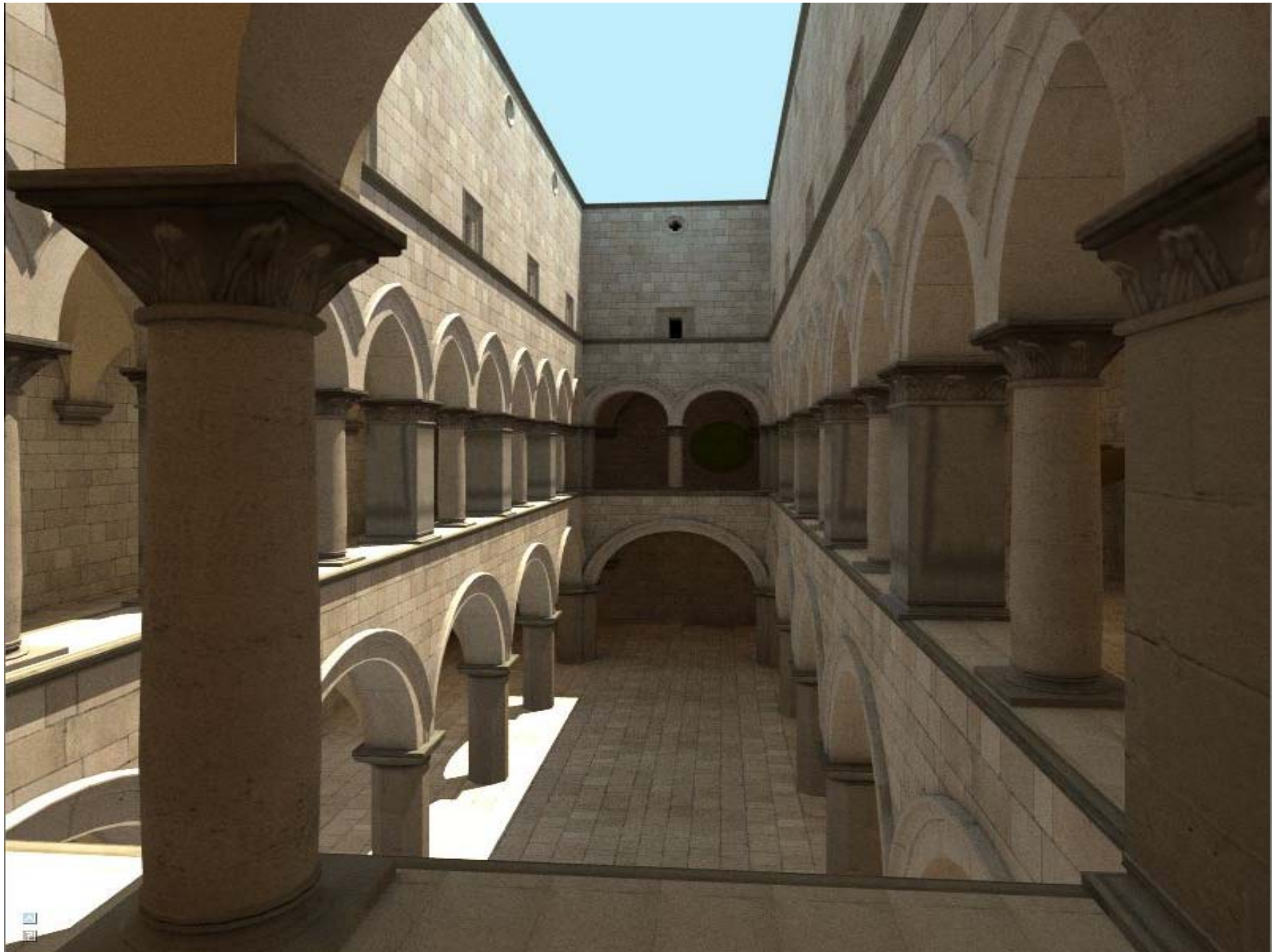


Ray Tracing

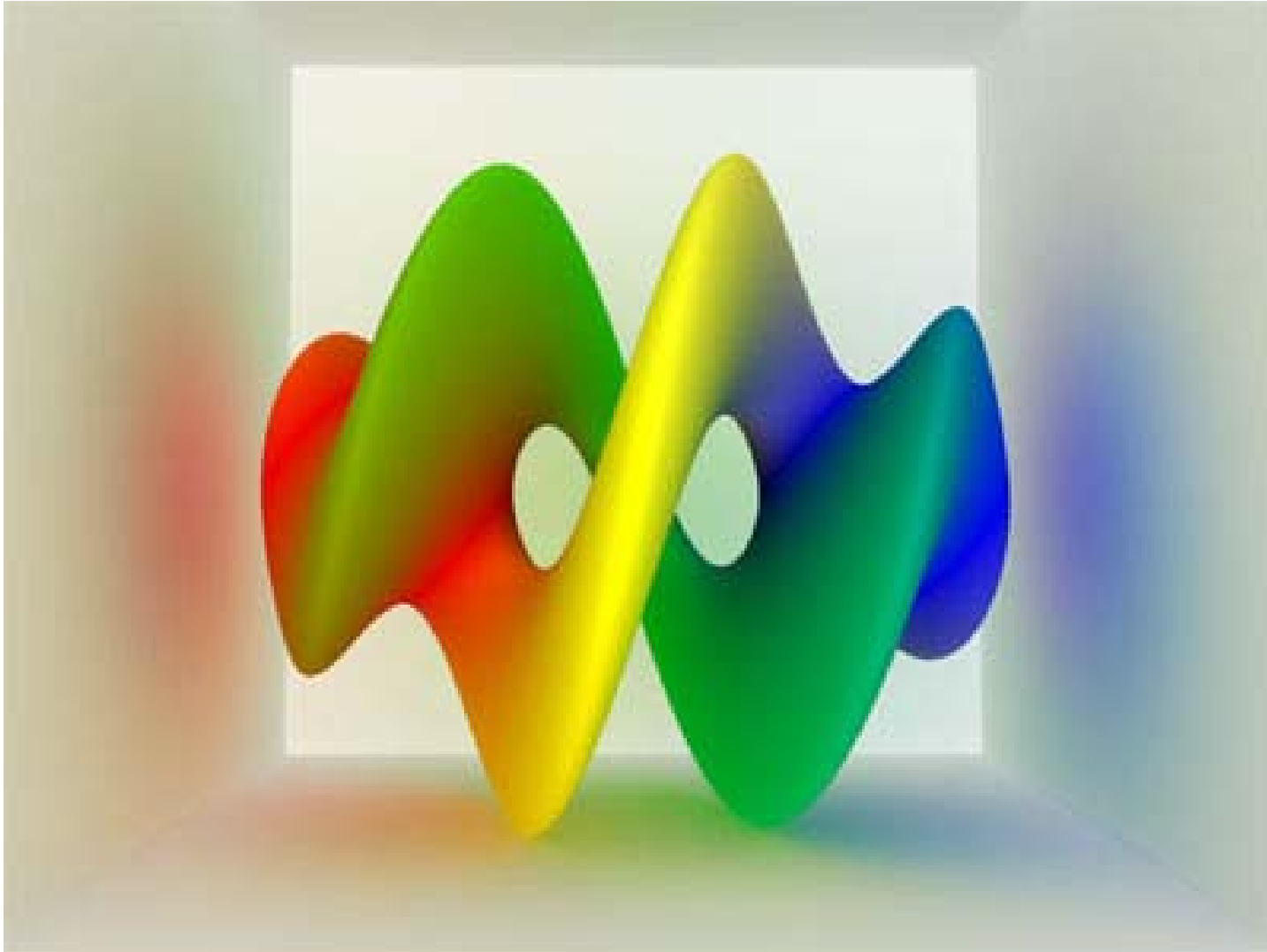
Carlos Andujar

Dec 2016

IL·LUMINACIÓ GLOBAL

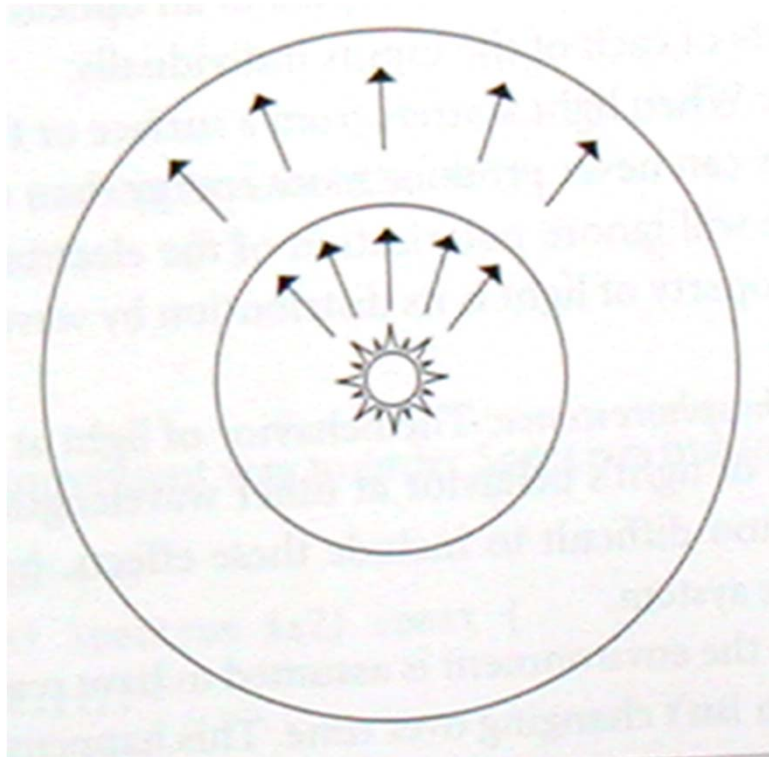


Color bleeding

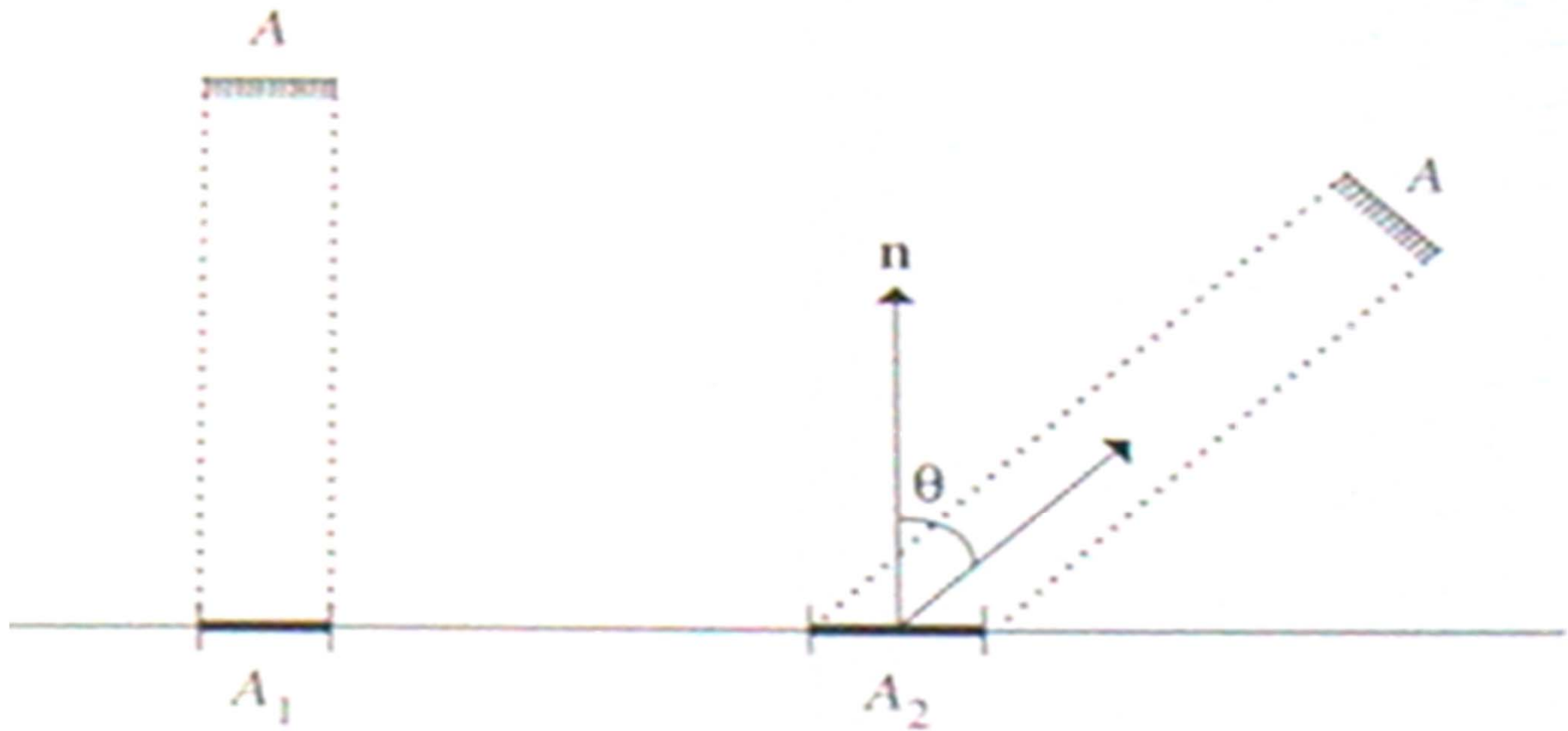


UNITATS RADIOMETRIA

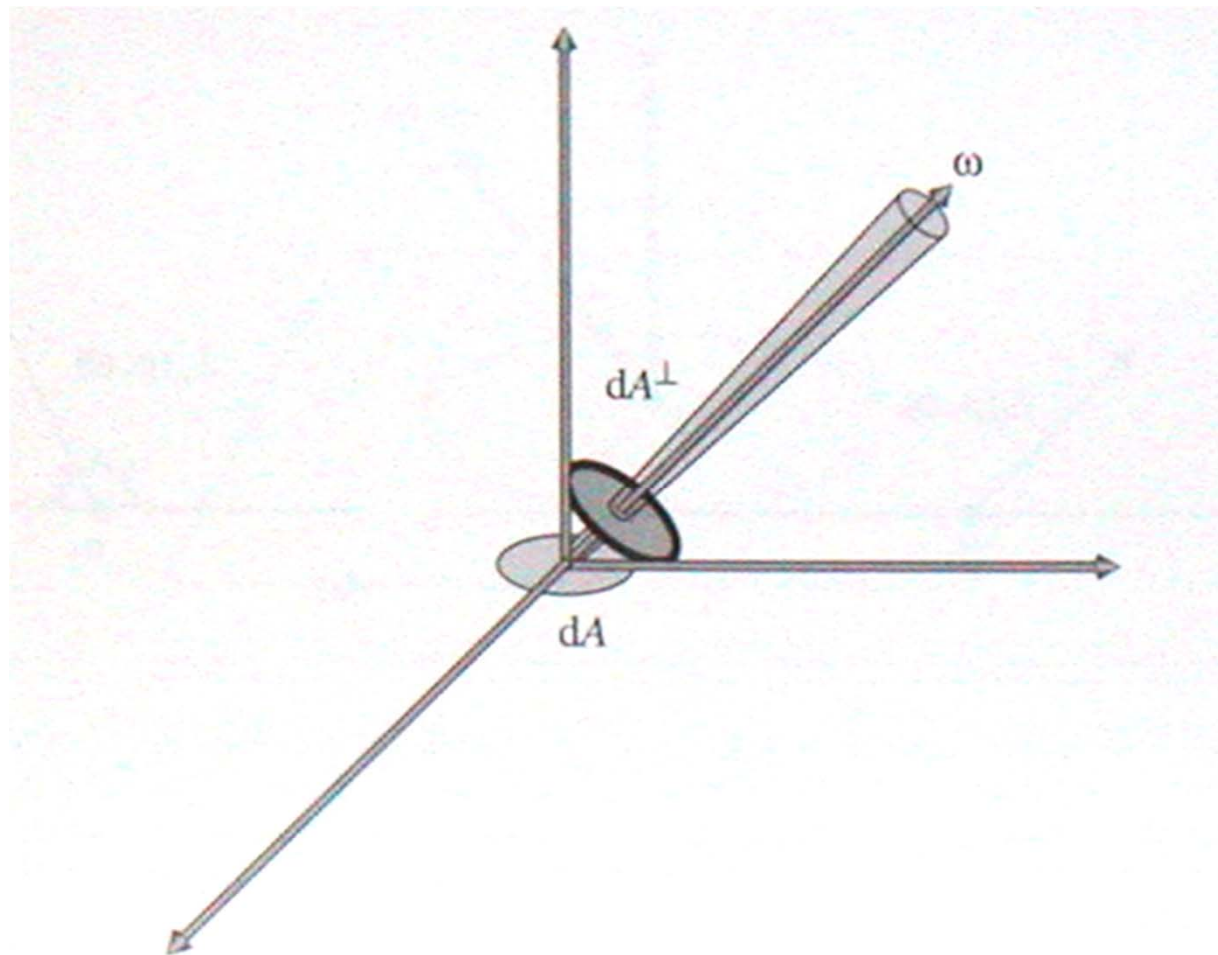
Flux radiant Φ



Irradiància E i Lley de Lambert



Radiància $L(p, \omega)$

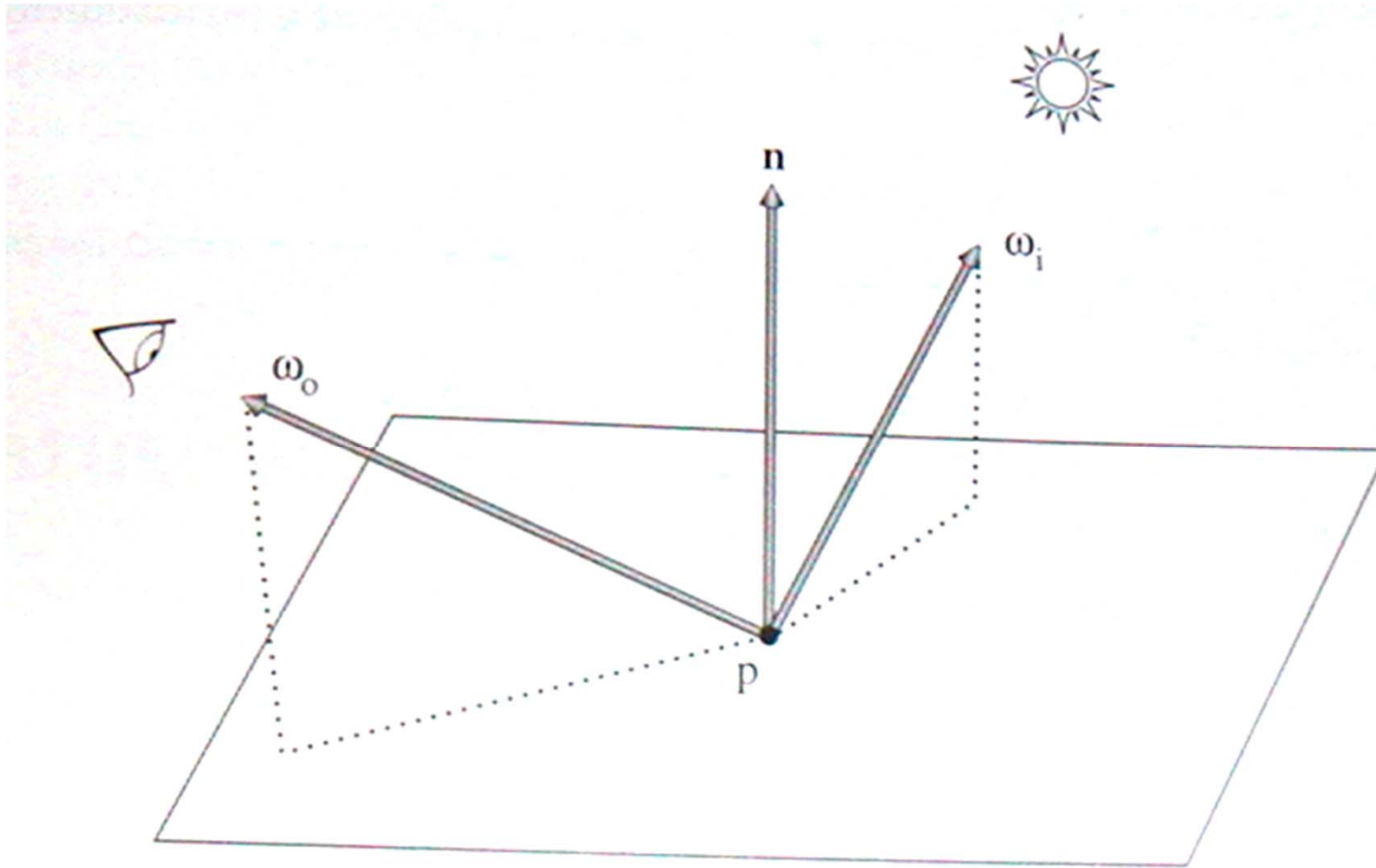


Resum fotometria

Sím.	Radiomet.	Fotometria	Definició	Ús
Φ	Fluxe (W)	Fluxe (lm)	Energia que travessa una superfície per unitat de temps	Energia total que emet una font de llum
