

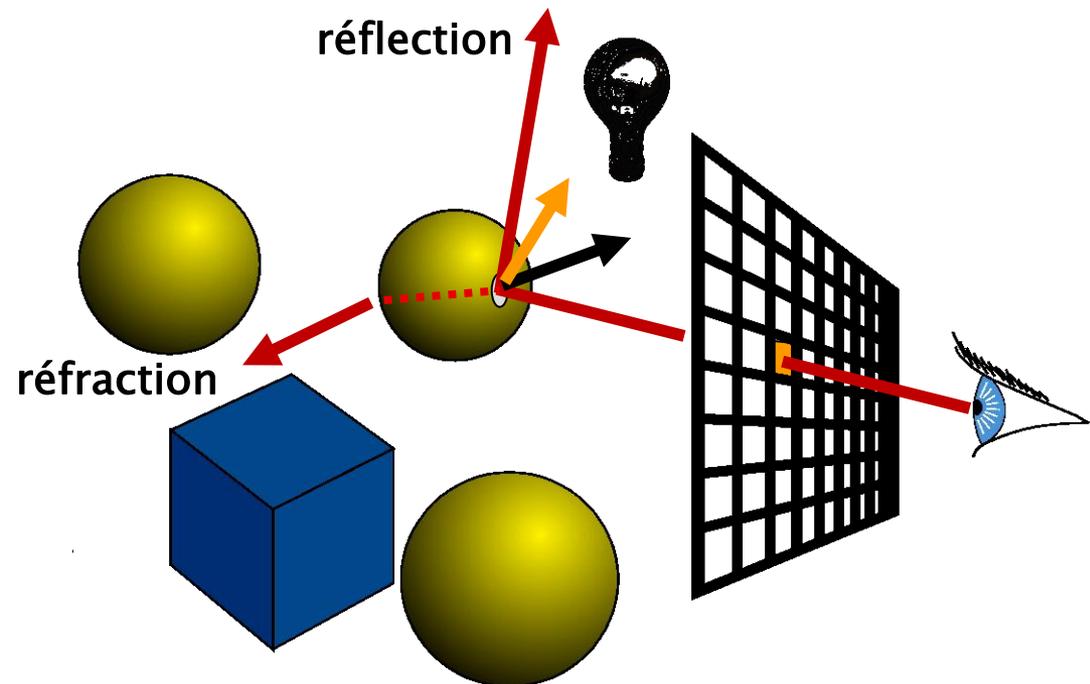
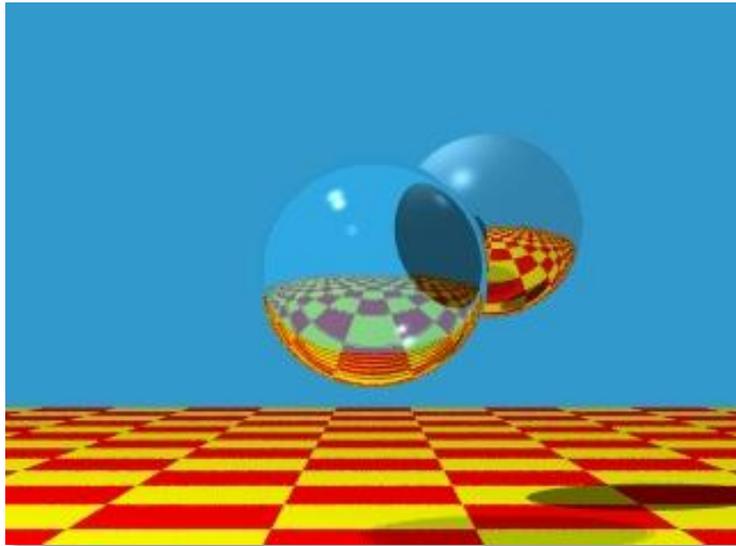
**Lancer de rayons**

