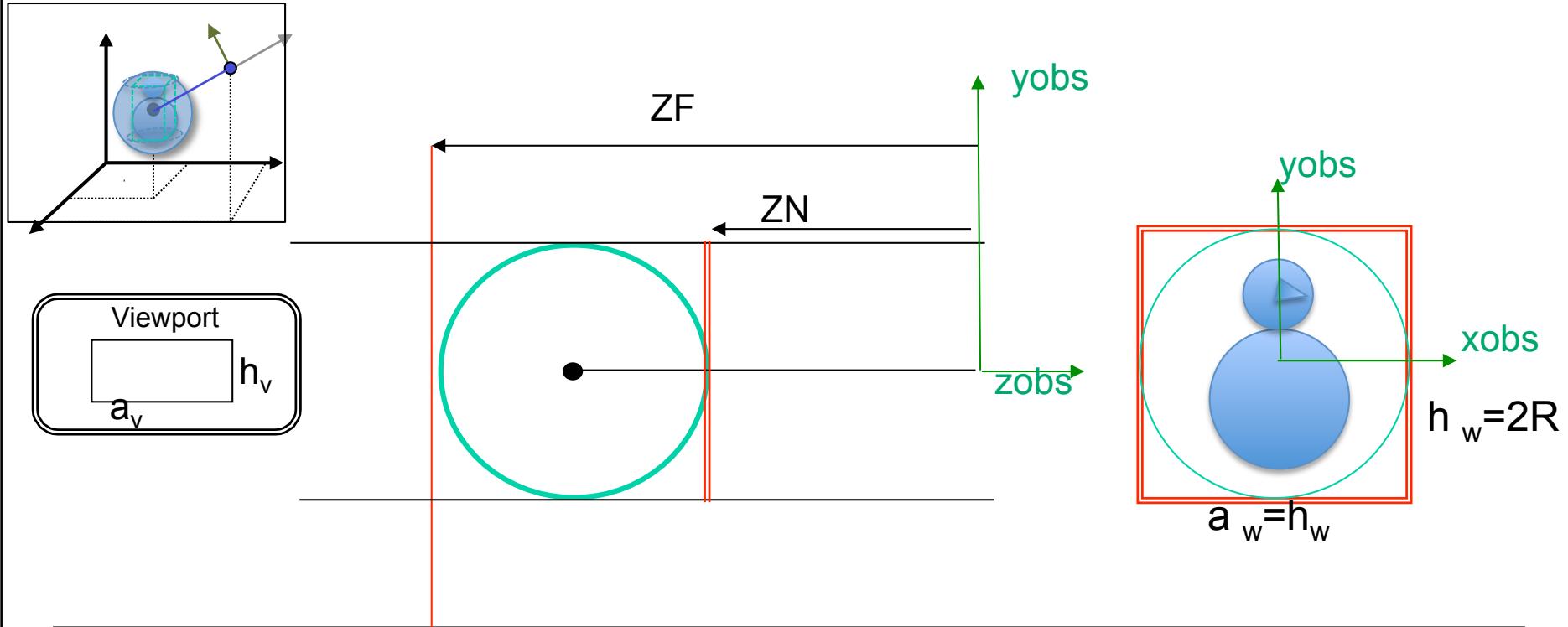
Òptica per càmera 3ra persona en ortogonal



Quins paràmetres de posició, orientació i òptica per càmera en 3ra persona? →
imatge inclogui tota l'escena, optimitzant l'espai del viewport.

Dada: capsa mínima contenidora $(x_{\min}, y_{\min}, z_{\min}) - (x_{\max}, y_{\max}, z_{\max})$

Tota escena en la vista, sense deformar i càmera ortogonal



- ZN i ZF mateix raonament que en càmera perspectiva.
- Window mínim requerit (central)= $(-R,-R,R,R)$ => una $ra_w = 1$ (per què ?)
- Si $ra_w \neq ra_v \Rightarrow$ deformació (per què?)
 - Si $ra_v > 1 \Rightarrow$ cal incrementar la $ra_w \Rightarrow$ modificar window
 com $ra_w = a_w/h_w \Rightarrow$ podem incrementar a_w o decrementar h_w (és retallaria esfera!!)
 Per tant:

$$a_w^* = ra_v * h_w = ra_v * 2R$$

$$\text{window} = (-R ra_v, R ra_v, -R, R)$$

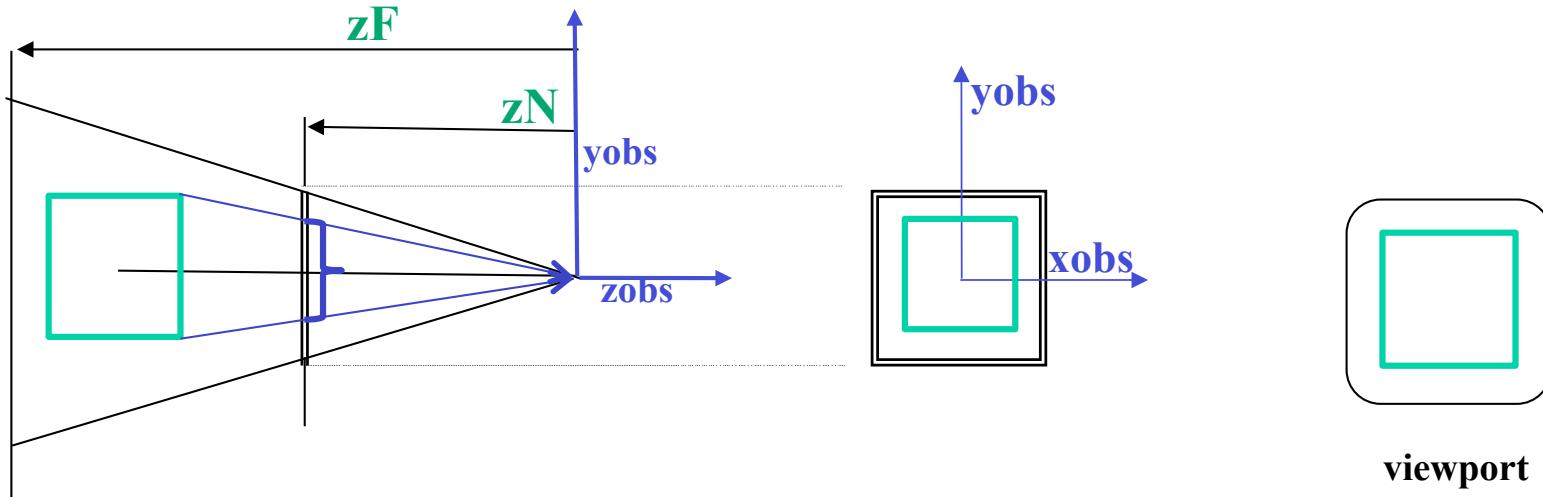
 - raonament similar per recalcular window quan $ra_v < 1$

Classe 5: contingut

- Òptica ortogonal: definició
- Zoom
- Exercicis

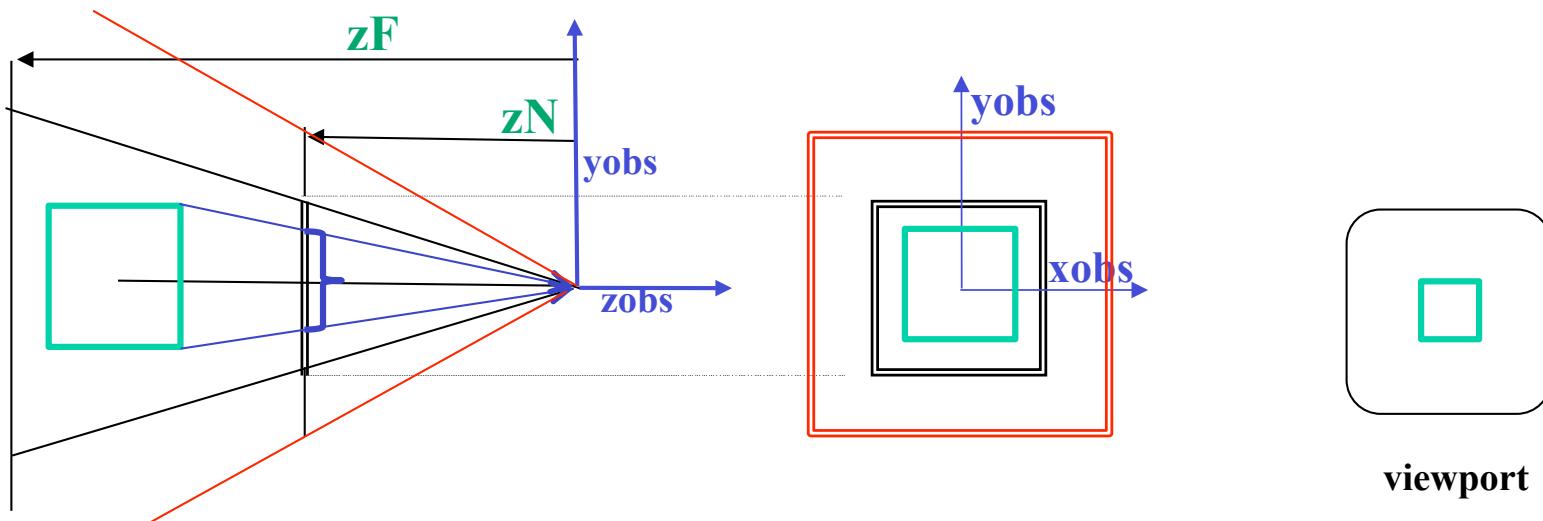
L'òptica i el ZOOM

- a) Modificar l'angle d'obertura (tot mantenint la *ra*) ;
Modificar window en ortogonal
- b) Modificar la distància de l'Obs al VRP
(modificant ZN i ZF adequadament)
- c) Modificar Obs i VRP en la direcció $-v \rightarrow$ travelling