E	Irradiancia (W/m ²)	Iluminància (lux=lm/m ²)	Fluxe per unitat d'àrea	Llum que incideix en un punt, des de qualsevol direcció
I	Intensitat (W/sr)	Intensitat (cd=lm/sr)	Fluxe per unitat d'angle sòlid	Distribució direccional d'una llum puntual
L	Radiància W/(sr·m ²)	Luminància (cd/m ²)	Fluxe per unitat d'àrea i unitat d'angle sòlid	Energia que travessa un punt en una determinada direcció

BRDF

$$\text{BRDF } f(p, \omega_o, \omega_i)$$

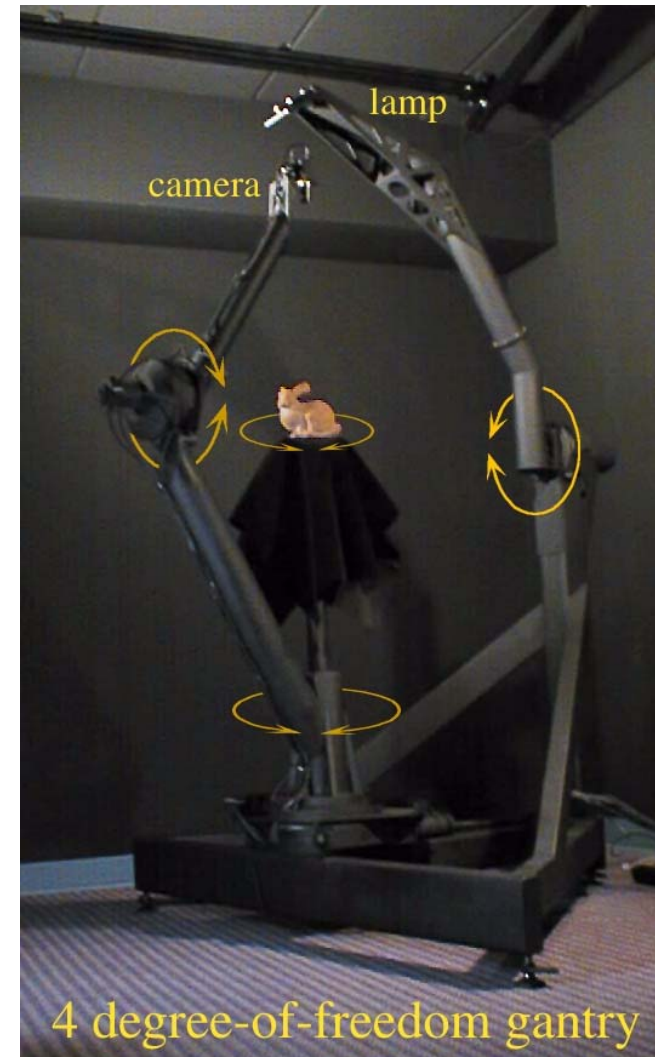
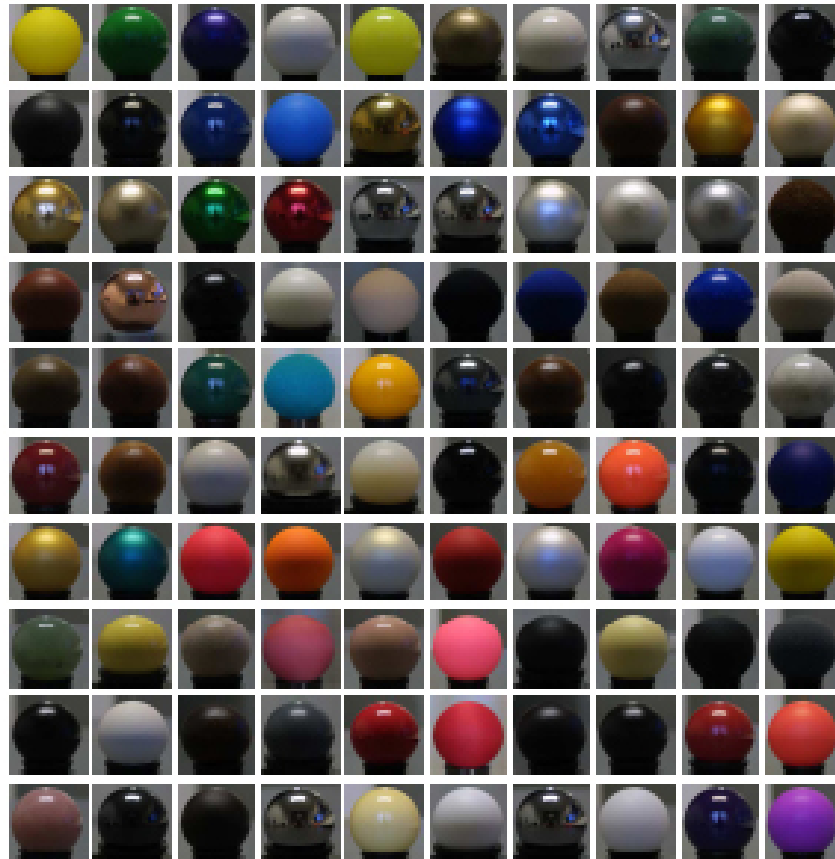


BRDF

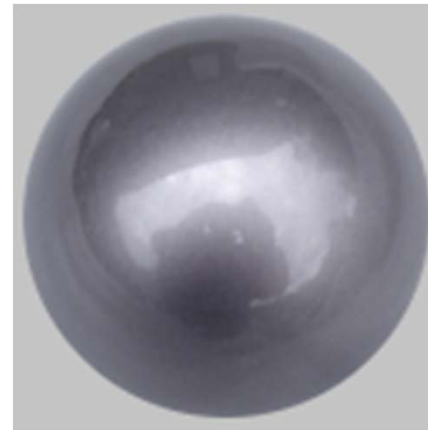
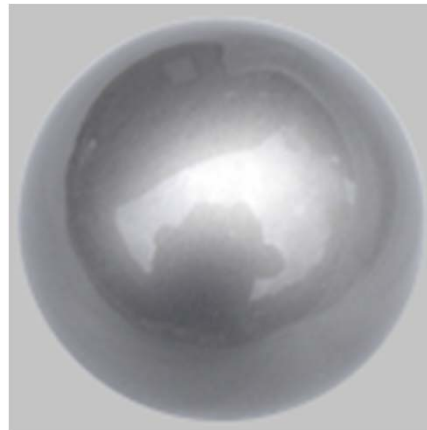
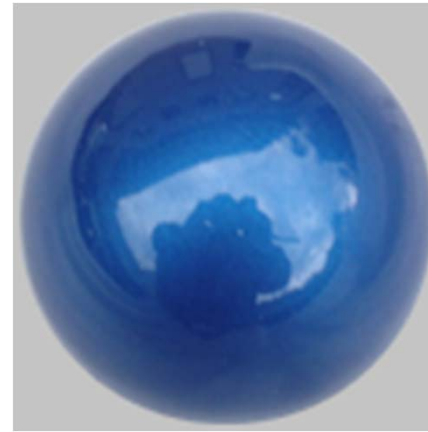


