

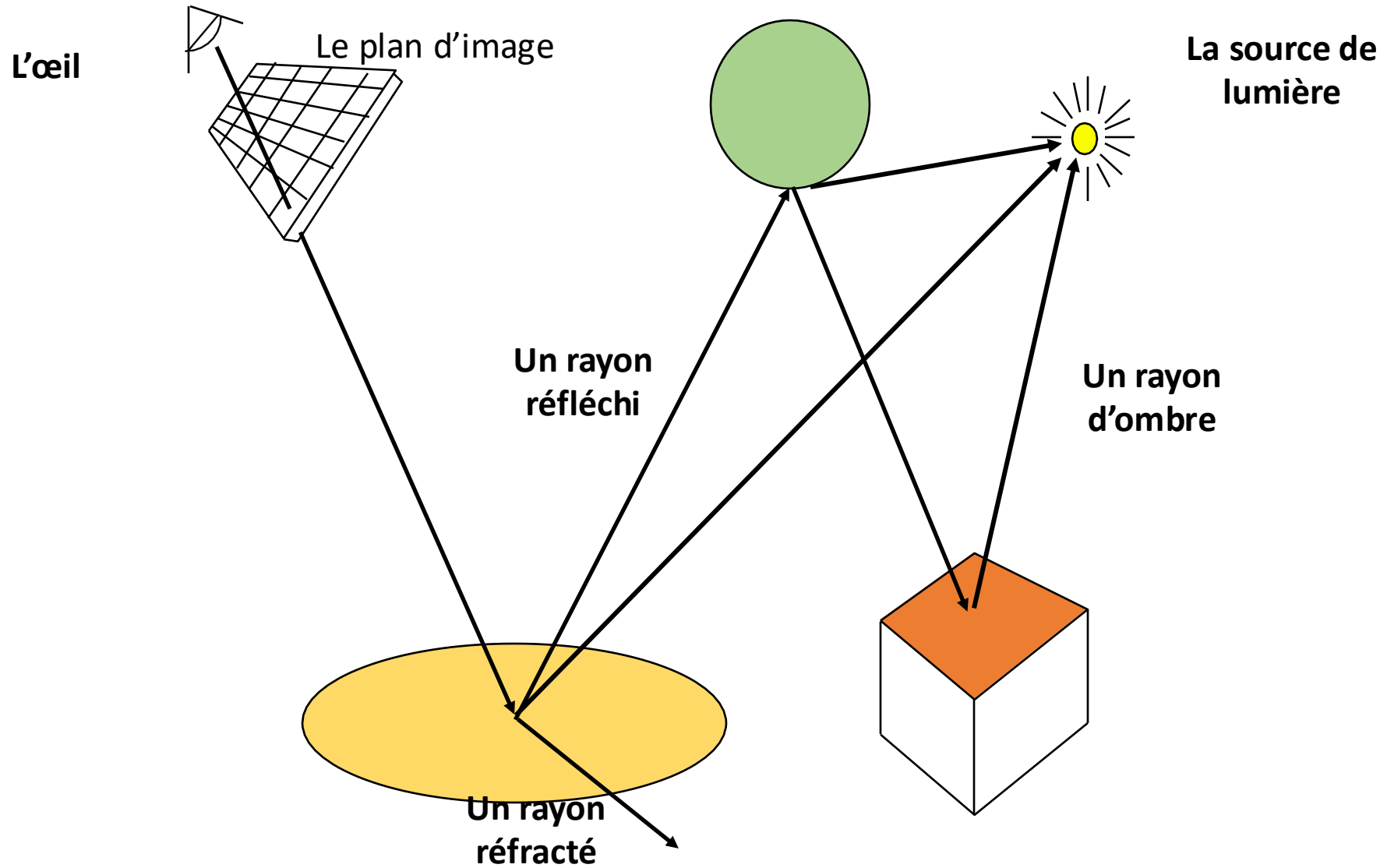
IFT 3355: INFOGRAPHIE L'ÉCLAIRAGE GLOBAL TRACER DE CHEMINS

Livre de référence: G: 20, S: 4

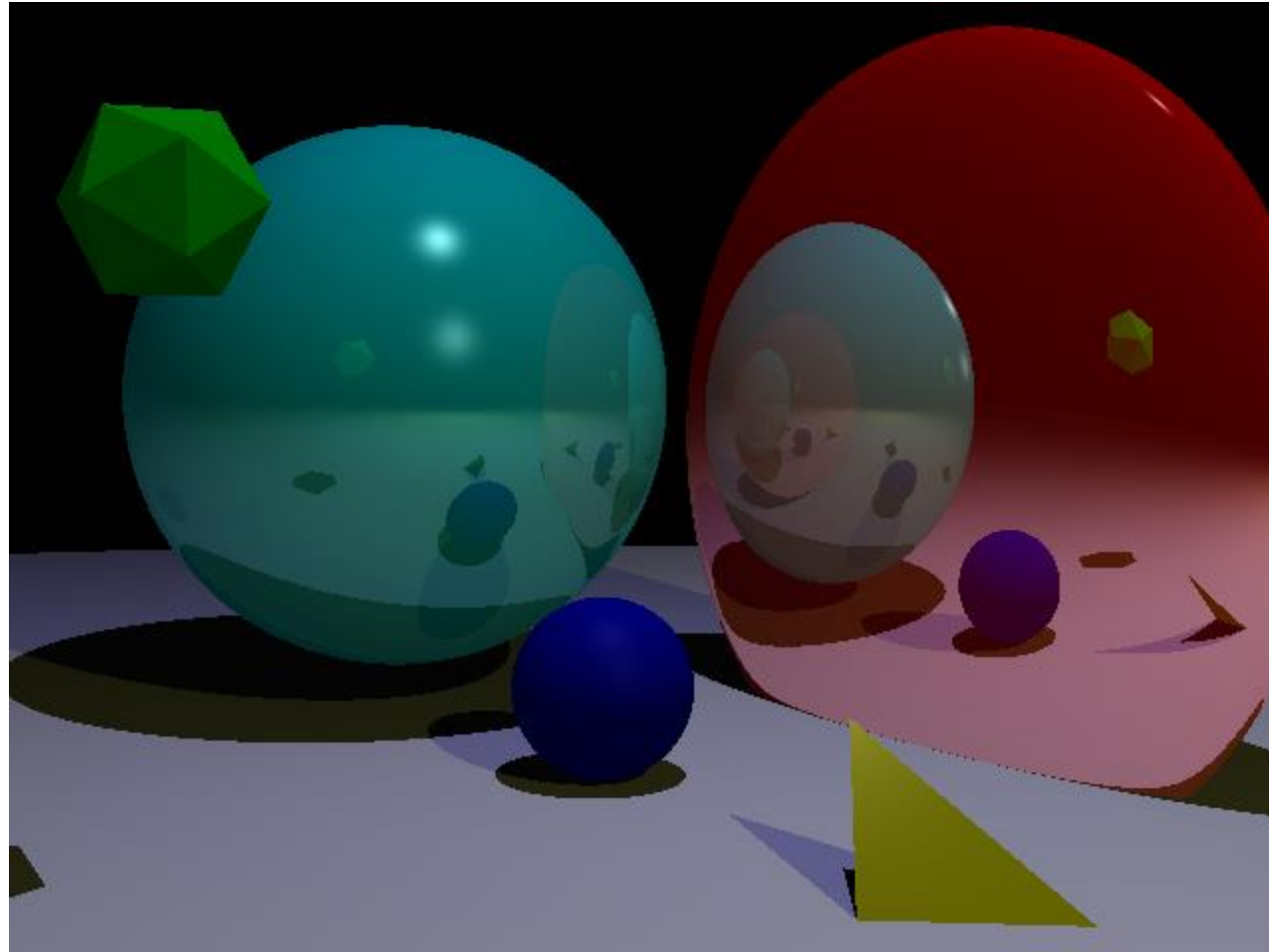
<http://tinyurl.com/ift3355>

Mikhail Bessmeltsev

LANCER DE RAYONS: DES PROBLÈMES?

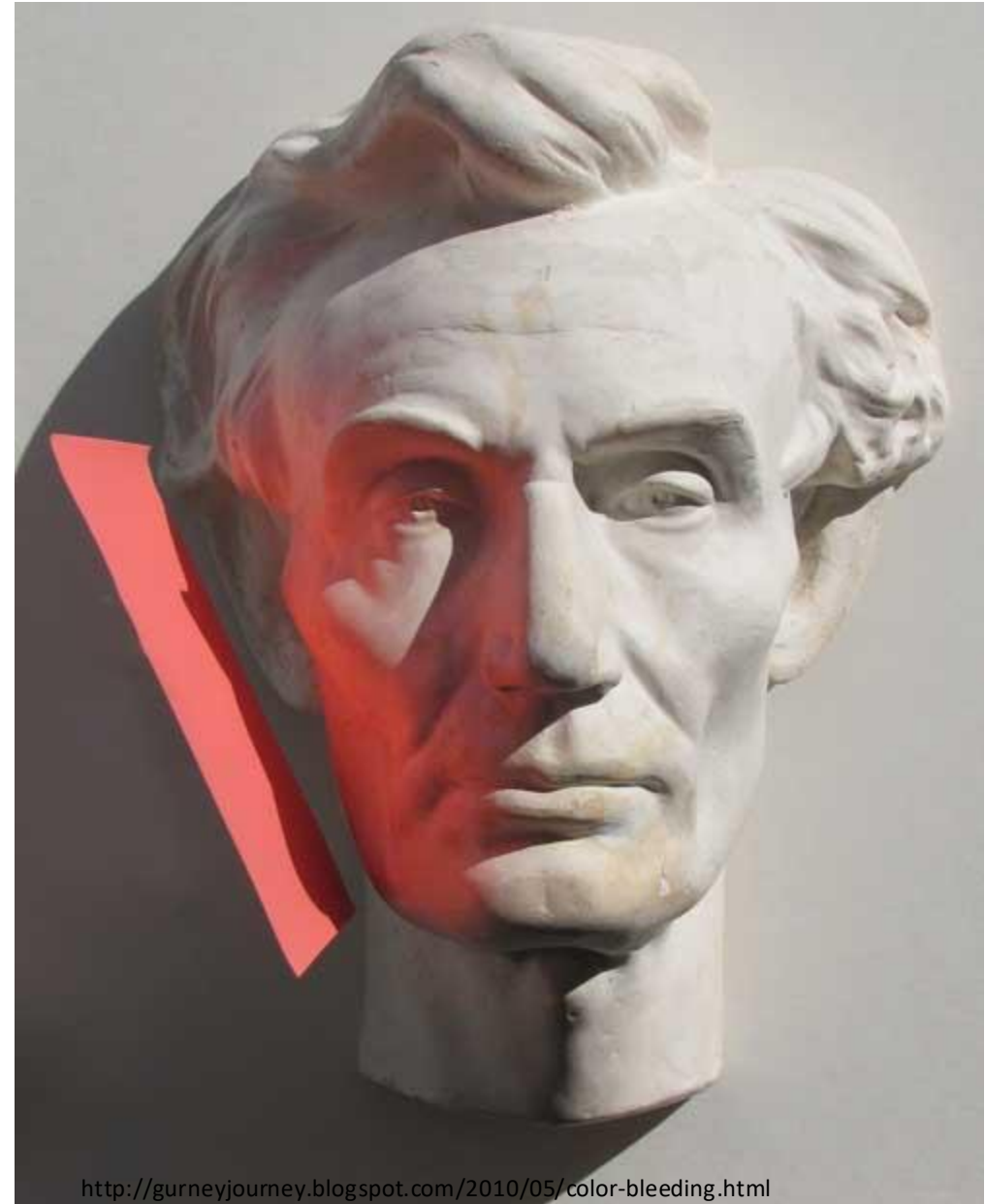


LANCER DE RAYONS: DES PROBLÈMES?



UNE LIMITATION

- Seulement les surfaces spéculaires considèrent les autres objets
 - Les réfractions aussi
- Les surfaces diffuses ou mates ne reflètent rien sauf les sources de lumière
- On utilise les modèles de **l'illumination directe**: Lambert et Phong
- Comment peut-on modéliser les objets diffus/mates dans un cadre similaire?

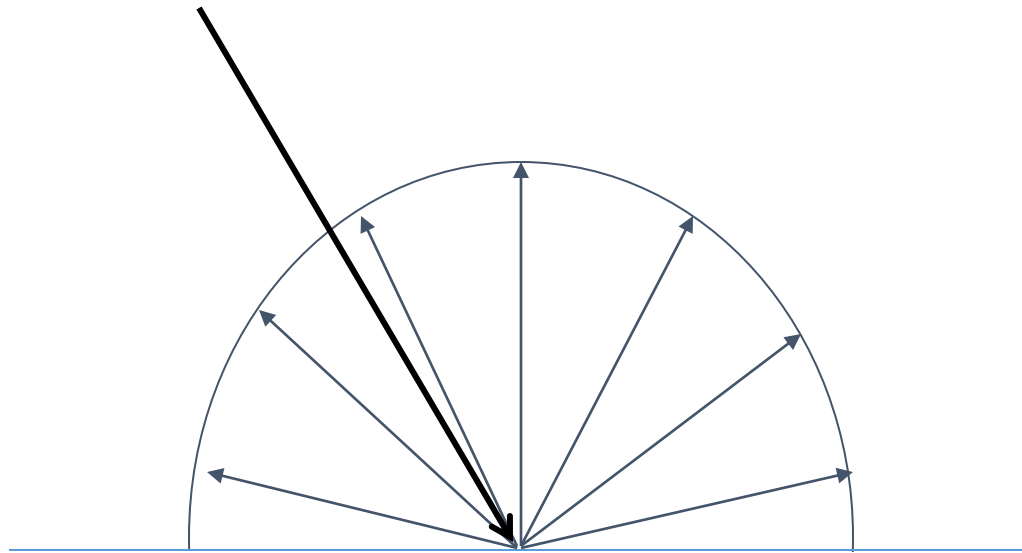


L'IDÉE

- Chaque objet est une source de lumière!

