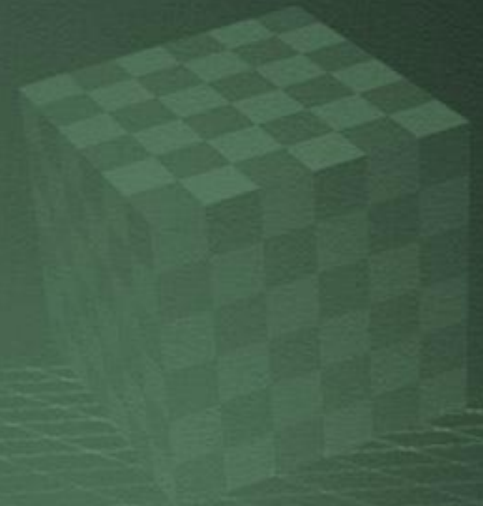


Visualisation de scènes massives et volumiques

Venceslas Biri
Séminaire LIRMM
17/04/13



Problématique générale

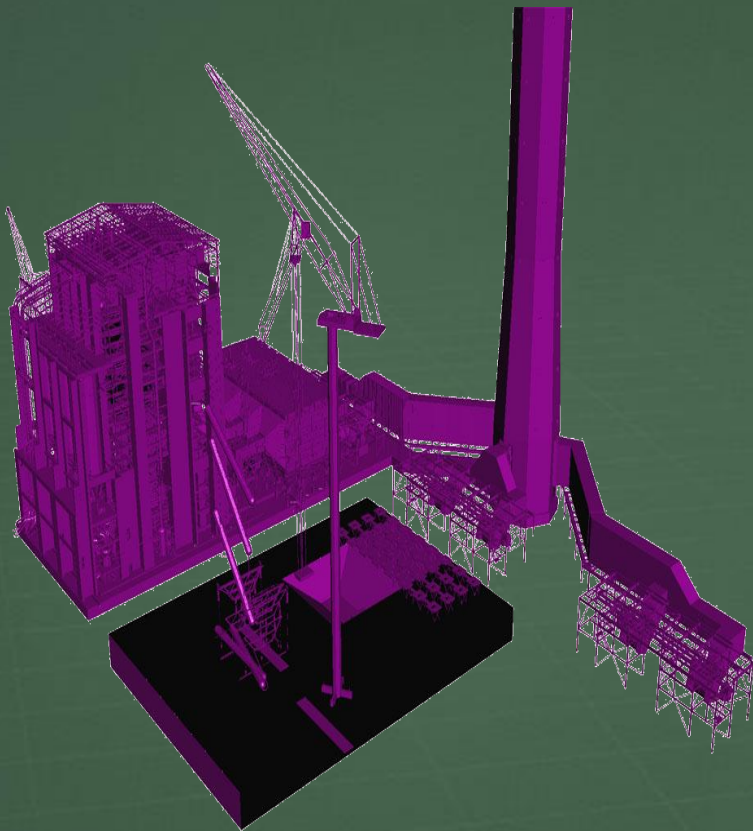
De grosses quantités de données :
exemple du cinéma d'animation



Artwork : Brave

Problématique générale

De grosses quantités de données : géométrie

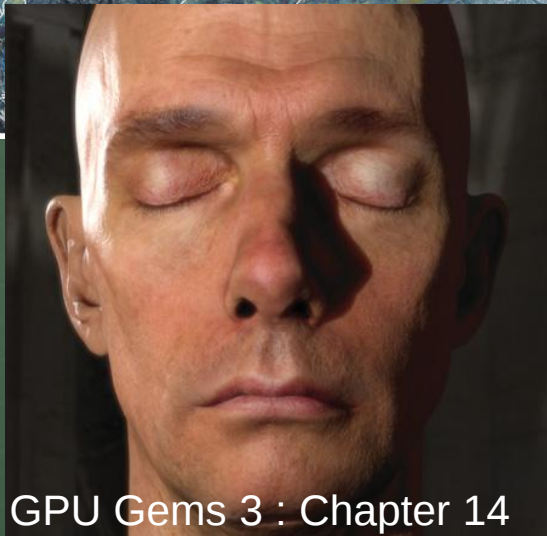
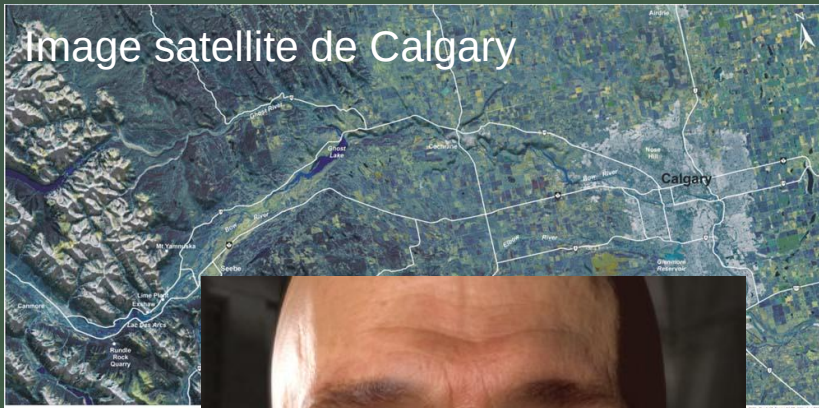


UNC Power Plant
+12 millions triangles
+11 millions sommets



Problématique générale

De grosses quantités de données : textures



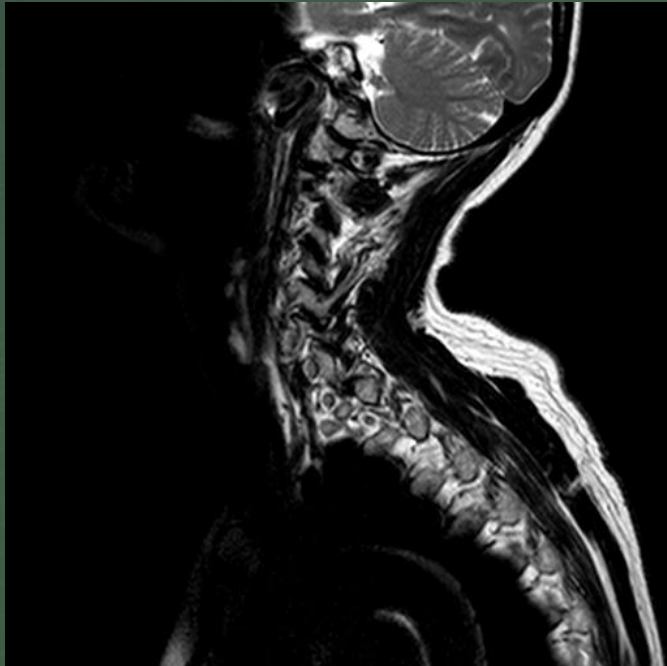
GPU Gems 3 : Chapter 14

"We also kept in mind the workflow of our texture painters, Nick Cosmi and Jennifer Stratton, to ensure they would be able to manage what came out to be over 380 material groups and over 3700 individual texture maps. It was a huge undertaking, not only to obtain a photo realistic look that matched the concept art and feel of a Giant Pacific Octopus, but to also manage the thousands of texture maps ranging from 1k-4k resolutions."

LookDev artist, Dante Quintana, A night in the museum, about the octopodus

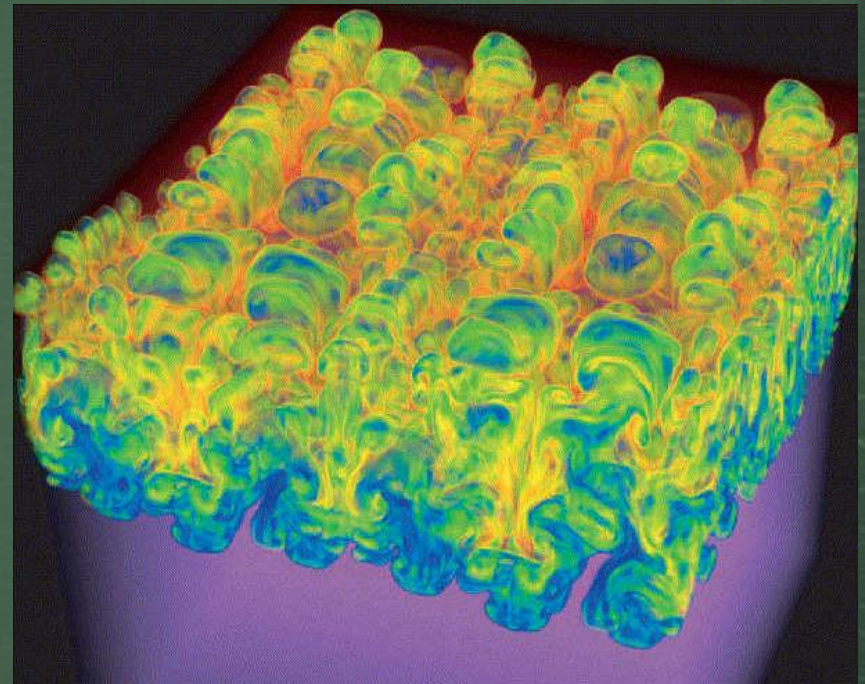
Problématique générale

De grosses quantités de données : volumes



“A typical image file size for MRI is 5-6MB, with 100 to 1,000+ images contained in 4-19 series.”, Telemedicine Article

Un champ vectoriel (en float) dans une grille 512^3 réclame 1,5 Go de mémoire



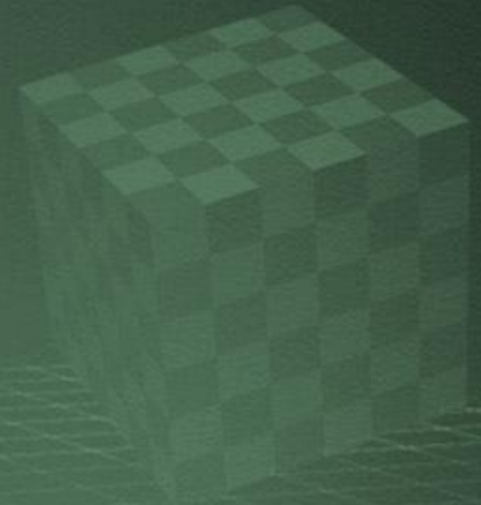
Problématique générale

- Quelques chiffres issus des productions à Ubisoft Motion Pictures :
 - Décors > 8Go
 - Animation > 200 frames / sequence
 - Objets > 5000 / décor
 - Multiples textures >4K / objet
 - Surfaces de subdivision



Problématique générale

- Comment charger les données ?
 - Streaming
 - Compression
 - Early Culling
- Comment visualiser les données ?
 - Qualité du rendu
 - Gérer l'illumination de volume

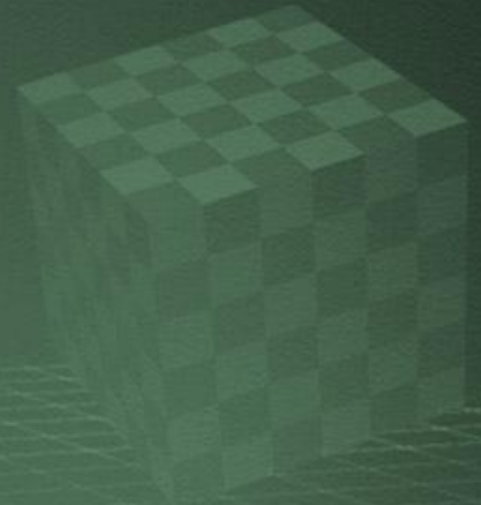


Plan

1. Rendu de données massives
 - a. Streaming de données
 - b. Deferred shading
 - c. Antialiasing
2. Rendu de volumes (milieux participants)
 - a. Différentes approches
 - b. Modélisation par ondelette
 - c. Occlusion Propagation Volume



RENDU DE DONNÉES MASSIVES



Plan

1. Rendu de données massives
 - a. Streaming de données
 - b. Deferred shading
 - c. Antialiasing
2. Rendu de volumes (milieux participants)
 - a. Différentes approches
 - b. Modélisation par ondelette
 - c. Occlusion Propagation Volume



Cas d'étude

- Situation : les données ne « rentrent » pas
 - dans la VRAM du GPU
 - dans la RAM du CPU
- Problématique concrète :
 - Travail des infographistes : aucun prédicat
 - Pas de structure accélératrice des données
 - Problème de la « théière dans un stade »
 - Animation & manipulation de données