MERL BRDF database

- www.merl.com/brdf/



BRDF de pinturas

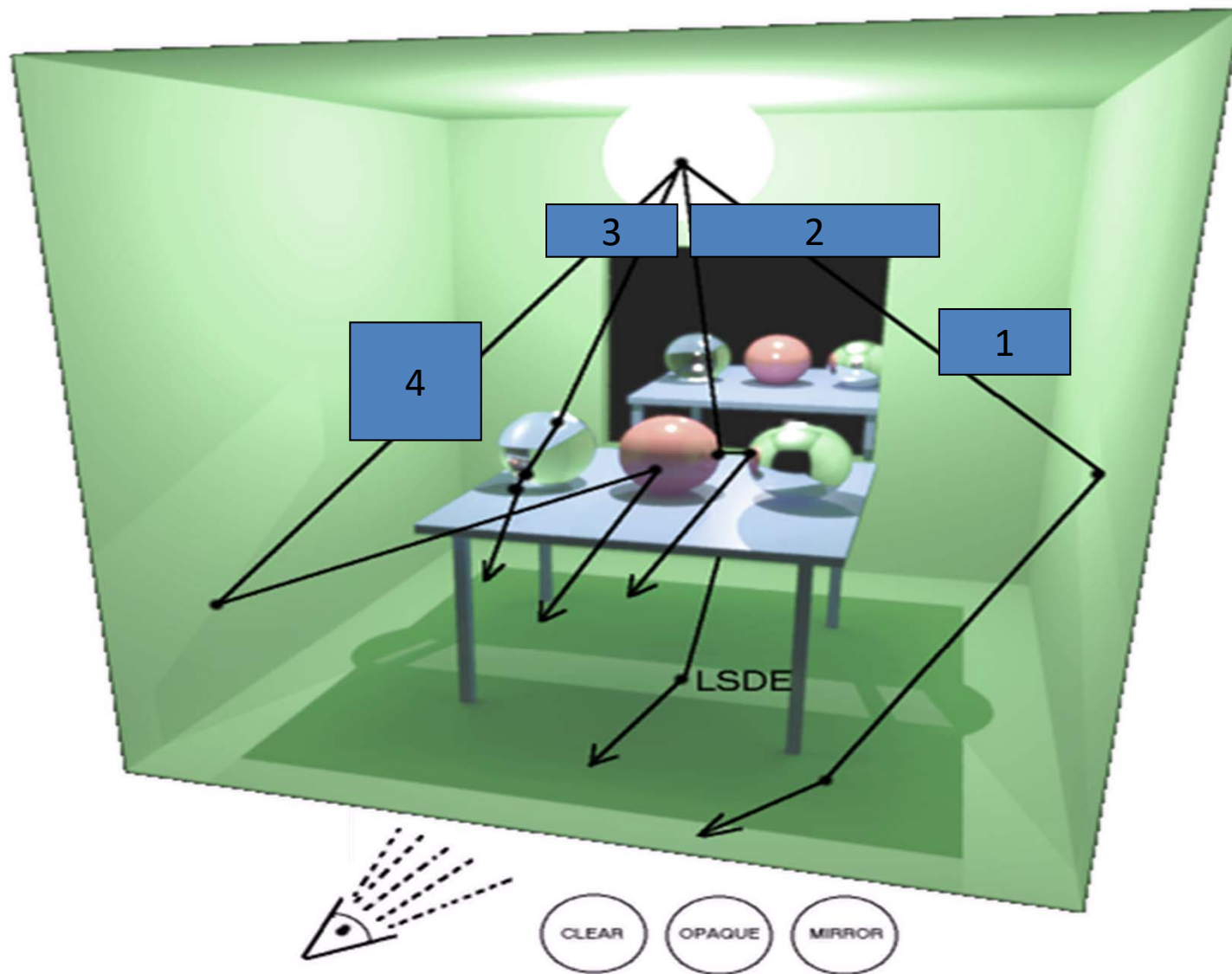


Phong vs BRDF mengukur

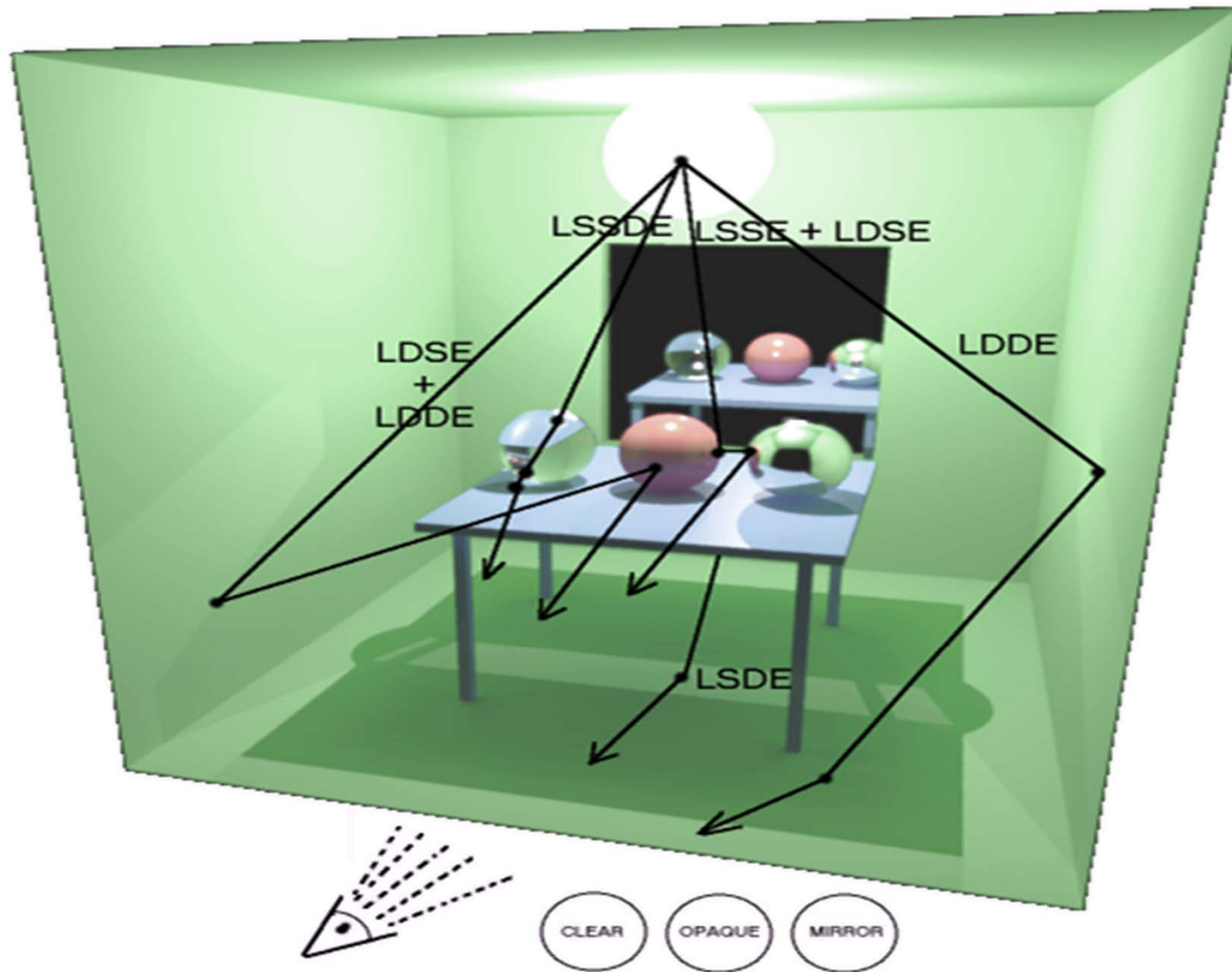


LIGHT PATHS

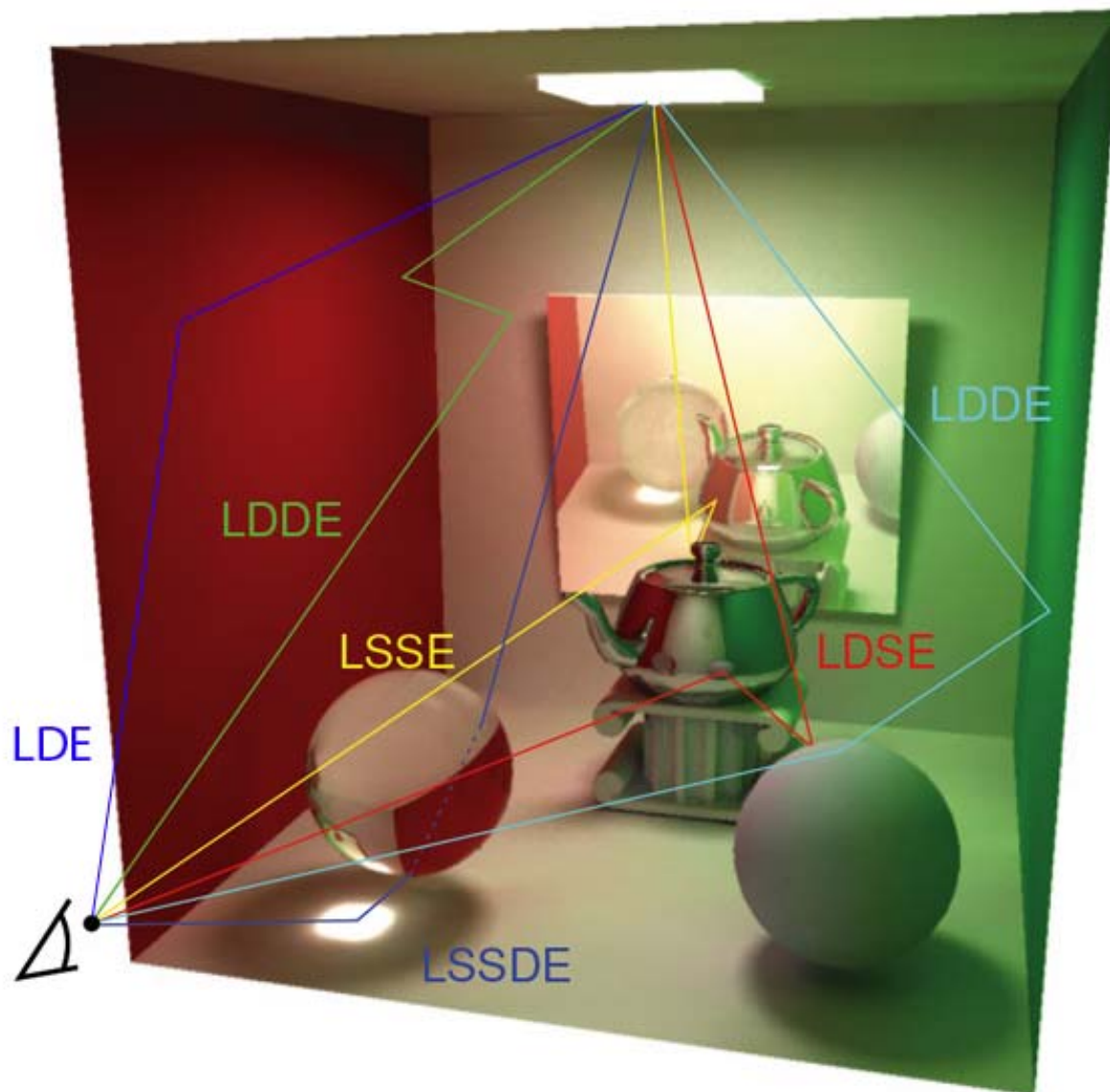
Light paths



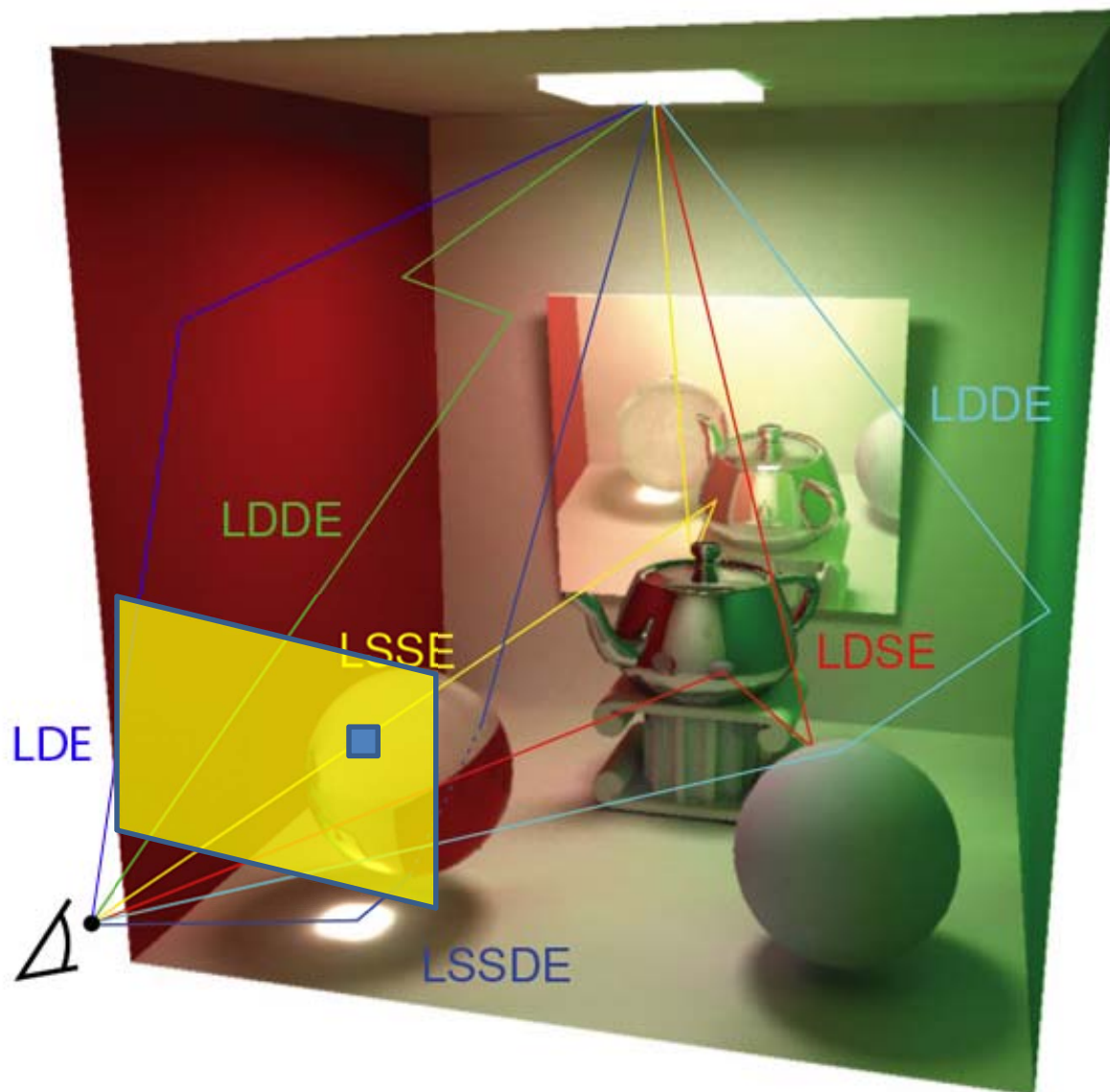
Light paths



Light paths



Light paths



RAY CASTING

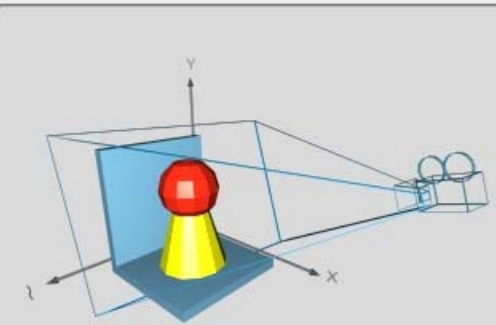
Ray-casting

accio ray_casting

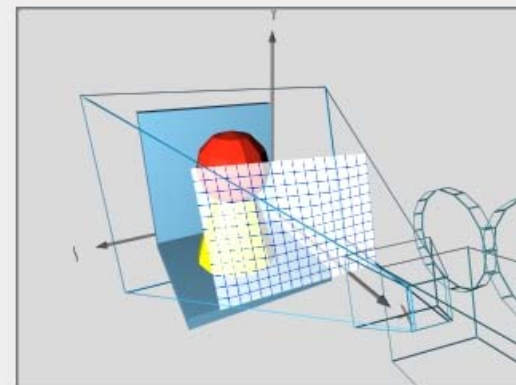
```
per y en [vyM...vym] fer
  per x en [vxm...vxM] fer
    r:=calcul_raig(x,y,Obs)
    per cada cara fer
      trobar_interseccions_raig_cara
    fper
  ordenar_interseccions
fper
fper
faccio
```

Anterior

Següent



1..5 - Presentació escena.



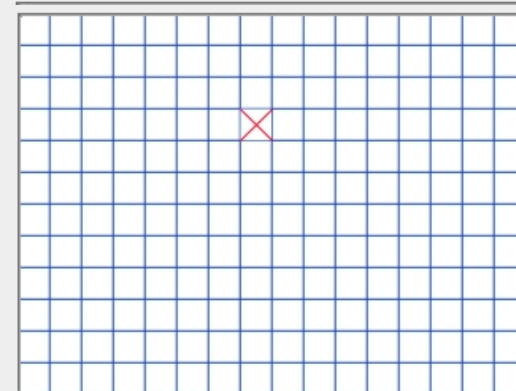
1..5 - Càlcul del window.

Ray-casting

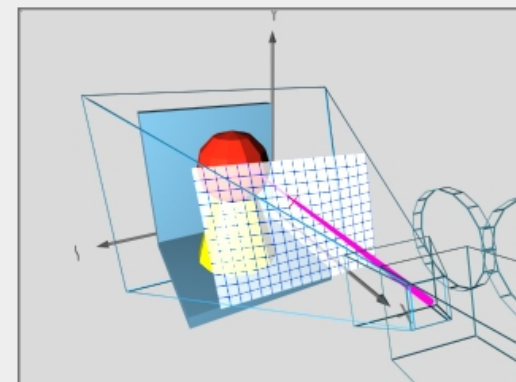
```
accio ray_casting
per y en [vyM...vym] fer
  per x en [vxm...vxdM] fer
    r:=calcul_raig(x,y,Obs)
    per cada cara fer
      trobar_interseccions_raig_cara
    fper
    ordenar_interseccions
  fper
fper
faccio
```

Anterior

Següent



2..5 - Donat un píxel.



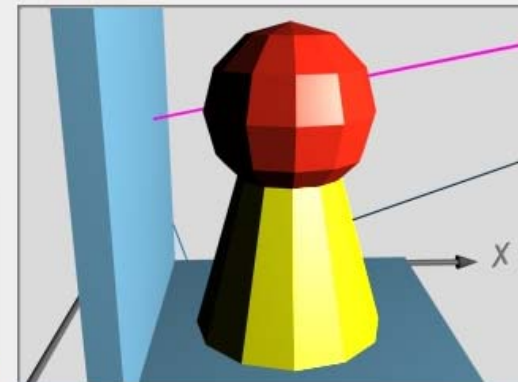
2..5 - Traçar raig per píxel.

Ray-casting

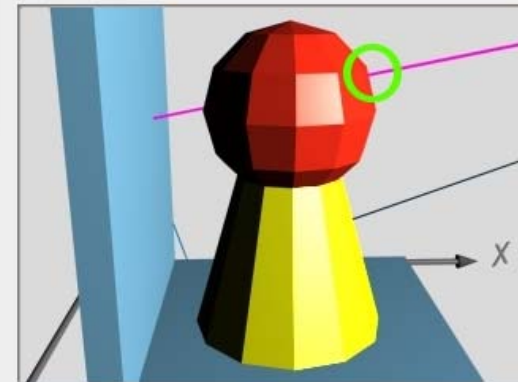
```
accio ray_casting
  per y en [vyM...vym] fer
    per x en [vxm...vxM] fer
      r:=calcul_raig(x,y,Obs)
    per cada cara fer
      trobar_interseccions_raig_cara
    fper
  ordenar_interseccions
fper
fper
faccio
```

Anterior

Següent



3.5 - Càlcul interseccions.



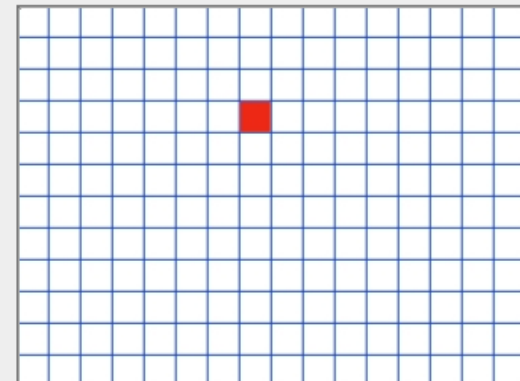
3.5 - Obtenir intersecció més propera.

Ray-casting

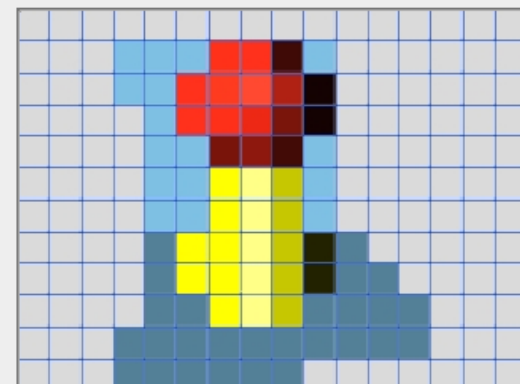
```
accio ray_casting
per y en [vyM...vym] fer
  per x en [vxm...vxdM] fer
    r:=calcul_raig(x,y,Obs)
    per cada cara fer
      trobar_interseccions_raig_cara
    fper
    ordenar_interseccions
  fper
fper
faccio
```

Anterior

Següent



4.5 - Pintar píxel del color de la intersecció.



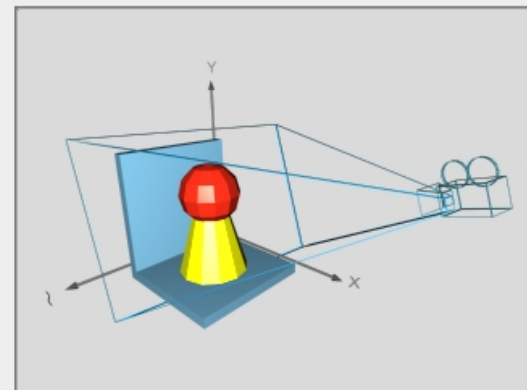
4.5 - Resultat en baixa resolució.

Ray-casting

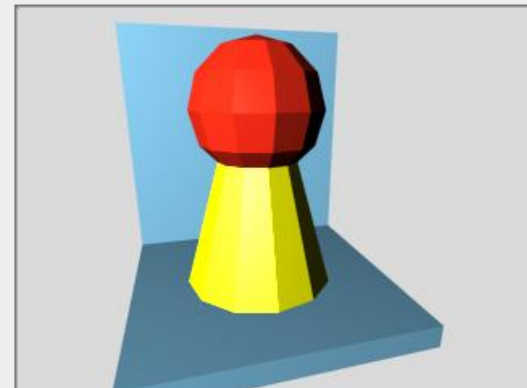
```
accio ray_casting
per y en [vyM...vym] fer
  per x en [vxm...vxdM] fer
    r:=calcul_raig(x,y,Obs)
    per cada cara fer
      trobar_interseccions_raig_cara
    fper
  ordenar_interseccions
fper
fper
faccio
```

Anterior

Següent



5..5 - Presentació escena.



5..5 - Visió final en resolució estàndard.

