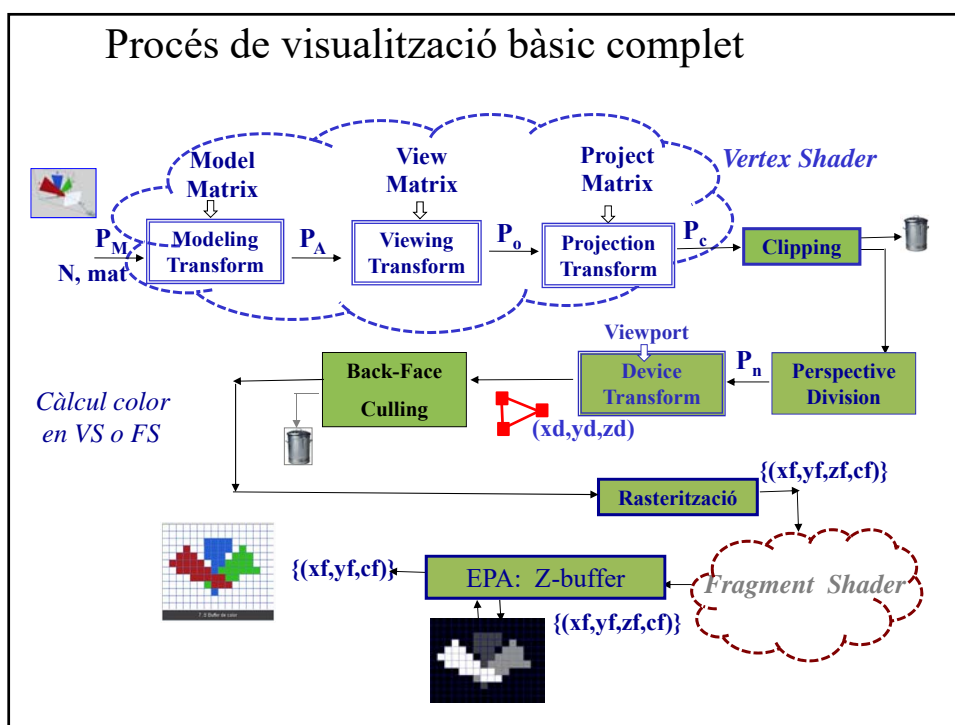
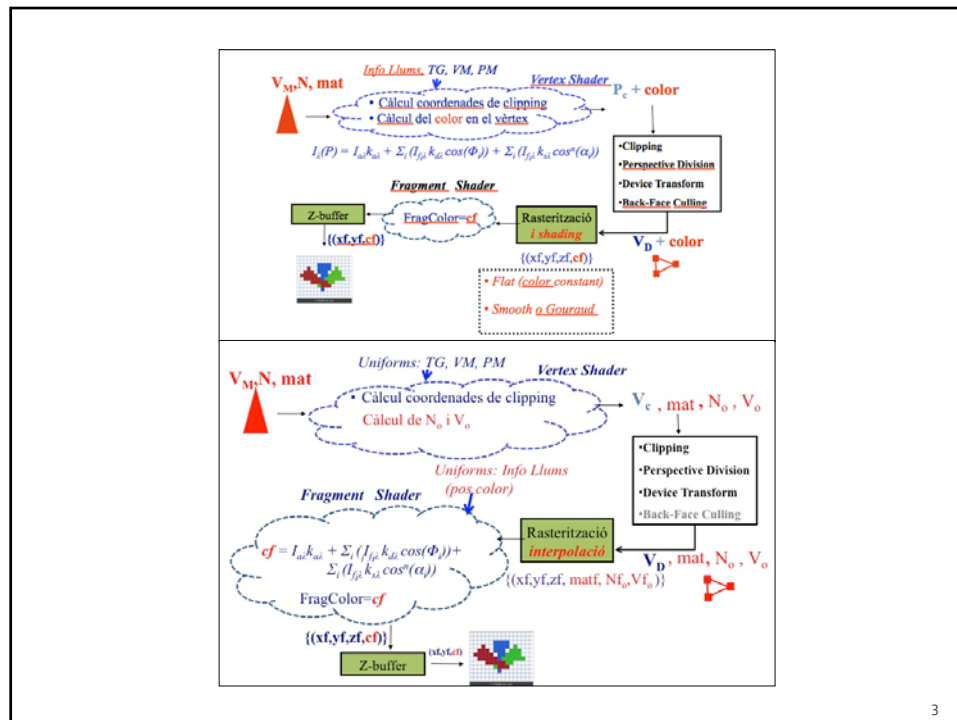


# Interacció i Disseny d'Interfícies





3

## Realisme

Per a poder veure les escenes amb un cert realisme s'activa el z-buffer i s'implementa il·luminació de Phong en el Vertex Shader. Per a què tot funcioni correctament quan es pinta l'escena:

- En fer el càlcul d'il·luminació cal que la normal i els vectors L, R i V estiguin normalitzats.
- En fer el càlcul d'il·luminació cal que la normal, el vector L i el vèrtex estiguin normalitzats.
- Si en comptes de z-buffer s'utilitza back-face culling, no cal que s'indiqui la normal per càlcul d'il·luminació.
- Si no s'ha fet cap escalat als objectes, no cal que la normal estigui normalitzada.