# Lancer de rayons

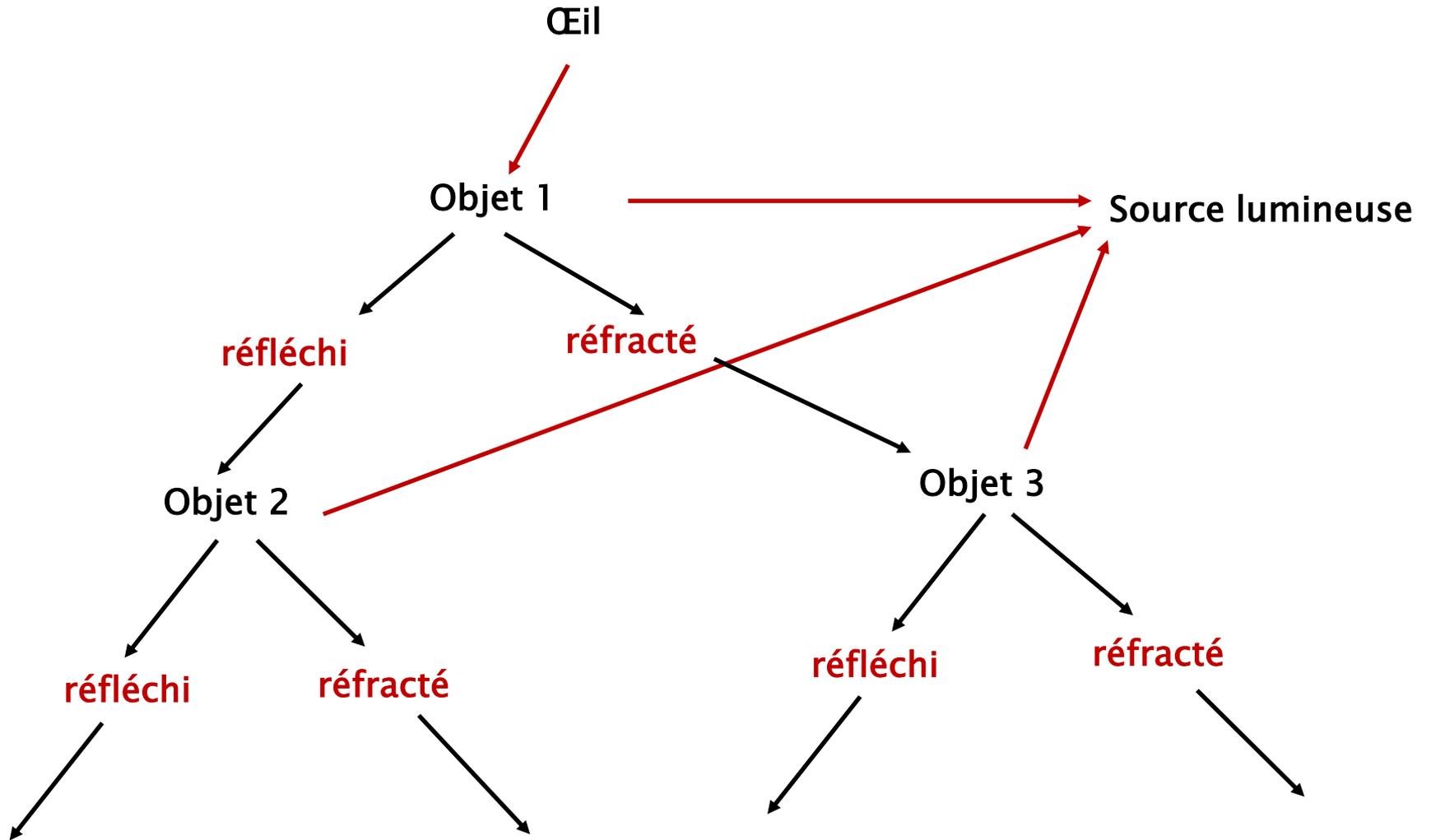
- ▶ **Ray casting, ray tracing** : rayons imaginaires du point d'observation vers la scène.
- ▶ Un rayon est lancé pour chaque pixel de l'image.
- ▶ La couleur du pixel est déterminée à partir de l'illumination du point de la surface intersectée par le rayon.
- ▶ On peut pour cela utiliser les modèles d'illumination locale (Gouraud, Phong).

# Extension du modèle

- ▶ Trois nouveaux rayons sont générés :  
un rayon **réfracté**, un rayon **réfléchi**,  
un rayon **d'ombre**