RAY TRACING

Ray-tracing clàssic



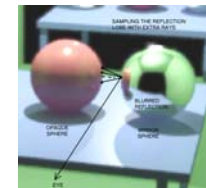
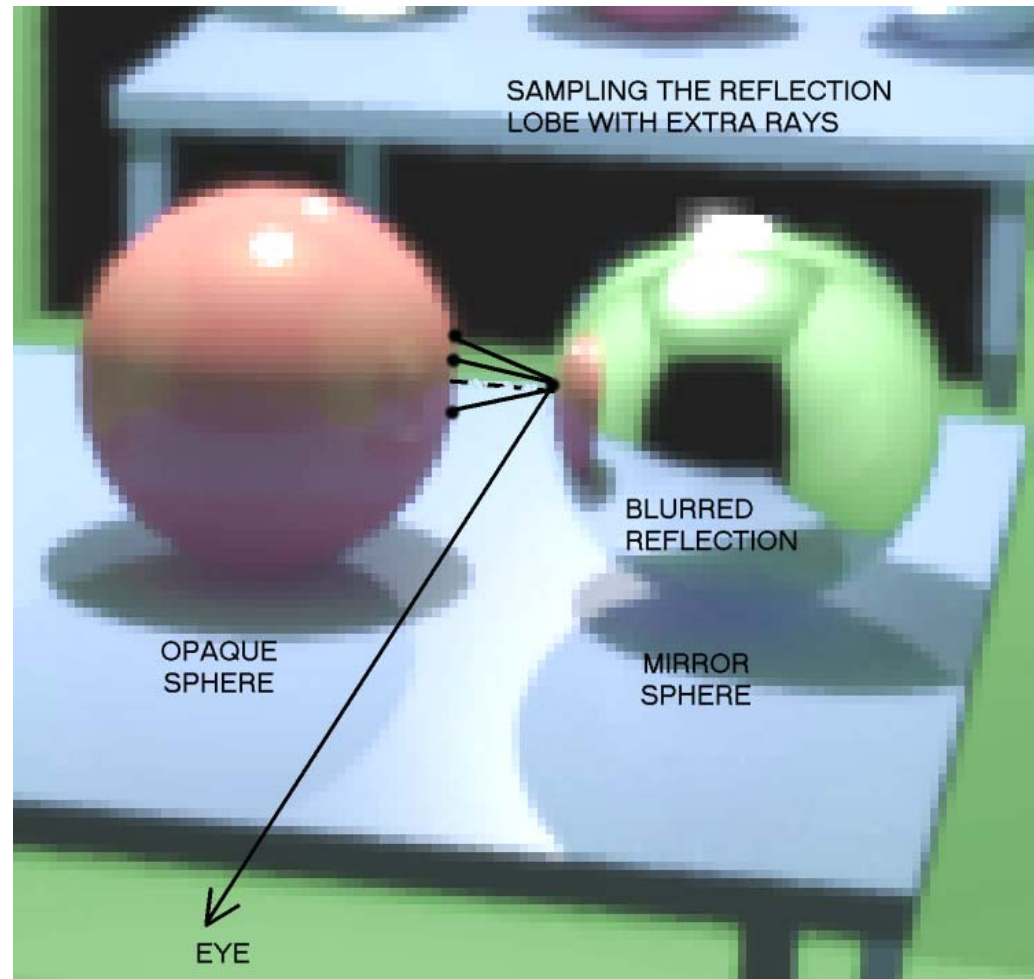
Observeu:

- Ombres ben definides (hard shadows)
- Refracció massa definida al gerro (sembla que tingui una banda negra en comptes de refractar l'ombra que té al darrera)

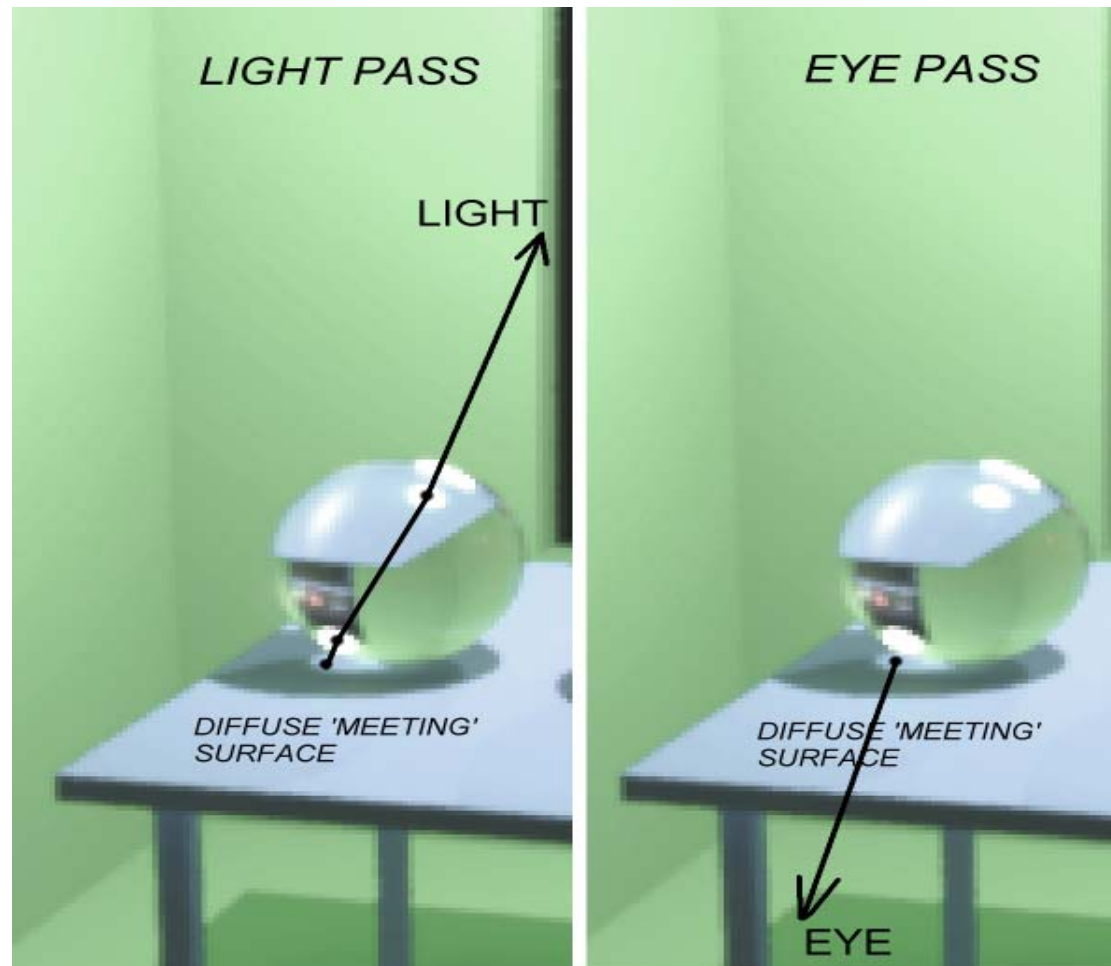
Path tracing



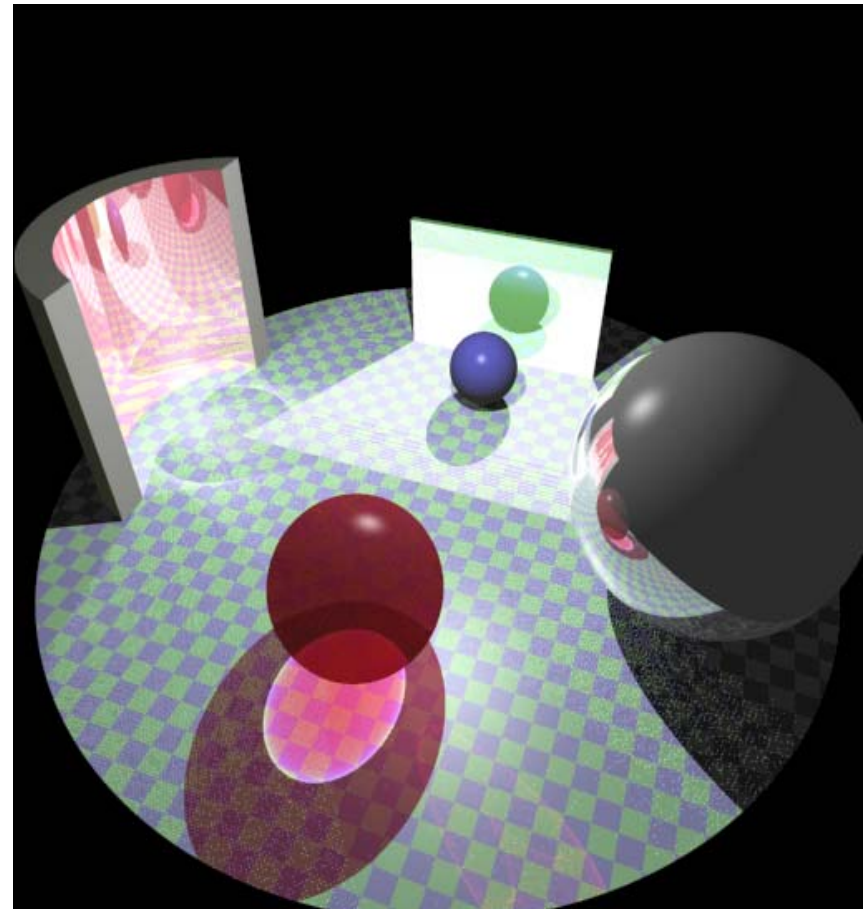
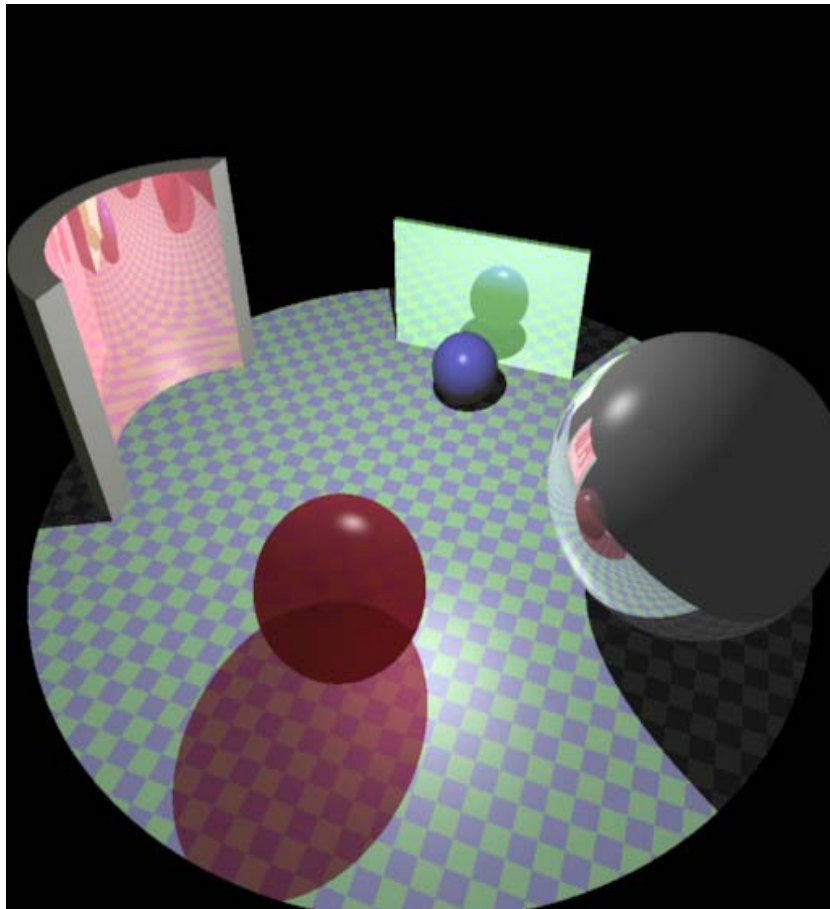
Distributed ray-tracing



Two-pass ray-tracing



Classic vs Two-pass ray-tracing



RAY TRACING CLÀSSIC

Color d'un punt P

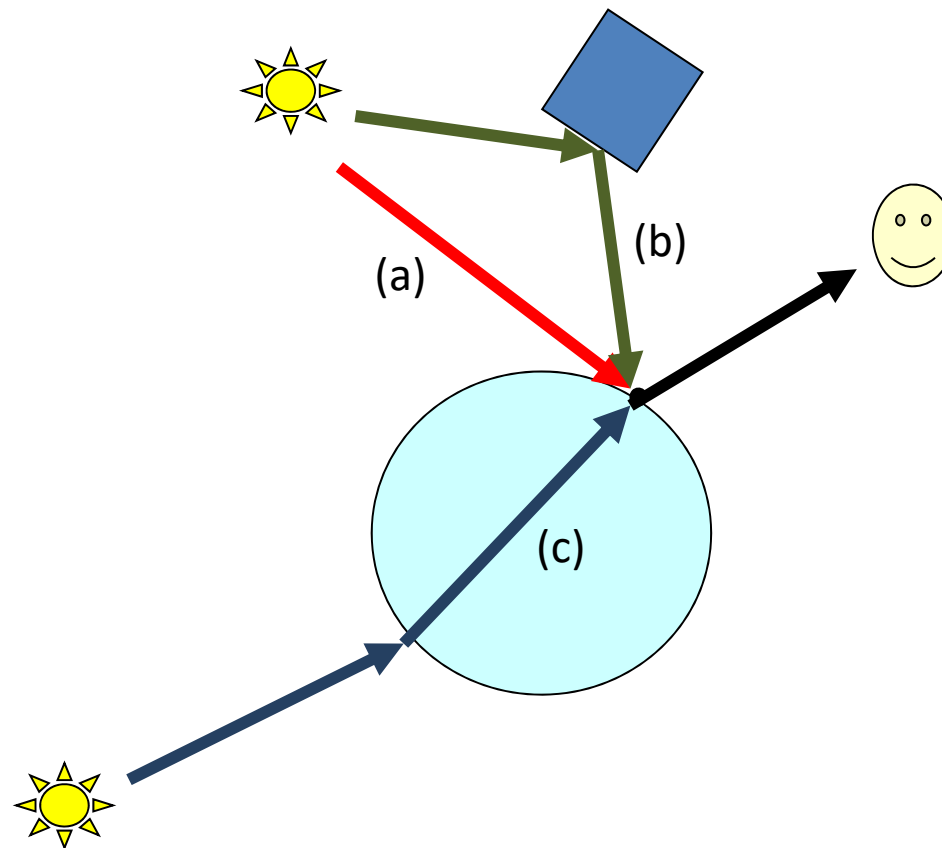
- El color amb que un determinat observador veu un cert punt P es calcula:

$$I(P) = I_D(P) + I_R(P) + I_T(P)$$

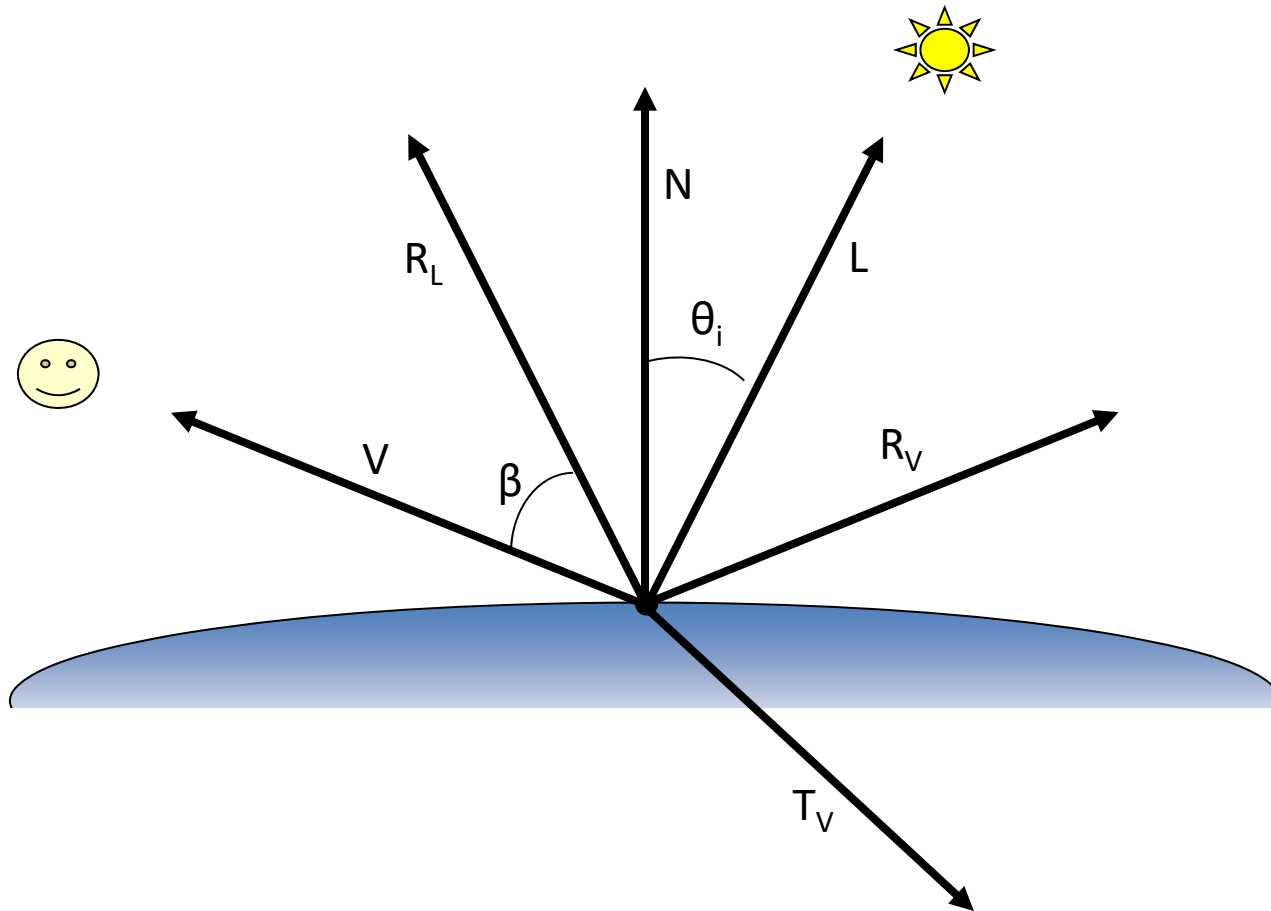
- $I_D(P)$ és color degut a la **llum directa** dels focus.
- $I_R(P)$ és color degut a la **llum indirecta que es reflecteix** a P en direcció cap a l'observador.
- $I_T(P)$ és color degut a la **llum indirecta que es transmet** des de P en direcció cap a l'observador.