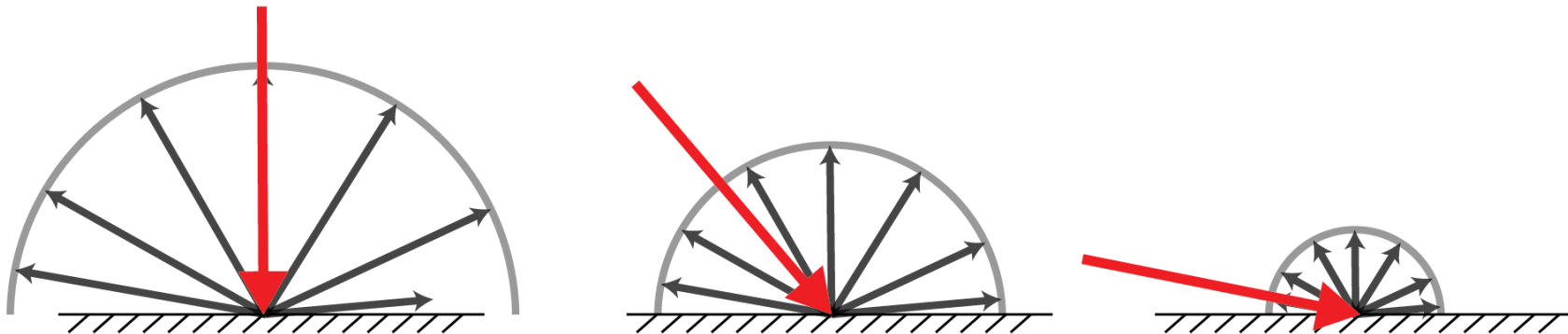
QUE SE PASSE-T-IL SUR UNE SURFACE DIFFUSE?

- Imaginez un photon qui frappe une surface
- S'il n'est pas absorbé ou réfracté, il devrait se refléter
- La direction de la réflexion est aléatoire
- Pour les surfaces lambertiennes, la distribution est uniforme



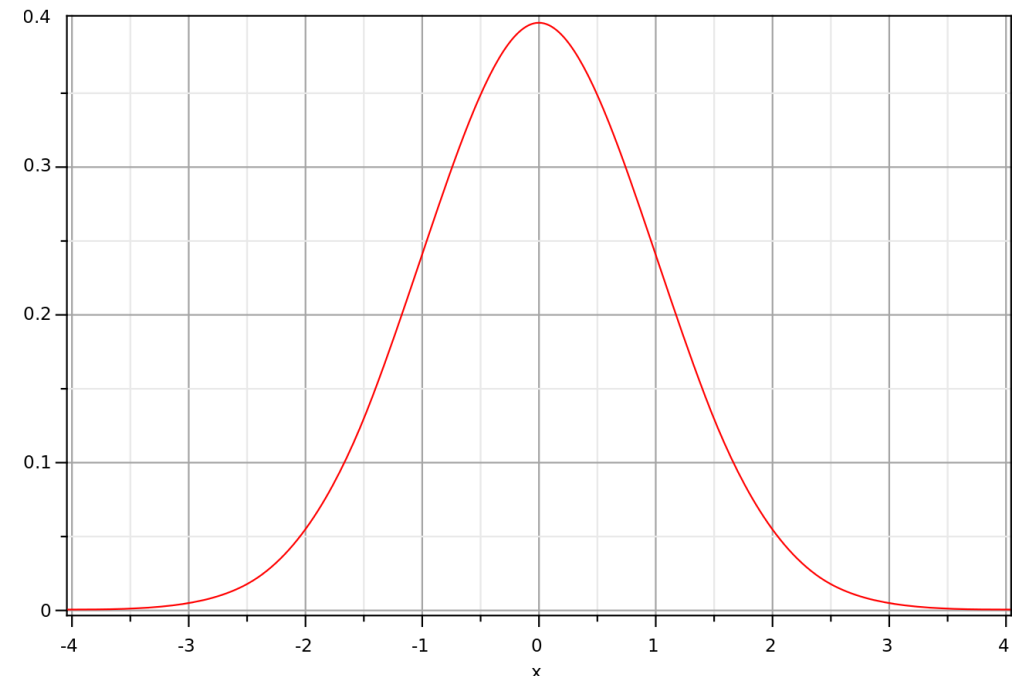
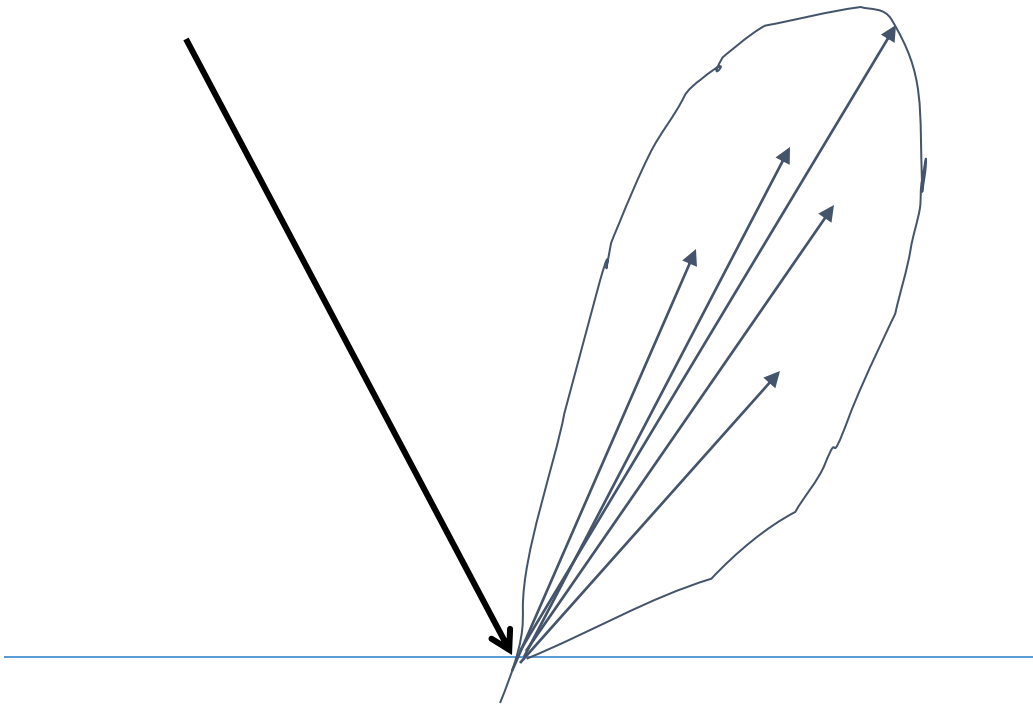
QUE SE PASSE-T-IL SUR UNE SURFACE DIFFUSE?

- Imaginer un photon qui frappe une surface
- S'il n'est pas absorbé ou réfracté, il devrait se refléter
- La direction de la réflexion est aléatoire
- Pour les surfaces lambertiennes, la distribution est uniforme
- L'énergie est proportionnelle au cosinus

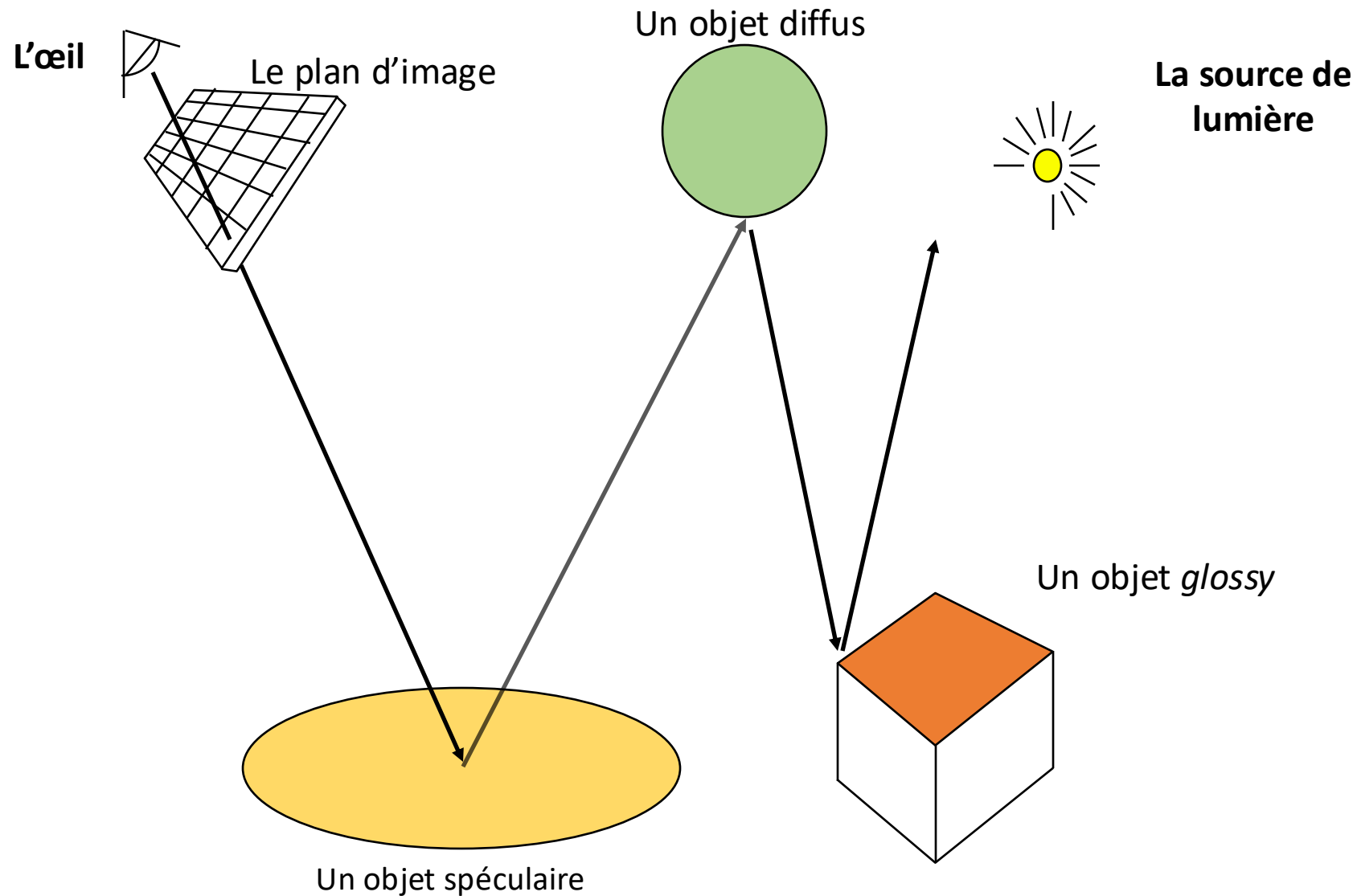


QUE SE PASSE-T-IL SUR UNE SURFACE GLOSSY/NON-DIFFUSE?