## Càmera

Tenim una escena formada per un "terra" de dimensions 10x10 ubicat en el pla  $Y=0$  i centrat en el  $(0,0,0)$ , i un objecte amb capsa contenidora de punt mínim  $(-2.5,0,-2.5)$  i punt màxim  $(2.5,10,2.5)$ . Volem definir una càmera en tercera persona que mostri l'escena centrada en el viewport, quina de les següents afirmacions és correcta?

- a) Es pot calcular la viewMatrix usant lookAt() amb el següent pseudocodi:  
`glm::vec3 OBS (0, 0, 20);`  
`glm::vec3 VRP (0, 0, 0);`  
`glm::vec3 up (0, 1, 0);`  
`VM = lookAt (OBS, VRP, up);`
- b) No es pot definir el posicionament de la càmera si no sabem el tipus d'òptica.
- c) Un possible càlcul de la viewMatrix (VM) seria amb el següent pseudocodi:  
`VM = Translate (0, 0, -20);`  
`VM = VM * Rotate (30, 1, 0, 0);`  
`VM = VM * Rotate (30, 0, 1, 0);`
- d) Cap de les altres respostes és correcta.

## Càmera

Un estudiant planteja el següent pseudocodi.

```
void painGL(){
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    glEnable (GL_DEPTH_TEST);
    Set_càmera();
    glViewport (0,0,500,500);
    pinta_escena();
    Set_càmera();
    glViewport (500,500,500,500);
    pinta_escena();}
```

Si la pantalla (finestra gràfica) és de 1000x1000, Quina afirmació és correcta?

- a) Es veuran dues imatges de l'escena a la pantalla, una a baix a l'esquerra i l'altre dalt a la dreta.
- b) No es pot modificar el *viewport* mentre s'està pintant l'escena, el resultat és indeterminat.
- c) El segon *viewport* no està definit correctament. La segona imatge es projectarà en un píxel.
- d) Si es vol pintar dues vegades l'escena, **només** es pot fer modificant les transformacions geomètriques que se li apliquen i pintant en un mateix *viewport*.

## Càmera

Quina de les següents afirmacions és incorrecta?

- a) Si tenim una càmera axonomètrica i reduïm el seu window (respectant la seva relació d'aspecte), estem fent un zoom-in.
- b) L'algorisme de retallat (clipping) és el mateix sigui la càmera perspectiva o axonomètrica.
- c) L'eix Y del sistema de coordenades de l'observador (SCO) sempre es projecta vertical (direcció Y) en el sistema de coordenades de dispositiu (SCD).
- d) Si incrementem el FOV de la càmera perspectiva, haurem d'incrementar la relació d'aspecte, per a què el window mantingui la seva proporció.

## II.luminació

Volem il·luminar un polígon de 10x10 ubicat sobre el pla XZ i centrat en l'origen, amb un focus de llum blanca ubicat en la posició (0,2,0). No hi ha llum ambient. La normal del polígon és (0,1,0). Les constants de material del polígon són  $K_d=(0,0.8,0)$ ,  $K_s=(1,1,1)$  i Shininess= 100. Indica quina de les següents afirmacions és la correcta:

- a) Com la llum ha d'estar fixa en l'escena, el càlcul de la il·luminació s'ha de fer obligatòriament en el vèrtex shader per a cada vèrtex del polígon.
- b) Si el càlcul de la il·luminació es realitza en el fragment shader, cal passar la posició de la llum i la normal a coordenades de dispositiu.
- c) Si el càlcul de la il·luminació es realitza en el vèrtex shader, cal que les posicions del vèrtex, del focus i la normal estiguin referenciades totes respecte al sistema de coordenades de l'aplicació o de l'observador.
- d) La imatge que s'obtindrà del polígon serà la mateixa tant si els càlculs es realitzen en el vèrtex com en el fragment shader; sempre que es realitzin en el sistema de coordenades adient.

## II.luminació

En un examen de laboratori d'OpenGL es demana posicionar un focus de llum d'escena.

Com podrà el professor detectar que realment l'estudiant ha definit un llum d'escena i no de càmera?

- a) Ho podrà detectar si modifica el FOV per a fer un zoom.
- b) Ho podrà detectar si modifica el color del focus de llum.
- c) Només ho podrà detectar si mira el codi però no executant l'aplicació, encara que pugui moure la càmera.
- d) Ho podrà detectar si mou la càmera al voltant de l'escena.

Professors d'IDI - UPC

## Interacció i Disseny d'Interfícies