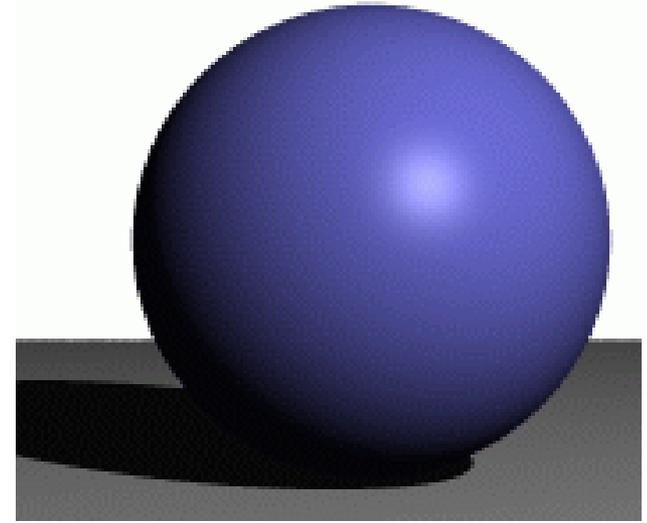
# L'arbre des rayons



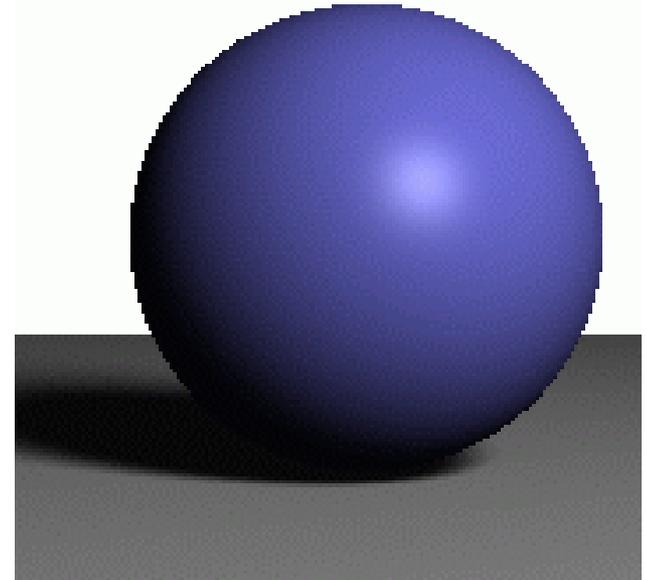
# Encore plus de rayons

- ▶ Ombres douces
  - plusieurs rayons par source de lumière étendue

source ponctuelle



source étendue

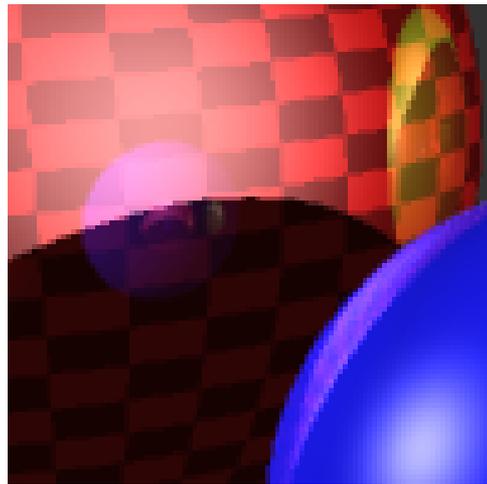


# Encore plus de rayons

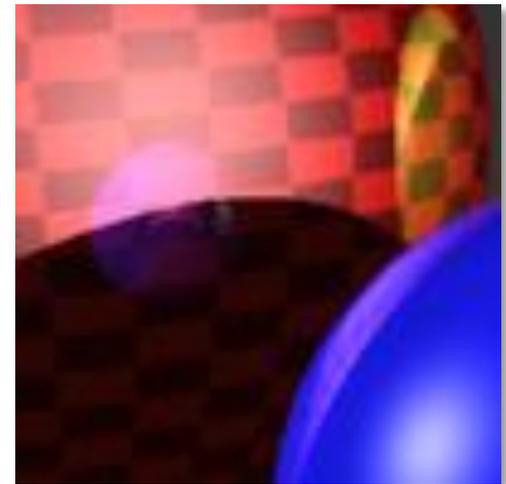
- ▶ Ombres douces
  - plusieurs rayons par source de lumière étendue
- ▶ Anti-aliasing
  - plusieurs rayons par pixel



1 rayon



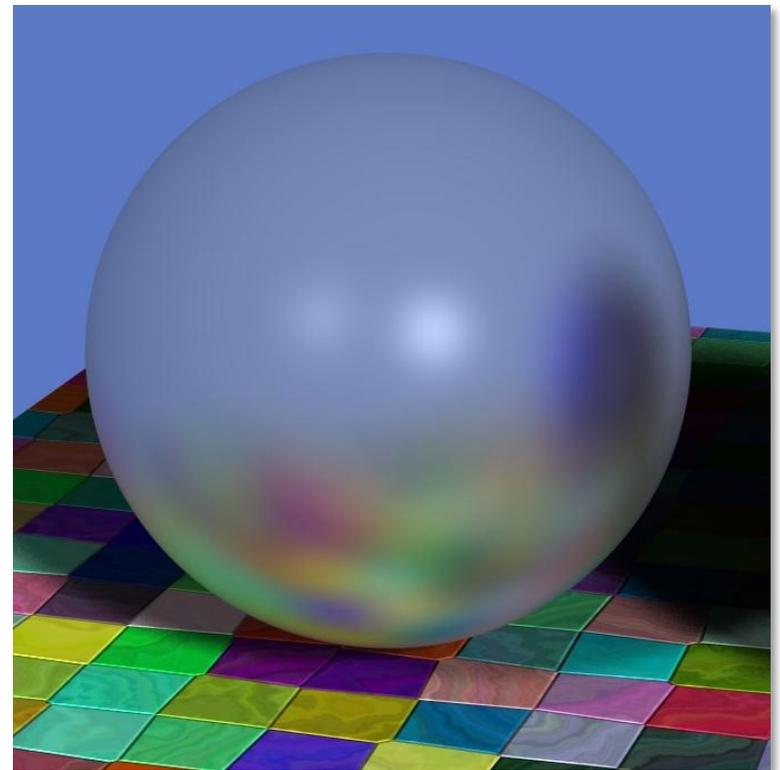
2 rayons



3 rayons

# Encore plus de rayons

- ▶ Ombres douces
  - plusieurs rayons par source de lumière étendue
- ▶ Anti-aliasing
  - plusieurs rayons par pixel
- ▶ Réflexion « glossy »
  - plusieurs rayons réfléchis