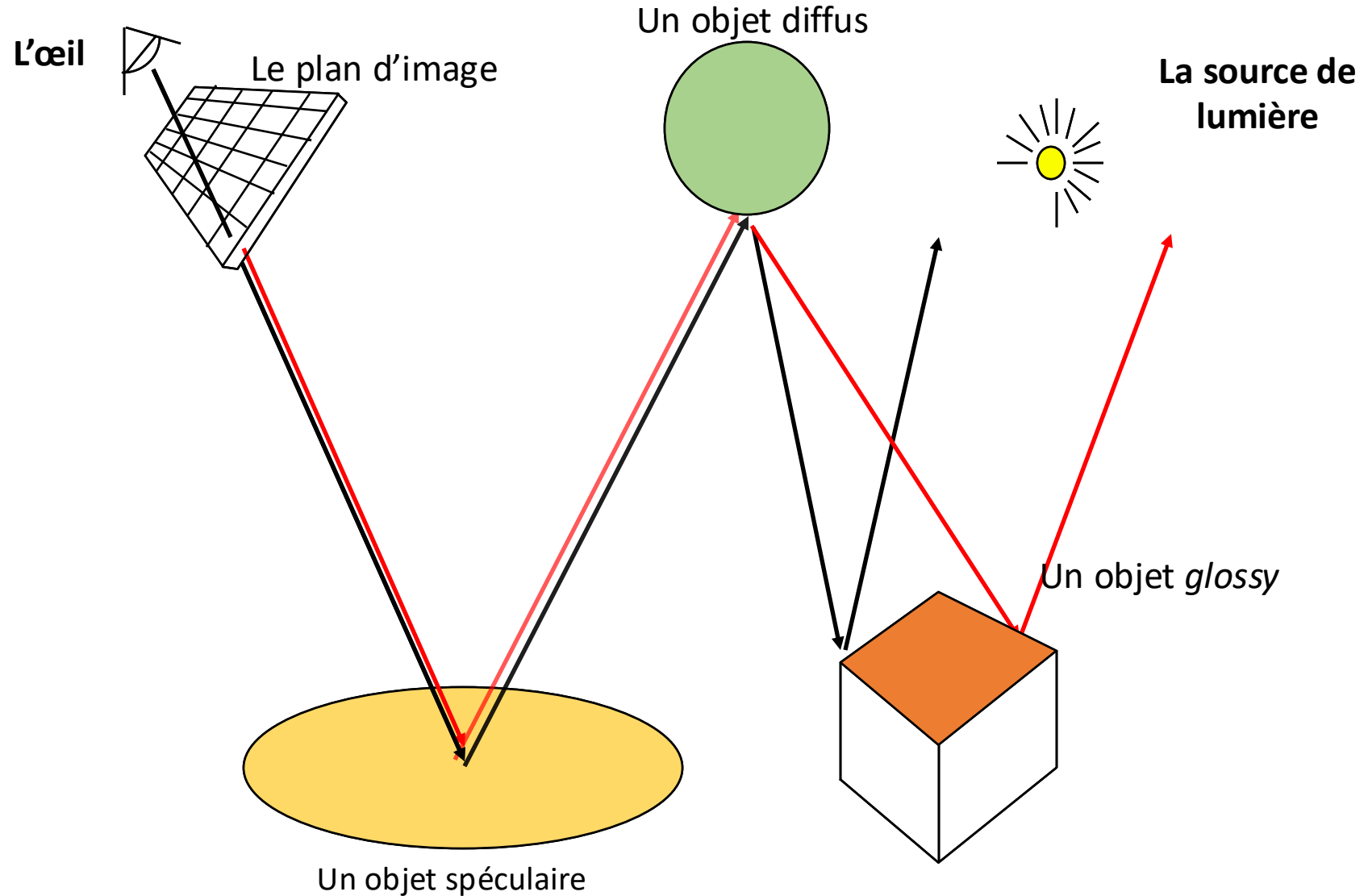
- La majorité de rayons sont reflété dans des directions similaires
- La distribution est gaussienne



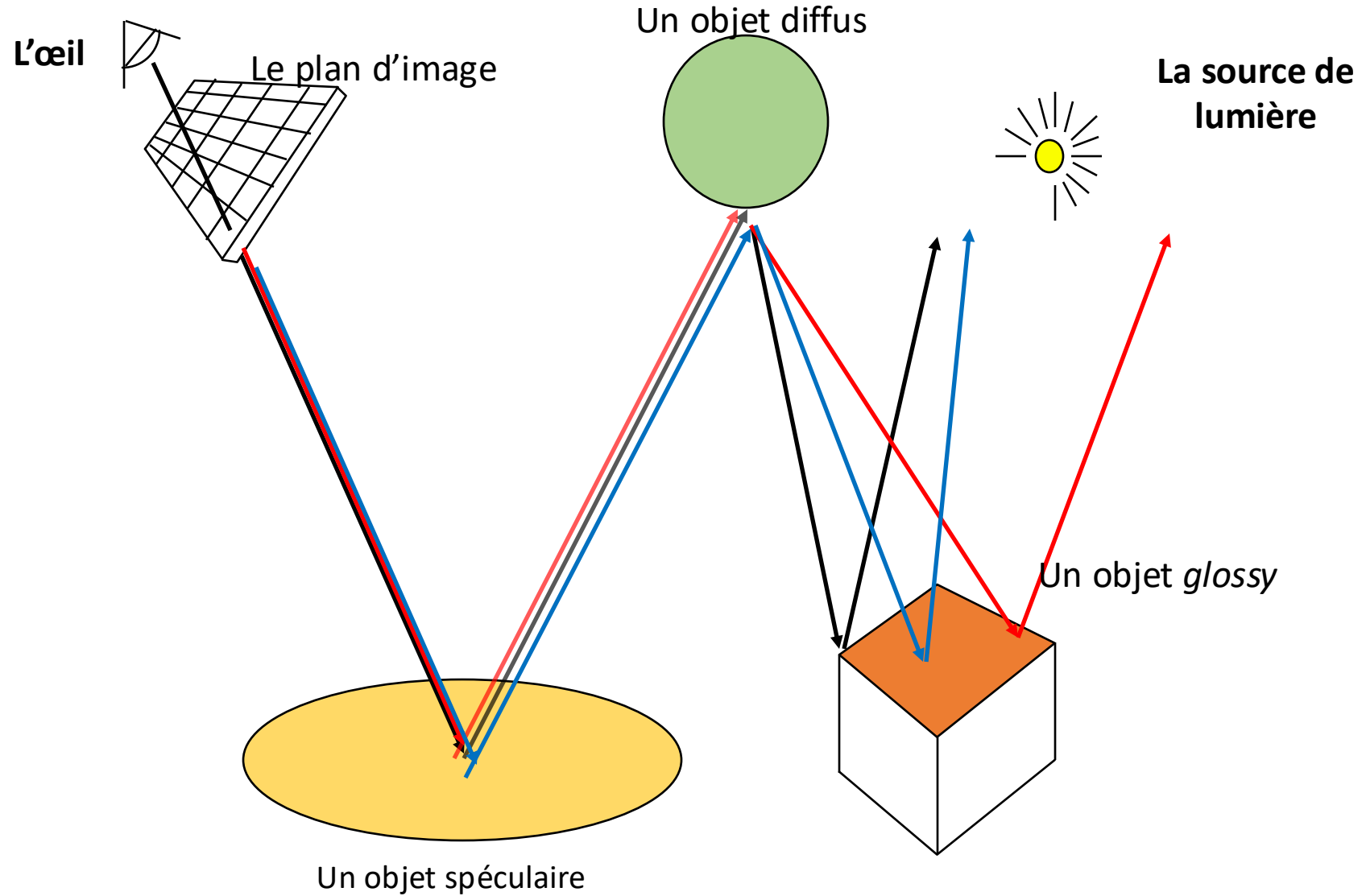
TRACER DE CHEMINS



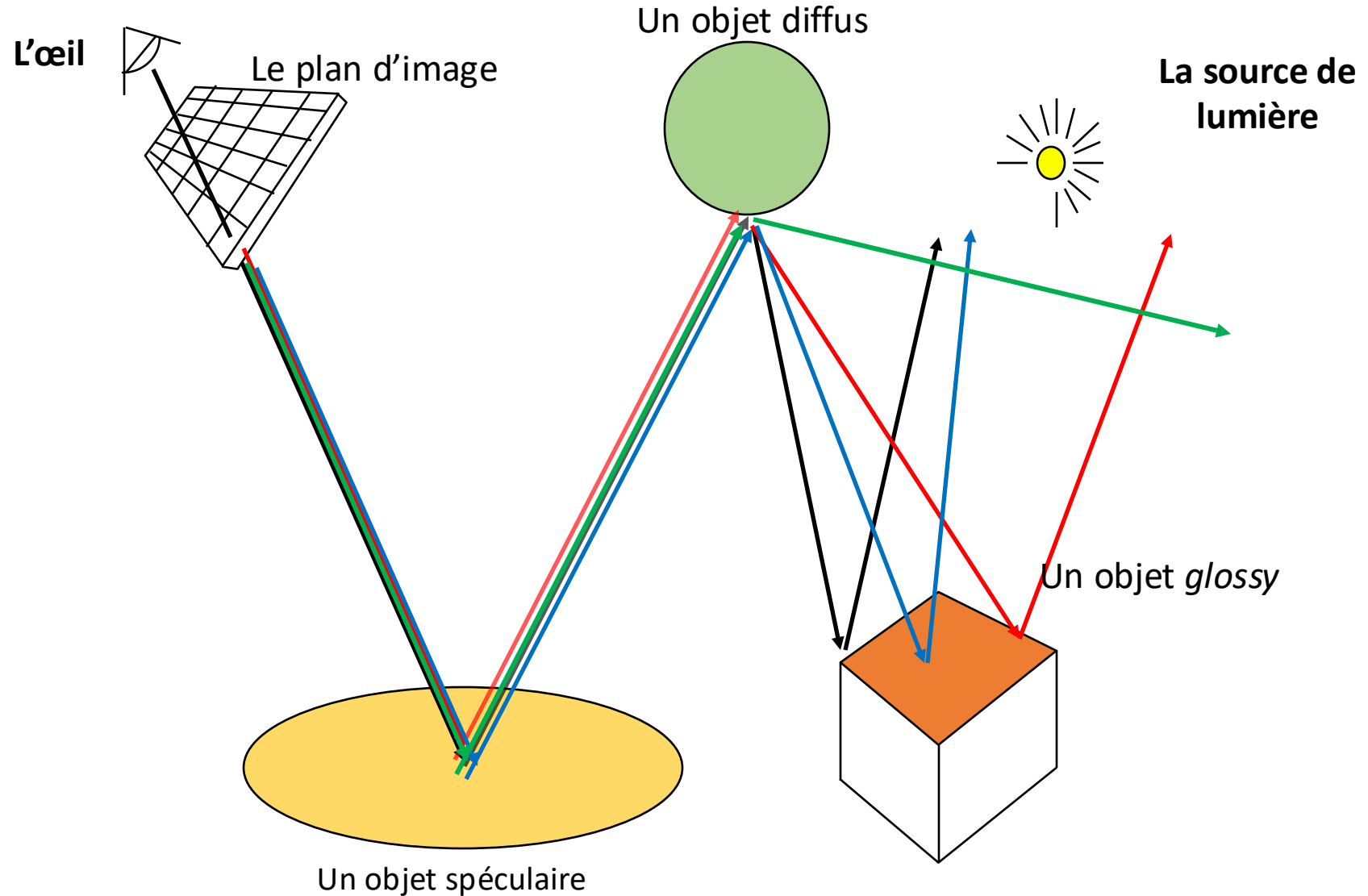
TRACER DE CHEMINS



TRACER DE CHEMINS



TRACER DE CHEMINS



TRACER DE CHEMINS

1. Tirer un rayon à travers (i, j) . Mettre l'*atténuation* à 1.0.
2. Trouver l'intersection la plus proche du rayon avec un objet
3. Choisir au hasard "*émission*" ou "*réflexion*"
 - a. Si "*émission*", retourner **emissionColor**;
 - b. Si "*réflexion*",
Choisir une direction de réflexion aléatoirement
rayWeight *= reflectance;
GOTO 2

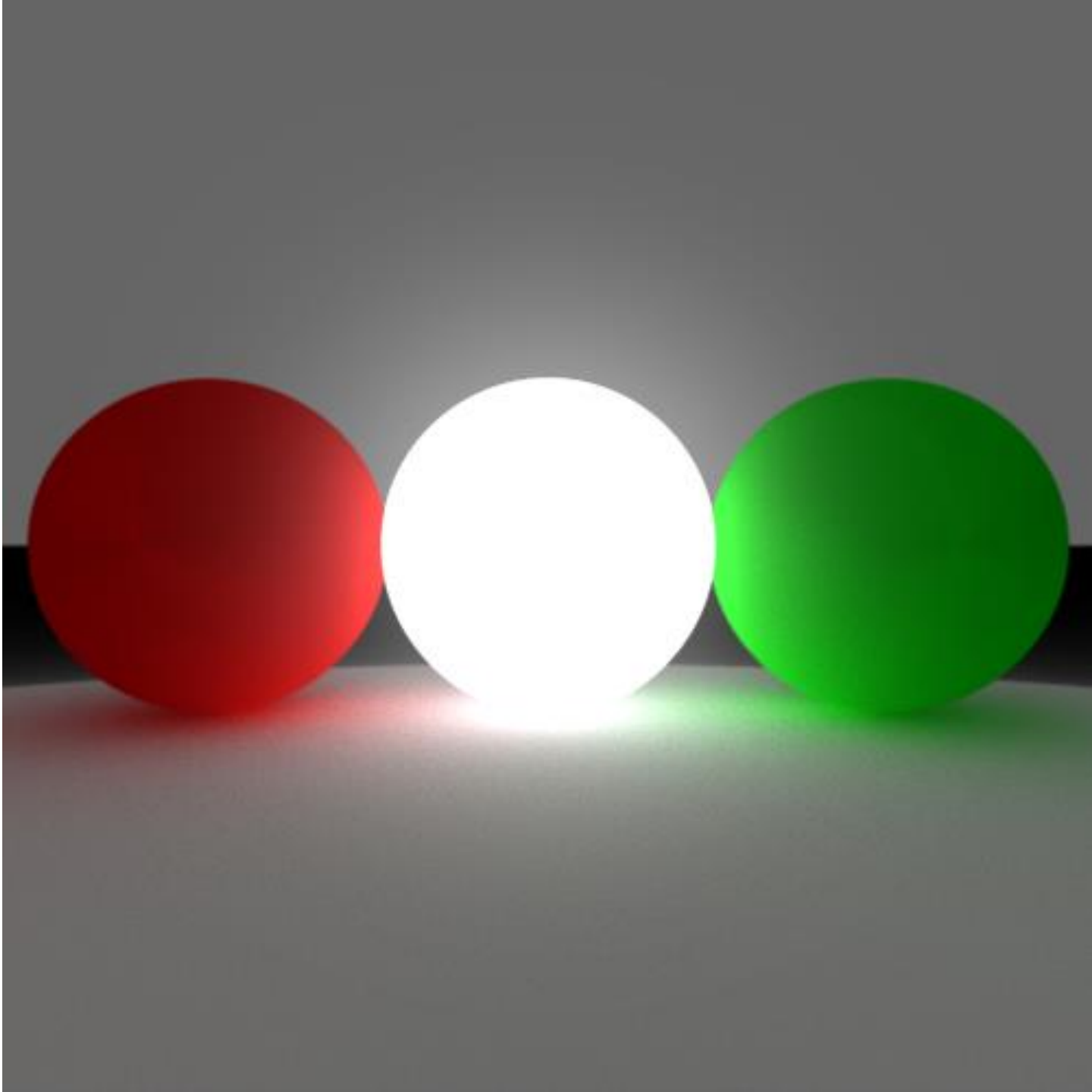
TRACER DE CHEMINS: LA VERSION LA PLUS SIMPLE

- Pour chaque pixel (i, j) :
 - For $k=1, \dots, N$:
 - Un rayon $r = \text{generateRay}(i, j)$;
 - $\text{PixelColor}(i, j) += \text{pathTrace}(r)/N$;

