

### Executar viewer:

~/assig/grau-g/Viewer/GLarenaSL

(dins del directori on volem que se'ns guardin els shaders que creem)

### Viewer Plugins:

/dades/nacho.llado.cortes/G/Viewer/GLarenaPL

### Activar Snippets:

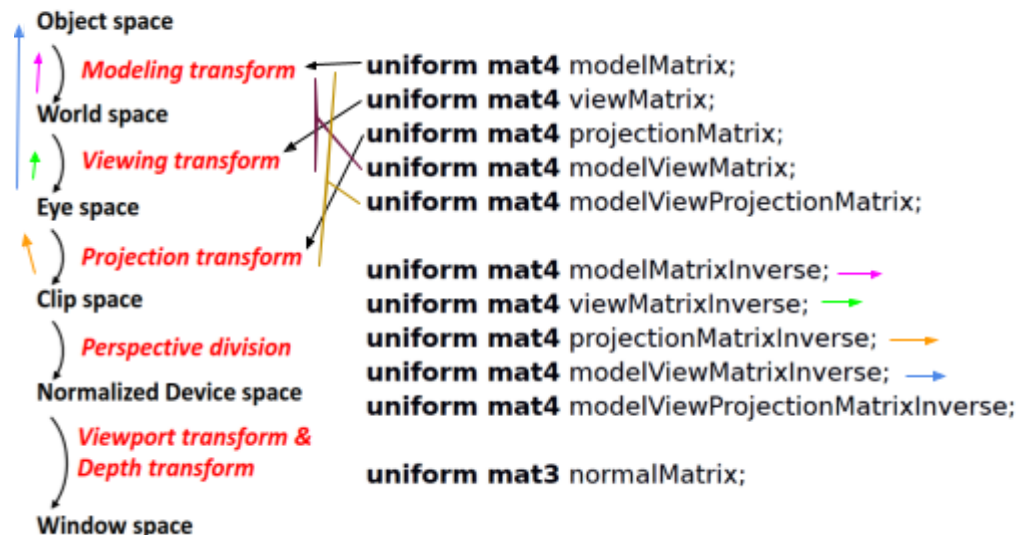
~/assig/grau-g/gedit-config

(activar plugin snippets i escriure defs[tab] per a que surti tot)

### Web Assignatura:

<https://www.cs.upc.edu/~virtual/G/>

## Sistemes de coordenades i matrius



### Model de Phong:

$N$  = vector normal unitari = `normalize(normalMatrix * normal);`

$L$  = vector unitari cap a la font de llum = `lightPosition.xyz - P;`

$P$  = vector unitari de la càmera cap al vèrtex = `(modelViewMatrix * vec4(vertex.xyz,1)).xyz;`

$V$  = vector unitari del vèrtex cap a la càmera = `-P`

$R$  = reflexió del vector  $L$  respecte  $N$  = `2 * (N * L) * N - L`

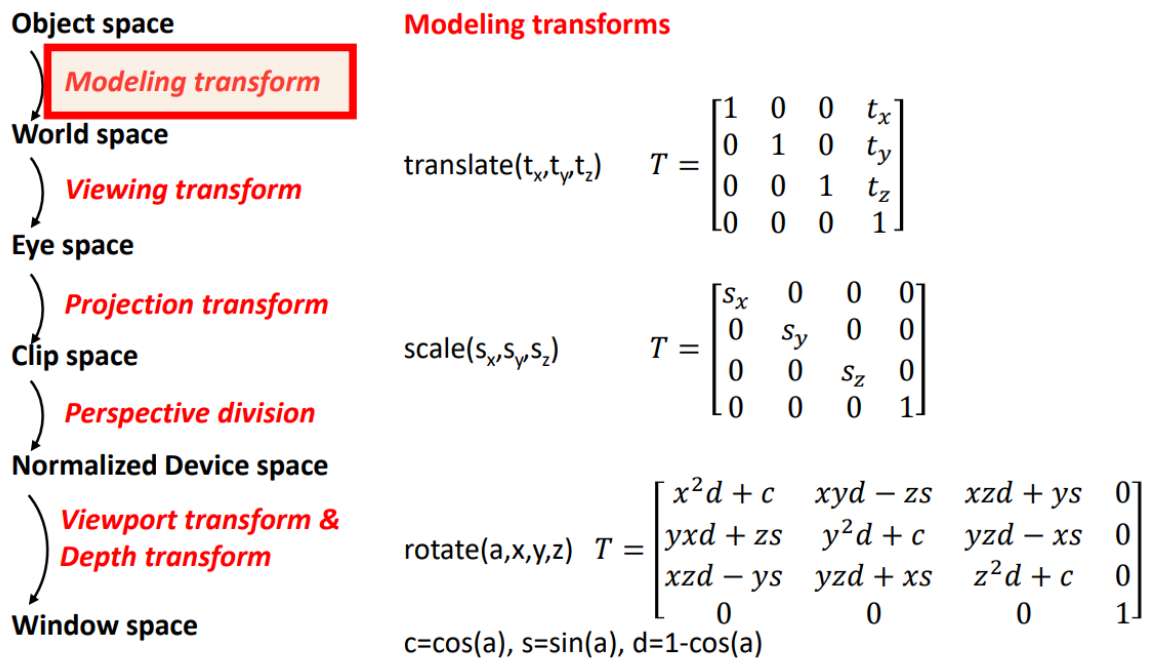
<https://learnopengl.com/Getting-started/Coordinate-Systems>

La normal a eye space multiplicar per la normalMatrix.

La resta entre el lightPosition i el punt V per obtenir el vector L es fa en eye space!

Per treballar les coordenades de textura quan hem d'agafar fragments, reescalar-les al domini que ens és fàcil treballar  $[0, 1] \rightarrow *6 \rightarrow [0, 6]$

# Transformacions bàsiques



# Transformacions bàsiques