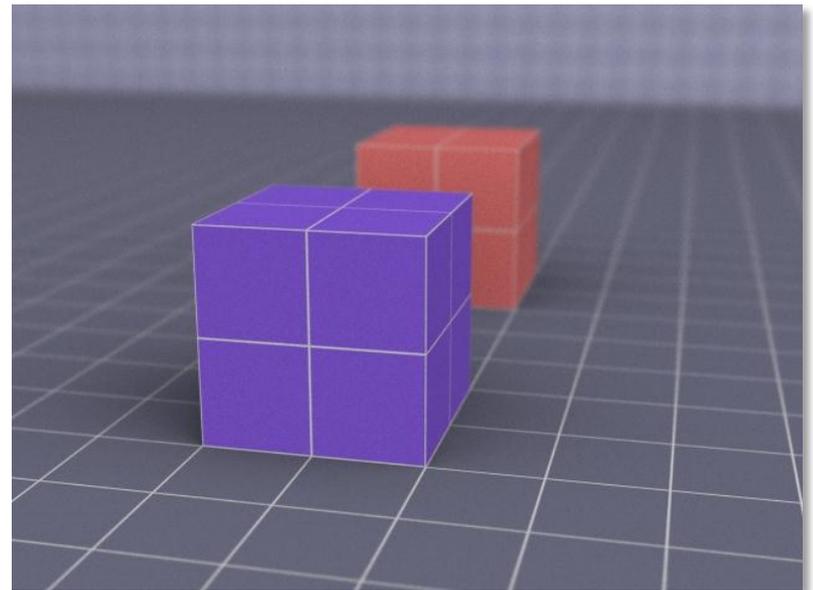
# Encore plus de rayons

- ▶ Ombres douces
  - plusieurs rayons par source de lumière étendue
- ▶ Anti-aliasing
  - plusieurs rayons par pixel
- ▶ Réflexion « glossy »
  - plusieurs rayons réfléchis
- ▶ Flou cinétique
  - plusieurs rayon au cours du temps



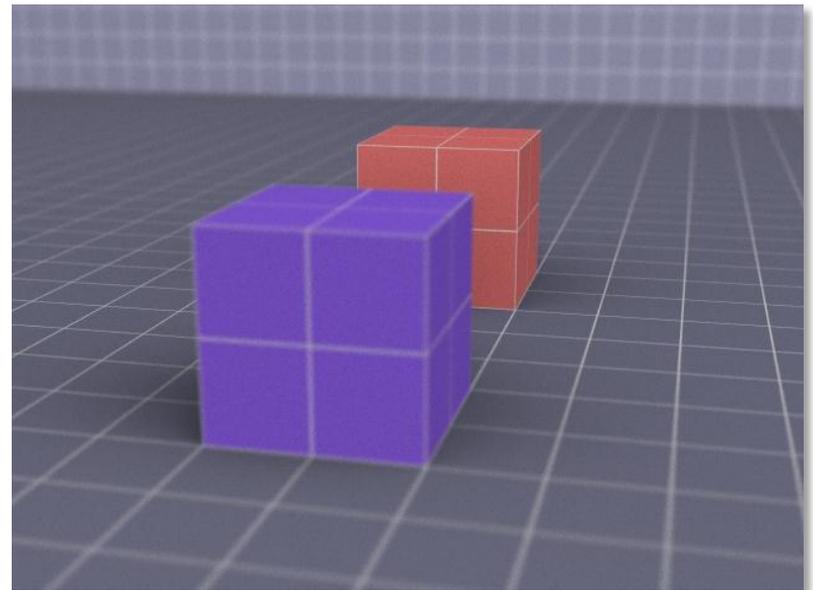
# Encore plus de rayons

- ▶ Ombres douces
  - plusieurs rayons par source de lumière étendue
- ▶ Anti-aliasing
  - plusieurs rayons par pixel
- ▶ Réflexion « glossy »
  - plusieurs rayons réfléchis
- ▶ Flou cinétique
  - plusieurs rayon au cours du temps
- ▶ Profondeur de champ
  - plusieurs rayons par pixel en considérant une lentille



# Encore plus de rayons

- ▶ Ombres douces
  - plusieurs rayons par source de lumière étendue
- ▶ Anti-aliasing
  - plusieurs rayons par pixel
- ▶ Réflexion « glossy »
  - plusieurs rayons réfléchis
- ▶ Flou cinétique
  - plusieurs rayon au cours du temps
- ▶ Profondeur de champ
  - plusieurs rayons par pixel en considérant une lentille