Color d'un punt P

$$I(P) = I_D(P) + I_R(P) + I_T(P)$$



Color d'un punt P: notació

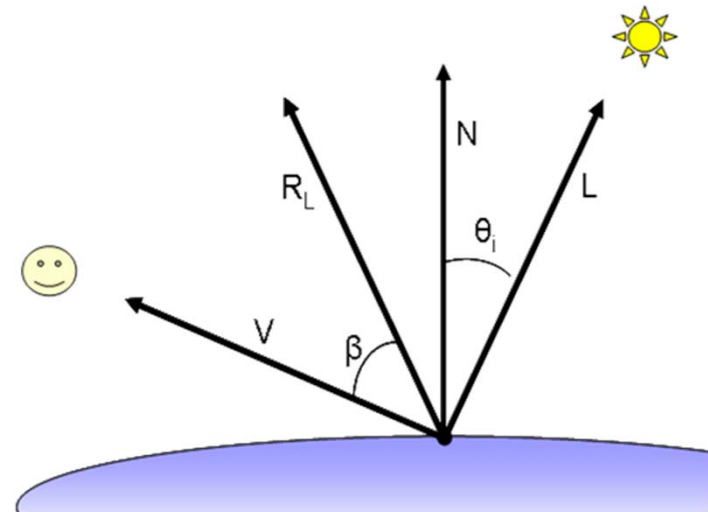


Contribució llum directa I_D

$$I(P) = I_D(P) + I_R(P) + I_T(P)$$

$$I_D(P) = K_a I_a + K_d \sum I_L \cos(\theta_i) + K_s \sum I_L \cos^n(\beta)$$

- $\cos(\theta_i) = N \cdot L$
- $\cos(\beta) = R_L \cdot V$
- El sumatori només considera les fonts de llum no ocluides (ombres)

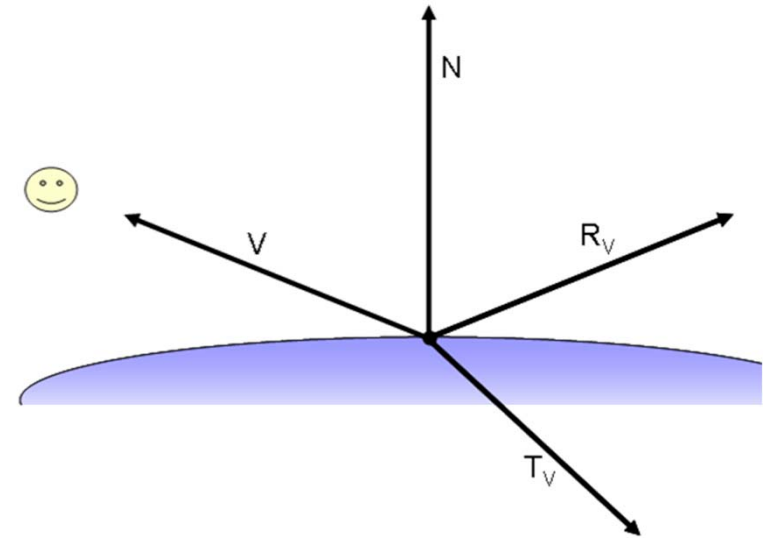


Contribución llum indirecta I_R, I_T

$$I(P) = I_D(P) + I_R(P) + I_T(P)$$

$$I_R(P) = K_R L_R$$

$$I_T(P) = K_T L_T$$



- K_R, K_T coeficients empírics de reflexió/transmissió especular
- L_R = llum que incideix en P en la direcció R_v
- L_T = llum que incideix en P en la direcció T_v

Es calculen recursivament, traçant un nou raig reflectit i un altre transmès

Algorisme

```
acció rayTracing
  per i en [0..w-1] fer
    per j en [0..h-1] fer
      raig:=raigPrimari(i, j, camera);
      color:=traçarRaig(raig, escena,  $\mu$ );
      setPixel(i, j, color);
    fper
  fper
faccio
```

Algorisme

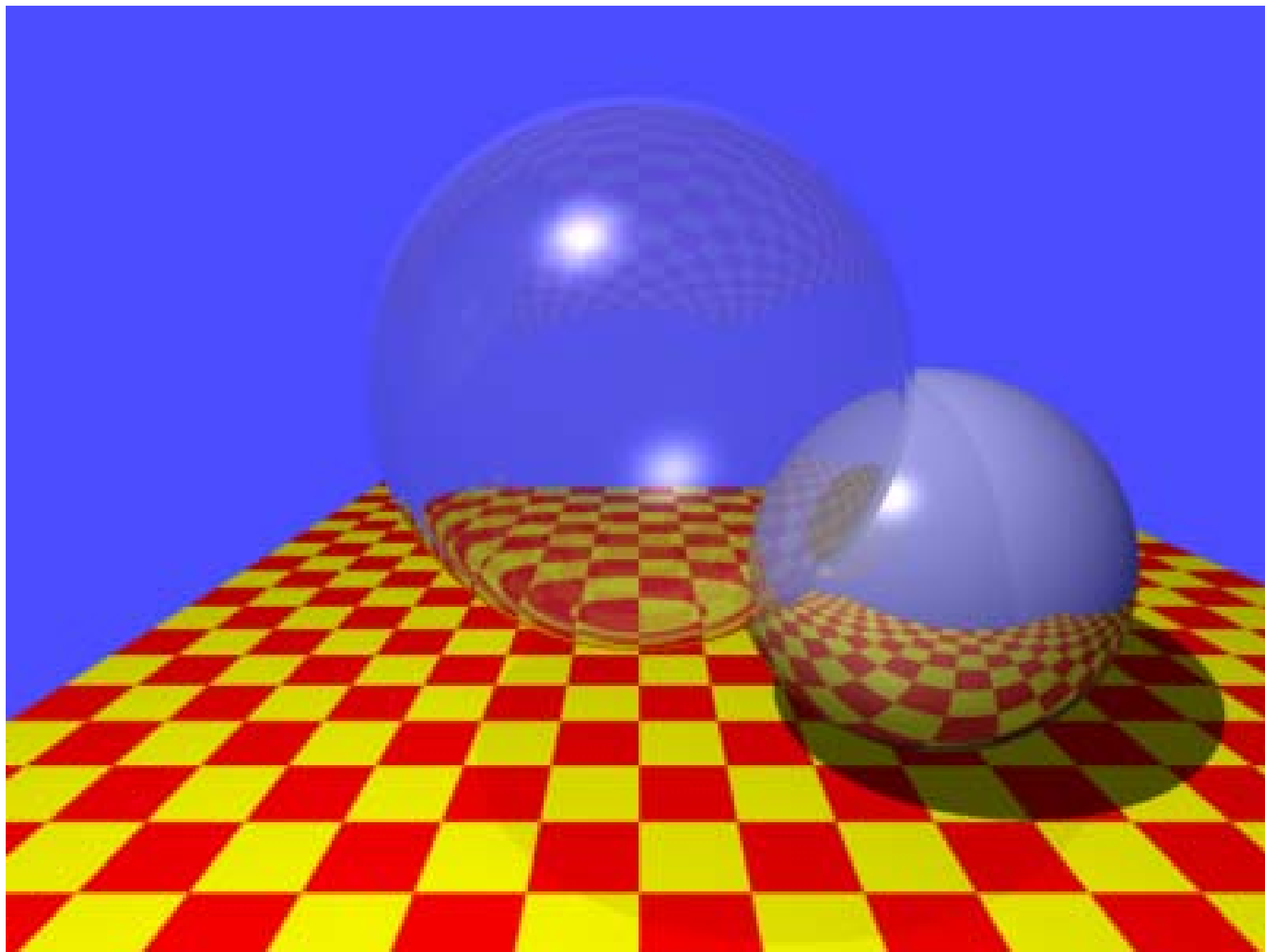
```
funció traçar_raig(raig, escena,  $\mu$ )
  si profunditat_correcta() llavors
    info:=calcula_interseccio(raig, escena)
    si info.hi_ha_interseccio() llavors
      color:=calcular_ID(info,escena); // ID
      si es_reflector(info.obj) llavors
        raigR:=calcula_raig_reflectit(info, raig)
        color+= KR*traçar_raig(raigR, escena,  $\mu$ ) //IR
      fsi
      si es_transparent(info.obj) llavors
        raigT:=calcula_raig_transmès(info, raig,  $\mu$ )
        color+= KT*traçar_raig(raigT, escena, info. $\mu$ ) //IT
      fsi
    sino color:=colorDeFons
  fsi
  sino color:=Color(0,0,0); // o colorDeFons
  fsi
  retorna color
ffunció
```

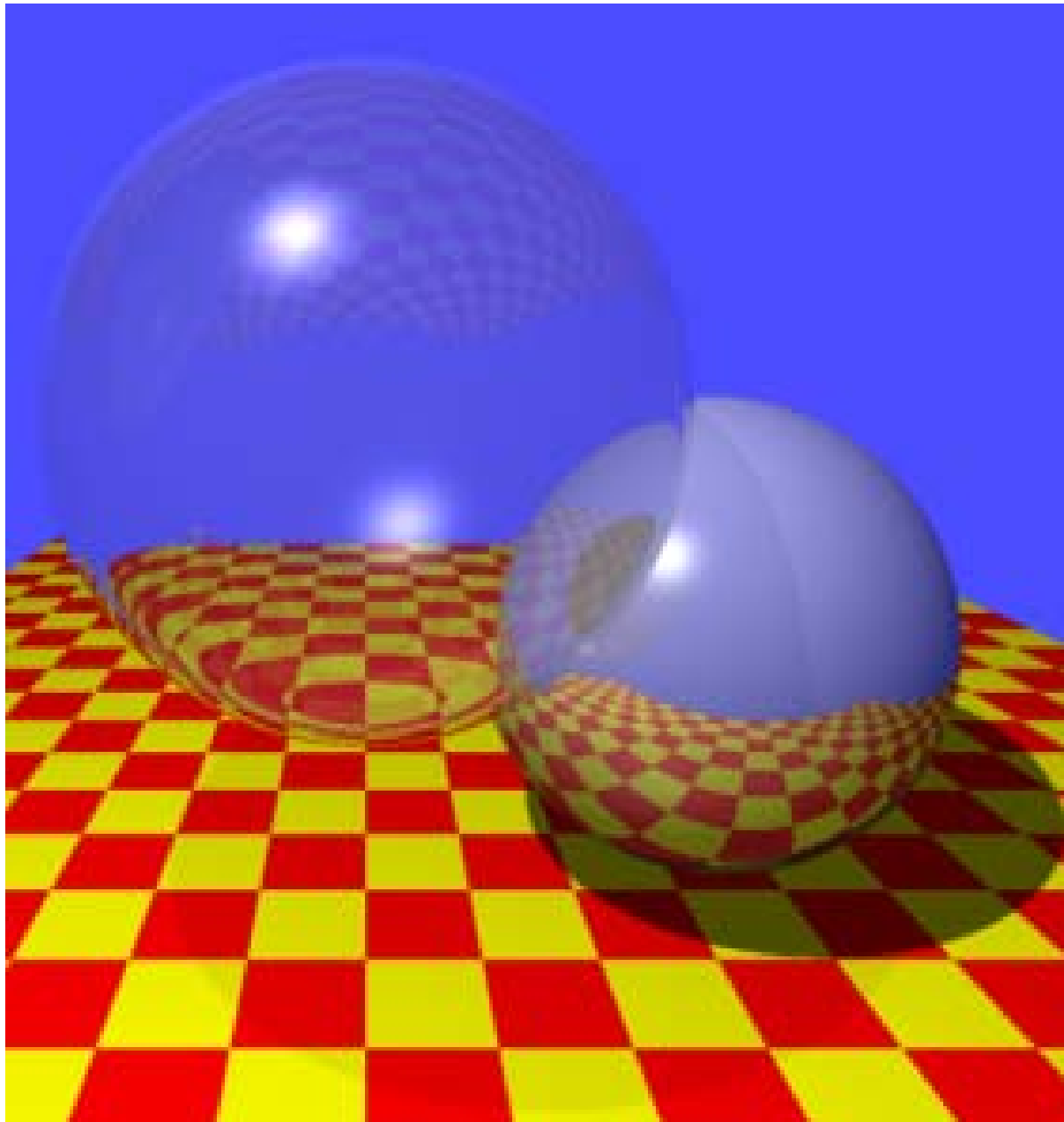

Shader (FS)

```
void main() {  
    vec3 obs = gl_ModelViewMatrixInverse[3].xyz;  
    vec3 dir = normalize(pos - obs);  
    Ray ray = Ray(obs,dir);  
    gl_FragColor = trace(ray, 1.0);  
}
```

Shader (FS)

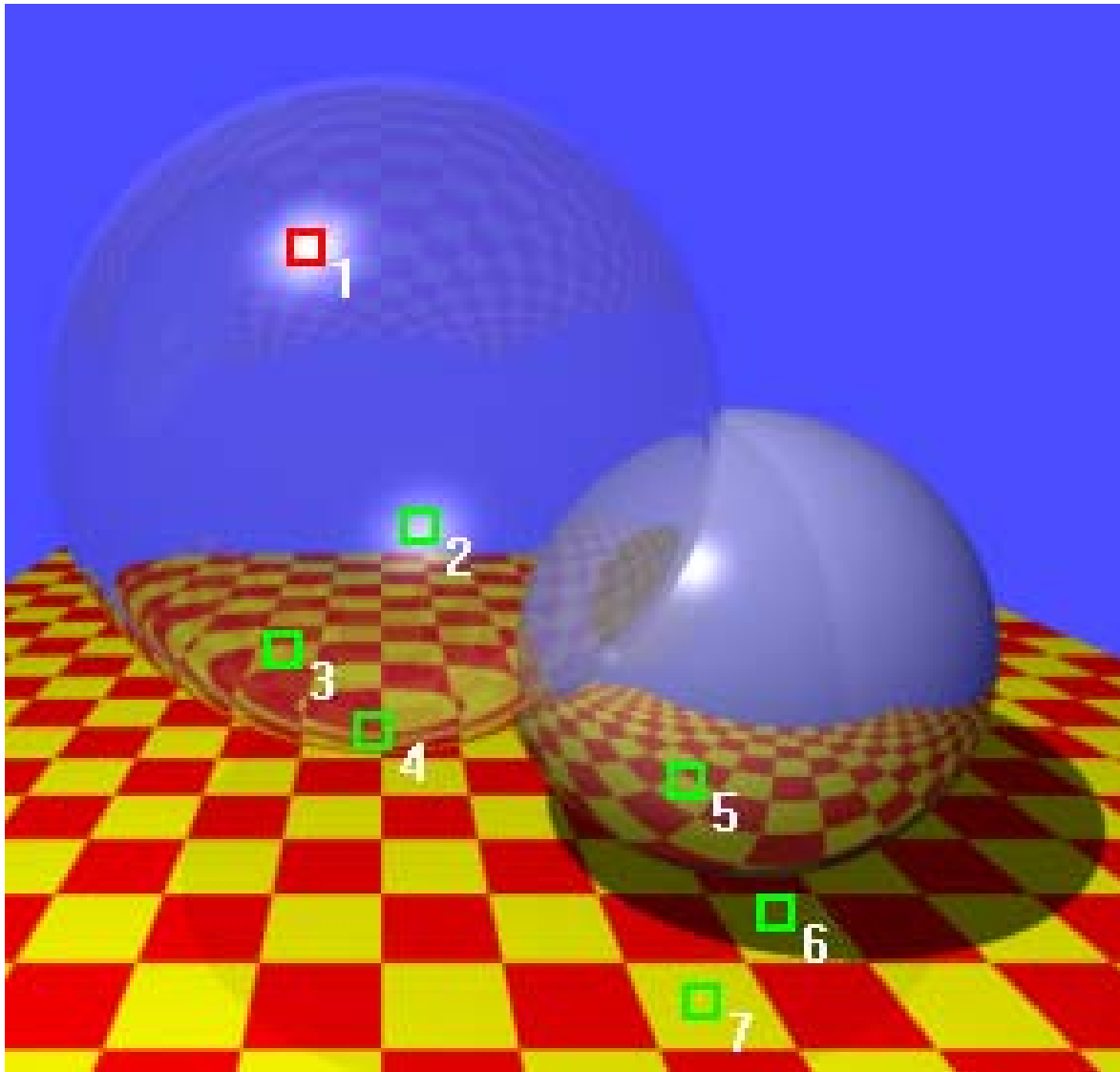
```
vec4 trace(Ray ray, float mu) {  
    if ( intersectScene(ray, Phit, Nhit,Kd, Kr, Kt) ) {  
        Ray shadowRay = Ray(Phit, normalize(lightPos-Phit));  
        bool inShadow = intersectScene(shadowRay, aux, aux,aux4, aux4, aux4);  
        if (inShadow) shadow = 0.2; else shadow = 1.0;  
        color+= shadow*light(Nhit, -ray.dir, lightPos-Phit, Kd, vec4(1.0));  
        vec3 R = reflect(ray.dir, Nhit);  
        color += Kr*trace1(Ray(Phit, R), mu);  
        if (mu==1.0) muHit=1.5;  
        else { muHit = 1.0; Nhit*=-1.0;}  
        vec3 T = refract(ray.dir, Nhit, mu/muHit);  
        if (length(T)>0.0) color+=Kt*trace1(Ray(Phit, T), muHit);  
    }  
    else color+=samplePanorama(ray.dir);  
    return color;  
}
```