⇒ **Nécessité du streaming de données**



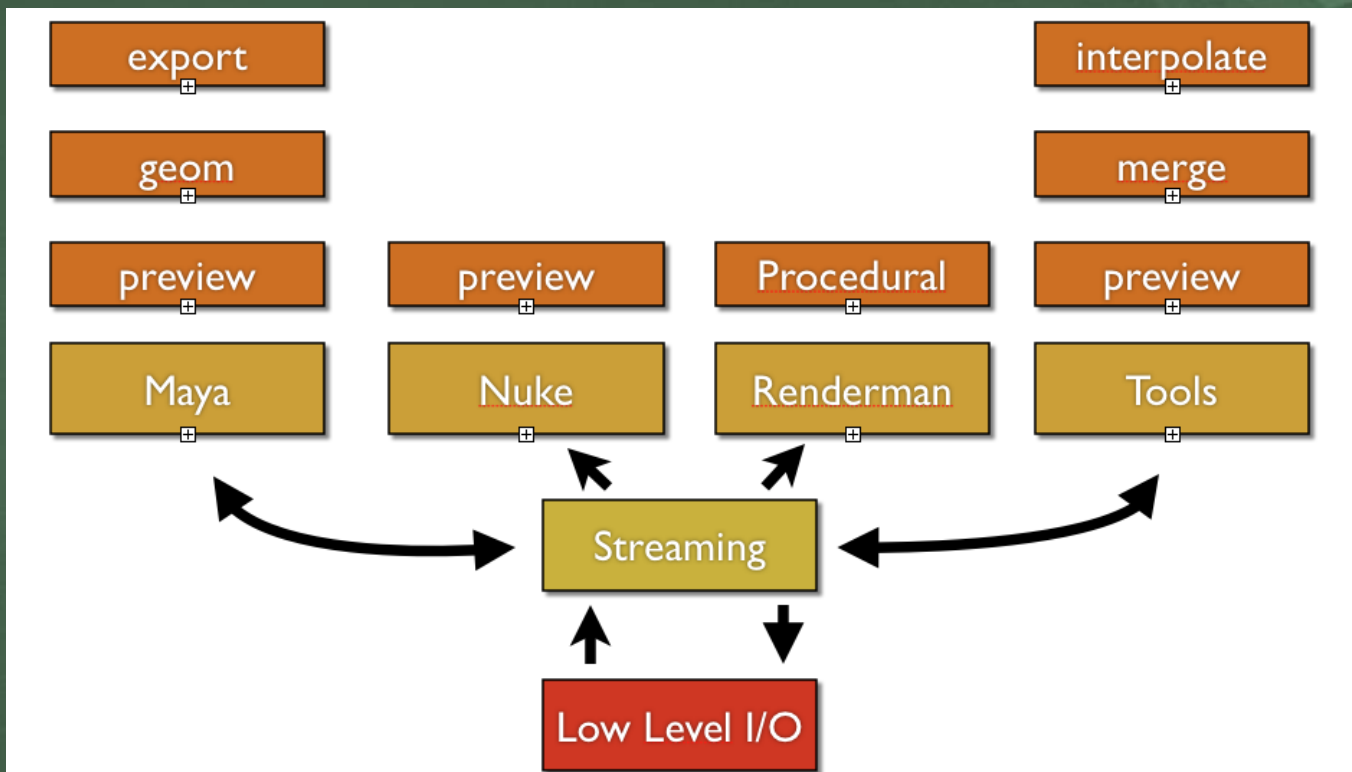
Streaming des données

- Format de fichier pour le streaming : manifesto
 - Stockage complet de la scène
 - Pas de « transformation » à la volée
 - Accès arbitraire aux données
 - Export de scènes
 - Simplicité et généricité
- Pas de format de fichier existant avec ces caractéristiques



Streaming des données

- Notre solution : Drone File
- Usage



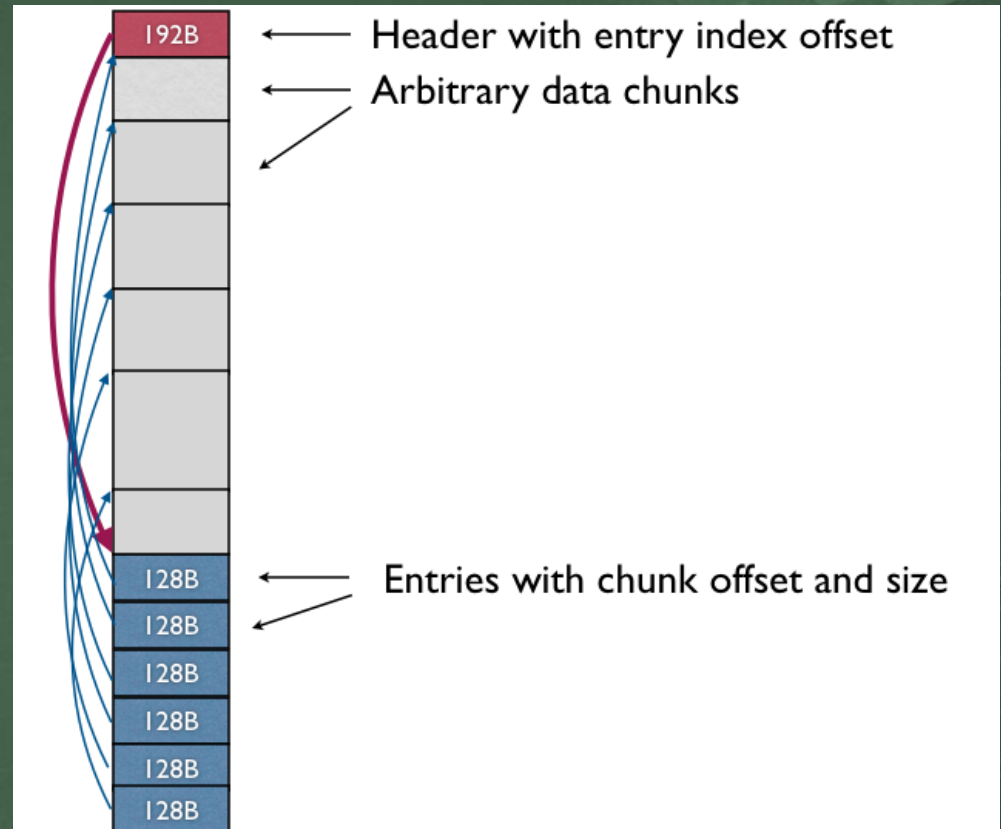
Streaming des données

- Notre solution : 1 API (Drone File Format & Drone API) pour stocker les données à bas niveau

Données binaires

Données représentées
en chunk

Données génériques



Streaming des données

- Notre solution : 1 API pour représenter la scène
 - Chunk order indépendant
 - Aspect sémantique ajouté. Un type de donnée = une structure avec accès aux chunk et chargement en RAM
 - Outils pour :
 - Réarranger les chunks
 - Concaténer plusieurs fichiers
 - Filtrer les animations
 - Analyse de scènes

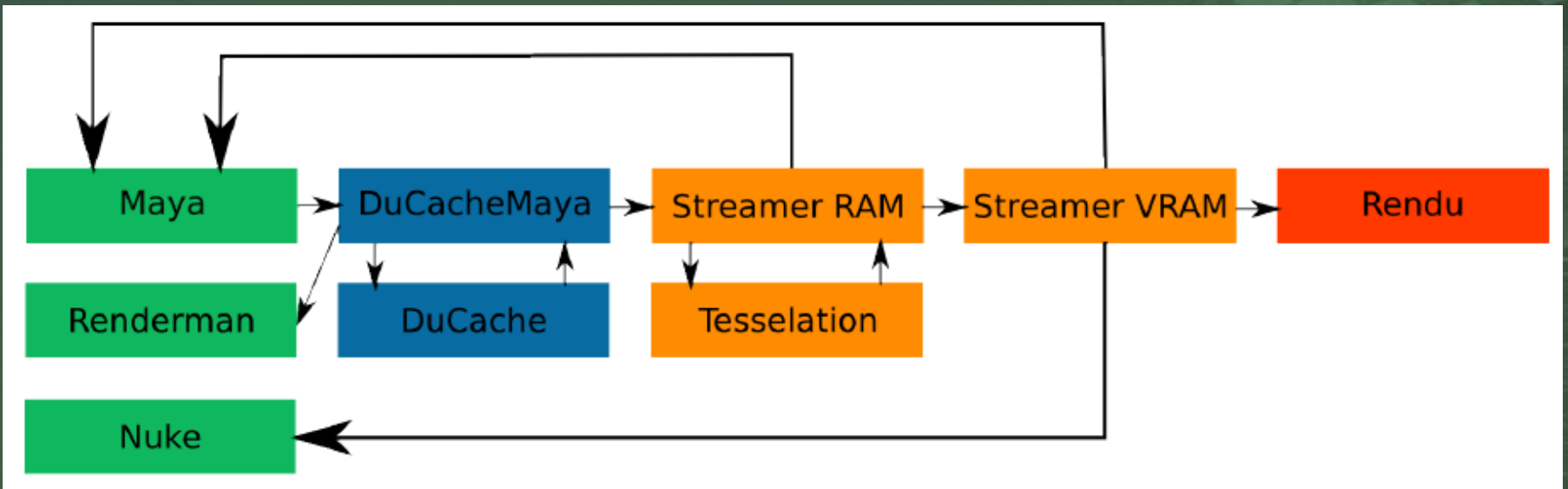


Streaming des données

- Code source sur OpenGL Insight



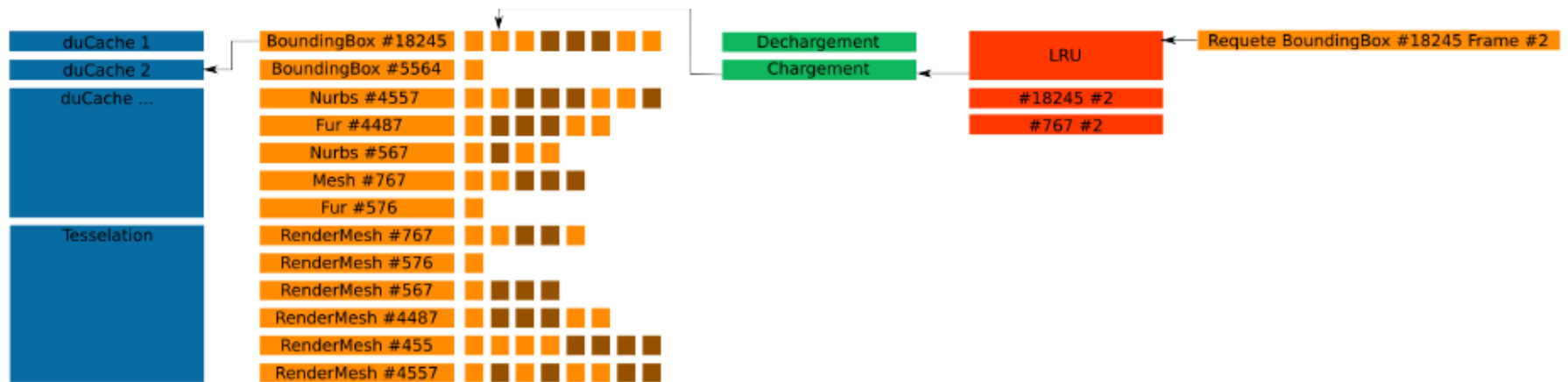
Pipeline



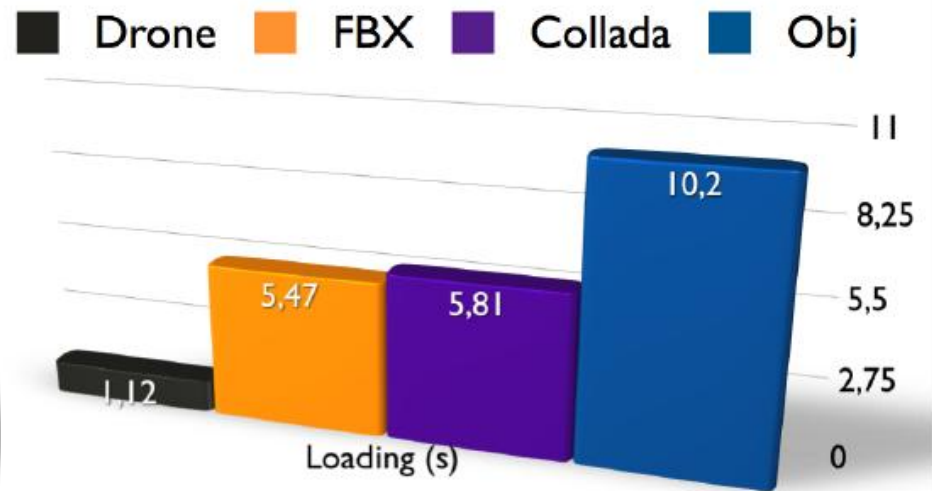
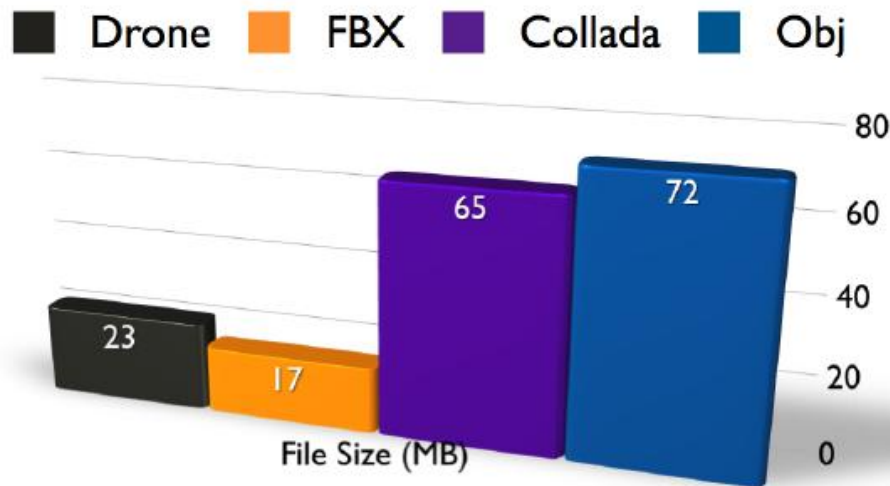
Streaming : chargement