# Intersection rayon-scène

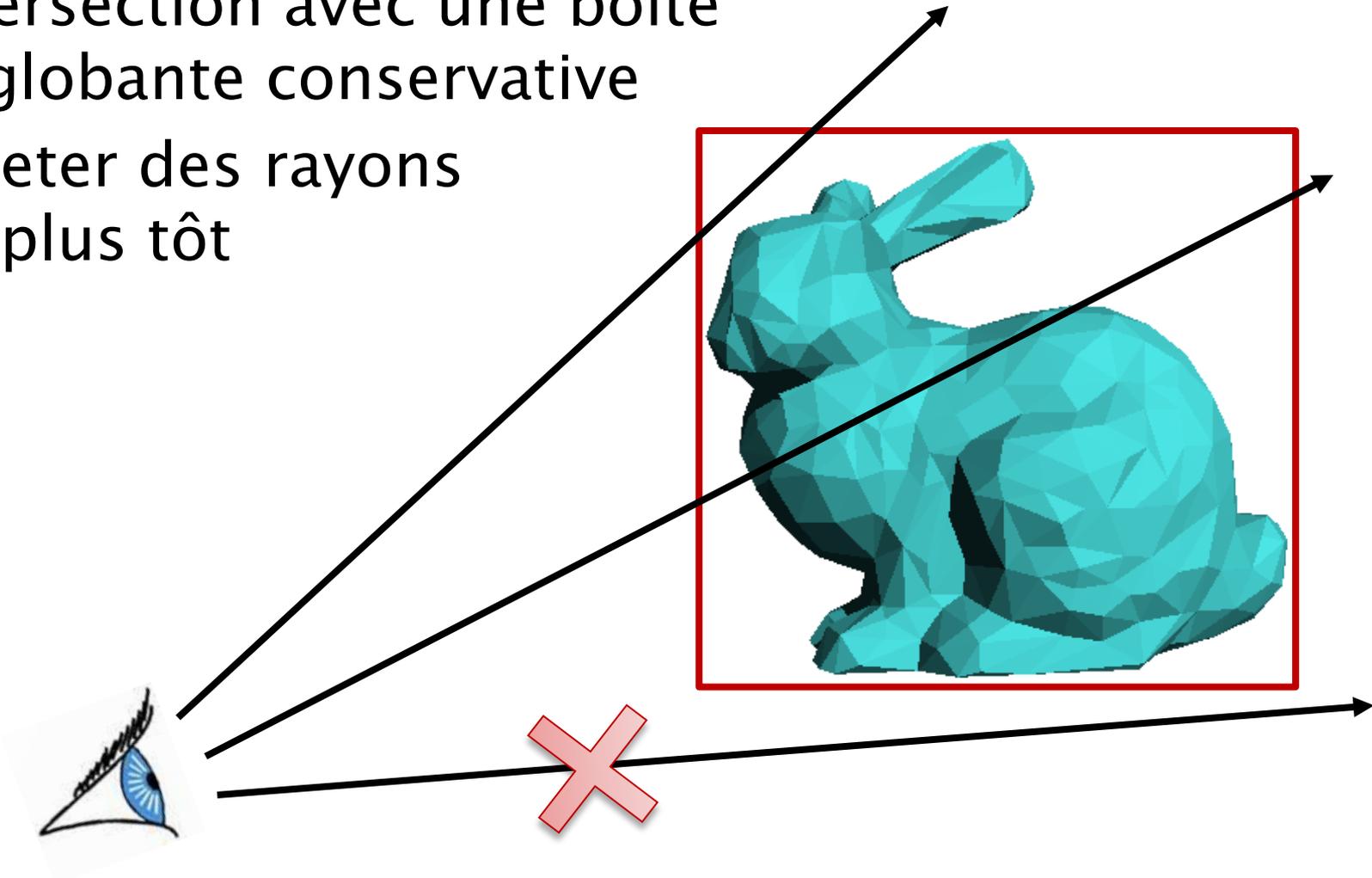
- ▶ Rayon-Sphère : distance point-ligne
- ▶ Rayon-Cylindre : distance ligne-ligne
- ▶ Rayon-Plan : intersection ligne-plan
- ▶ Rayon-Polygone :
  - intersection ligne-plan
  - test si l'intersection est dans le polygone

# Intersection rayon-scène

- ▶ 99 % du temps est utilisé pour les intersections
- ⇒ Accélération : réduire le nombre d'intersection
  - Boîtes englobantes
  - Grilles uniformes (voxels)
  - Grille adaptative : octrees

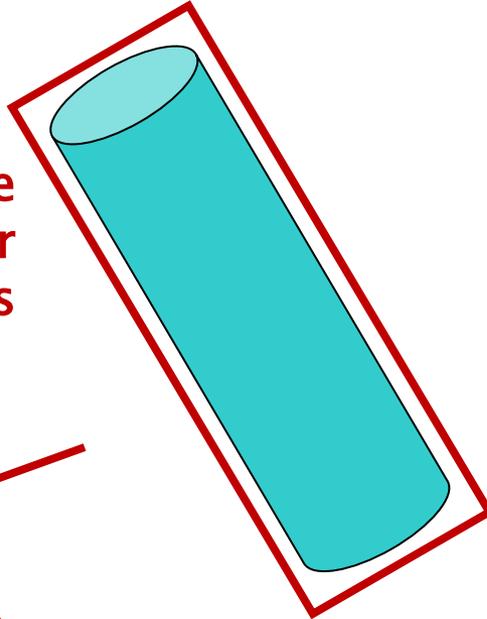
# Boites englobantes

- ▶ Intersection avec une boite englobante conservative
- ▶ Rejeter des rayons au plus tôt