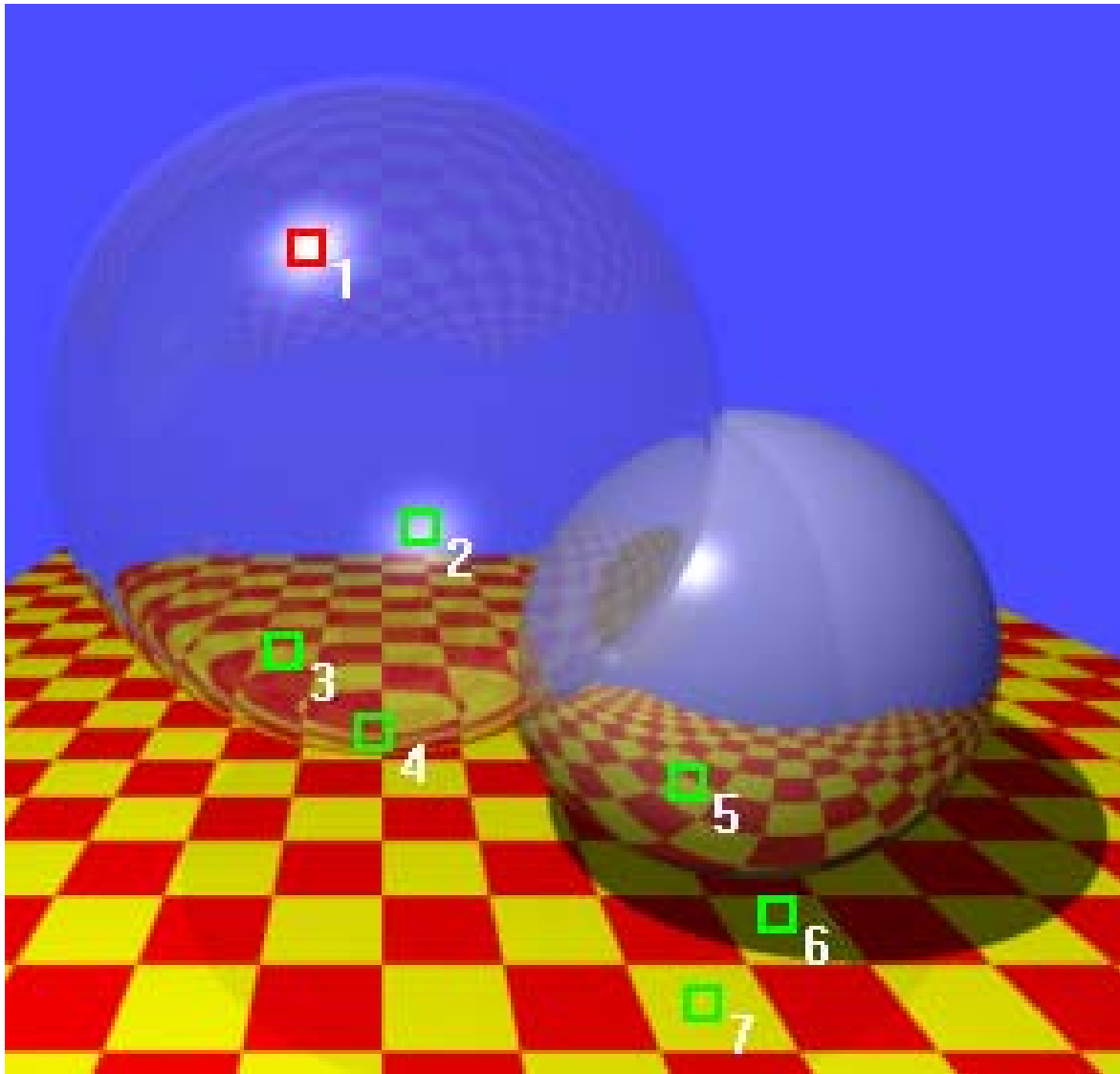
Escena:

- Esfera **buida** transparent.
- Esfera opaca blanca mirall.
- Tauler escacs
- Fons blau
- Llum blanca



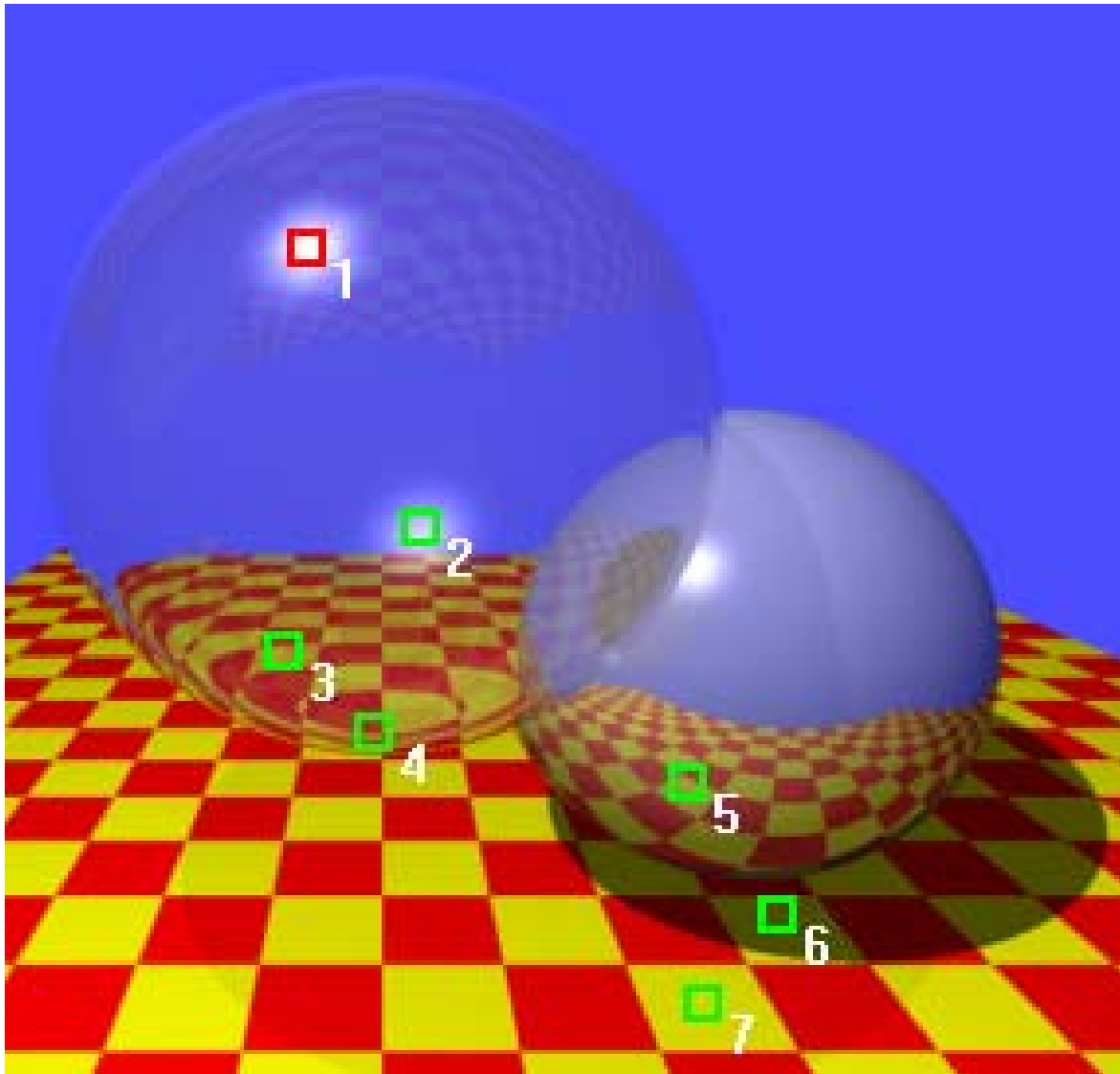
Raig 1

- El raig primari gairebé coincideix amb la reflexió del vector L : podem veure una taca especular (specular highlight).
- La contribució principal és la component especular de $I_D(P)$.



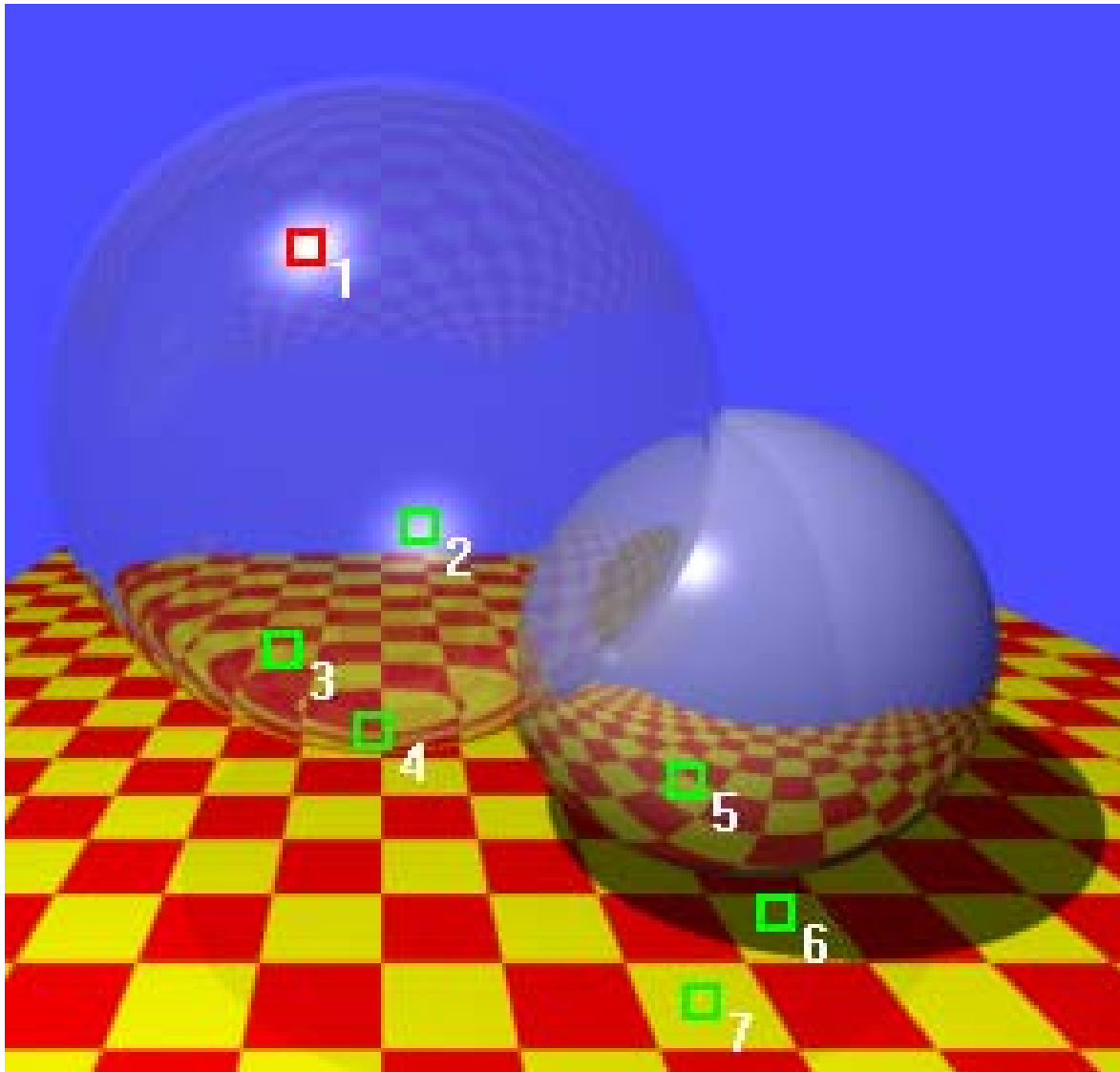
Raig 2

- És la mateixa situació que el raig 1, però ara la taca especular apareix a la paret interna de l'esfera (que és buida).
- Aquest raig mostra un error acceptat en ray-tracing: el raig de la font de llum travessa l'esfera sense refracció (només comparem L , N ignorant el fet que som a dins de l'esfera). Per tant, la taca especular està en una posició errònia, però no tenim cap intuïció de la posició correcta.



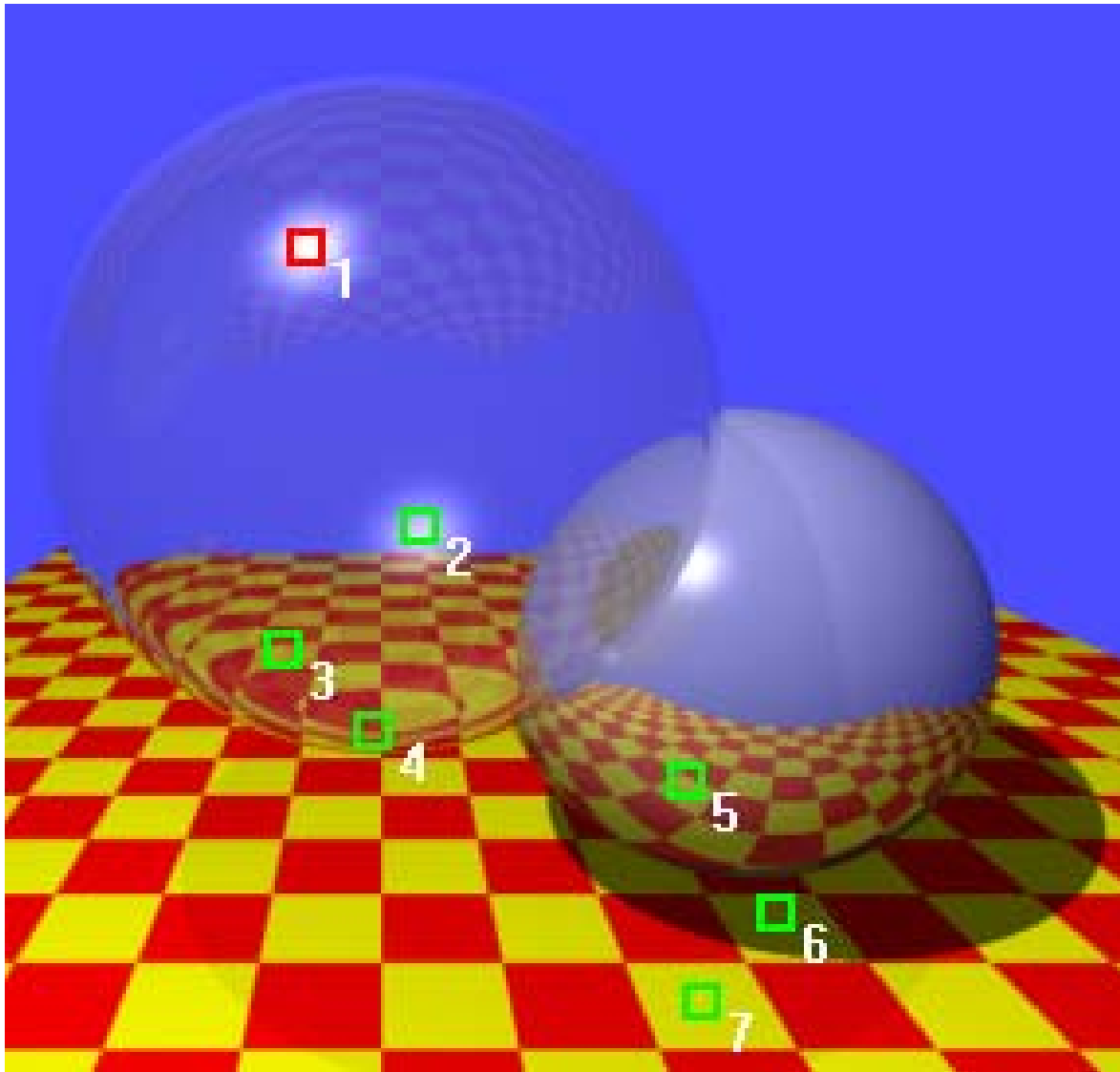
Raig 3

- En totes les interseccions d'aquest raig amb l'esfera la contribució local és nula.
- La contribució dominant és el tauler vermell-groc.
- Hi ha una lleugera distorsió deguda a la refracció.
- Hi ha una barreja de dos taulers: el transmès i el reflectit a la superfície exterior de l'esfera.



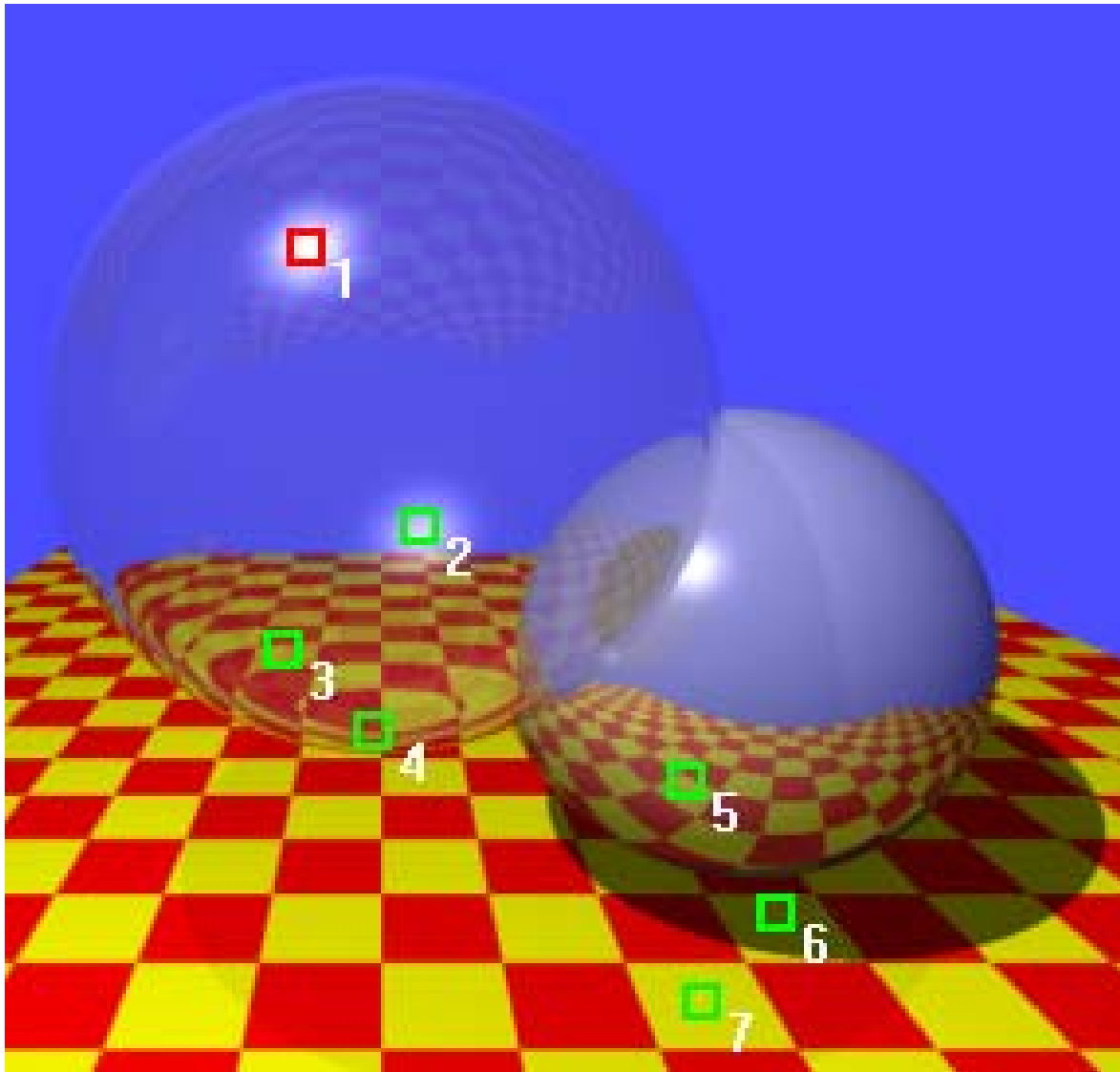
Raig 4

- Igual que el raig anterior, però ara la distància recorreguda dins el cristall és més significativa, amb la qual cosa l'efecte de la refracció és més notori.