Streaming : chargement



Résultats



Plan

1. Rendu de données massives

- a. Streaming de données
- b. Deferred shading
- c. Antialiasing

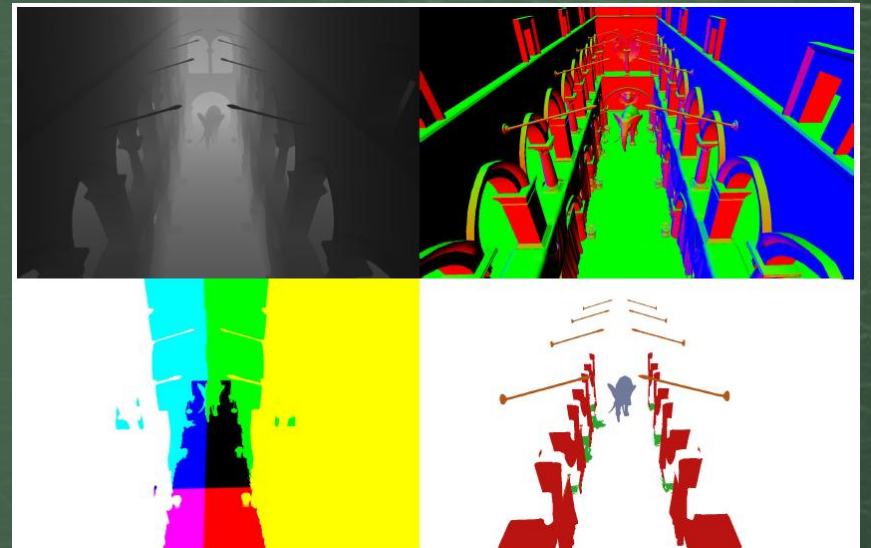
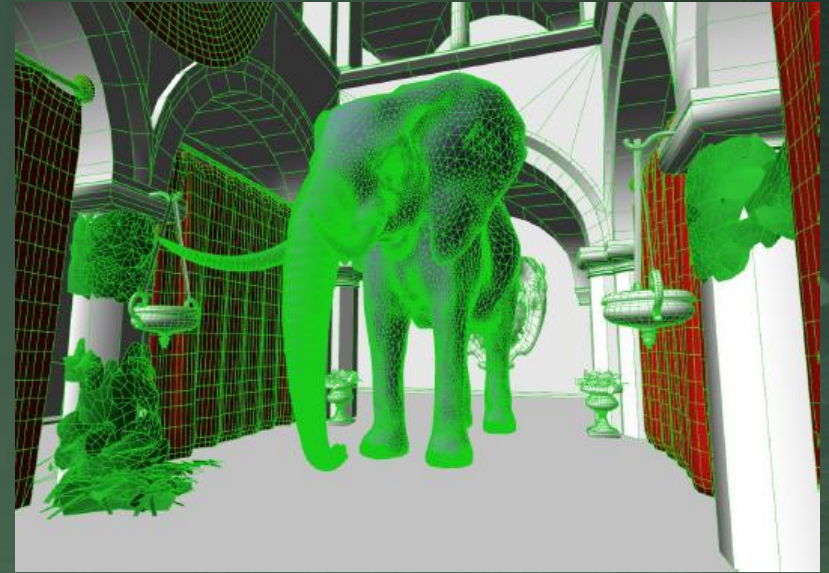
2. Rendu de volumes (milieux participants)

- a. Différentes approches
- b. Modélisation par ondelette
- c. Occlusion Propagation Volume



Le deferred shading

- Technique de rendu en rasterisation
 - Objectif : un unique rendu de la scène
 - Informations nécessaires à la visualisation stockées dans des buffers images
 - Travail en espace « image »
 - Opposé au forward rend.



Le deferred shading

- Intérêt : rendu = requête multidimensionnelle
Vue x Géométrie x Matériau x Lumière
 - Découpler lumière & géométrie
 - 1 seul rendu des objets
 - Peut gérer beaucoup de lumière (massive lighting)
- Limitations :
 - Seules les informations visibles sont exploitables
 - Stockage des G-Buffers
 - Beaucoup d'accès textures (souvent redondants)
 - Antialiasing !



Le deferred shading

- Pleins de variantes :
 - Choix des G-Buffers
 - Combinaison avec le forward shading & le culling
 - *Tiled Deferred Shading* [OA11],
Clustered Deferred and Forward Shading [OBA12]



Plan

1. Rendu de données massives

- a. Streaming de données
- b. Deferred shading
- c. **Antialiasing**

2. Rendu de volumes (milieux participants)

- a. Différentes approches
- b. Modélisation par ondelette
- c. Occlusion Propagation Volume



MLAA

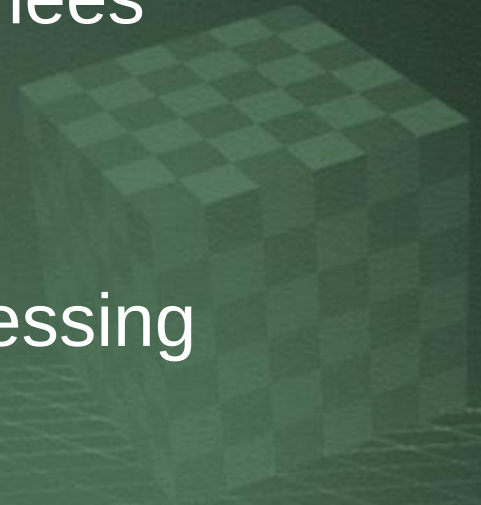
- Deferred Shading & Antialiasing classique
 - Peu performant
- Solution : Antialiasing en post-processing
 - MLAA



Présentation MLAA

Conclusion

- Modèle simple de chargement de données
 - Pour CPU & GPU
 - Prévisualisation rapide
- Algorithme d'antialiasing en post-processing
 - Implémenté en GPU
 - Extension : Topological MLAA
 - Trou d'1 pixel



Conclusion & perspectives

- Réutilisation des G-Buffers entre frames
 - G-Buffers par bounding box
 - Réutilisation des G-Buffers au temps N-1
 - Problème d'heuristique des Bbox / G-Buffers à réutiliser
 - Nombre de pixel couvrant
 - Distance
 - Visibilité
 - Animations, taille mémoire

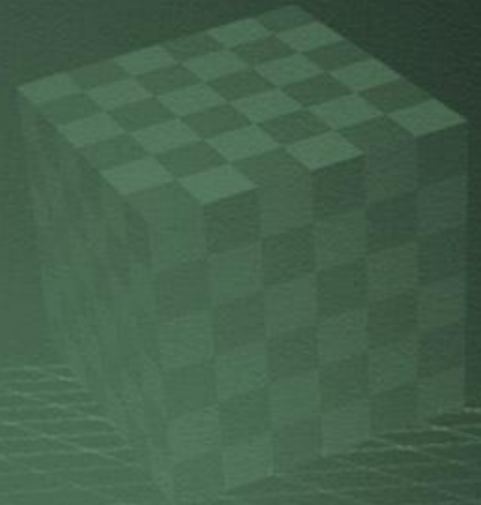


Conclusion & perspectives

- Massive lighting
 - Relighting des G-Buffers
- Antialiasing
 - Aller plus loin que le MSAA
 - Cohérence temporelle à moindre cout par détection (et suivi) de segment discret



RENDU DE VOLUMES



Contexte

- On s'intéresse
 - A des milieux participants
 - Dans des scènes contenant des surfaces
 - Le milieu peut être large ou contraint
- 2 problématiques :
 - La modélisation du volume
 - Le rendu du volume



Plan

1. Rendu de données massives
 - a. Streaming de données
 - b. Deferred shading
 - c. Antialiasing
2. Rendu de volumes (milieux participants)
 - a. Différentes approches
 - b. Modélisation par ondelette
 - c. Occlusion Propagation Volume

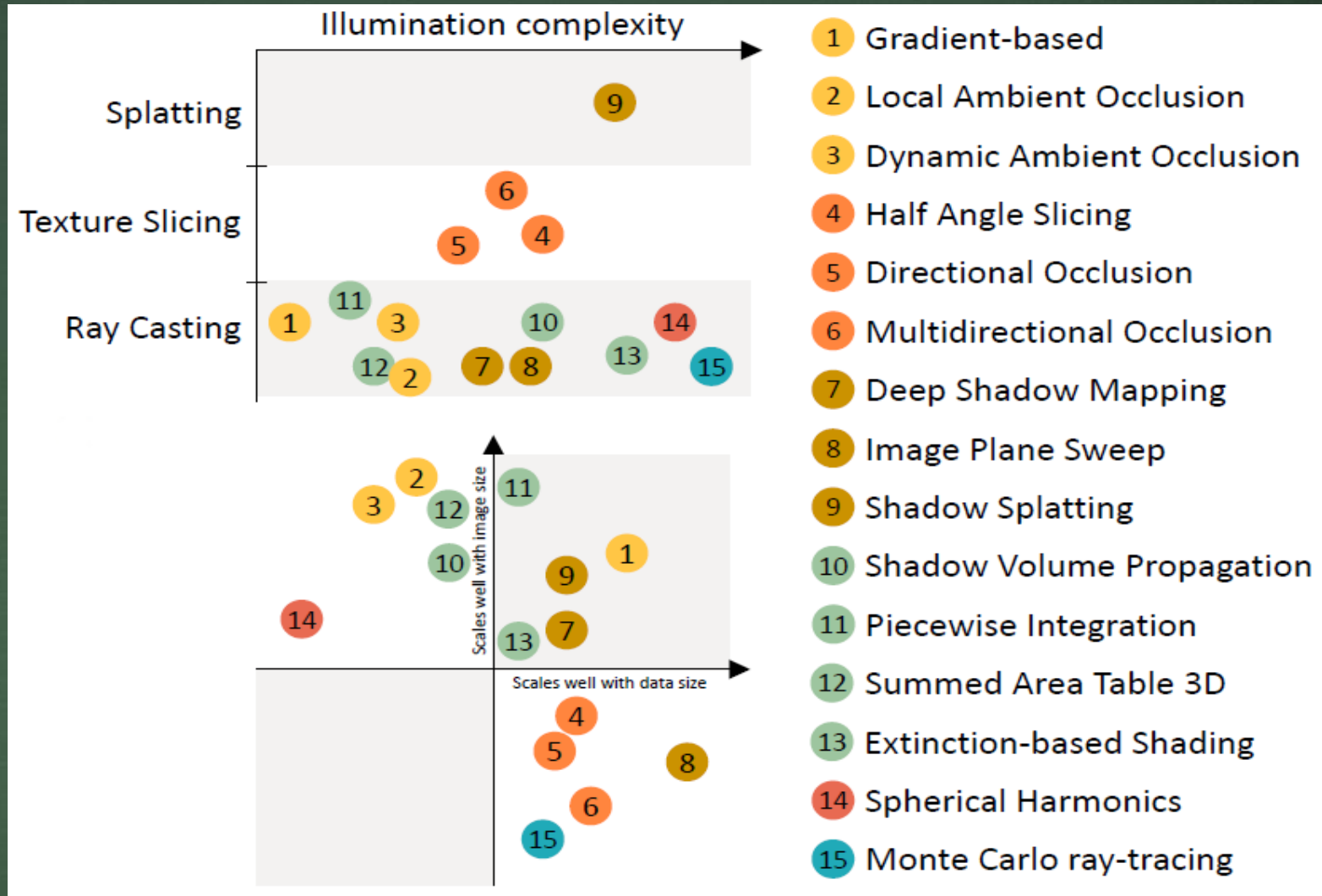


Différentes approches

- Sur le rendu :
 - Approches en DVR
 - Cf. STAR EG 2012
 - D'autres approches plus « synthèse d'images »
 - PPB
 - VBL
 - LPV