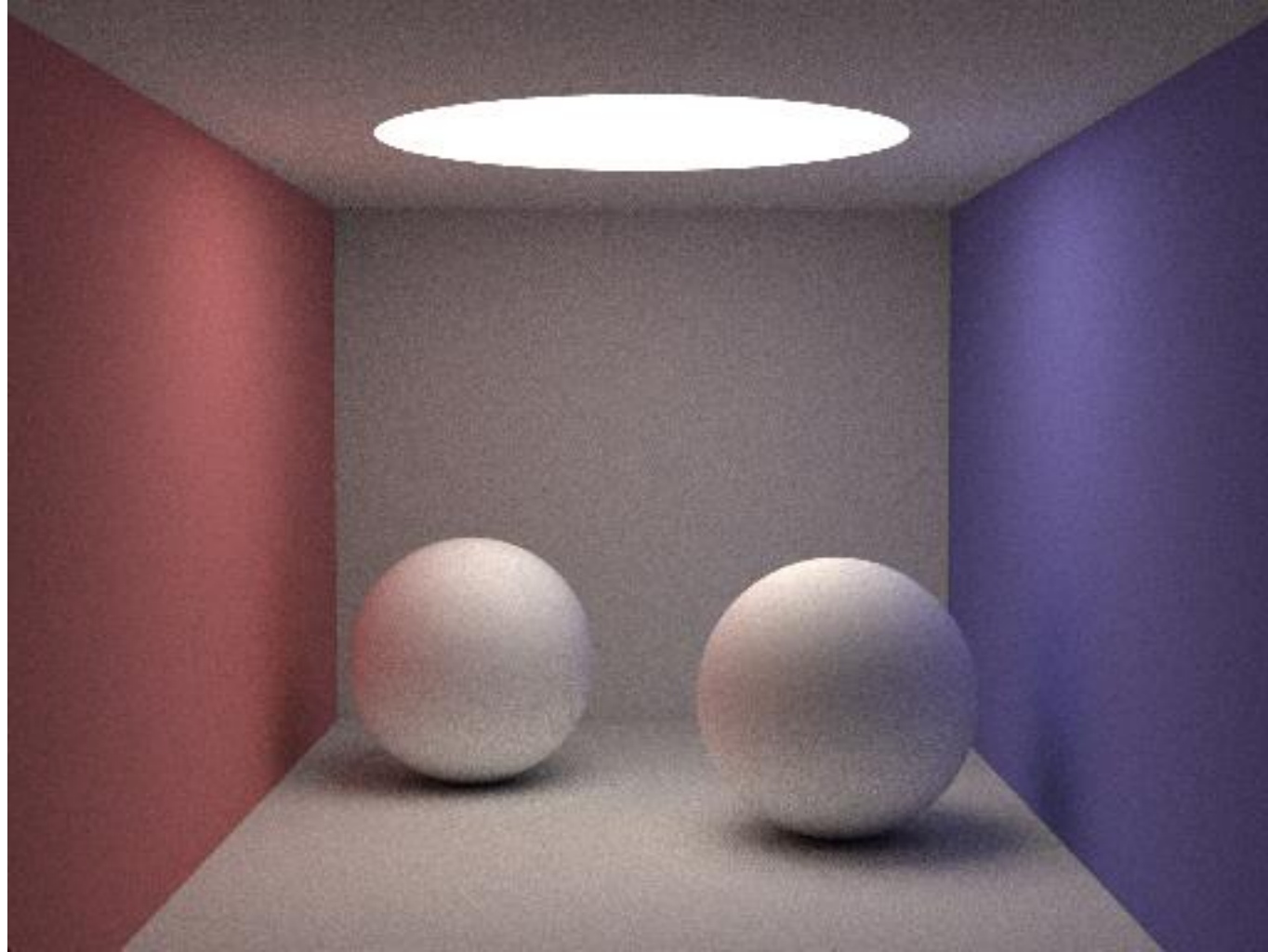
TRACER DE CHEMINS: LA VERSION LA PLUS SIMPLE

```
PathTrace(Ray r) {  
    P = closestIntersection(r);  
    if (random(emit, reflect) == emit)  
        return EmissionColor;  
    else {  
        Ray v = {intersectionPt,  
            randomDirectionInHemisphere(r.normalWhereObjWasHit)};  
        double cos_theta = dot(v.direction, r.normalWhereObjWasHit);  
        return PathTrace(v)*cos_theta*reflectance;  
    }  
}
```

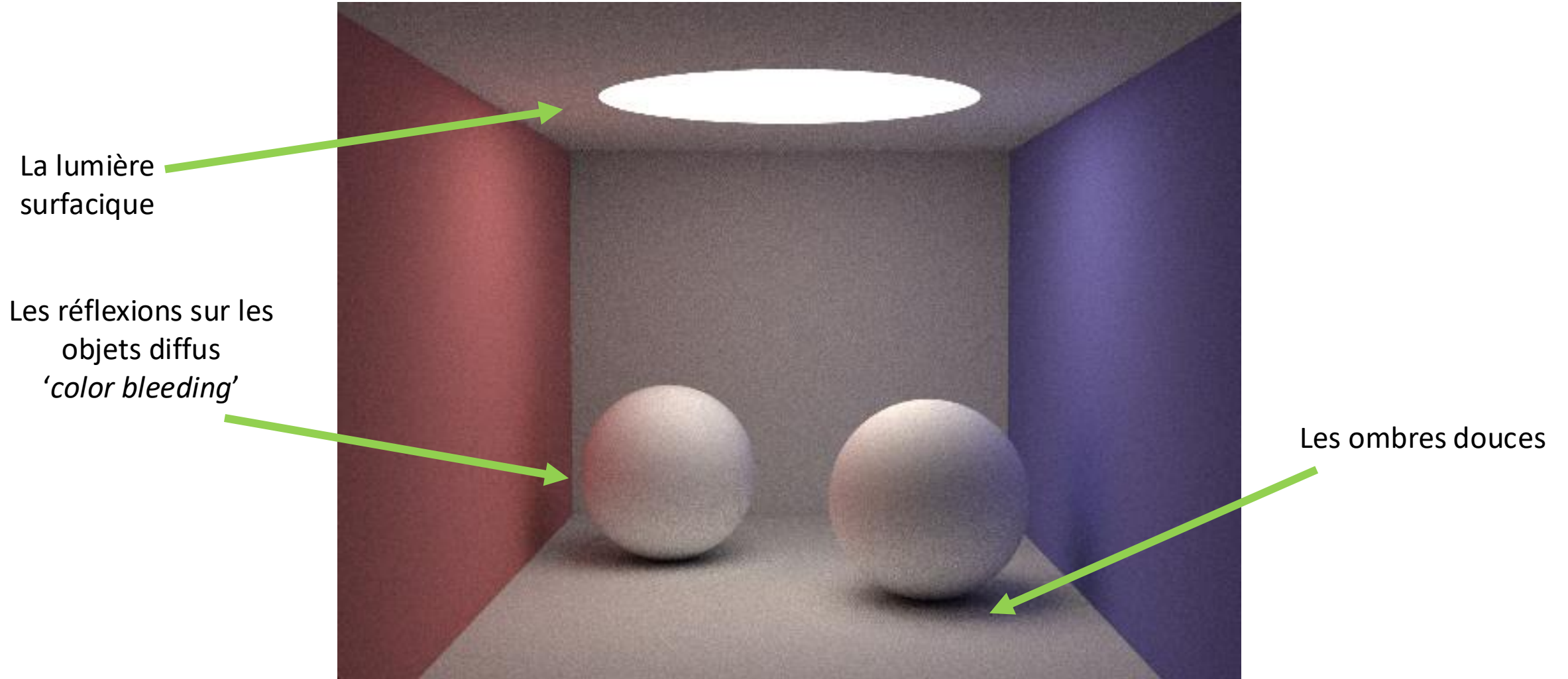
EXAMPLE



ON OBTIENT



ON OBTIENT



Le bruit



LES METHODES DE MONTE CARLO

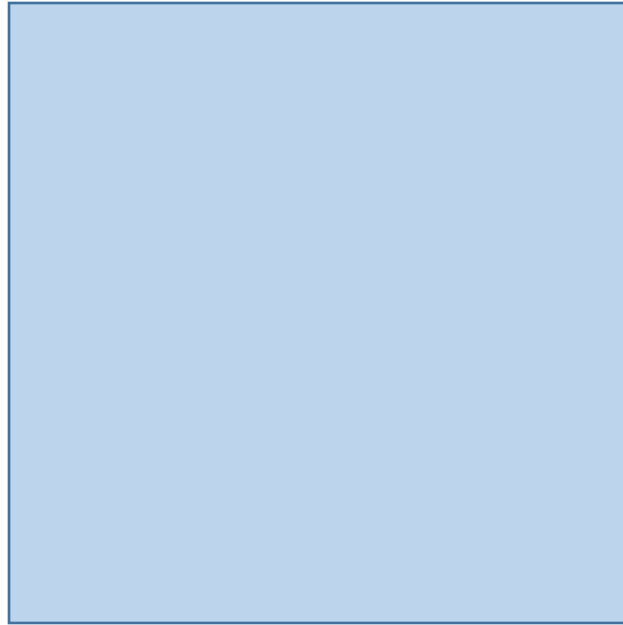
- L'idée: calculer quelque chose en utilisant un échantillonnage aléatoire
- E.g. calculer les intégrales difficiles
- E.g. les aires ou volumes
 - Il peut être difficile de les calculer analytiquement
 - Mais il est normalement facile de déterminer si un point est à l'intérieur
- L'erreur de l'approximation est en fonction du nombre des échantillons

GÉNÉRER DES POINTS ALÉATOIRES

- Souvent on a une seule fonction `random()`;
- Qui génère un chiffre entre 0 et 1
- Avec une distribution uniforme

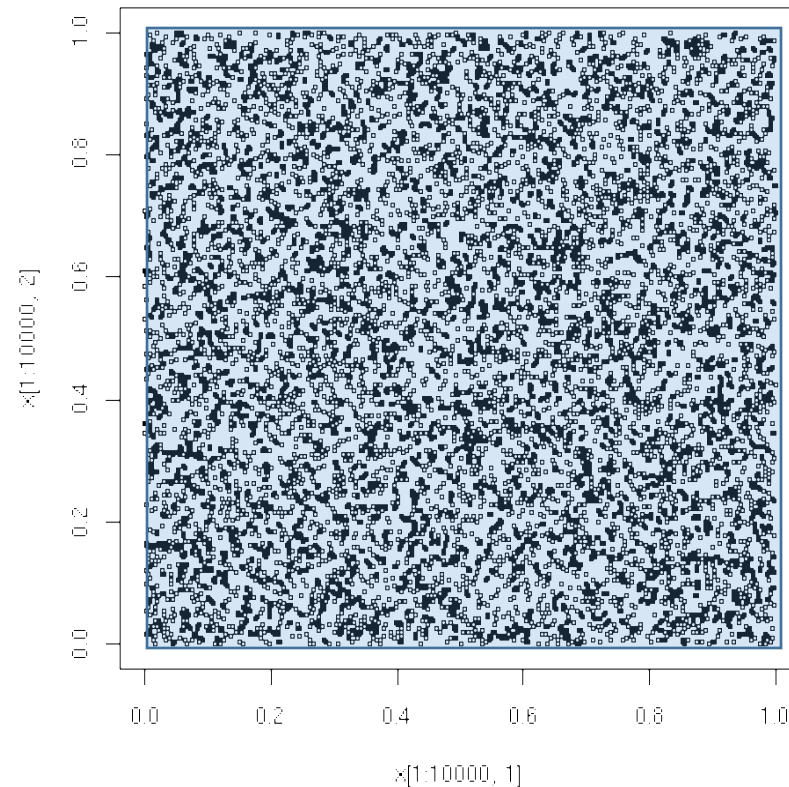
GÉNÉRER DES POINTS ALÉATOIRES

- Comment générer un point à l'intérieur d'un carré?