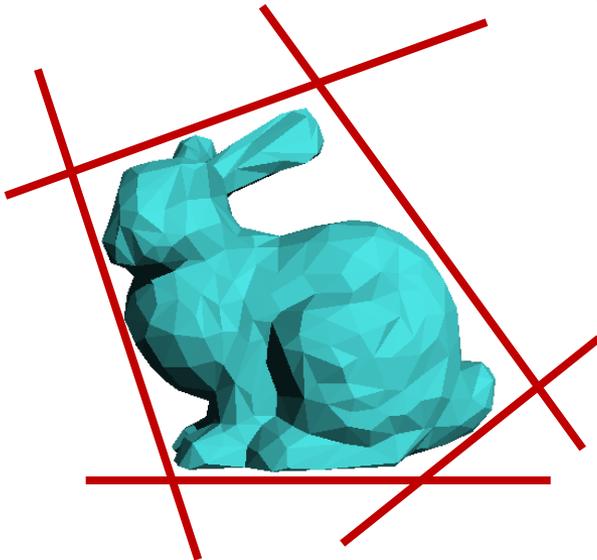


# Boîtes englobantes

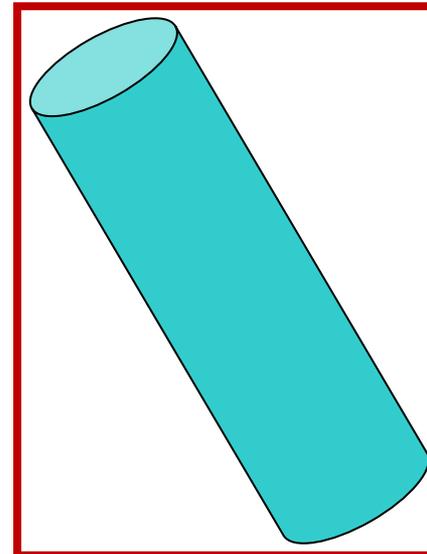
boîte englobante  
non-alignée sur  
les axes