DVR

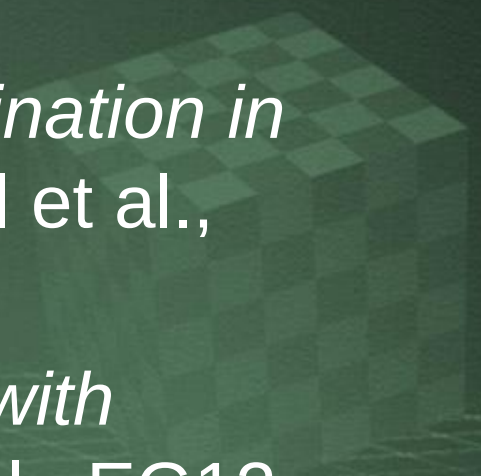


State of The Art Report on Interactive Volume Rendering with Volumetric Illumination, EG 12,

DVR

- Particulièrement :

- *Extinction-based Shading and Illumination in GPU Volume Ray-Casting*, Schlegel et al., TVCG 2011
- *Interactive direct volume rendering with physically-based lighting*, Kroes et al., EG12



Des algorithmes de rendu



Progressive
Photon
Beams
[JNT+11]



Virtual Beam
Lights
[NNDJ12]

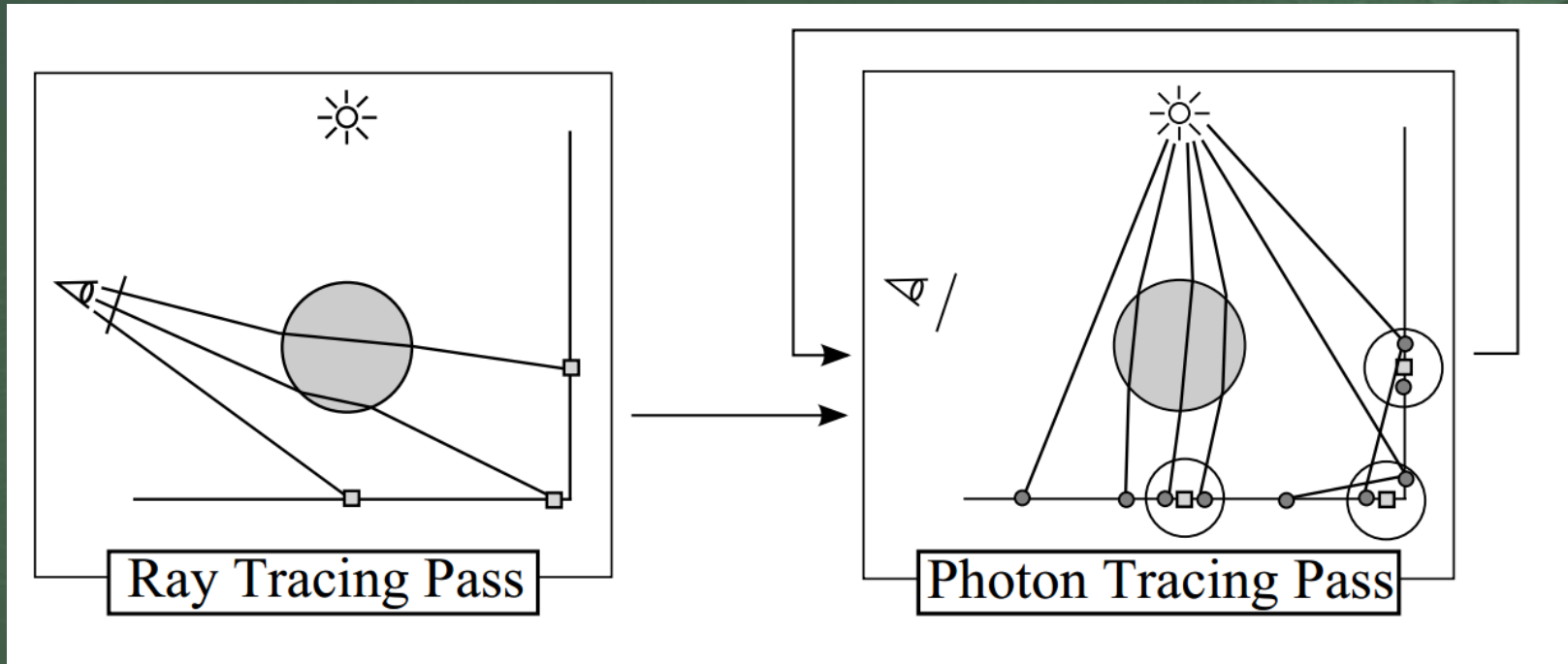


Light
Propagation
Volumes
[KD10]

Qualité / Rapidité

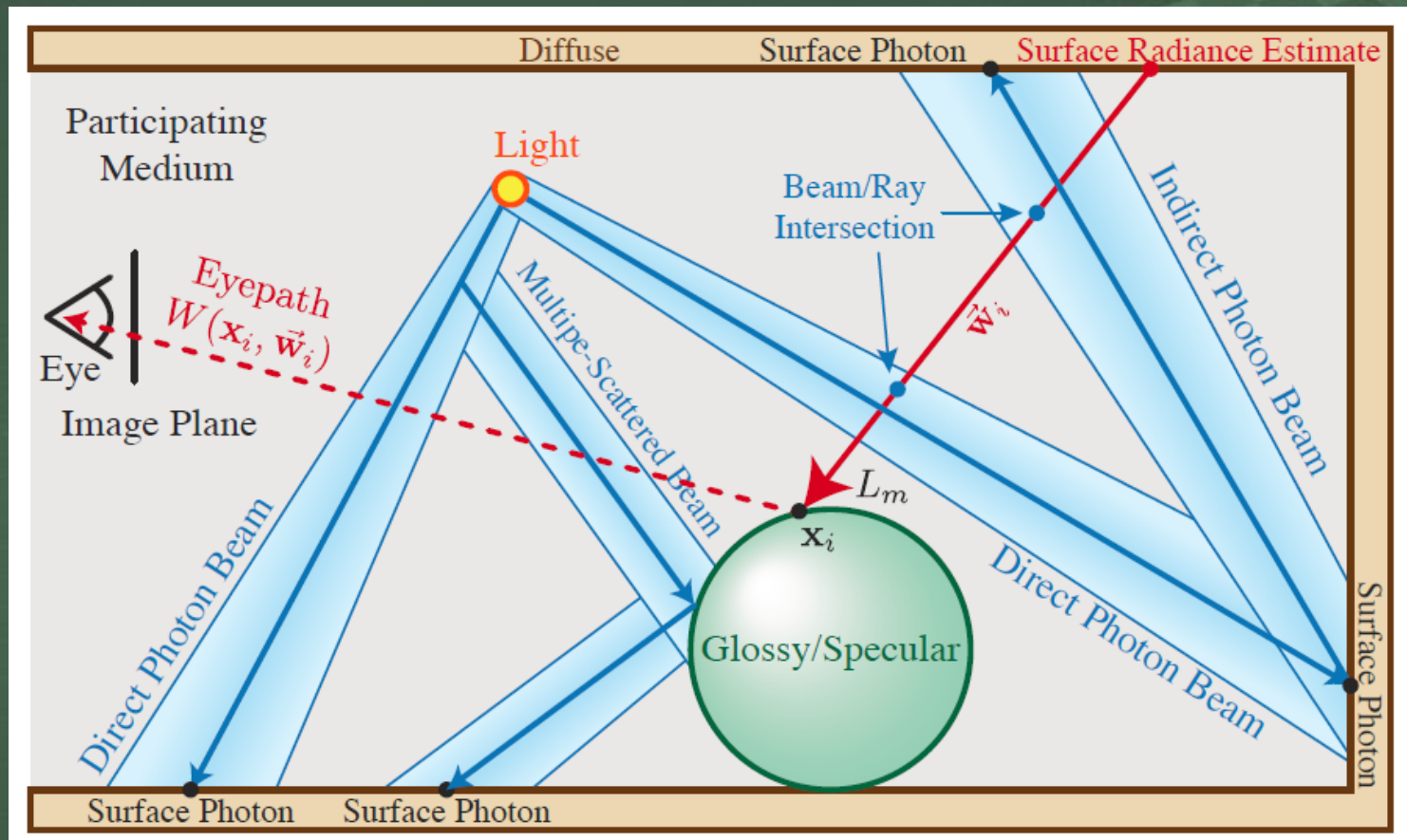
Progressive Photon Beams

- Du Progressive Photon Mapping ...

