



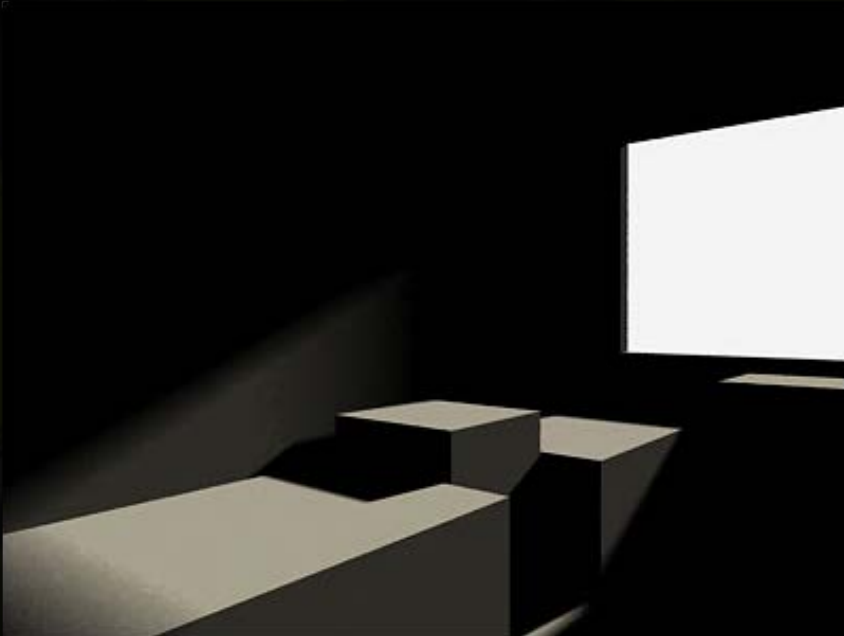
# Voxel Cone Tracing and Sparse Voxel Octree for Real-time Global Illumination

Cyril Crassin  
*NVIDIA Research*

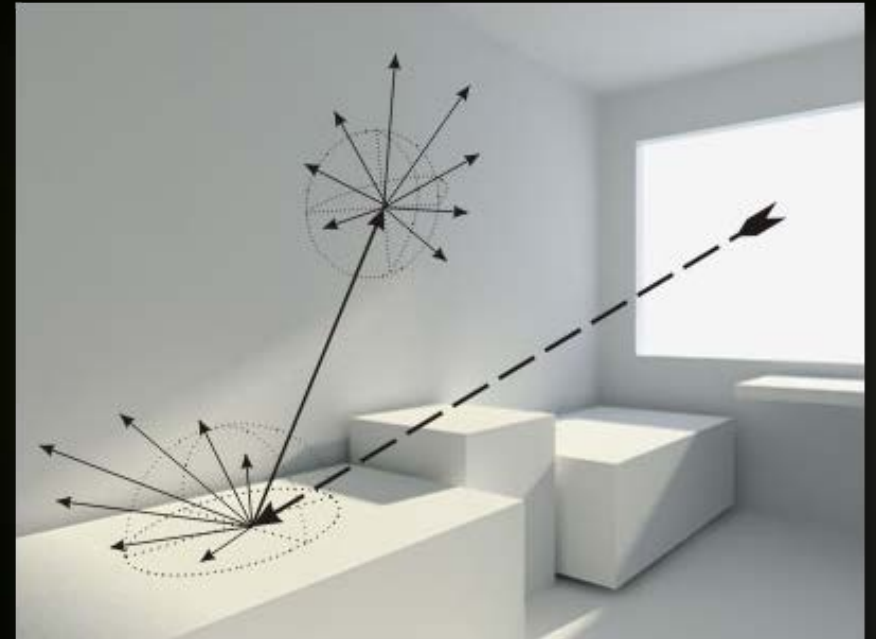


# Global Illumination

- Indirect effects
- Important for realistic image synthesis



Direct lighting



Direct+Indirect lighting



Settings:

Off

Voxel-Based GI

8ms @ 512x512 – 27ms @ 720p - 62ms @ 1080p





# Voxel-based GI

- It can run in a game !
  - EPIC Games : SVOgi





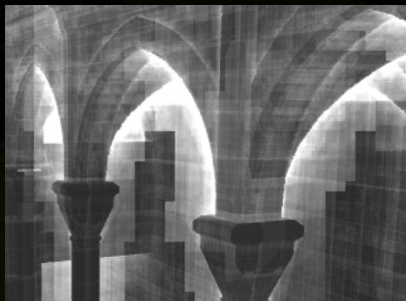
DEMO



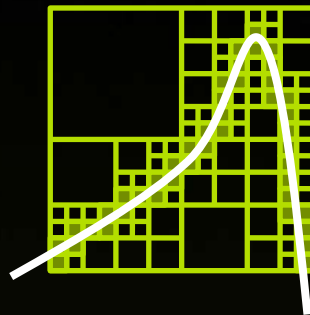
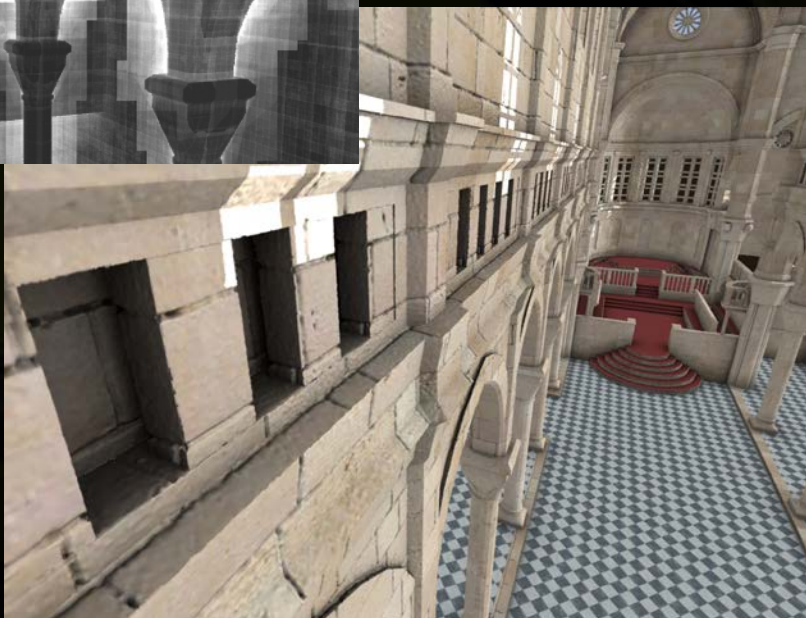


# Sparse Voxel Octree

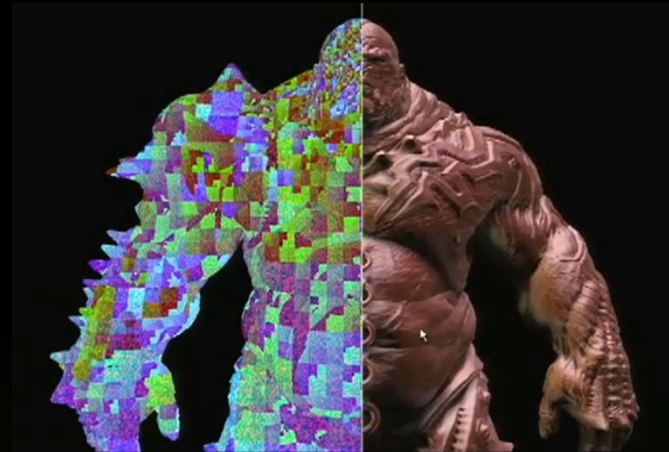
- Detailed geometry rendering
  - Structured LODs