GÉNÉRER DES POINTS ALÉATOIRES

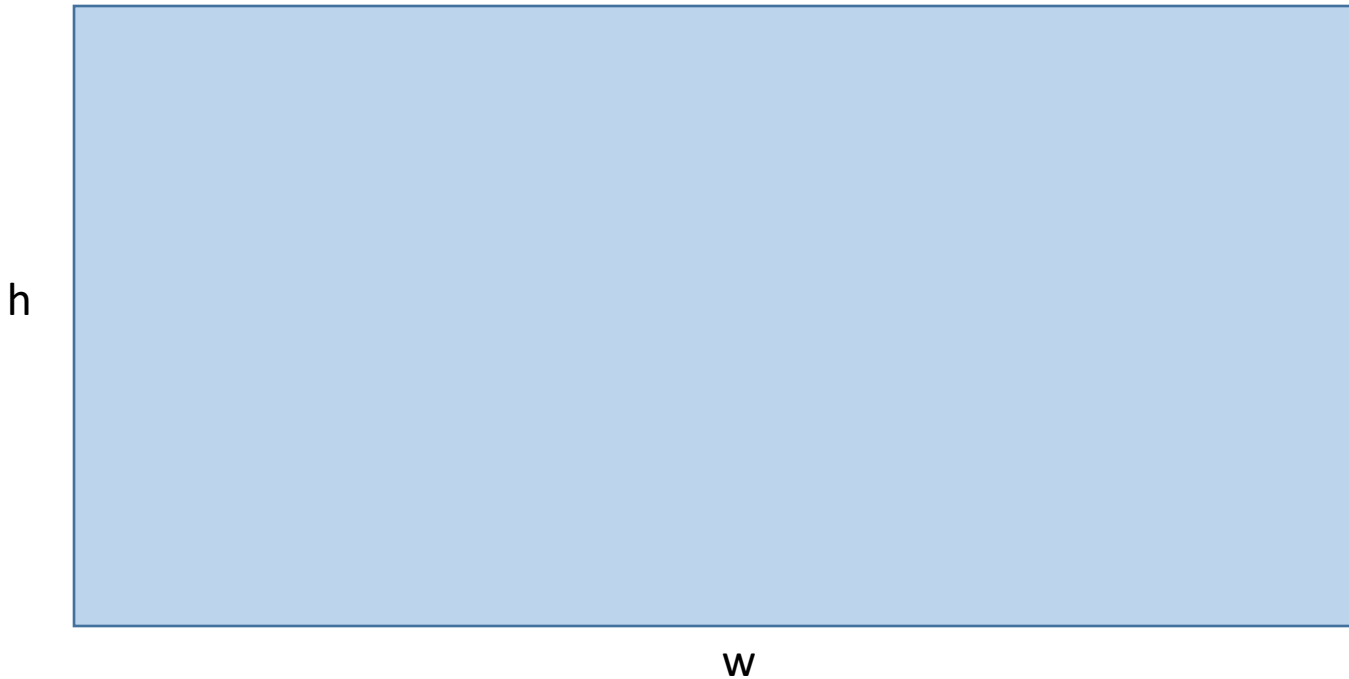
- Comment générer un point à l'intérieur d'un carré?



```
for (i=0...N)
{
    x = rand();
    y = rand();
}
```

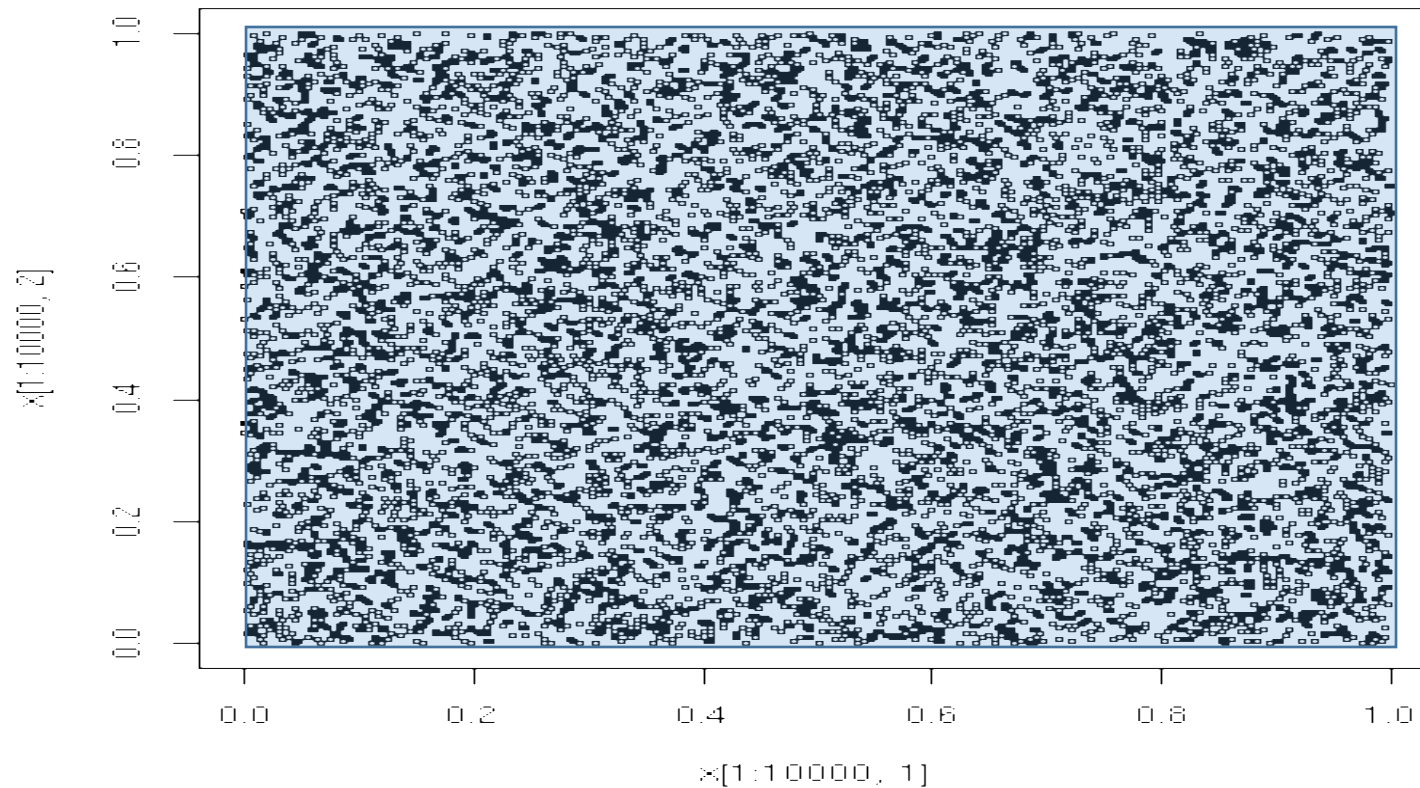
GÉNÉRER DES POINTS ALÉATOIRES

- Comment générer un point à l'intérieur d'un rectangle?



GÉNÉRER DES POINTS ALÉATOIRES

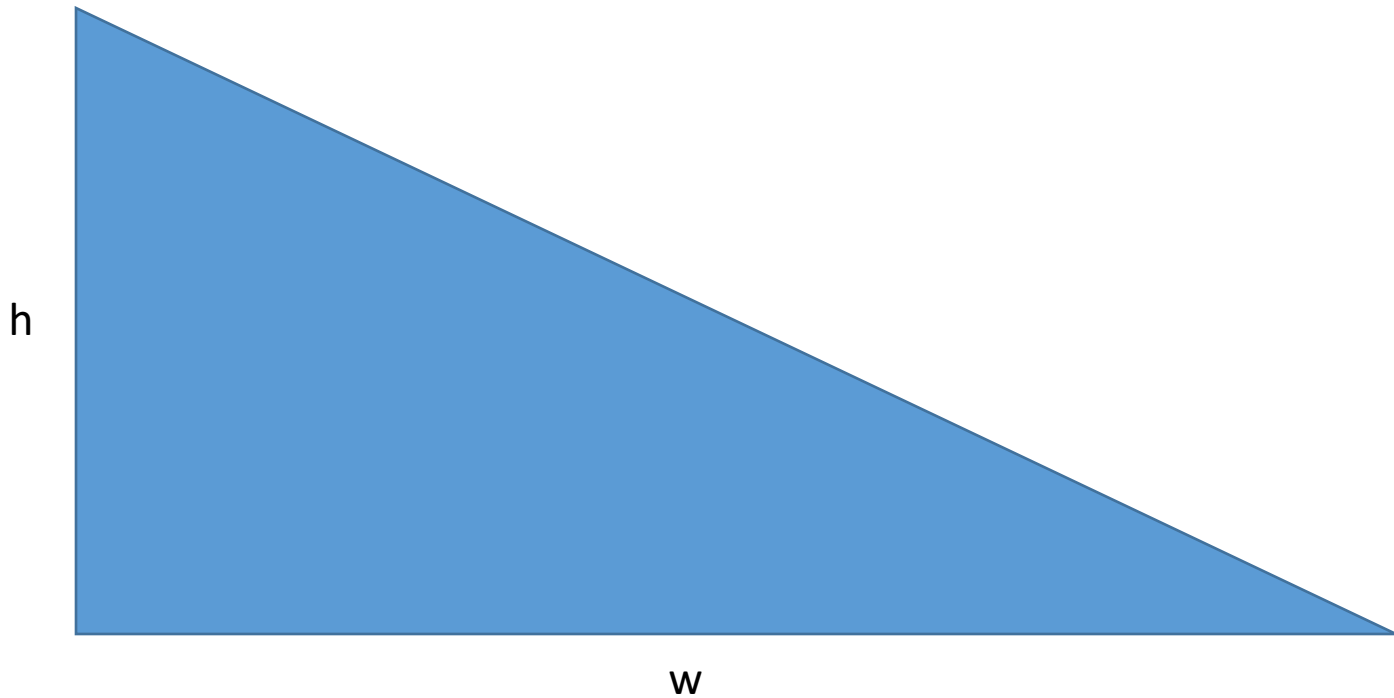
- Comment générer un point à l'intérieur d'un rectangle?



```
for (i=0...N)
{
    x = w*rand();
    y = h*rand();
}
```

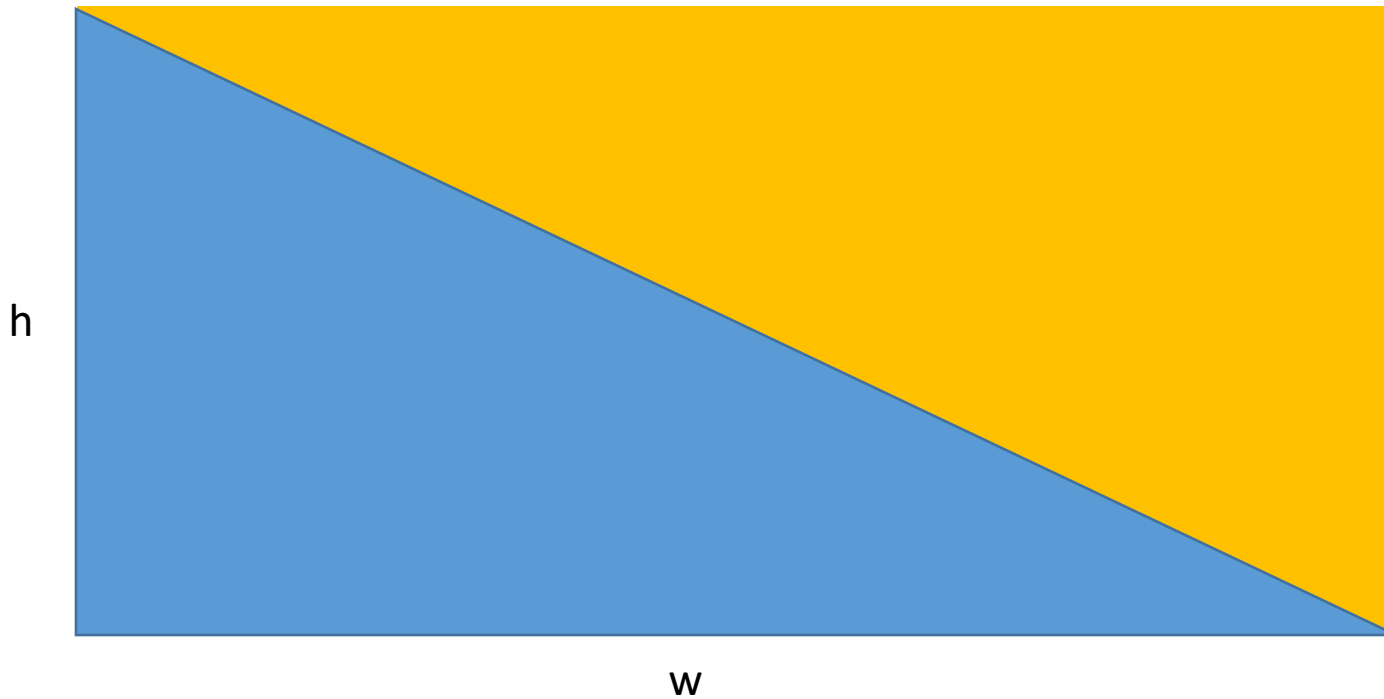
GÉNÉRER DES POINTS ALÉATOIRES

- À l'intérieur d'un triangle?