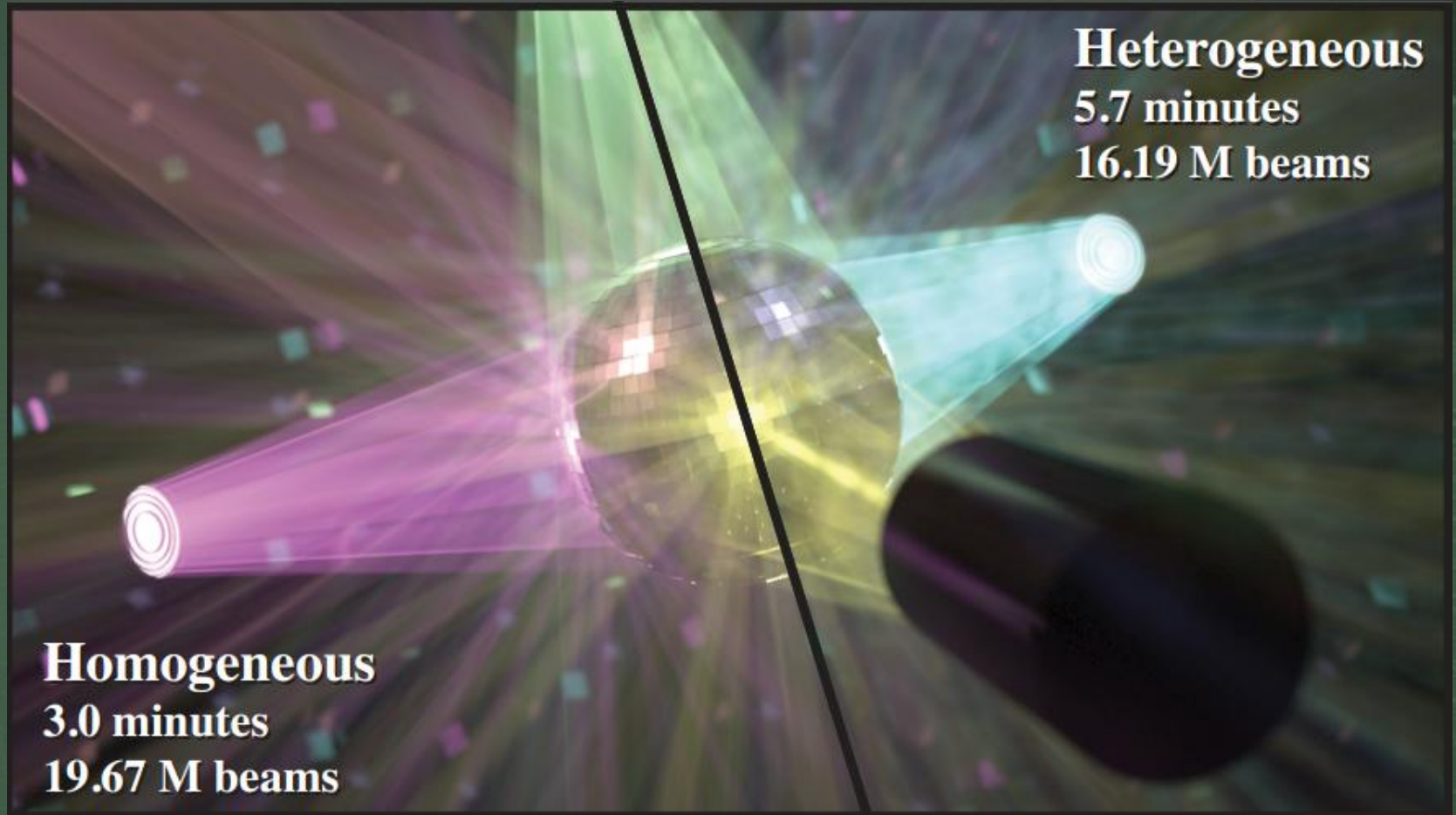


Progressive Photon Beams

- ... mais avec des Beams



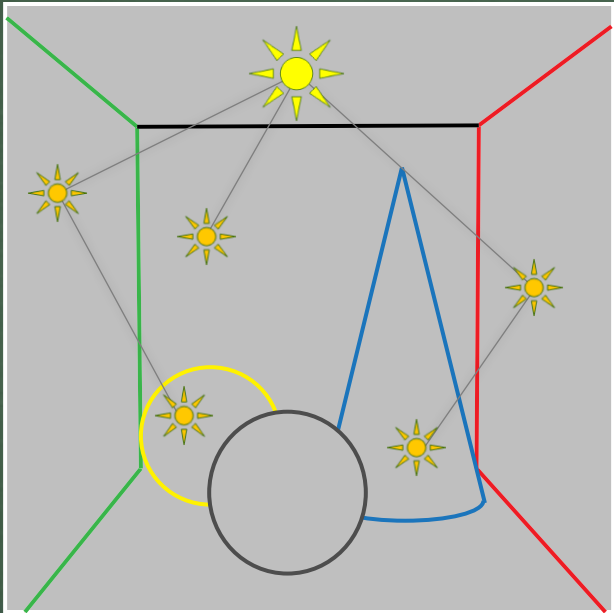
Progressive Photon Beams



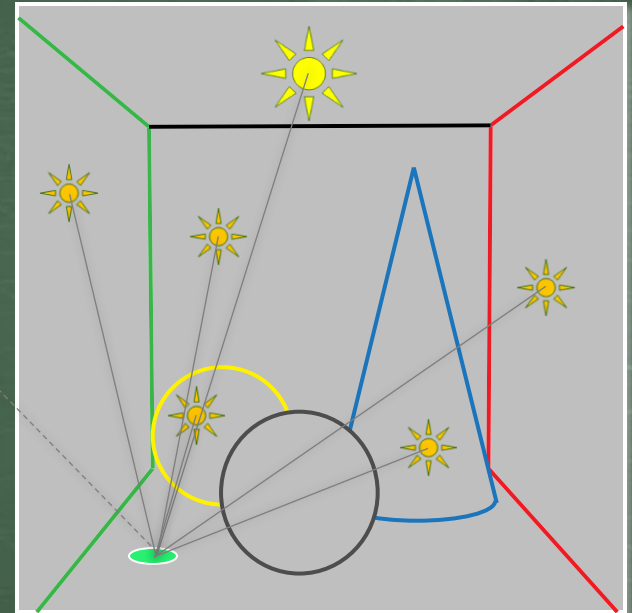
Virtual Beam Lights

- L'idée des Virtual Point Lights (VPL) ...
 - Approcher l'illumination indirecte
 - Origine : *Instant Radiosity* [Keller97]

1. Générer les VPLs



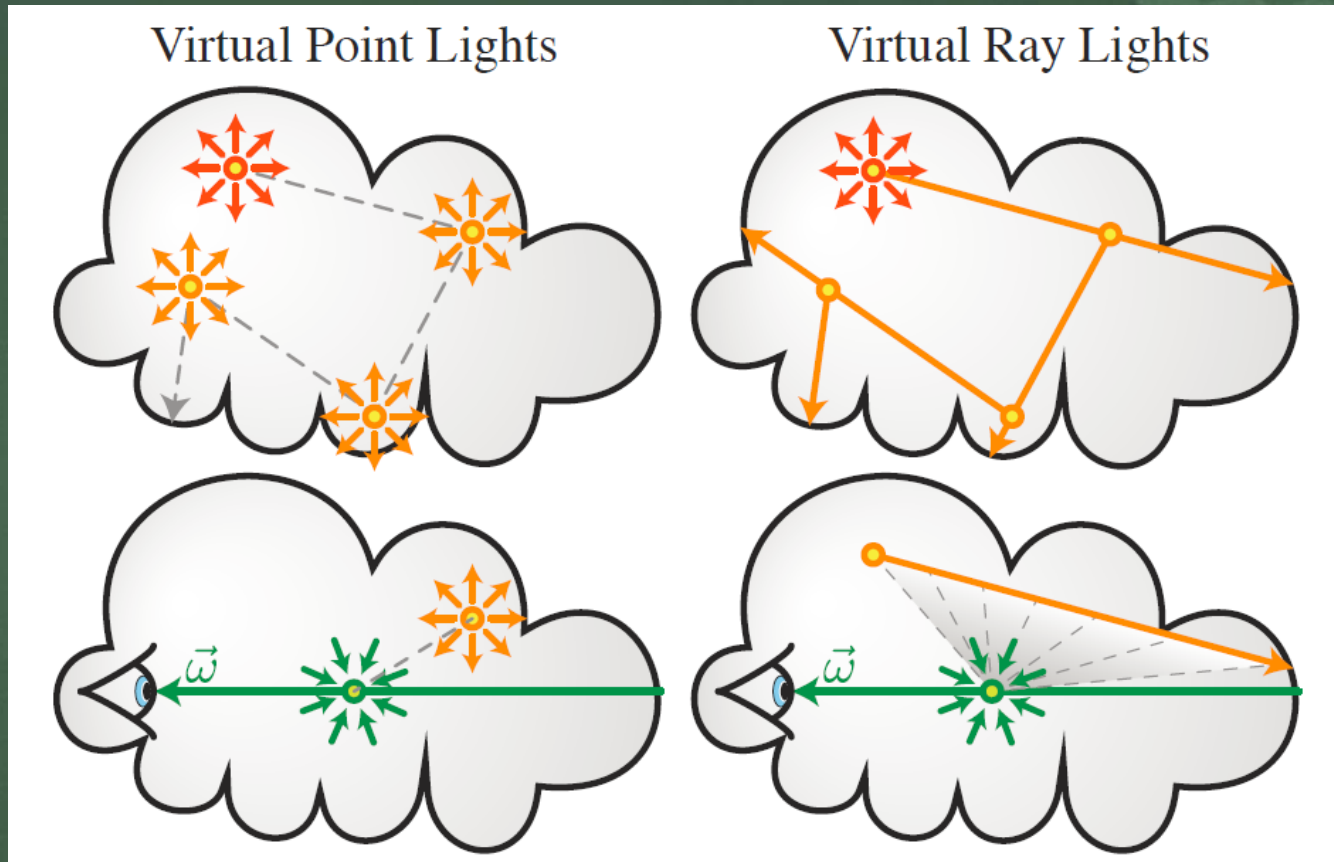
2. Rendu avec VPLs



« Optimizing Realistic Rendering with Many Light Method », Krivanek J.

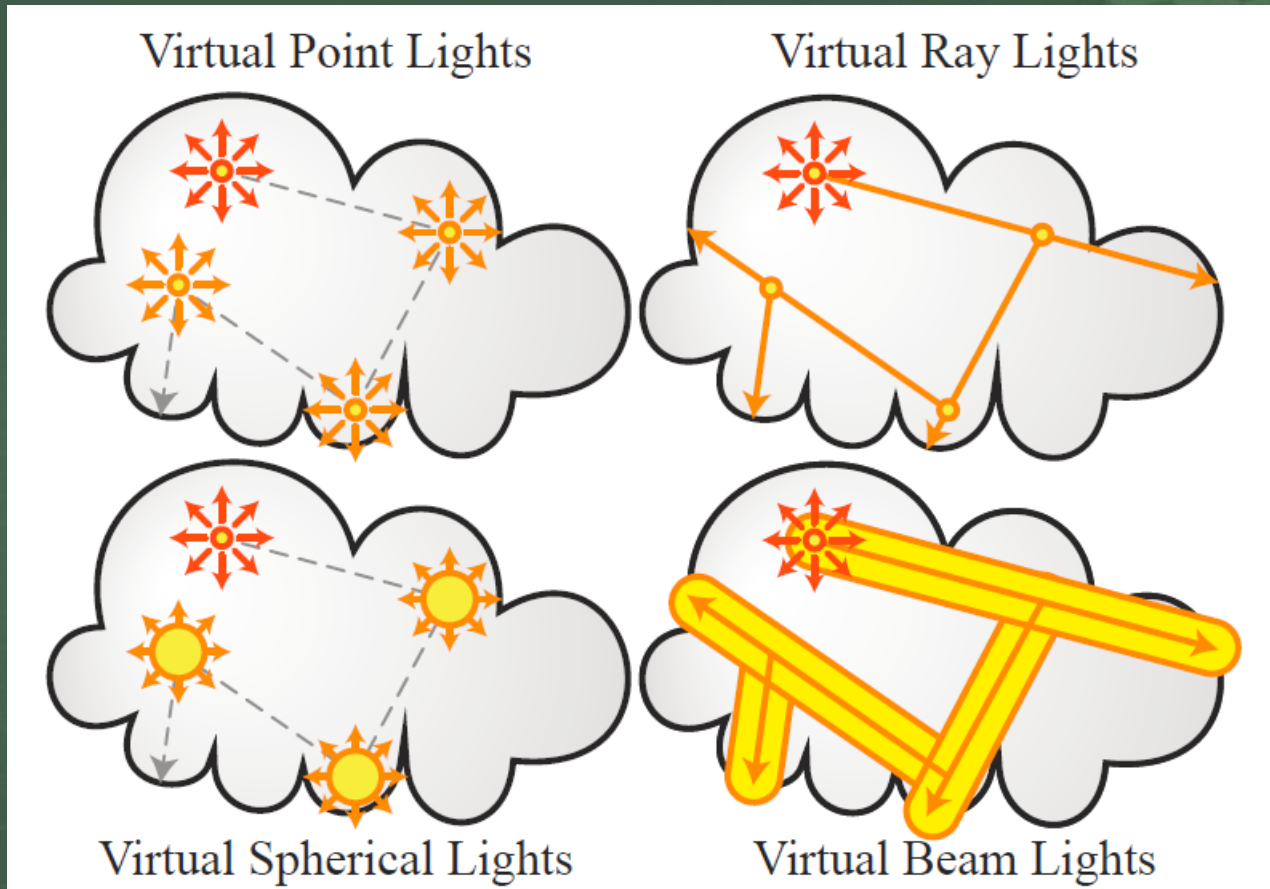
Virtual Beam Lights

- ... avec des rayons ...