sphère  
englobante

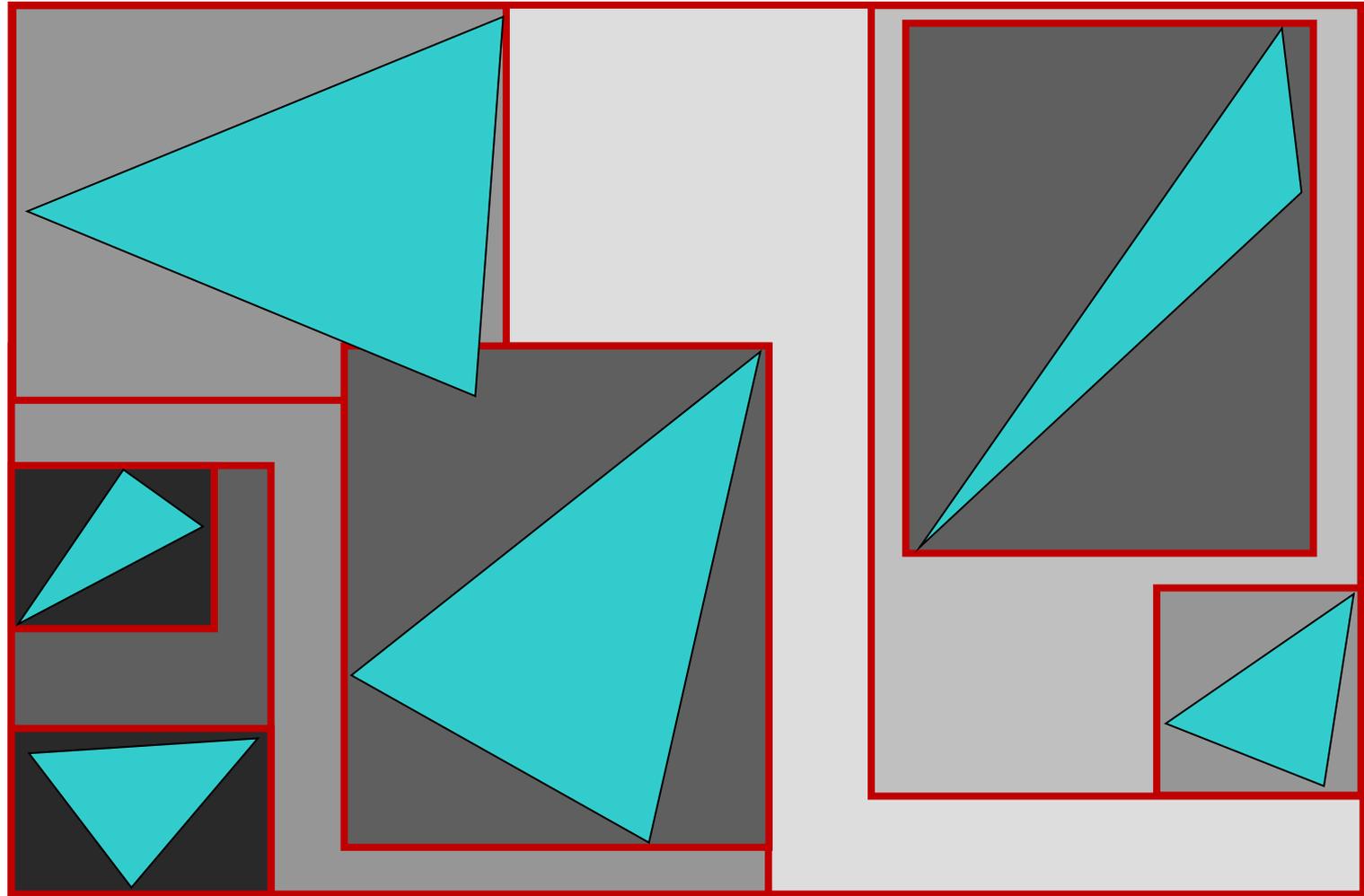


région convexe arbitraire

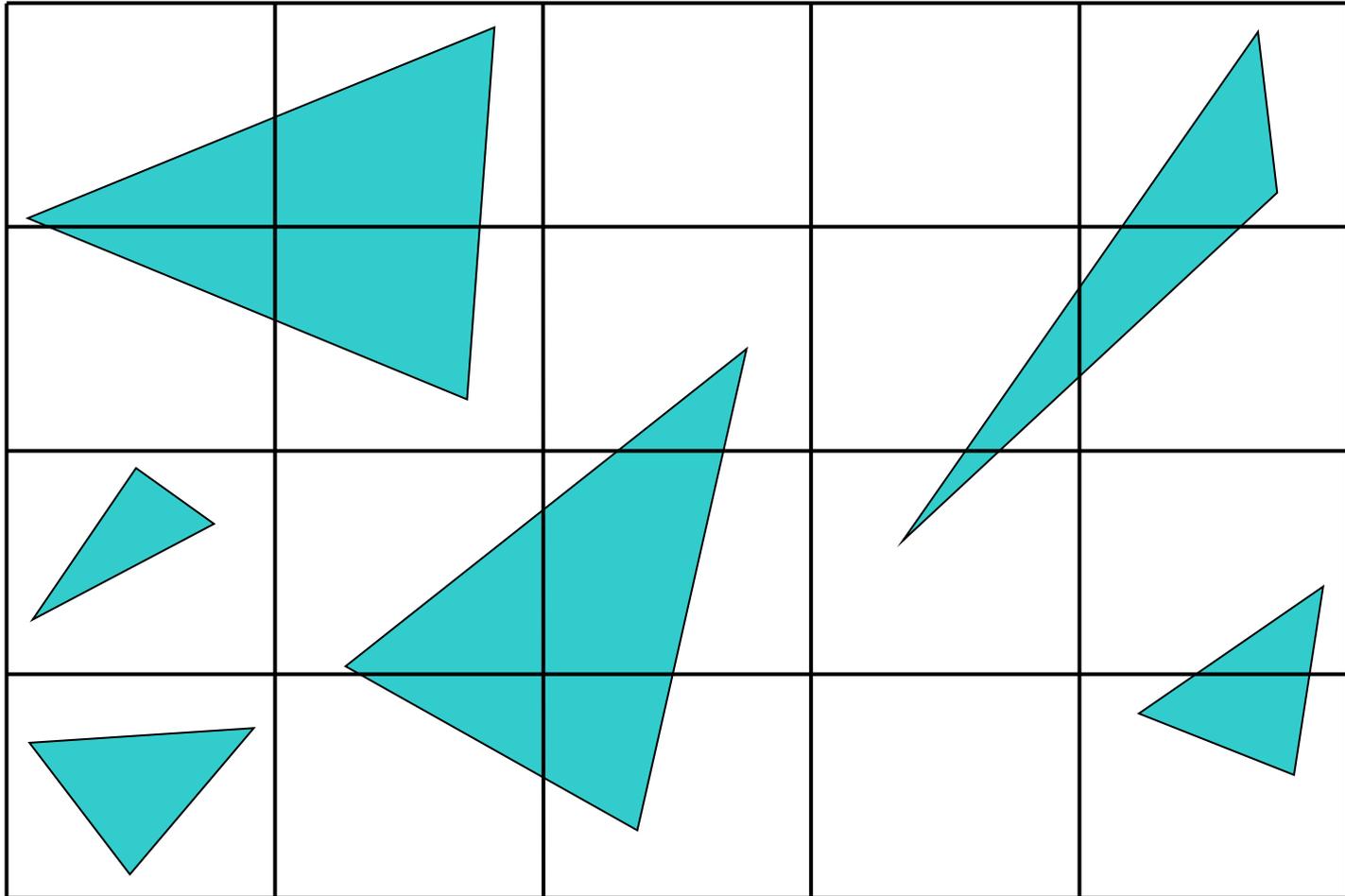


boîte englobante  
alignée sur les  
axes

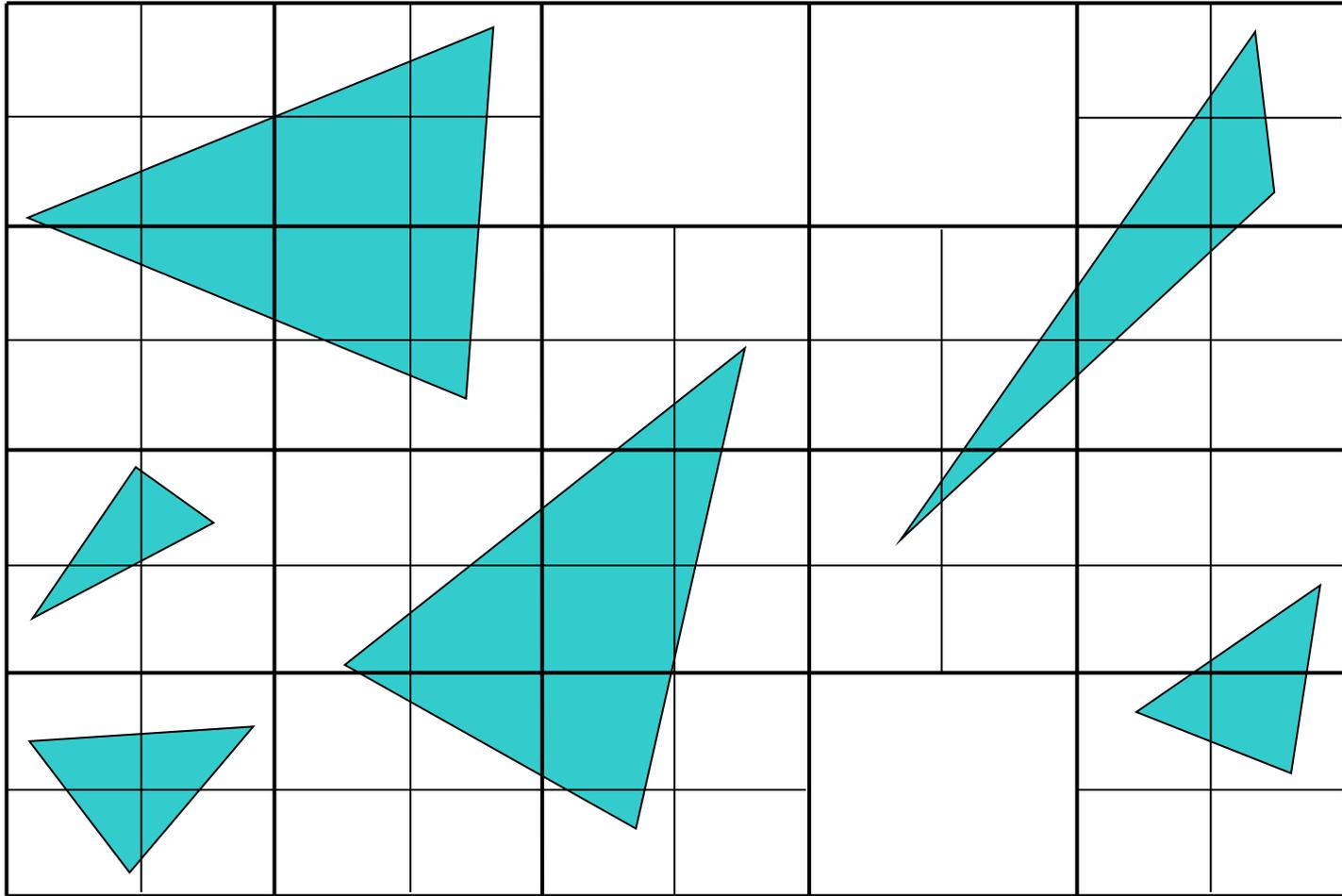
# Hiérarchie de boîtes englobantes



# Grille uniforme



# Grille adaptative : Octree



# Question – 3 mn avec vos voisins

- ▶ Comparer les 3 accélérations :
  - Boites englobantes
  - Grille uniforme
  - Grille adaptative : Octree



# Comparaison

- ▶ Boites englobantes:
  - longue initialisation, requêtes rapides.
- ▶ Grille uniforme :
  - Initialisation rapide, requêtes rapides... si résolution bien choisie.
- ▶ Octrees :
  - Initialisation rapide, requêtes plus longues.

# Lancé de rayons : avantages

- ▶ Lent, mais pas de calculs supplémentaires pour :
  - élimination des parties cachées
  - ombres
  - transparence
  - plaquage de textures (y compris procédurales)
- ▶ Inter-réflexions spéculaires entre objets
- ▶ Primitives graphiques quelconques

# Lancé de rayons : problèmes

- ▶ L'arbre est limité à une certaine profondeur
  - Les objets complexes peuvent avoir un problème (diamant, cristal...)
- ▶ Limité à Snell–Descartes
  - Tous les objets sont métalliques
  - Pas d'inter-réflexion entre objets diffus