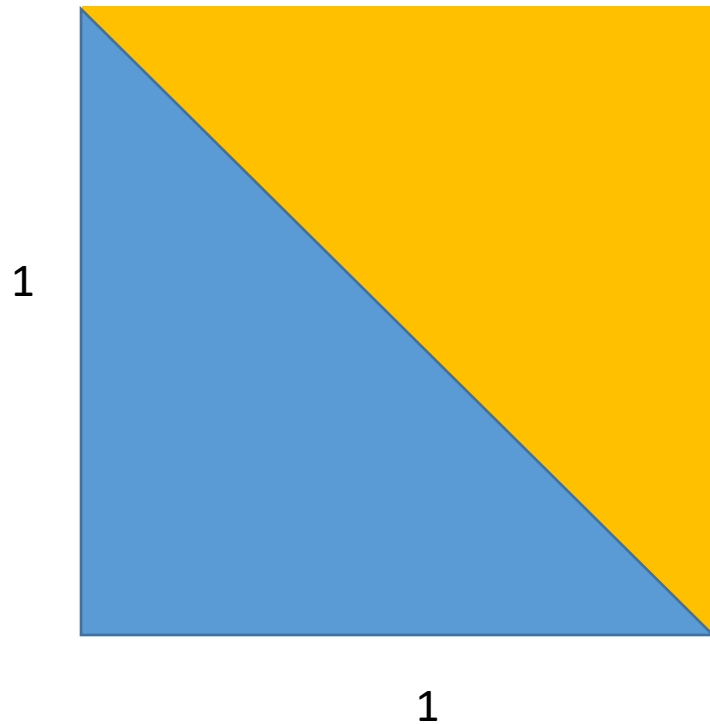
GÉNÉRER DES POINTS ALÉATOIRES

- À l'intérieur d'un triangle rectangle?



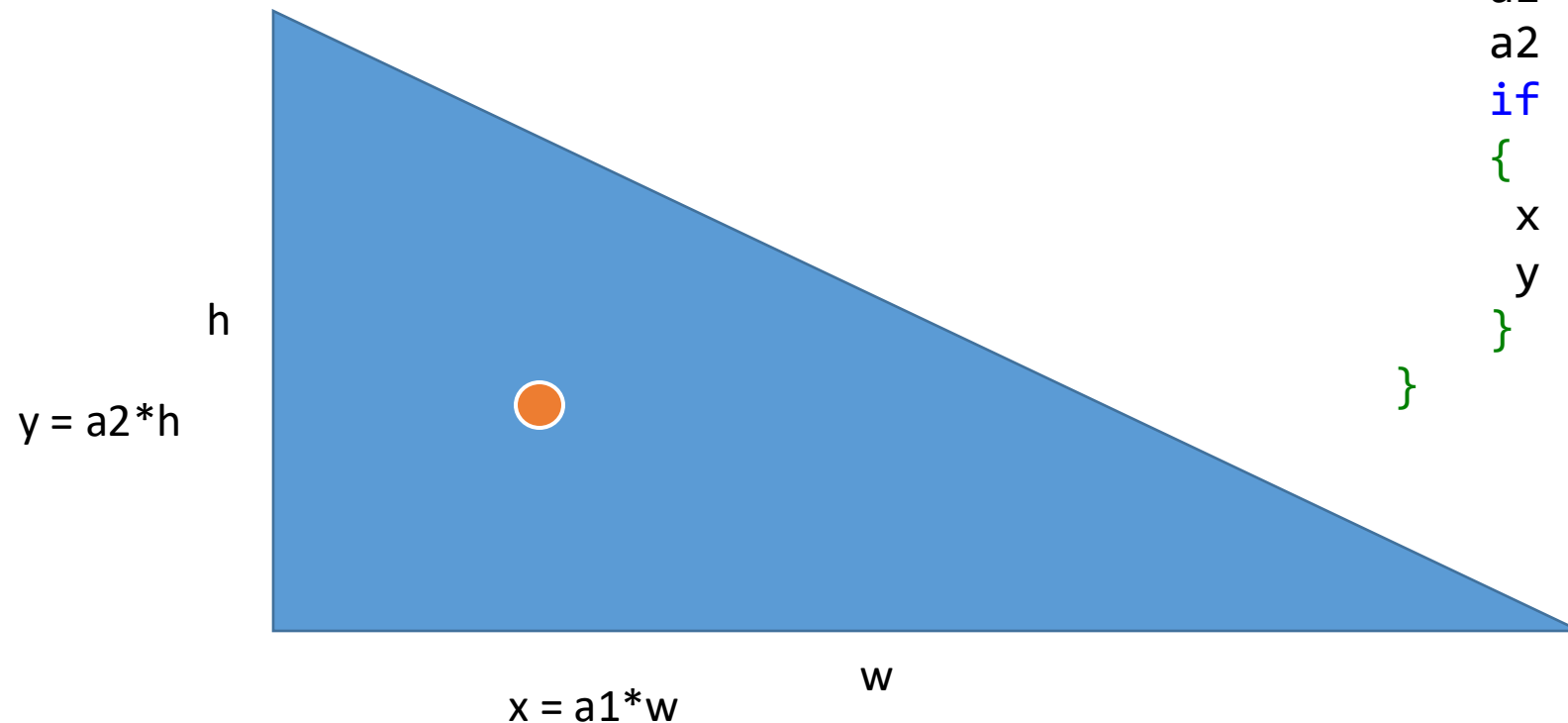
GÉNÉRER DES POINTS ALÉATOIRES

- À l'intérieur d'un triangle rectangle?



GÉNÉRER DES POINTS ALÉATOIRES

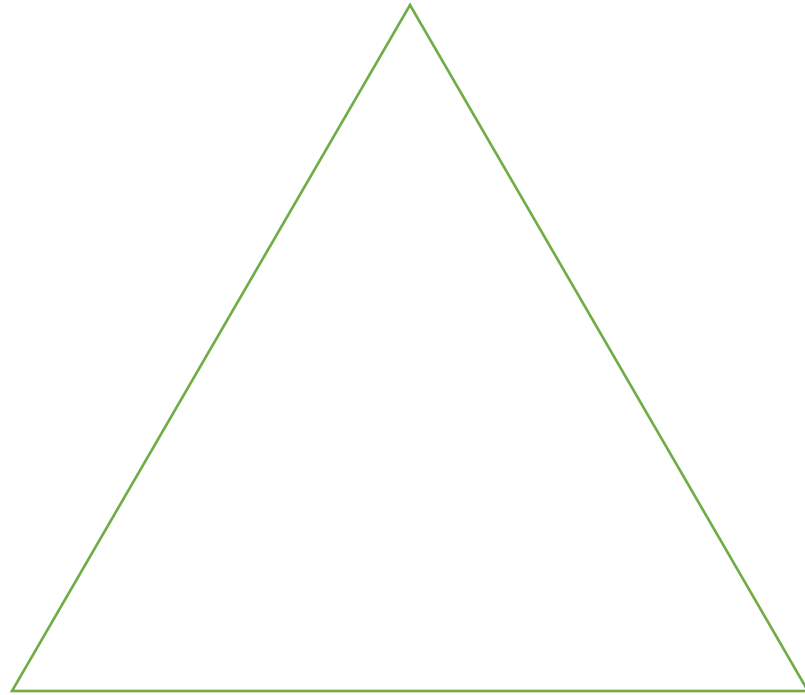
- À l'intérieur d'un triangle rectangle?



```
for (i=0...N)
{
    a1 = rand();
    a2 = rand();
    if (a1+a2<1)
    {
        x = a1*w;
        y = a2*h;
    }
}
```

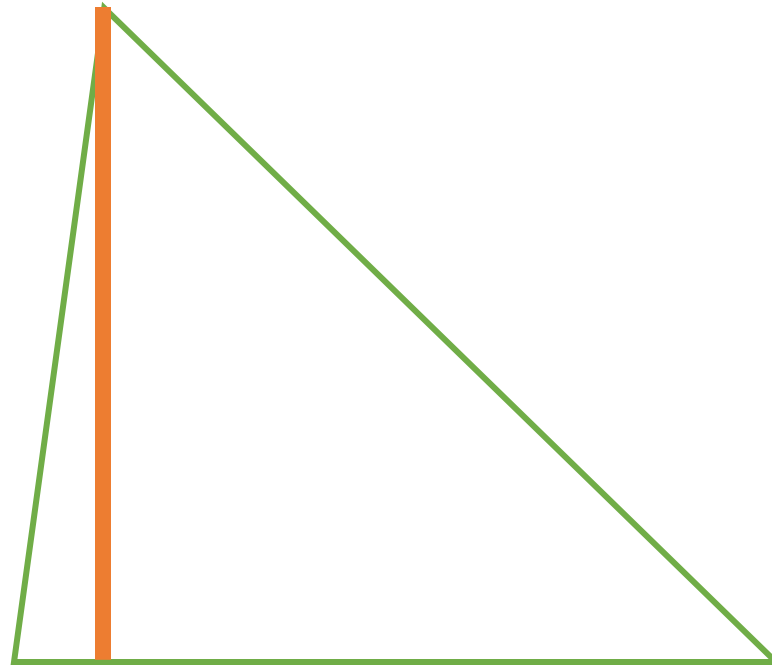
GÉNÉRER DES POINTS ALÉATOIRES

- À l'intérieur d'un triangle?



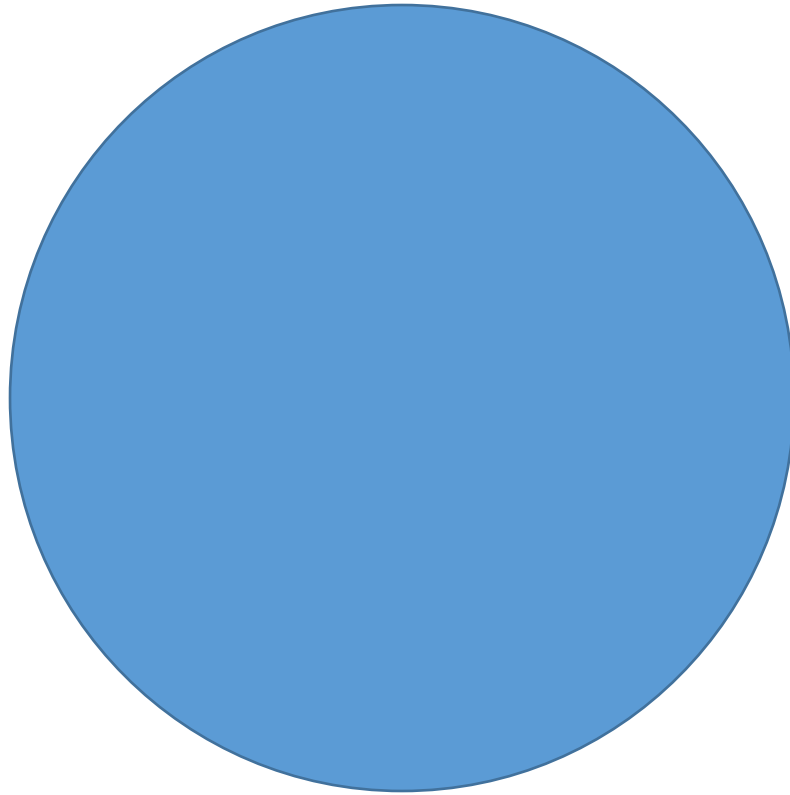
GÉNÉRER DES POINTS ALÉATOIRES

- À l'intérieur d'un triangle?



GÉNÉRER DES POINTS ALÉATOIRES

- À l'intérieur d'un cercle?

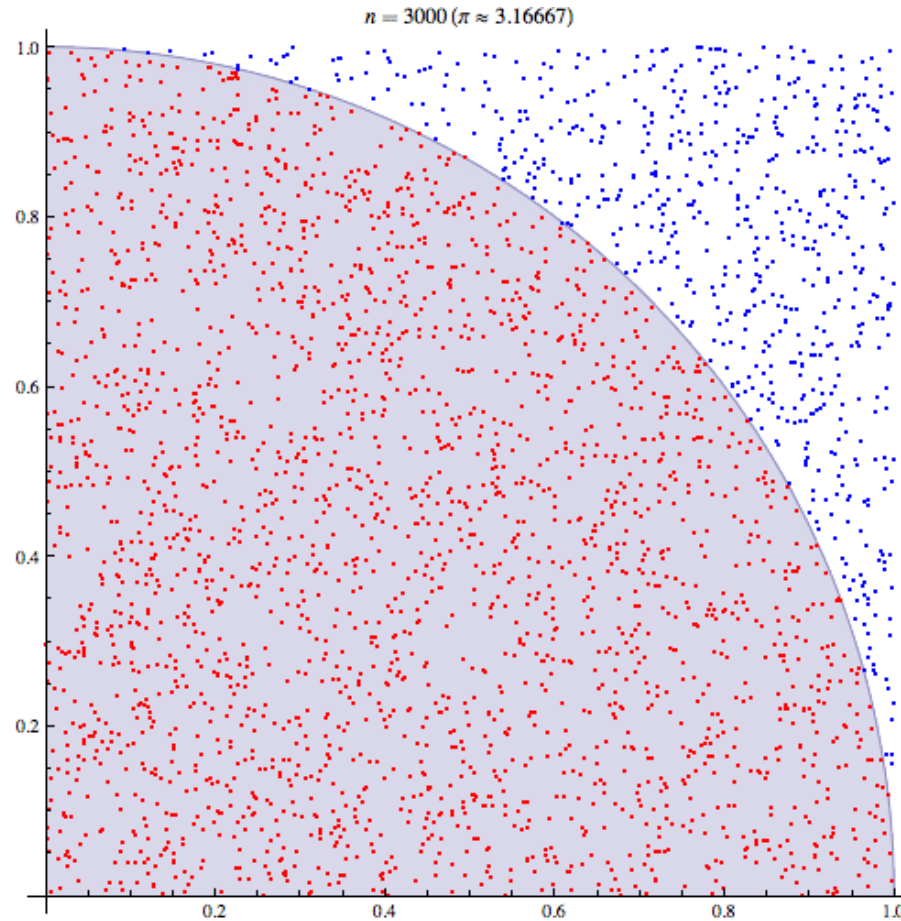


L'ÉCHANTILLONNAGE DE REJET

- Soit une forme compliquée et on veut générer des points dedans
- Une façon élégante est l'échantillonnage de rejet:
 - Générer un point dans une boîte englobante
 - S'il n'est pas dedans, rejeter
- Combien de points seront rejetés (le ratio)?