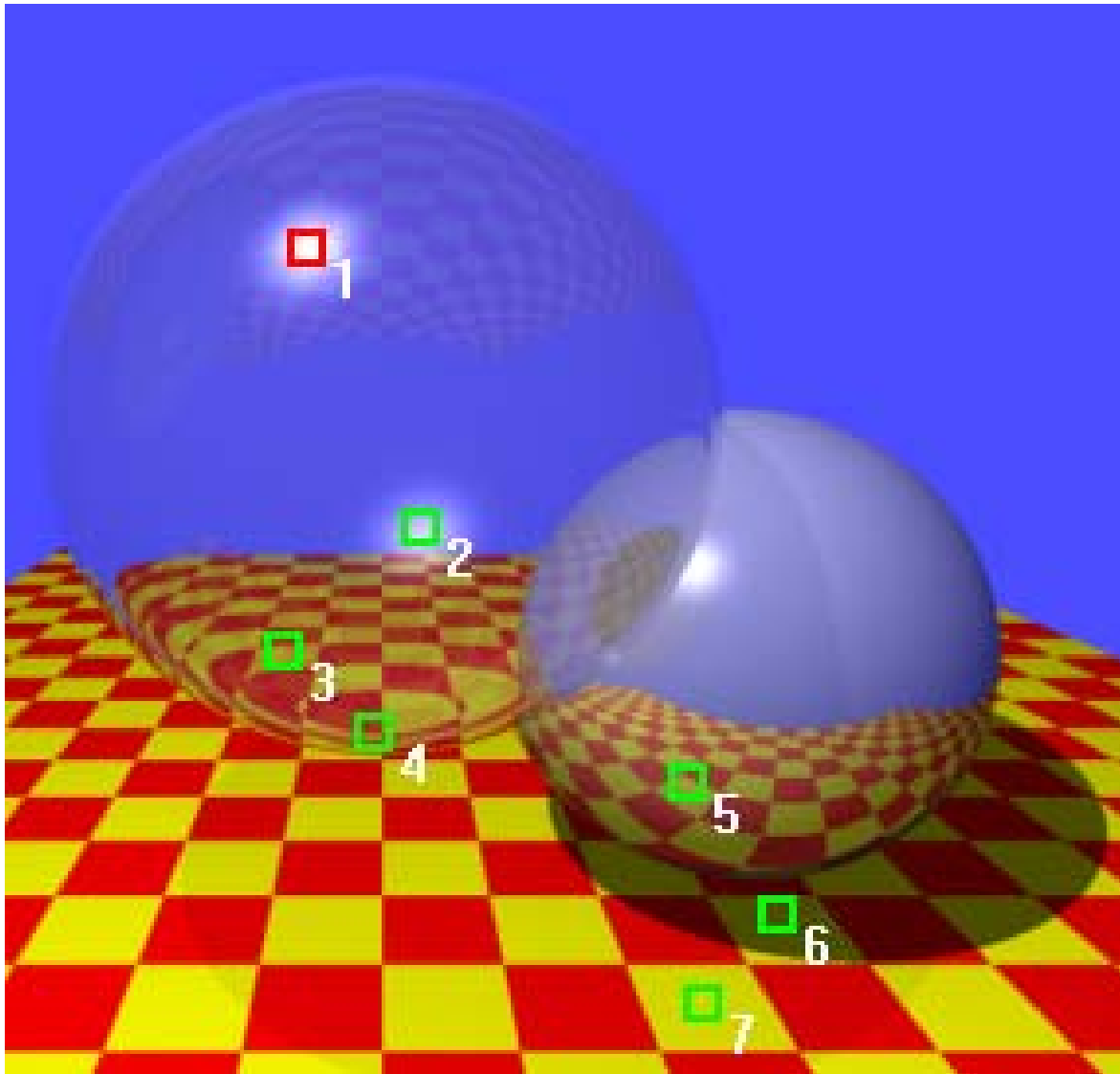
Raig 5

- El color resultant és la barreja del tauler (raig reflectit) amb el blanc de l'esfera (lleugerament difosa).



Raig 6

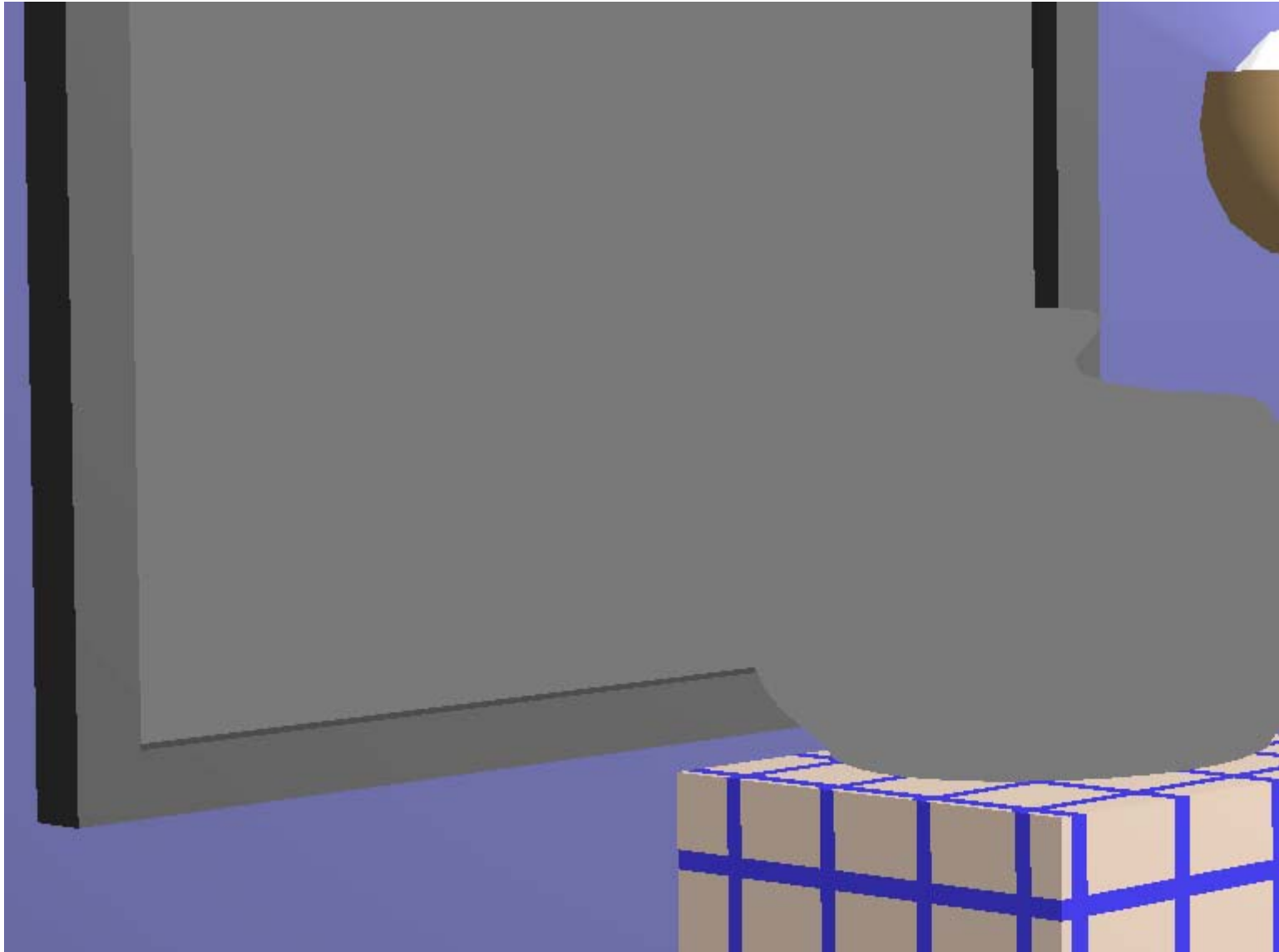
- El raig intersecta el tauler; el punt d'intersecció està a l'ombra (el shadow ray intersecta l'esfera opaca).



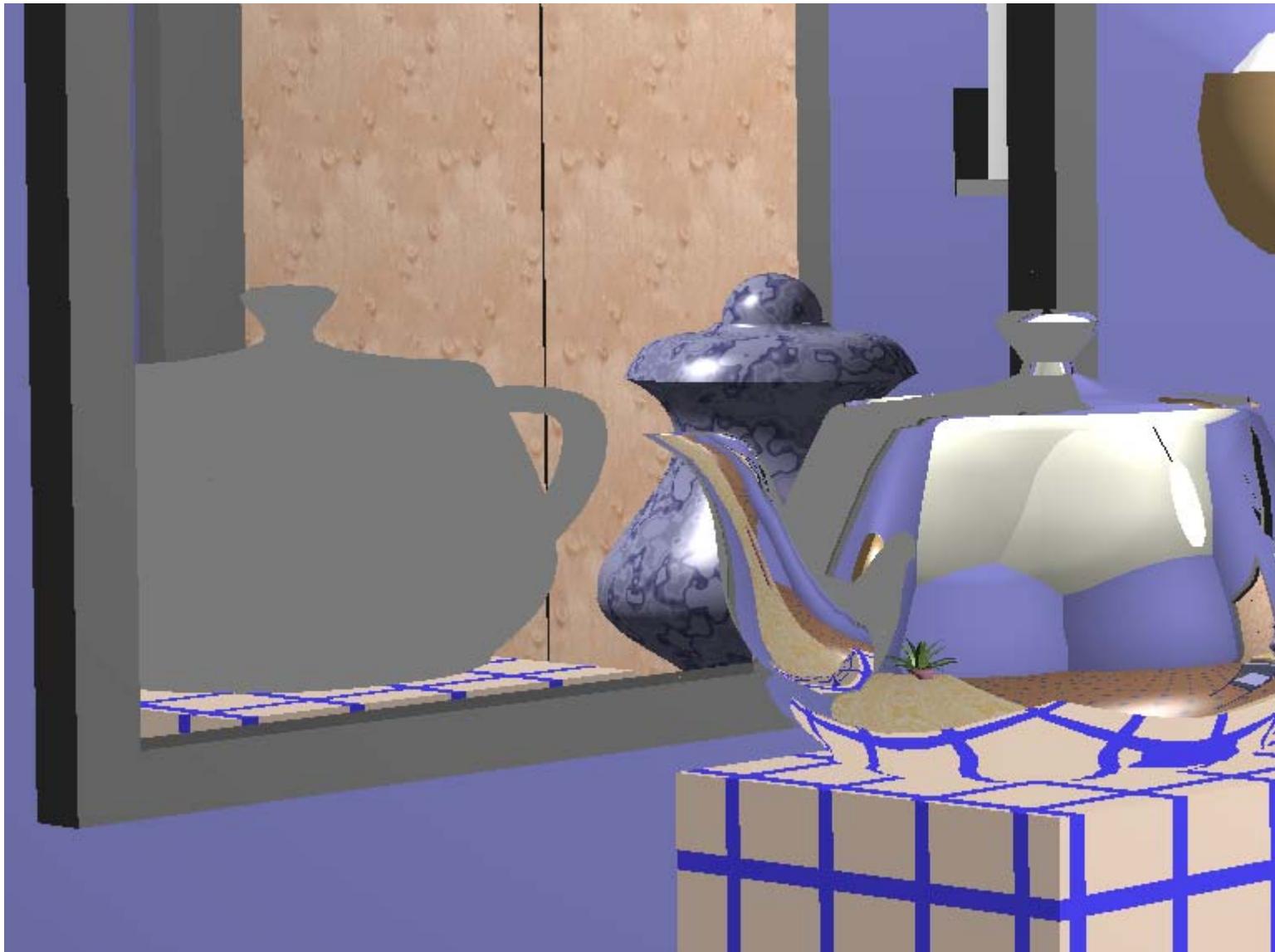
Raig 7

- Exactament igual que abans però ara el shadow ray intersecta l'esfera transparent i per tant gairebé no reduïm la contribució de la font de llum.
- Novament, no estem tenint en compte que la llum es refractaria a l'esfera i per tant el contorn de l'ombra no està a la posició correcta.