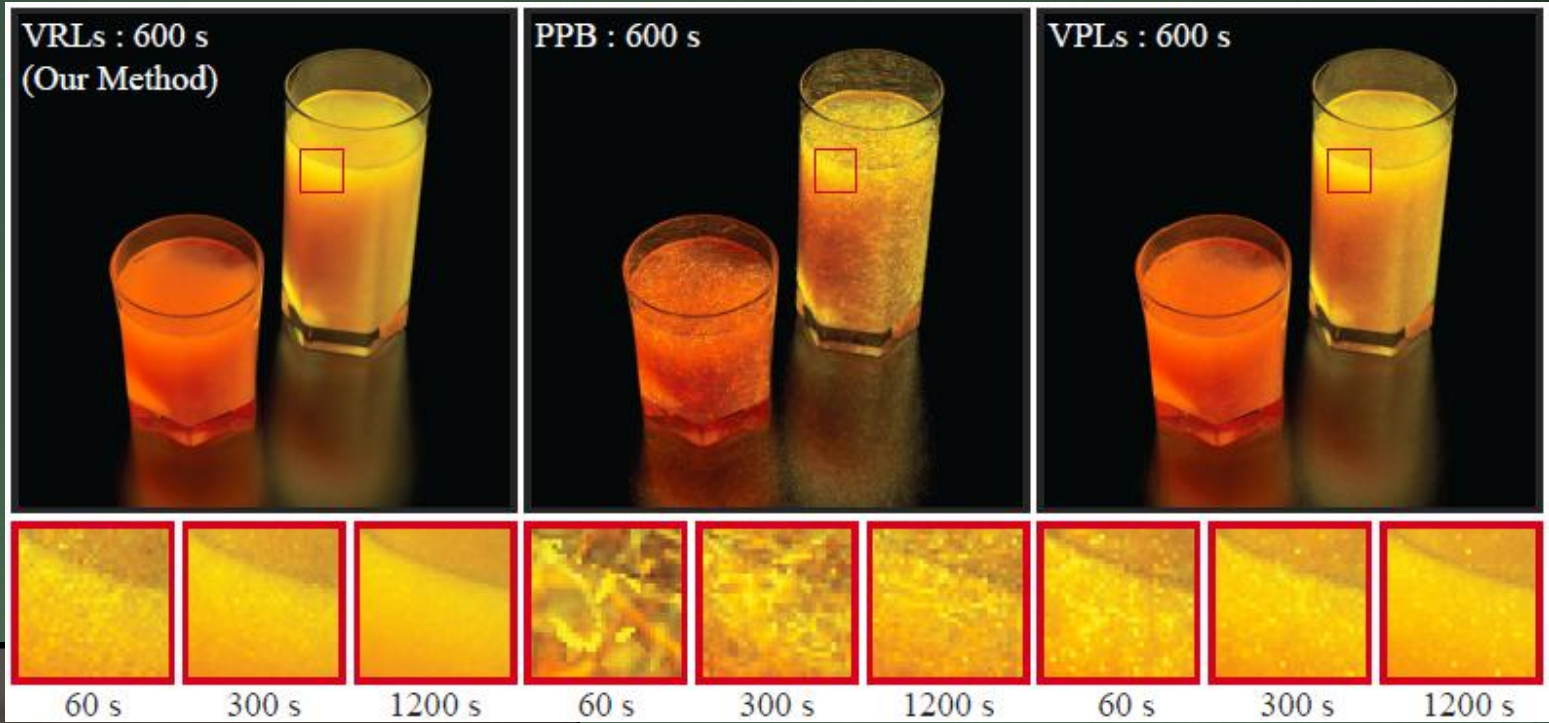


Virtual Beam Lights

- ... ou plutôt des faisceaux de lumière

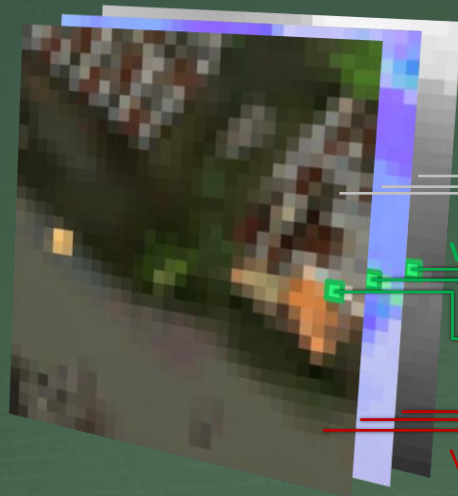


Virtual Beam Lights



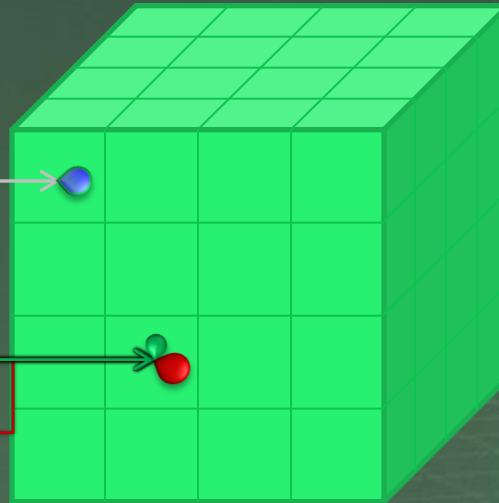
Light Propagation Volume

Reflective shadow maps



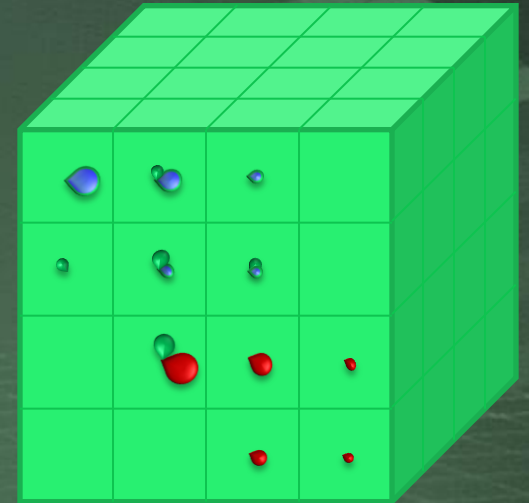
A set of regularly sampled VPLs of the scene from light position

Radiance volume gathering



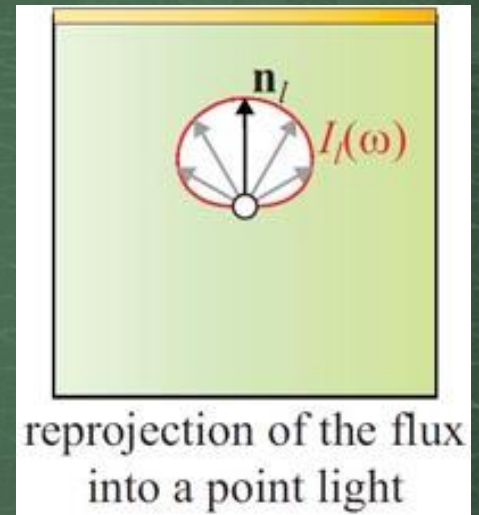
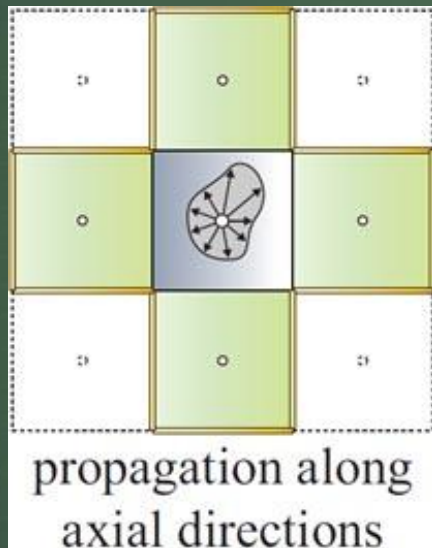
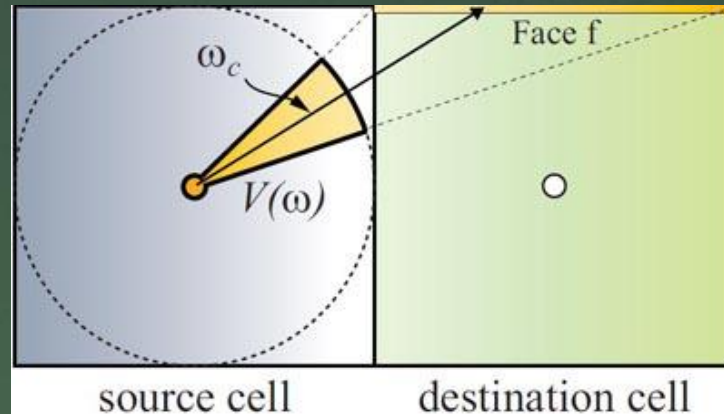
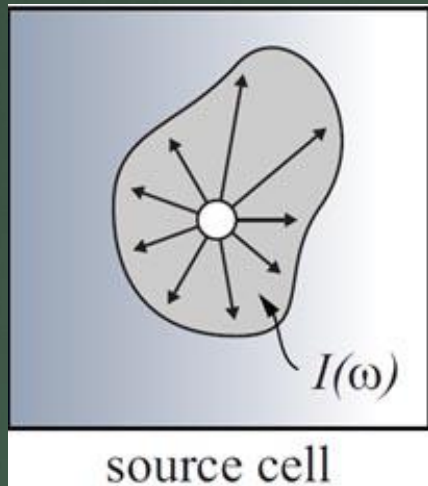
Discretize initial VPL distribution by the regular grid and SH

Iterative propagation



Propagate light iteratively going from one cell to another

Light Propagation Volume



Light Propagation Volume



Environ 4 ms

Plan

1. Rendu de données massives

- a. Streaming de données
- b. Deferred shading
- c. Antialiasing

2. Rendu de volumes (milieux participants)

- a. Différentes approches
- b. Modélisation par ondelette
- c. Occlusion Propagation Volume



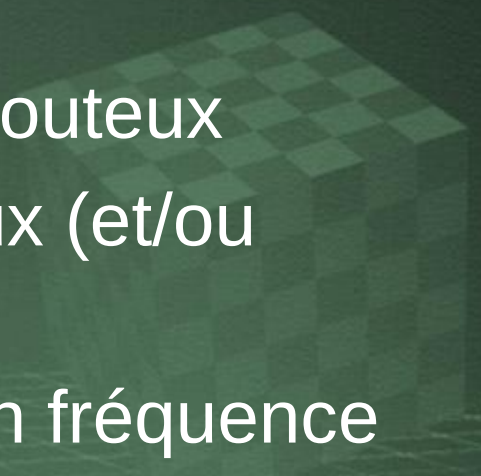
Contexte

- Modélisation de milieu participant hétérogène de type brouillard
- Sur une large scène



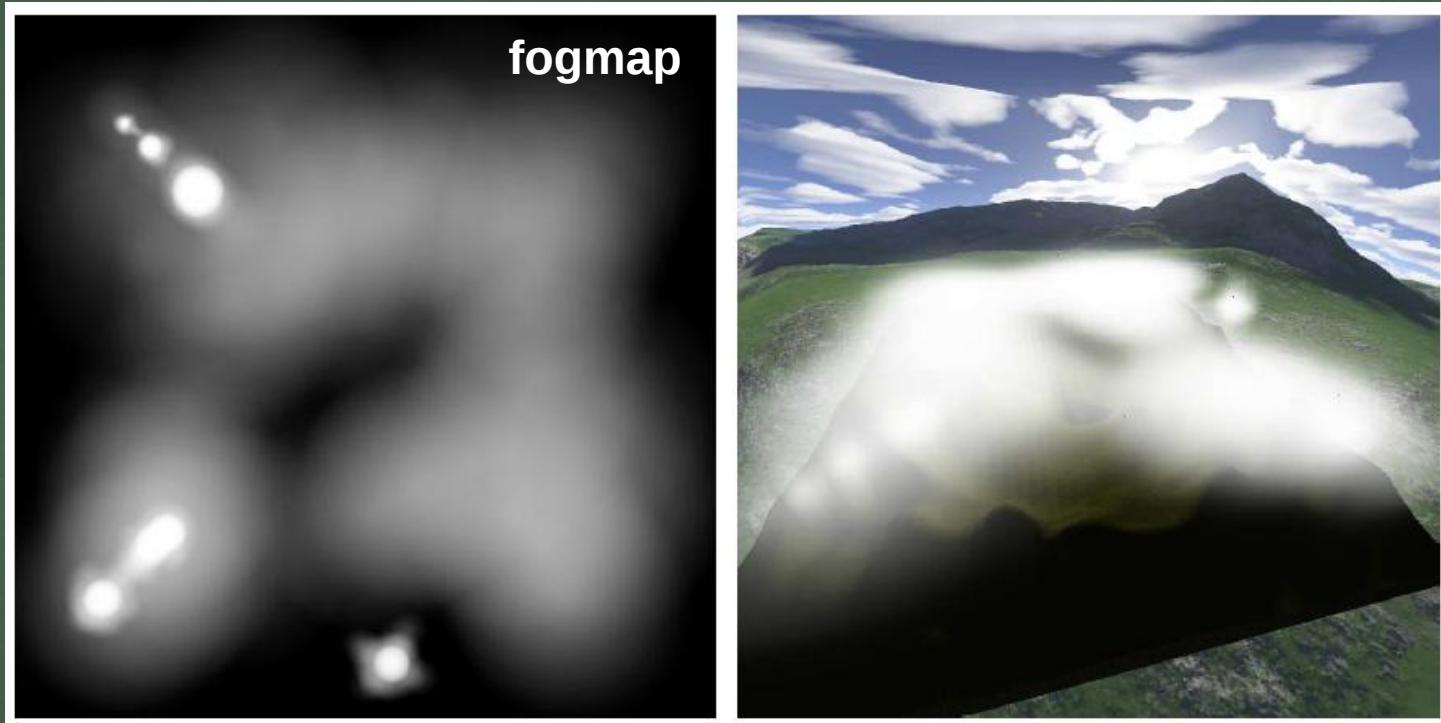
Modélisation du milieu

- Problématique :
 - Représentation par grille uniforme coûteux
 - Représentation particulière fastidieux (et/ou coûteux) pour de large scène.
- Il faut une décomposition spatiale et en fréquence
 - Les ondelettes sont un bon outil mathématique pour ce problème