L'òptica i el ZOOM

- a) Modificar l'angle d'obertura (tot mantenint la *ra*) ;
Modificar window en ortogonal
- b) Modificar la distància de l'Obs al VRP
(modificant ZN i ZF adequadament)
- c) Modificar Obs i VRP en la direcció $-v \rightarrow$ travelling

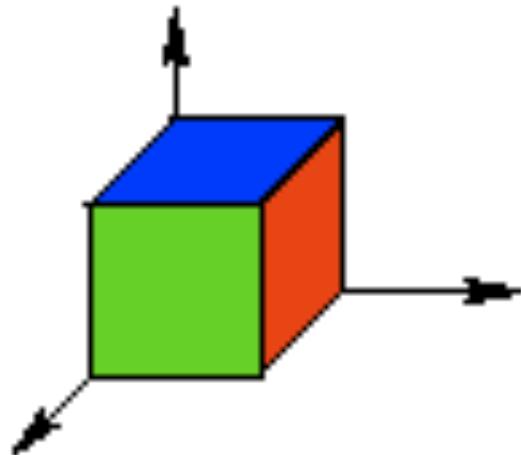


Classe 5: contingut

- Òptica ortogonal: definició
- Zoom
- Exercicis

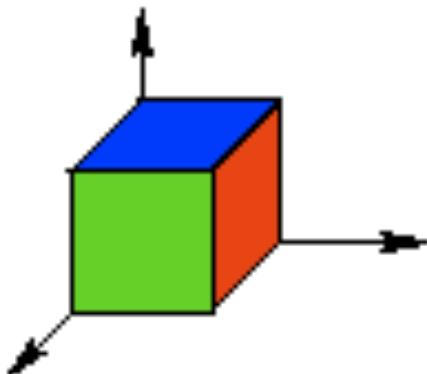
77. (2015-2016P Q1) Tenim una escena amb un cub de costat 2 orientat amb els eixos i de manera que el seu vèrtex mínim està situat a l'origen de coordenades. La cara del cub que queda sobre el pla $x=2$ és de color vermell, la cara que queda sobre el pla $z=2$ és de color verd i la resta de cares són blaves.

a) Indica TOTS els paràmetres d'una càmera perspectiva que permeti veure completes a la vista només les cares vermella i verda. La relació d'aspecte del viewport (vista) és 2. Fes un dibuix indicant la imatge final que s'obtindria.

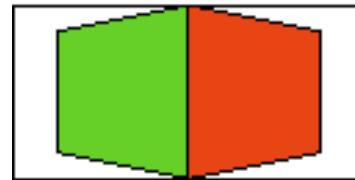


77. (2015-2016P Q1) Tenim una escena amb un cub de costat 2 orientat amb els eixos i de manera que el seu vèrtex mínim està situat a l'origen de coordenades. La cara del cub que queda sobre el pla $x=2$ és de color vermell, la cara que queda sobre el pla $z=2$ és de color verd i la resta de cares són blaves.

- a) Indica TOTS els paràmetres d'una càmera perspectiva que permeti veure complertes a la vista només les cares vermella i verda. La relació d'aspecte del viewport (vista) és 2. Fes un dibuix indicant la imatge final que s'obtindria.
- b) Quin efecte tindria en la imatge final modificar l'òptica ortogonal? Defineix la càmera ortogonal.



$$\begin{aligned} \text{VRP} &= (2,1,2) \\ \text{OBS} &= (3,1,3) \\ \text{Up} &= (0,1,0) \end{aligned}$$



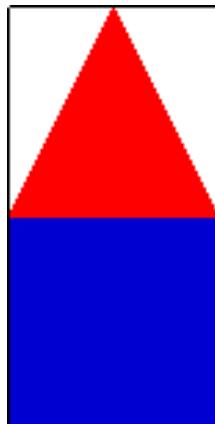
Exercici 45. Una esfera de radi 1 es visualitza en un viewport quadrat de 400 per 400, amb una càmera posicionada correctament per poder veure tota l'esfera, i on el mètode per a definir la projecció de la càmera utilitza la següent crida:

```
TP = Perspective (60.0, 1.0, 1.0, 10.0);  
projectMatrix (TP);
```

L'usuari ha redimensionat la finestra a 500 d'amplada per 400 d'alçada. Digues què cal canviar de la càmera per tal que es vegi l'esfera correctament (sense retallar-la ni deformar-la).