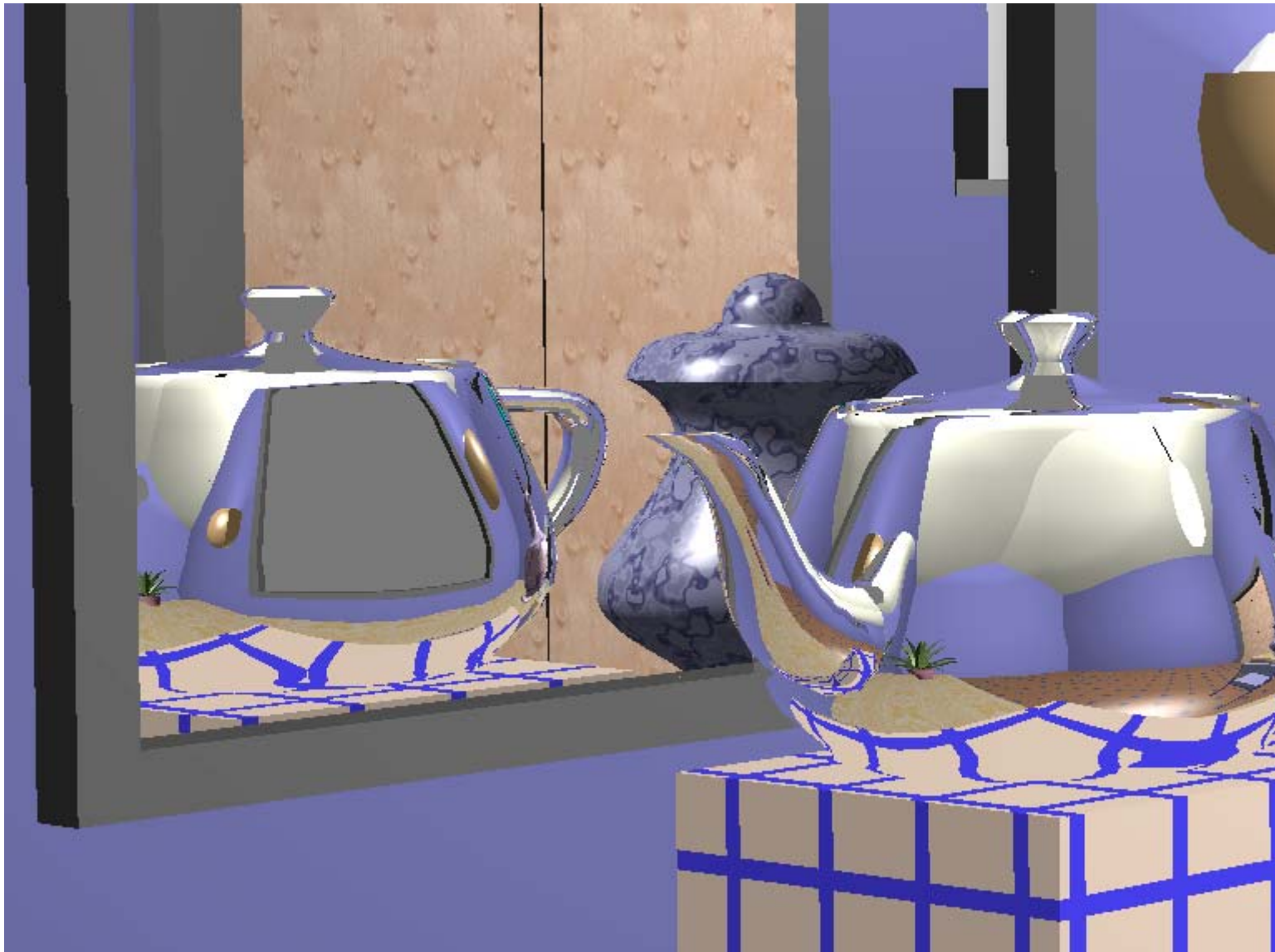
Ray-tracing clàssic. Profunditat 0



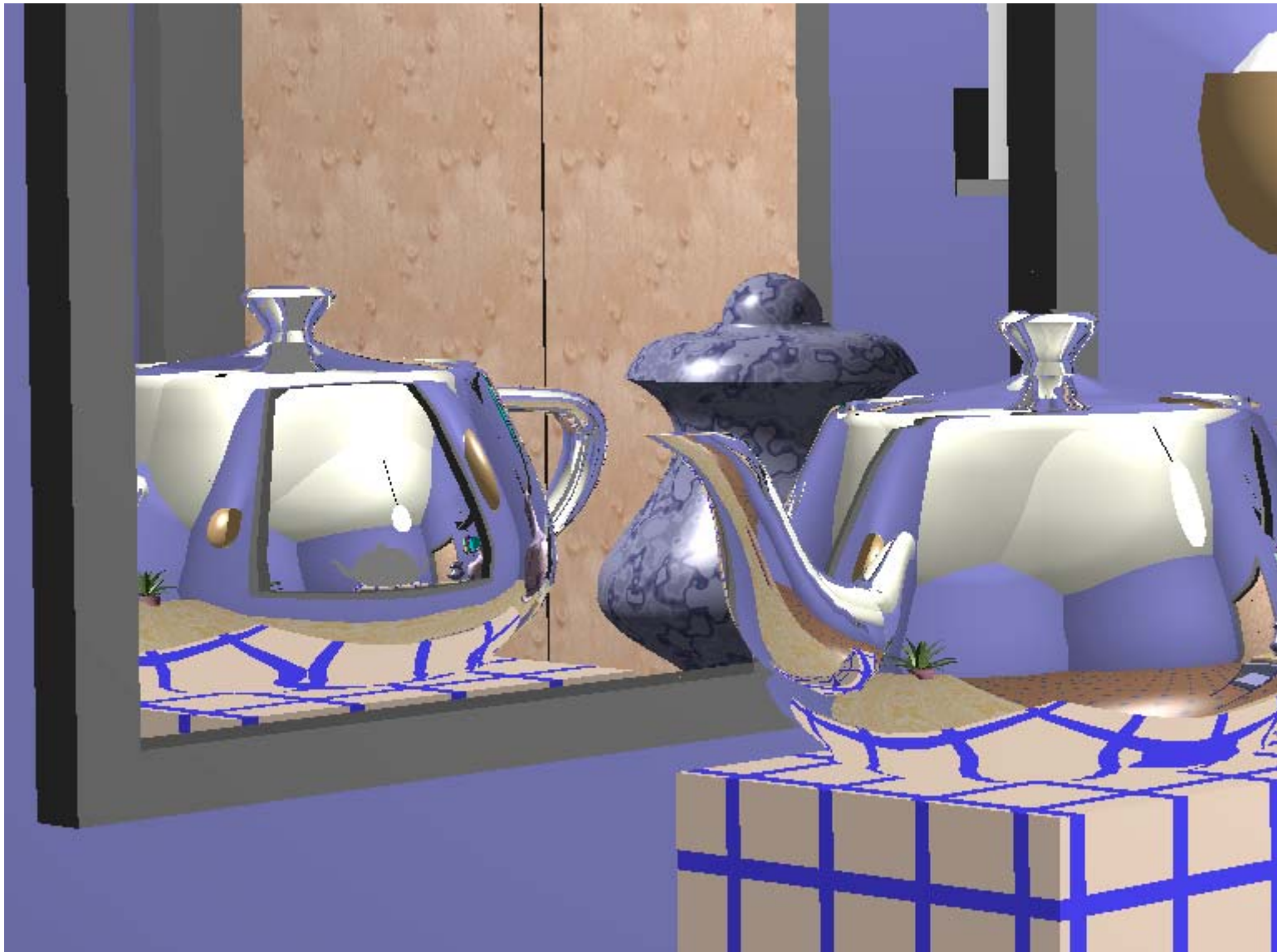
Ray-tracing clàssic. Profunditat 1



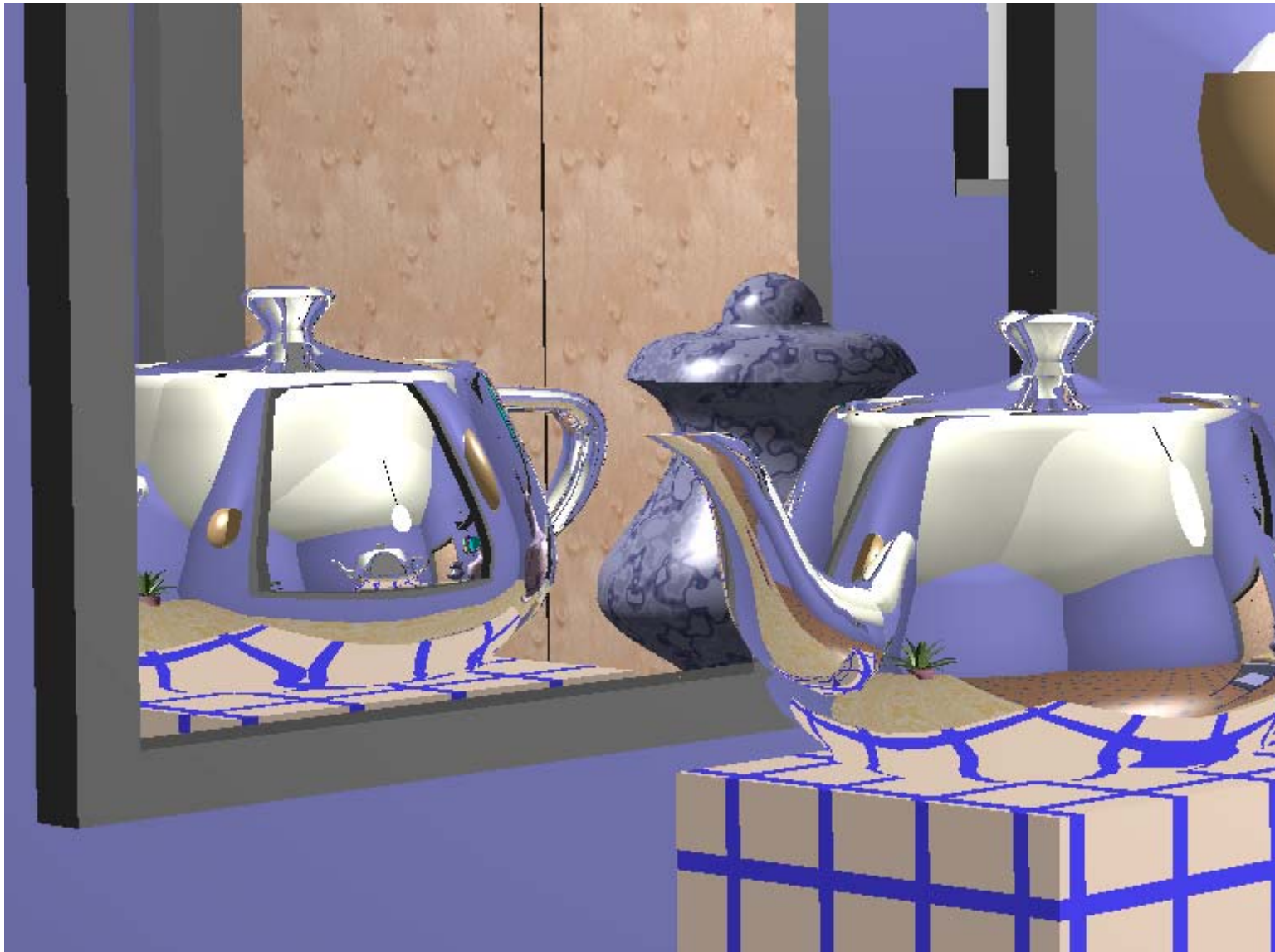
Ray-tracing clàssic. Profunditat 2



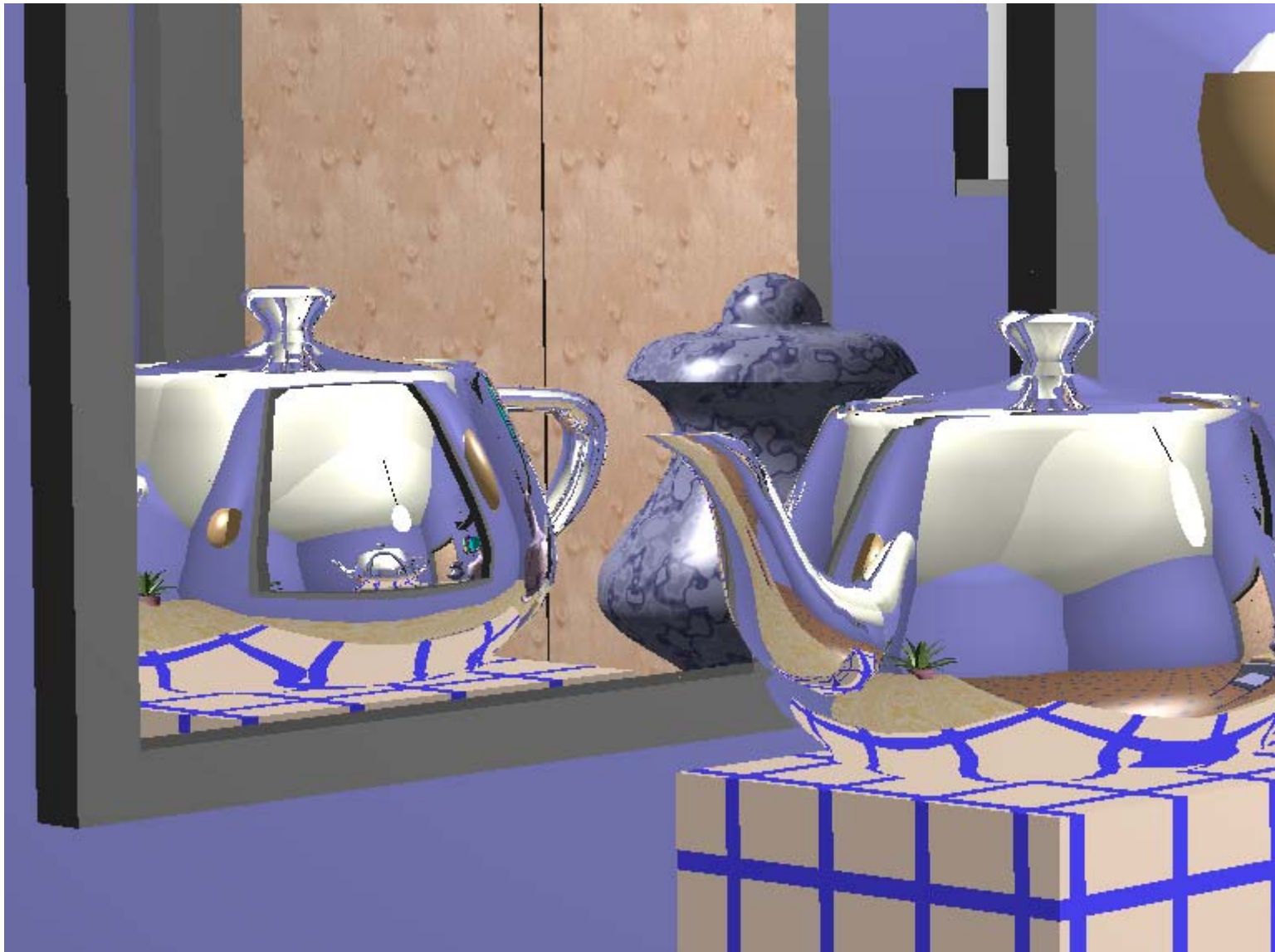
Ray-tracing clàssic. Profunditat 3



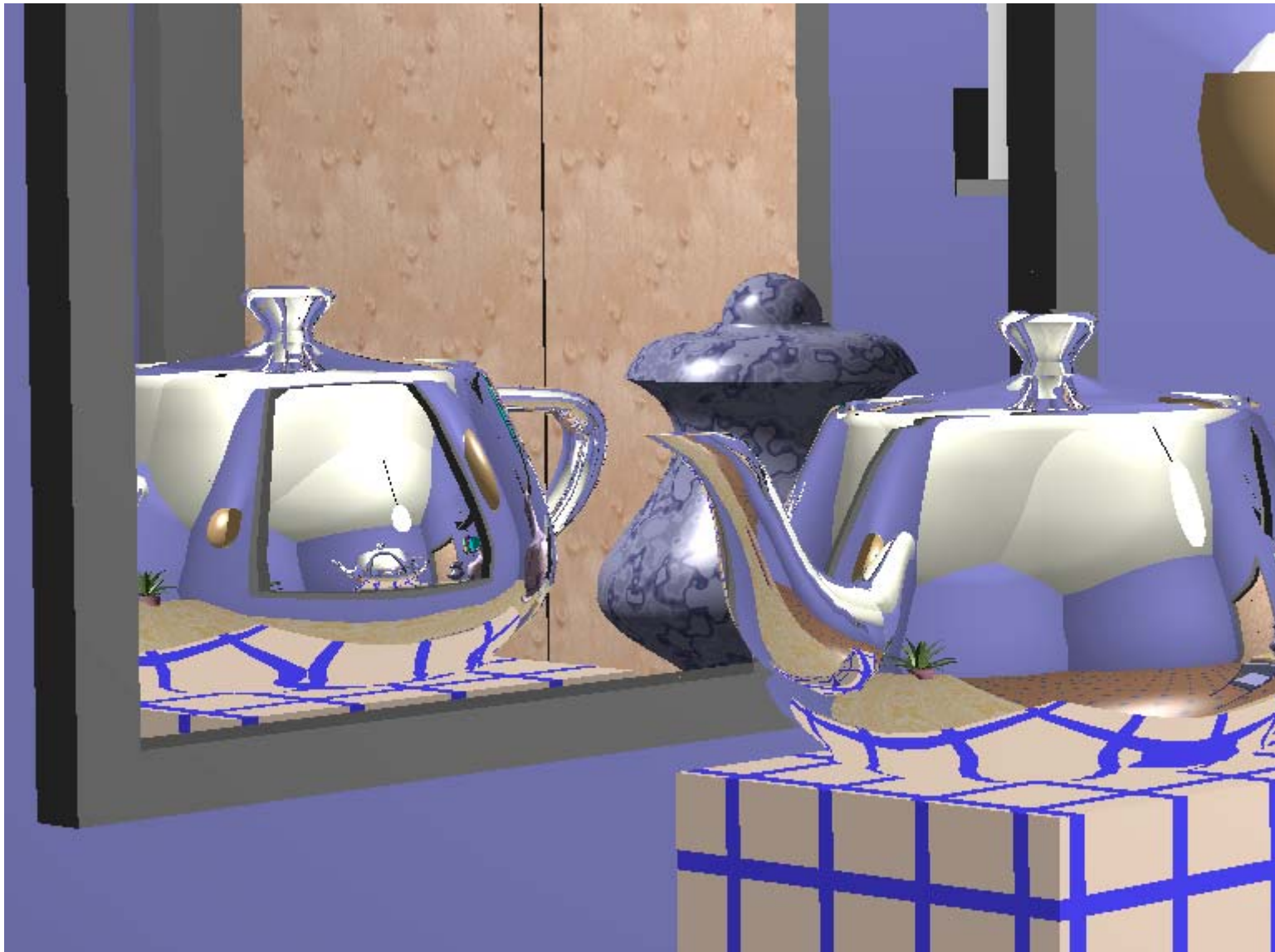
Ray-tracing clàssic. Profunditat 4



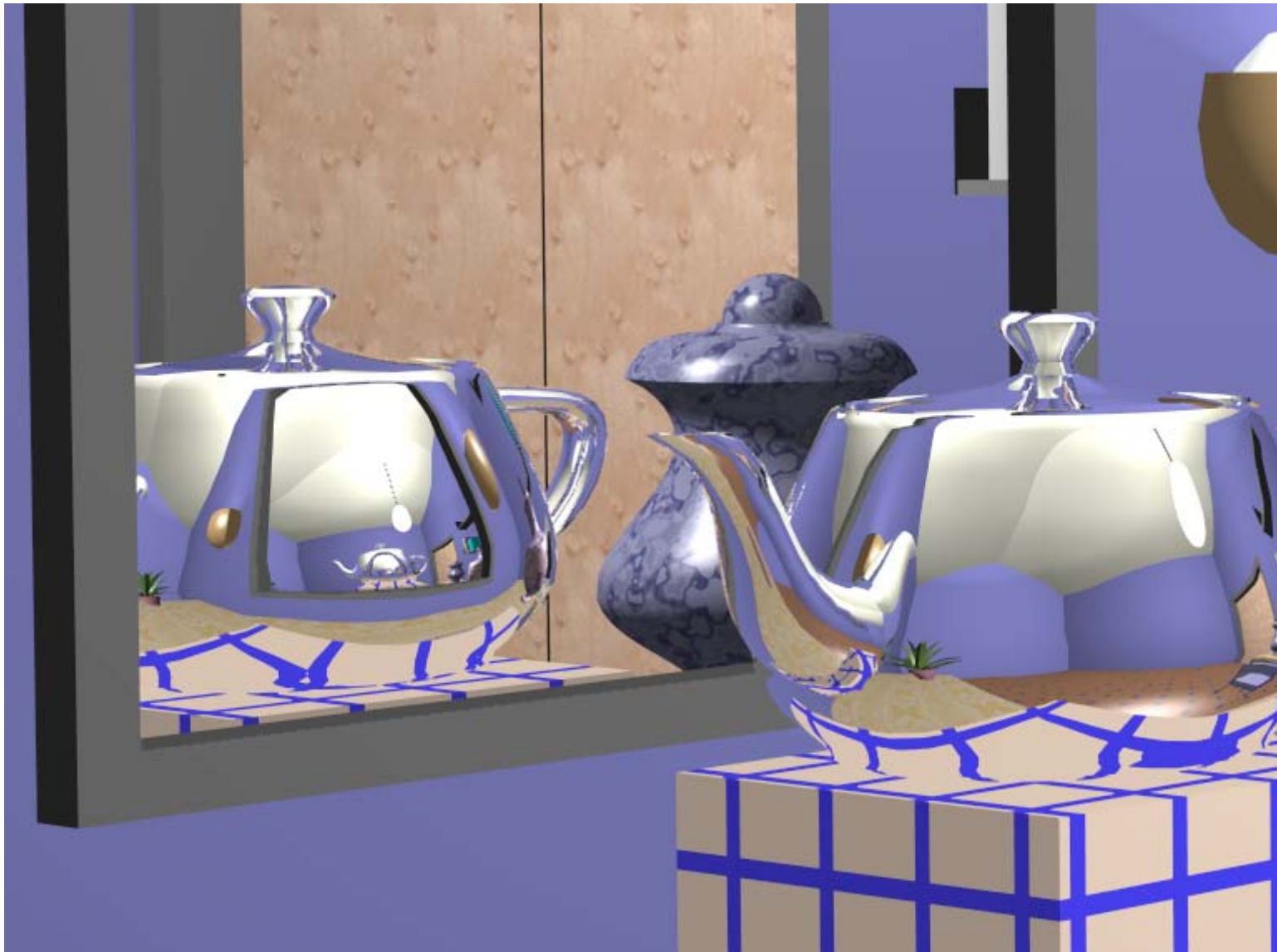
Ray-tracing clàssic. Profunditat 6



Ray-tracing clàssic. Profunditat 8



Ray-tracing clàssic. P. 8 + AA



Contribució local i global