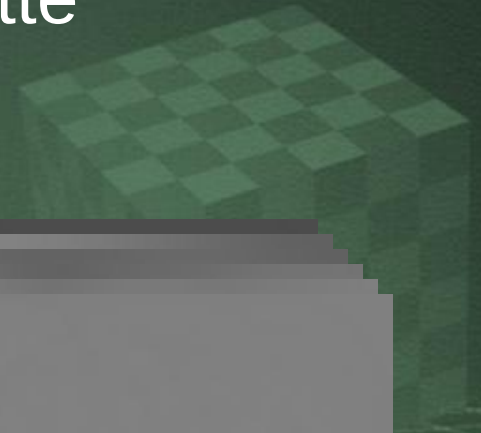
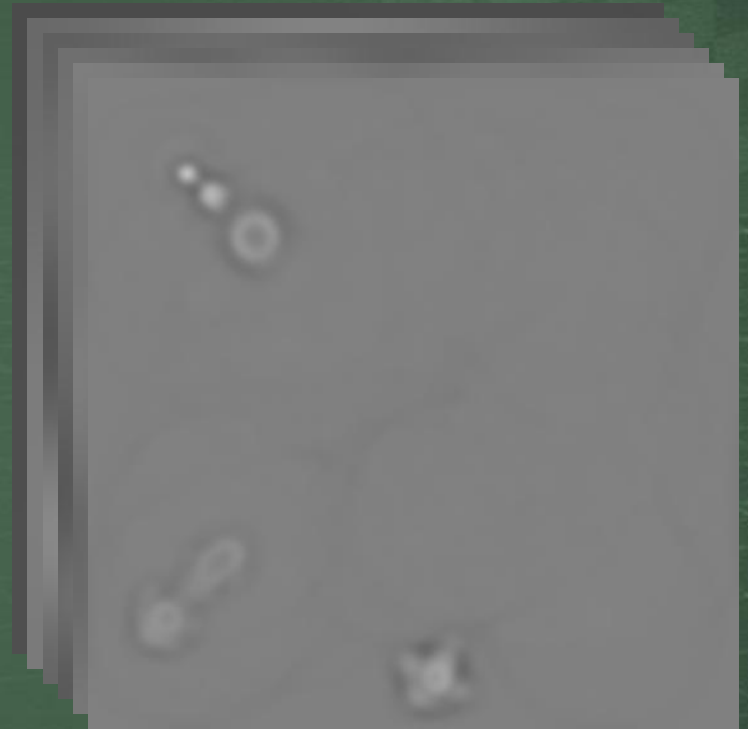
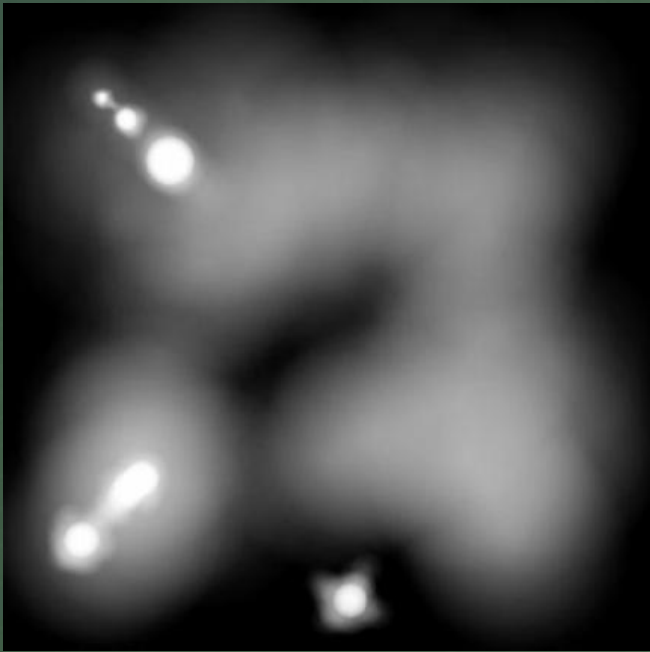
Modélisation du milieu

- Choix de modélisation
 - Brouillard hétérogène 2D, par couche



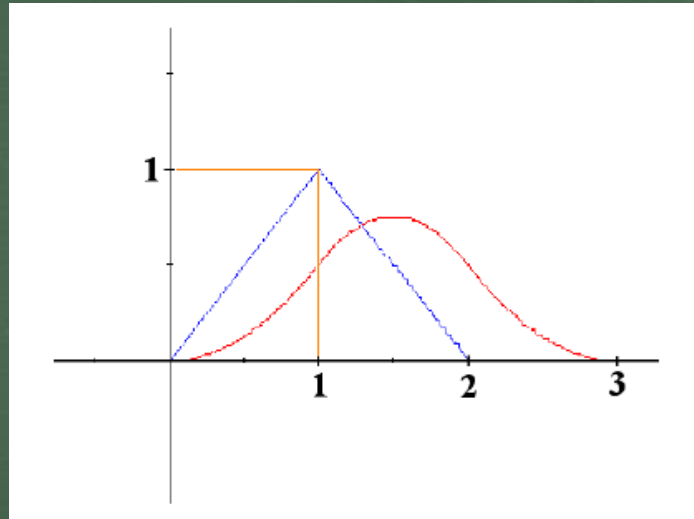
Modélisation du milieu

- Compression des données par ondelette
 - Images de coefficients



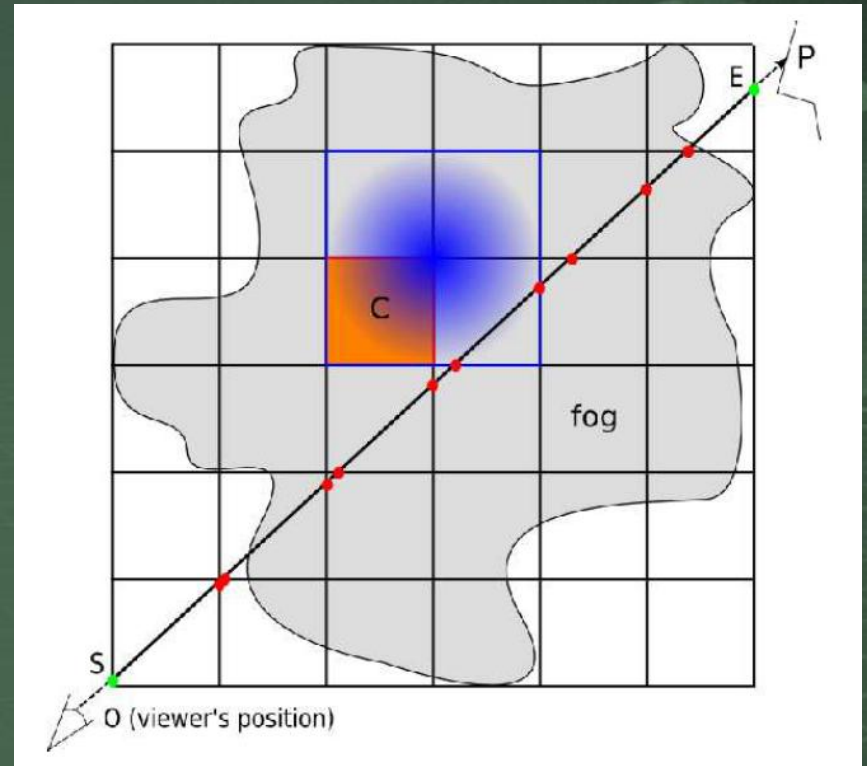
Modélisation du milieu

- Compression des données par ondelette
 - Choix de la base d'ondelette
 - Haar, Linear, Quadratic



Rendu du milieu

- Rendu absorption seule
- Calcul exact
 - Précalcul sur chaque *patch* de l'évaluation de chaque fonction de base
- Calcul approché
 - Raymarching le long du rayon
- Attention, support non local des fonctions



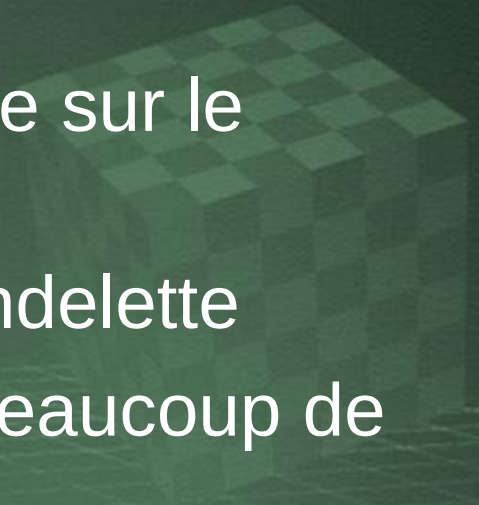
Rendu du milieu

- Avantages de la décomposition en ondelette
 - Décomposition en fréquence
 - On peut éliminer les coefficients les + faibles
 - Décomposition en espace
 - Le point précédent peut se faire partout
 - Désavantages
 - Support non local des fonctions
 - Structure de données + complexe
- ⇒ Difficile de battre, en GPU, les approches « brute force »



Conclusion & perspectives

- Mieux exploiter les ondelettes
 - Stocker la structure de type quadtree sur le GPU
 - Modélisation 3D du brouillard par ondelette
 - Attention, passer en 3D génère beaucoup de fonctions différentes
 - Faire la composition / décomposition à la volée sur GPU



Plan

1. Rendu de données massives

- a. Streaming de données
- b. Deferred shading
- c. Antialiasing

2. Rendu de volumes (milieux participants)

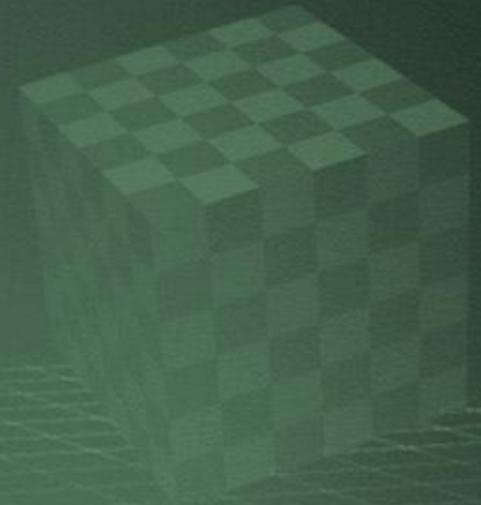
- a. Différentes approches
- b. Modélisation par ondelette
- c. Occlusion Propagation Volume



Occlusion Propagation Volume

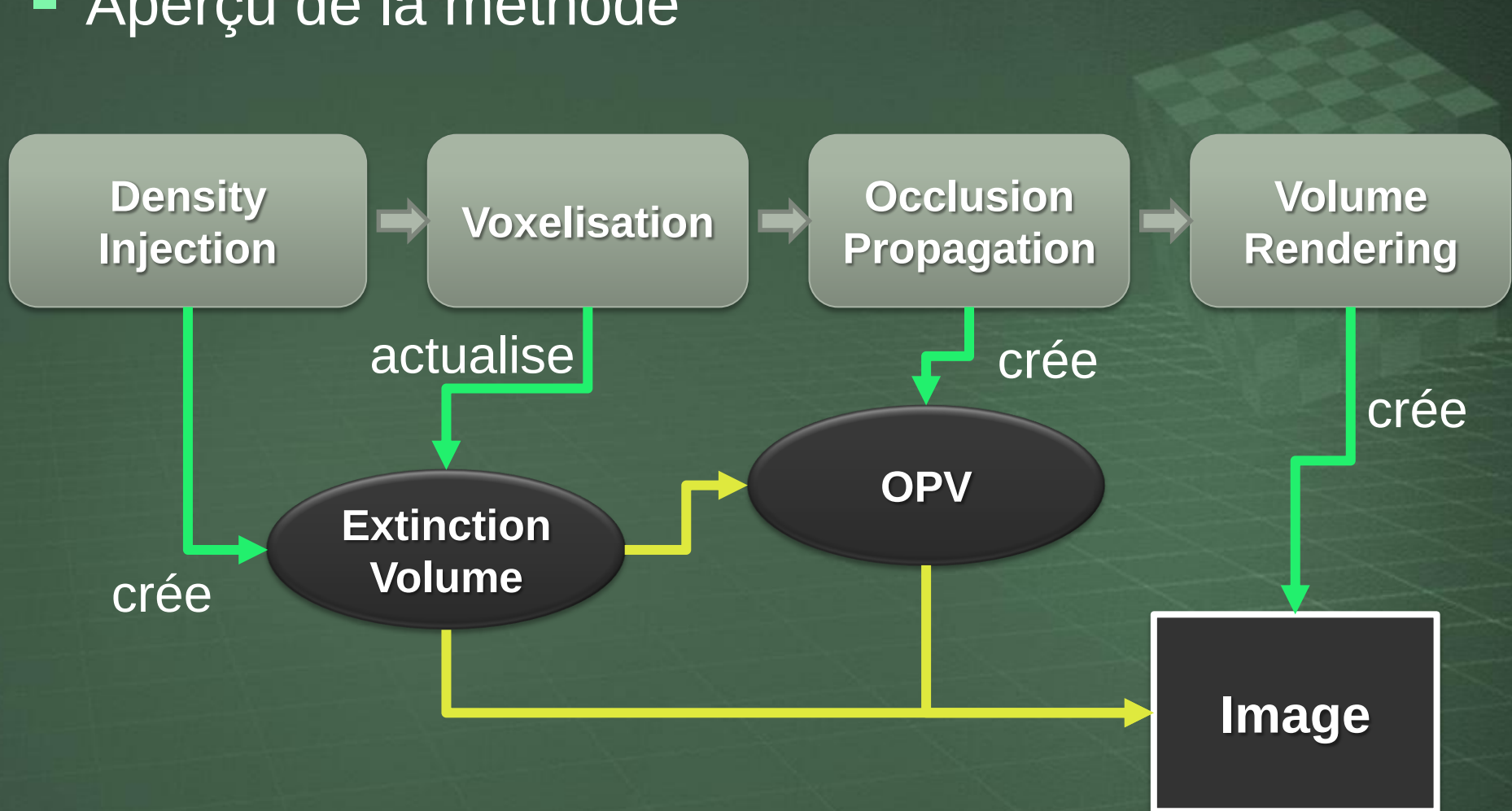
■ Problématique :