LES MÉTHODES DE MONTE CARLO

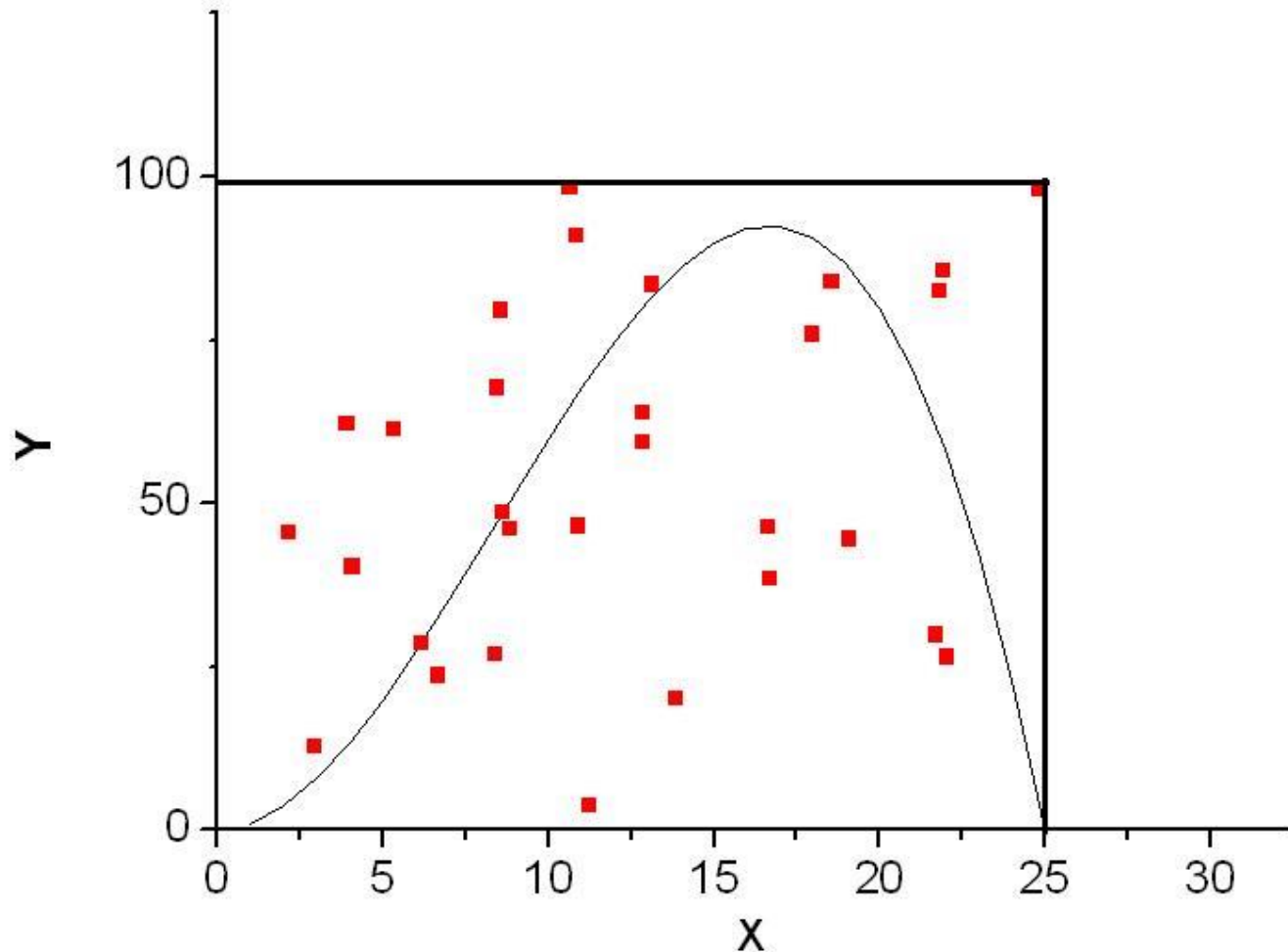
- L'exemple: approcher la valeur de π :



"Pi 30K" by CaitlinJo - Own work. This mathematical image was created with Mathematica. Licensed under CC BY 3.0 via Commons - https://commons.wikimedia.org/wiki/File:Pi_30K.gif#/media/File:Pi_30K.gif

LES MÉTHODES DE MONTE CARLO

- L'exemple: calculer une intégrale difficile



"Pi 30K" by CaitlinJo - Own w

https://commons.wikimedia.org/wiki/File:Pi_30K.gif#/media/File:Pi_30K.gif

LANCER DE RAYONS: MONTE CARLO

- Dans le lancer de rayons, on peut générer les rayons aléatoirement depuis une source de lumière surfacique
- Au lieu de générer d'un seul rayon d'ombre
 - Générer plusieurs rayons vers la lumière aléatoirement
 - Générer un point sur la lumière
 - Lancer un rayon vers ce point
 - Faire la moyenne de leurs contributions
- Les ombres douces en lancer de rayons!

TRACER DE CHEMINS: ALGORITHME DE BASE

```
PathTrace(Ray r) {  
    P = closestIntersection(r);  
    if (random(emit, reflect) == emit)  
        return EmissionColor;  
    else {  
        Ray v = {intersectionPt,  
            randomDirectionInHemisphere(r.normalWhereObjWasHit)};  
        double cos_theta = dot(v.direction, r.normalWhereObjWasHit);  
        return PathTrace(v)*cos_theta*reflectance;  
    }  
}
```

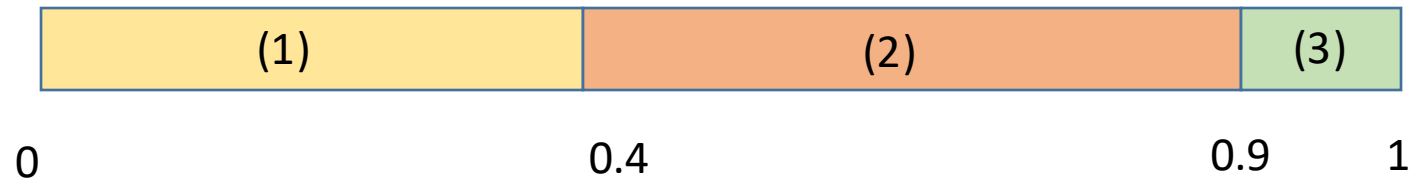
COMMENT CHOISIR PARMI 3 ACTIONS

1. “Aller écouter IFT 3355”, $P = 0.4$
2. “Déjeuner”, $P = 0.5$
3. “Dormir tard”, $P = 0.1$

COMMENT CHOISIR PARMI 3 ACTIONS

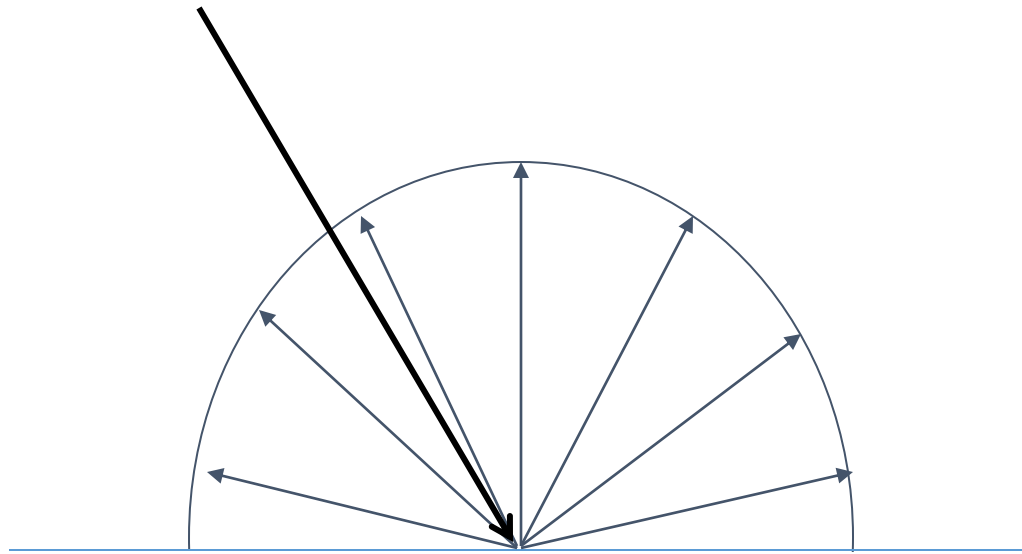
1. “Aller écouter IFT 3355”, $P = 0.4$
2. “Déjeuner”, $P = 0.5$
3. “Dormir tard”, $P = 0.1$

```
x = random();  
if (x < 0.4)  
    return 1;  
else if (x < 0.9)  
    return 2;  
else  
    return 3;
```



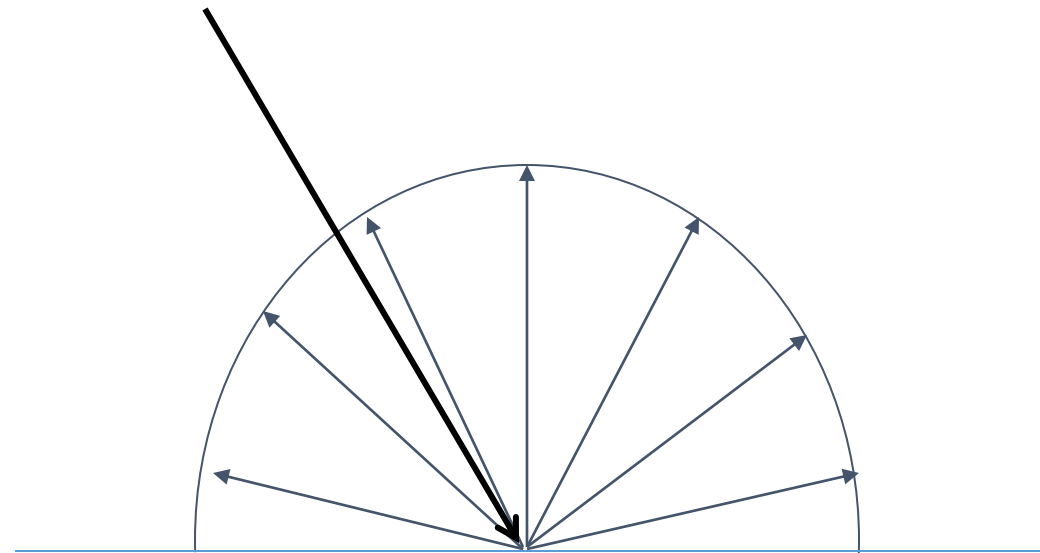
COMMENT GÉNÉRER UNE DIRECTION SUR UN HÉMISPÈRE ALÉATOIREMENT?

- Uniformément?



COMMENT GÉNÉRER UNE DIRECTION SUR UN HEMISPHERE ALÉATOIREMENT?

- Uniformément?
- Option 1:
 - $\phi = \text{random}(1.0);$
 - $\theta = \text{random}(1.0);$
 - $$\text{dir} = \begin{pmatrix} \cos(\theta) \sin(\phi) \\ \sin(\theta) \sin(\phi) \\ \cos(\theta) \end{pmatrix}$$



COMMENT GÉNÉRER UNE DIRECTION SUR UN HEMISPPHÈRE ALÉATOIREMENT?

- Uniformément?

- Option 1:

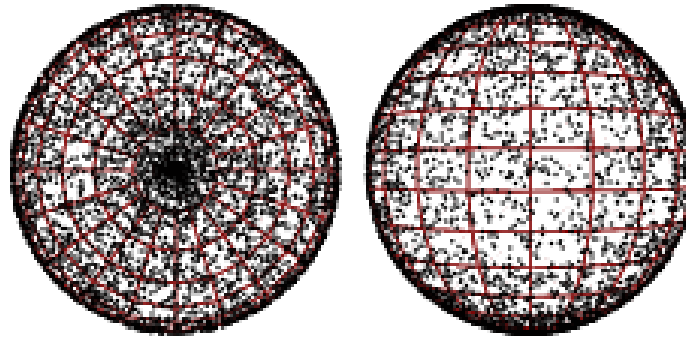
- $\phi = \text{random}(1.0);$

- $\theta = \text{random}(1.0);$

- $$\text{dir} = \begin{pmatrix} \cos(\theta) \sin(\phi) \\ \sin(\theta) \sin(\phi) \\ \cos(\theta) \end{pmatrix}$$

top view

side view



incorrectly distributed points

- Ceci n'est pas une distribution uniforme!

COMMENT GÉNÉRER UNE DIRECTION SUR UN HEMISPPHÈRE ALÉATOIREMENT?

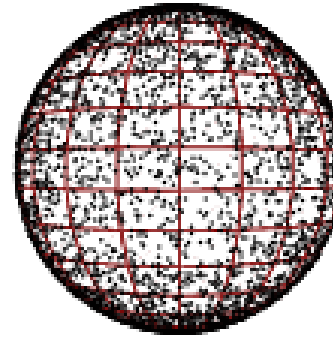
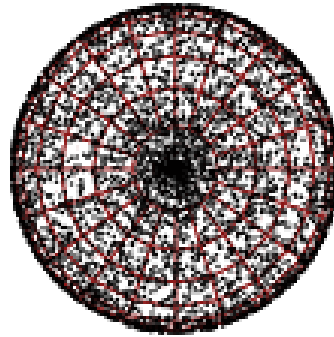
- Uniformément?