- a. Incrementar l'angle d'obertura vertical (FOV) i la relació d'aspecte del window.
- b. Augmentar la relació d'aspecte del window i la distància al ZNear.
- c. Només augmentar la relació d'aspecte del window.
- d. Només canviar l'angle d'obertura vertical (FOV).

Tenim una piràmide de base quadrada de costat 5, amb la base centrada al punt (0,0,2.5) i alçada de la piràmide 5 amb l'eix en direcció Z+. A l'escena tenim també un cub de costat 5 centrat a l'origen. El viewport esta definit amb glViewport (0,0,400,800). Si a la vista es veu la imatge que teniu al dibuix (caseta), quines inicialitzacions d'una càmera ortogonal (posició+orientació i òptica) permetrien veure aquesta imatge? Tots els angles estan en graus.



<pre>PM=perspective (90, 1, 5, 10); projectionMatrix (PM) VM=translate (0,0,-10); VM=VM*rotate (90,1,0,0); VM=VM*translate (0,0,-2.5); viewMatrix (VM); pinta_escena ();</pre>	<pre>PM=ortho (-2.5, 2.5, -5, 5, 5, 10); projectionMatrix (PM) VM=translate (0,0,-7.5); VM=VM*rotate (-90,0,0,1); VM=VM*rotate (90,0,1,0); VM=VM*translate (0,0,-2.5); viewMatrix (VM); pinta_escena ();</pre>
<pre>PM=ortho (-2.5,2.5,-5,5,5, 10); projectionMatrix (PM) VM=translate (0,0,-7.5); VM=VM*rotate (90,0,0,1); VM=VM*rotate (90,0,1,0); VM=VM*translate (0,0,-2.5); viewMatrix (VM); pinta_escena ();</pre>	<pre>PM=ortho (-5, 5, -5, 5, 5, 10); projectionMatrix (PM) VM=translate (0,0,-7.5); VM=VM*rotate (90,0,0,1); VM=VM*rotate (90,0,1,0); VM=VM*translate (0,0,-2.5); viewMatrix (VM); pinta_escena ();</pre>

Exercici 43: Indica quina de les inicialitzacions de l'òptica perspectiva és més apropiada per a una càmera que porta un observador que camina per una escena fent fotos amb una òptica constant. Esfera englobant d'escena té radi R , d és la distància entre OBS i VRP. Observació: ra_v és la relació d'aspecte del *viewport*

- a) $\text{FOV} = 60^\circ$, $ra = ra_v$, $zNear = 0.1$, $zFar = 20$
- b) $\text{FOV} = 60^\circ$, $ra = ra_v$, $zNear = R$, $zFar = 3R$;
essent R el radi de l'esfera contenidora de l'escena.
- c) $\text{FOV} = 2 * (\arcsin(R/d) * 180/\text{PI})$, $ra = ra_v$, $zNear = R$, $zFar = 3R$;
essent R el radi de l'esfera contenidora de l'escena i d la distància d'OBS a VRP.
- d) $\text{FOV} = 2 * (\arcsin(R/d) * 180/\text{PI})$, $ra = ra_v$, $zNear = 0$, $zFar = 20$;
essent R el radi de l'esfera contenidora de l'escena i d la distància d'OBS a VRP

Exercici 48: Disposem d'una càmera ortogonal amb els següents paràmetres:

OBS=(0.,0.,0.), VRP=(-1.,0.,0.), up=(0.,1.,0.), window de (-5,-5) a (5,5), ra=1, zn=5, zf=10.

Indiqueu quin conjunt de paràmetres d'una càmera perspectiva defineix un volum de visió que conté l'anterior (és a dir, garanteix que es veurà, coma mínim, el mateix que amb la càmera axonomètrica):

- a) FOV= 90, ra=1, zn= 5, zf=10
- b) FOV= 60, ra=1, zn=5, zf=10
- c) FOV= 60, ra= 2, zn=6, zf=11
- d) FOV= 90, ra= 0.5, zn=5, zf=10

Exercici 35: Defineixen la mateixa VM?:

Codi 1

```
VM= LookAt ((0,80,0),(0,50,0),(1,0,0));
```

Codi 2

```
VM = Translate(0, 0, -80);  
VM = VM * Rotate (90, 0, 0, 1);  
VM = VM * Rotate (90 , 1, 0, 0);  
VM = VM * Rotate(-90, 0, 1, 0);
```

Preguntes:

- a) Defineixen la mateixa càmera?
- b) Quina vista de l'escena?
- c) Es poden optimitzar les TGs del codi 2?

Hi ha una llista de molts exercicis, alguns a fer:

33, 58, 63, 53, 67, 73, 85,...