- Single scattering en temps réel
- MP hétérogène relativement local
- On veut tout l'ombrage
 - Des volumes sur les objets
 - Des objets sur le volume
 - Dans le volume (optical depth)
- Pas d'élément statique : lumière, géométrie, milieu



Occlusion Propagation Volume

- Aperçu de la méthode



Density Injection

- Ajout de la densité du milieu
 - Code localement la valeur du coefficient d'extinction
 - On a choisi ici une modélisation particulière avec des RBF
- Stocké dans une grille 3D
 - Calcul par plane sweep



Voxelisation

- Il existe des méthodes temps-réel
 - *Fast Parallel Surface and Solid Voxelization on GPUs*, Schwarz, Seidel, SIG Asia 10
 - *Real-Time Hierarchical Binary-Scene Voxelization*, Forest et al., JGT 09
 - Nous avons utilisé une méthode « maison » moins rapide ☹
- Réactualisation de la grille d'extinction
 - Là où il y a un objet, l'extinction est « totale »



Occlusion propagation

- Similaire au LPV mais :
 - On s'intéresse ici à l'éclairage direct
 - On doit « diffuser » de manière concentrique et précisément
- Notre stratégie
 - On ne diffuse que l'occlusion
 - Propagation tenant compte de l'origine de l'émission
 - Par gathering, sur GPU



Rendering

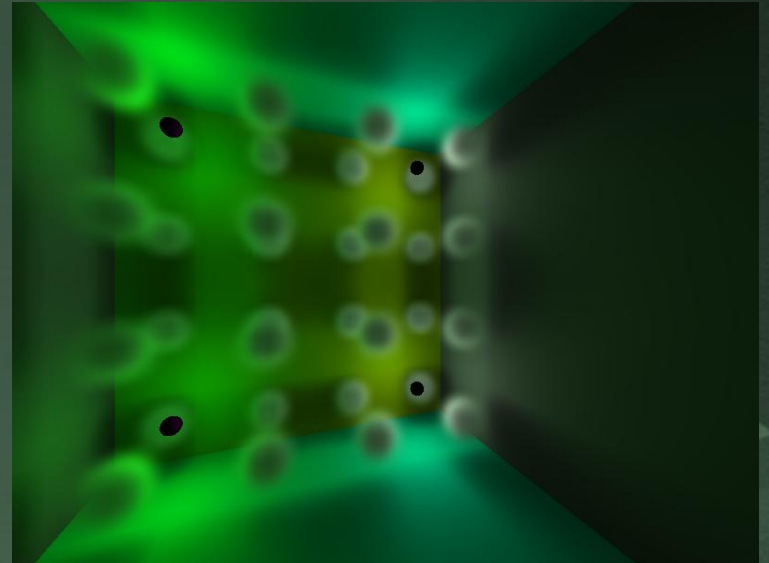
- Par ray marching

- On récupère les informations d'extinction et d'occlusion directement des grilles 3D
- On calcule séparément la contribution de la lumière
- Ombrage « automatique »



Résultats

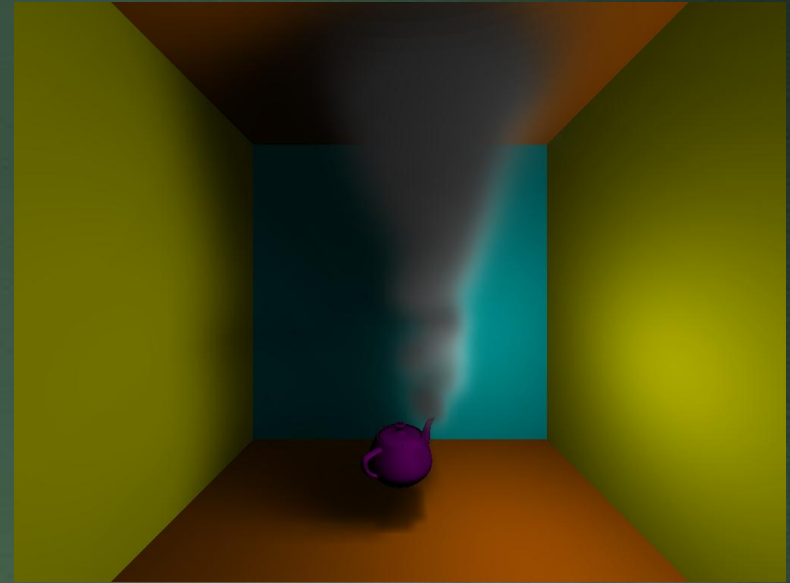
- Rapidité (fps)



Résolution	1 light	2 lights	4 lights
24³	166	87	42
32³	111	58	29
40³	78	40	19
64³	35	17	8
80³	23	11	5

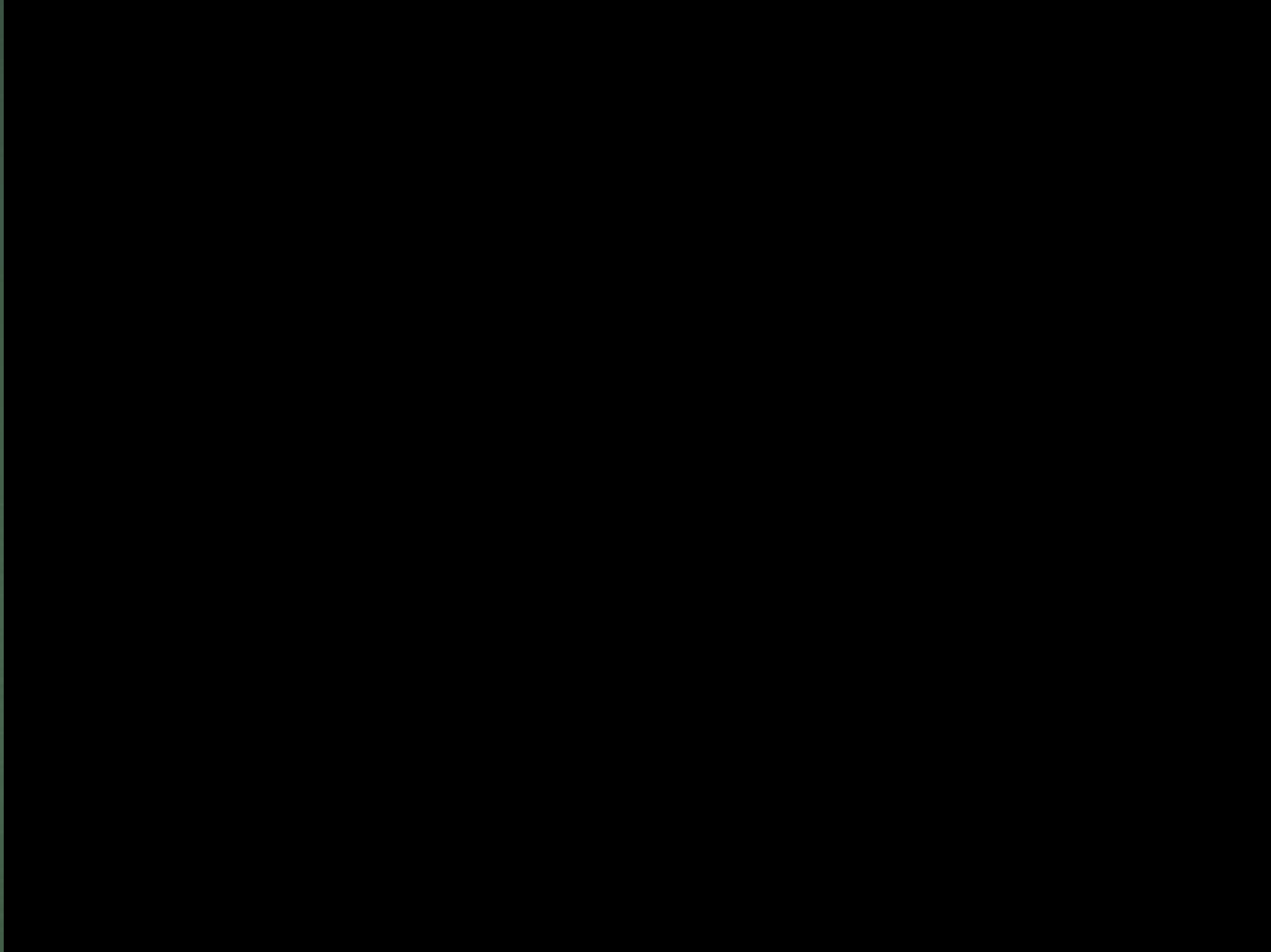
Résultats

- Pour l'animation (fps)



Résolution	All dynamic	Dynamic light & medium	Rendering only
24³	13.4	156	277
32³	9.9	106	227
40³	8	76	192
64³	4.8	34	135
80³	3.6	24	111

Résultats



Conclusion & Perspectives

- Une méthode temps réel d'illumination
 - Mélange surfaces & volumes
 - Aucun élément statique
 - Reste des artefacts (occlusion) dus à la grille
- Perspectives :
 - Multi résolution de la grille (cascaded OPV ?)
 - Etendre au *multiple scattering*
 - En adaptant la technique des LPV



Conclusion générale

- Visualisation de données massives 2D...
 - Streaming de données
 - Deferred shading
 - Effet en post-processing
- ... mais aussi purement 3D
 - Propagation d'information dans les grilles
 - Visualisation de volumes



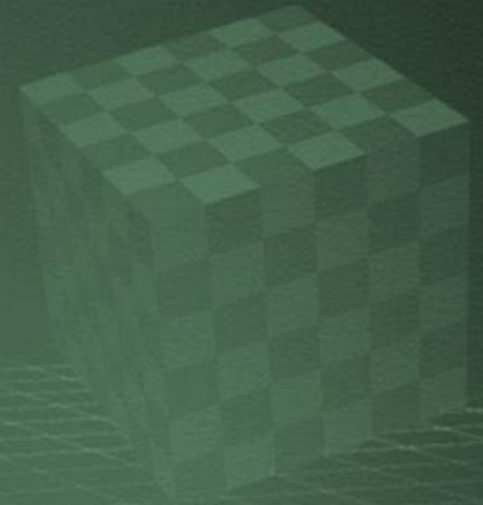
Perspectives générales

- Sur les scènes massives
 - Jouer sur la reprojection, travail en espace image
 - Segmentation de scène 3D
- Sur les volumes
 - Représentation par ondelette
 - Représentation unifiée surface – volume
 - Aborder les techniques de type *cone tracing in sparse voxel octree*



QUESTIONS ?

biri@univ-mlv.fr



Précisions sur le streaming



Résultats ondelettes

Fog resolution	Haar	Linear	Quadratic
16×16	199	142	71
32×32	124	83	31
64×64	90	45	15

Table 8.1: FPS results with our ray-marching without optimizations.

μ Nb levels	1	0.8	0.6	0.4	0.2	0
0	45	-	-	-	-	-
1	35	39	47	55	66	83
2	31	45	55	71	90	124
3	27	39	66	76	99	166

Table 8.2: FPS results with our optimization, using a 64×64 linear B-Spline fogmap.

OPV technique