• ~~Option 1:~~

- ~~$\phi = \text{random}(1.0);$~~
- ~~$\theta = \text{random}(1.0);$~~
- ~~$$\text{dir} = \begin{pmatrix} \cos(\theta) \sin(\phi) \\ \sin(\theta) \sin(\phi) \\ \cos(\theta) \end{pmatrix}$$~~

top view

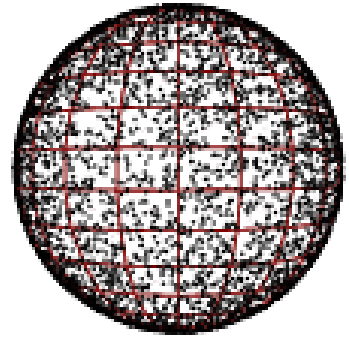
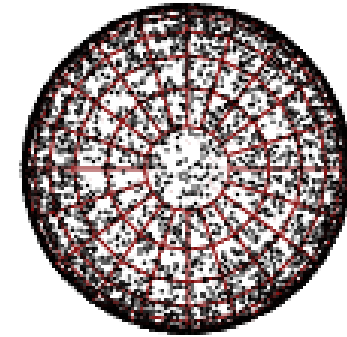
side view



incorrectly distributed points

top view

side view



correctly distributed points

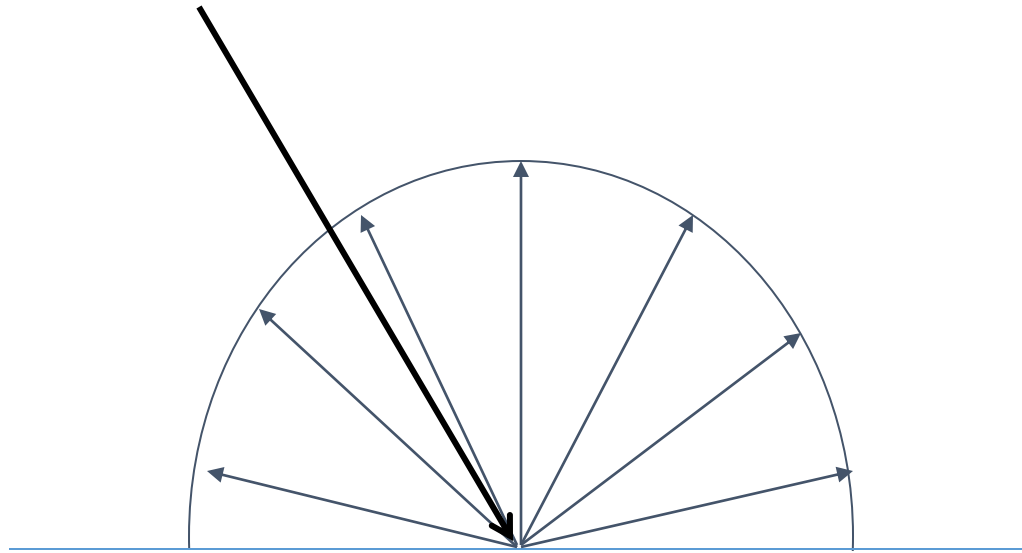
• Option 2:

- Générer z uniformément
- Générer θ uniformément

- $$\text{dir} = \begin{pmatrix} \sqrt{1 - z^2} \cdot \cos(\theta) \\ \sqrt{1 - z^2} \cdot \sin(\theta) \\ z \end{pmatrix}$$

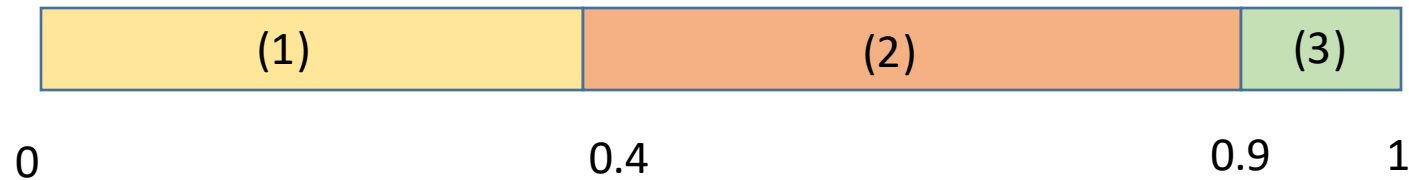
RETOUR À TRACER DE CHEMINS

- Maintenant, on sait comment réfléchir un rayon lorsqu'il intersecte une surface diffuse



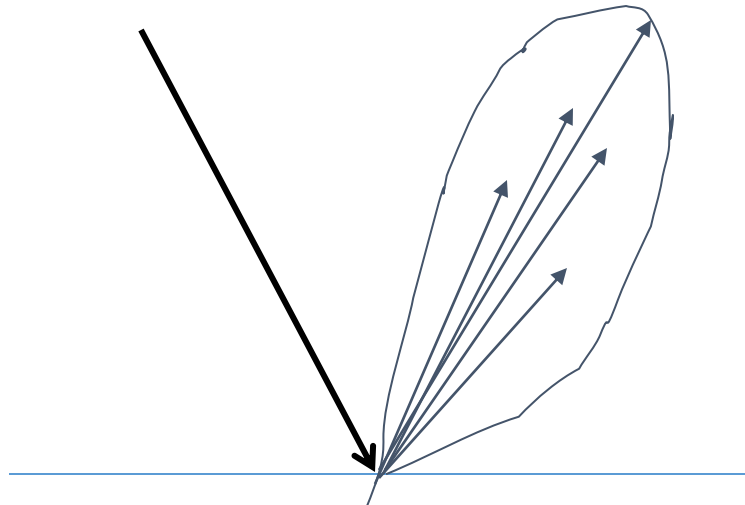
RETOUR À TRACER DE CHEMINS

- Et on sait comment choisir entre les actions « Émettre » et « Disperser »
- De même, on peut ajouter les actions “absorber”, “réfléchir”, “réfracter”

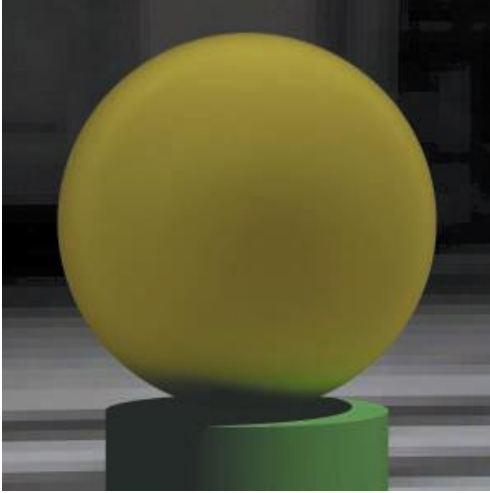


COMMENT GÉNÉRER LES RÉFLEXIONS POUR UNE SURFACE BRILLANTE (*GLOSSY*)

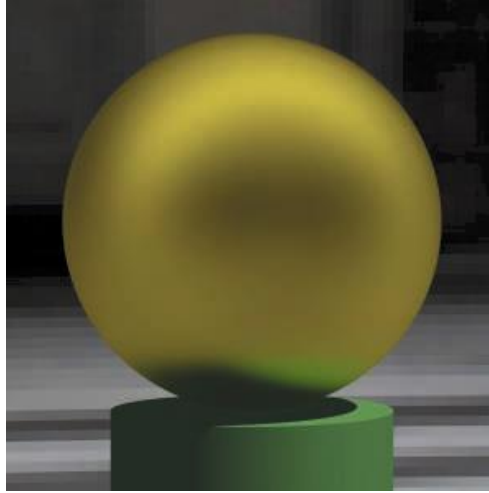
- Choisir aléatoirement entre « diffusion » et « réflexion »
- Si diffusion, utiliser une direction aléatoire sur un hémisphère (uniformément, centré à la normale)
- Si réflexion, utiliser la loi de Snell-Descartes
- Les détails: <https://blog.demofox.org/2020/06/06/casual-shadertoy-path-tracing-2-image-improvement-and-glossy-reflections/>



COMMENT GÉNÉRER LES RÉFLEXIONS POUR UNE SURFACE BRILLANTE



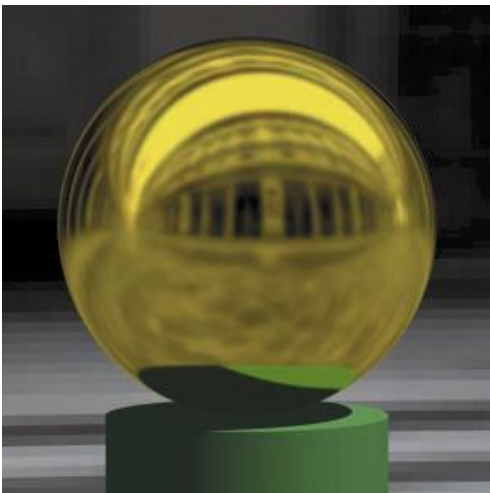
(a)



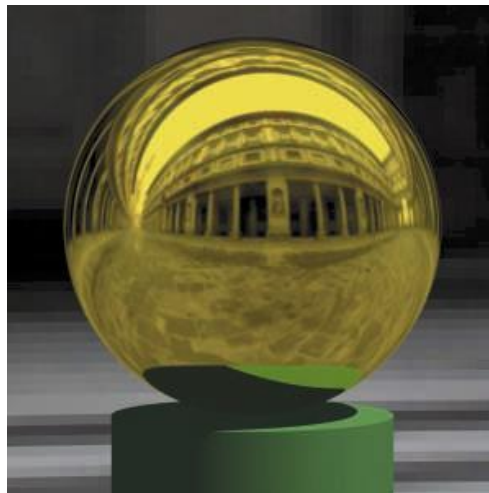
(b)



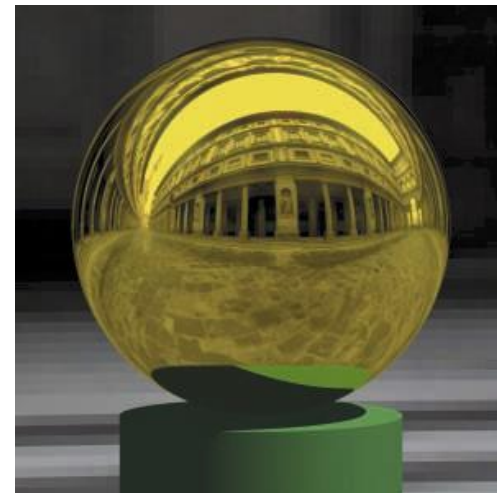
(c)



(d)



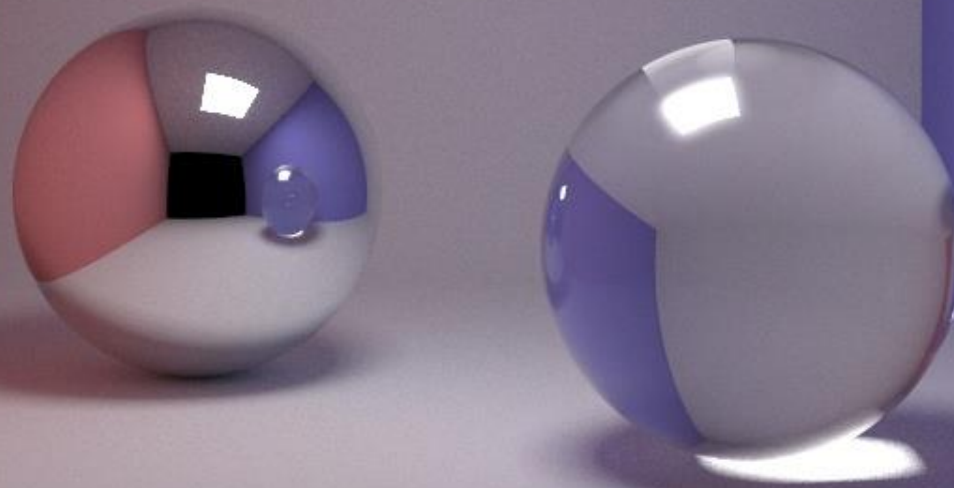
(e)



(f)

RETOUR À TRACER DE CHEMINS

- Et on sait comment choisir entre les actions « Émettre » et « Diffuser »
- De même, on peut ajouter les actions “absorber”, “réfléchir”, “réfracter”



LANCER DE RAYONS VS TRACER DE CHEMINS

- Les algorithmes d'éclairage global
- Les rayons sont tirés de la caméra
- Lancer de rayons
 - Un rayon principal par pixel
 - L'éclairage indirect des surfaces spéculaires seulement
 - Pas de « *color bleeding* »
 - Lancer des rayons d'ombre pour calculer l'éclairage direct
- Tracer de chemins (*plus photoréaliste*)
 - Plusieurs rayons par pixel, leurs contributions forment une moyenne
 - À chaque itération, la direction d'un rayon est aléatoire suivant une certaine distribution
 - Il n'y pas (beaucoup) de différence entre les lumières et les objets
 - Les ombres douces, les matériaux complexes, etc.
 - L'éclairage indirect