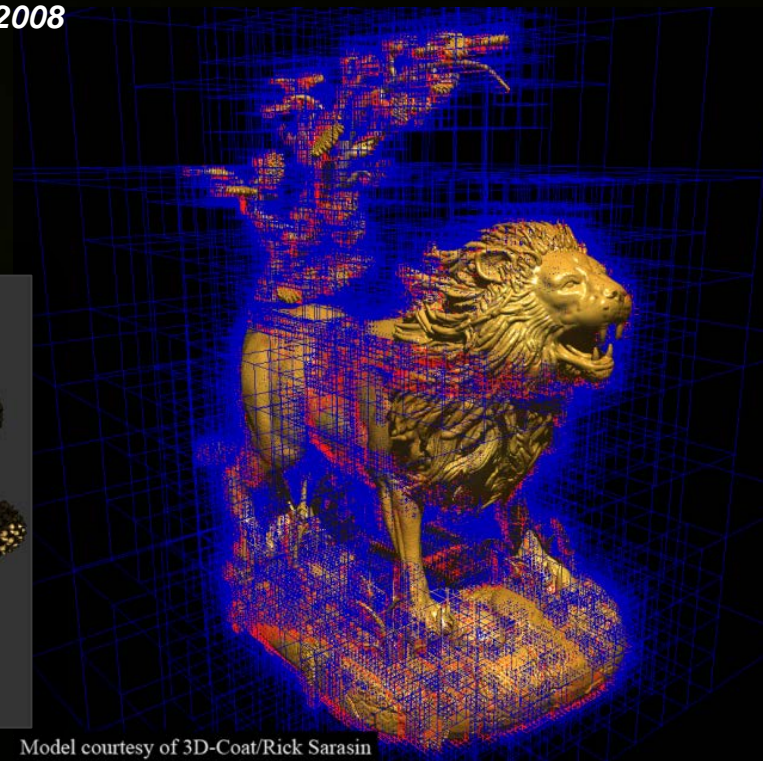
Laine and Karras (NVIDIA)  
2010



Crassin et al. 2009  
(GigaVoxels)

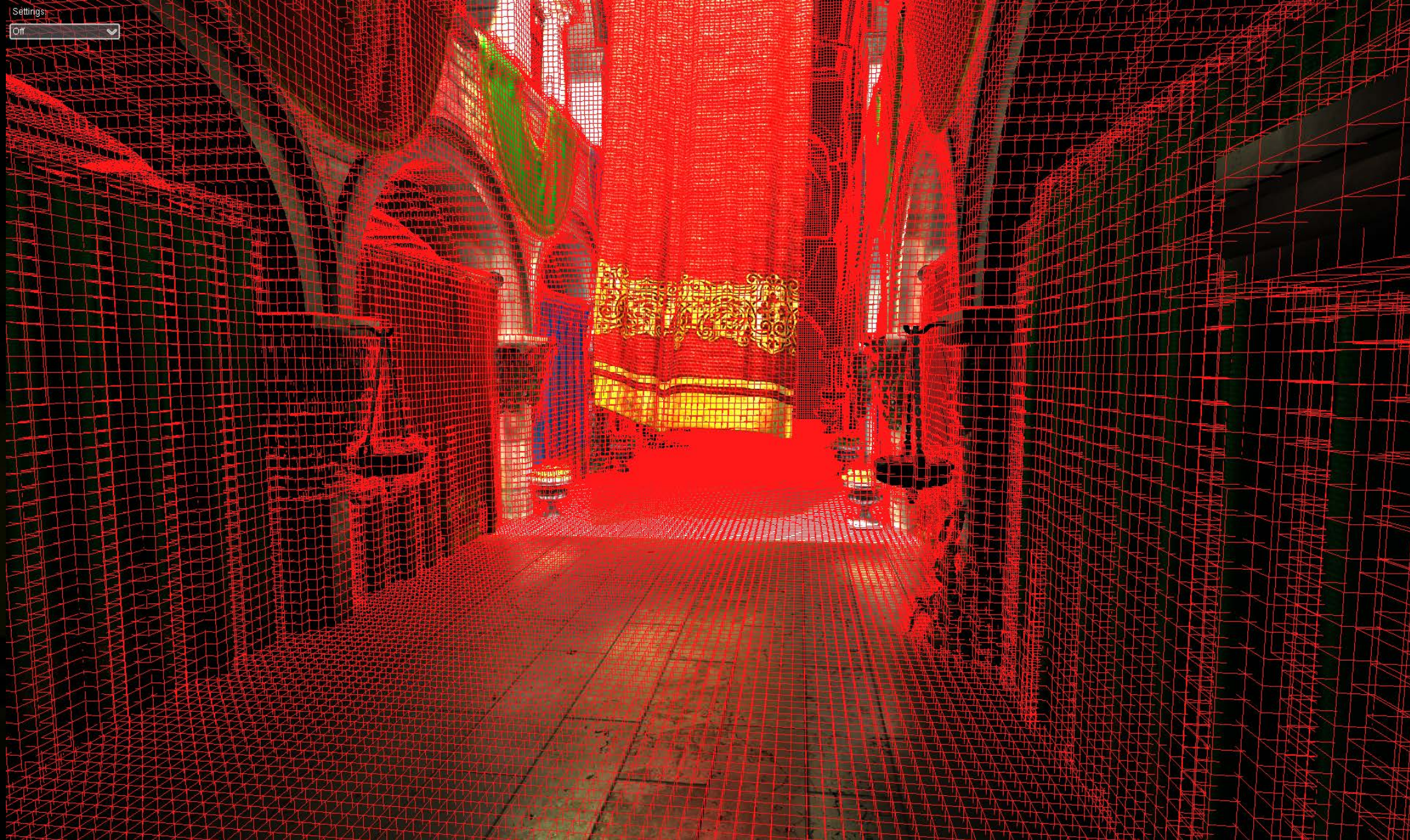


Olick. 2008



Model courtesy of 3D-Coat/Rick Sarasin

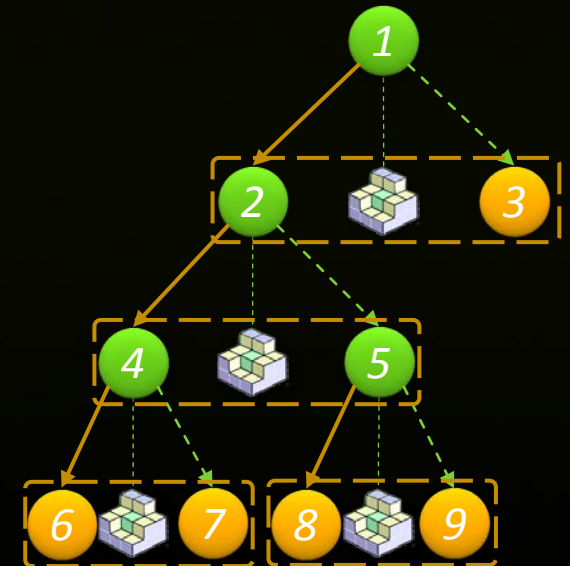
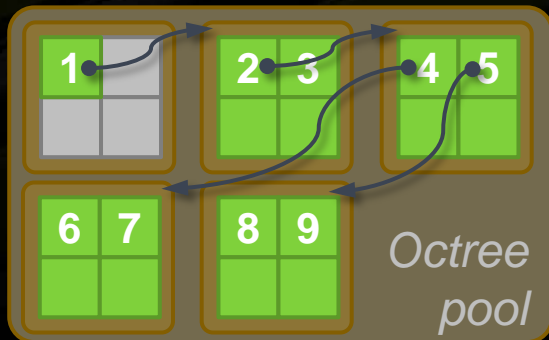






# GPU Voxel Octree

- Linked nodes in **linear video memory** (*Octree Pool*)
  - 2x2x2 nodes tiles
  - 1 pointer per node to a node-tile
- Voxels stored into a **3D texture** (*Brick Pool*)
  - Allows hardware tri-linear interpolation

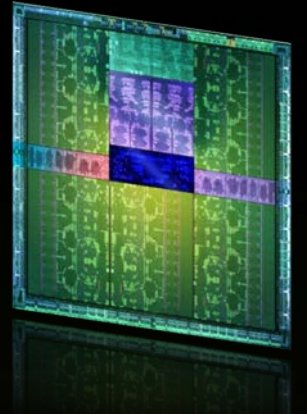




# Hybrid rendering pipeline



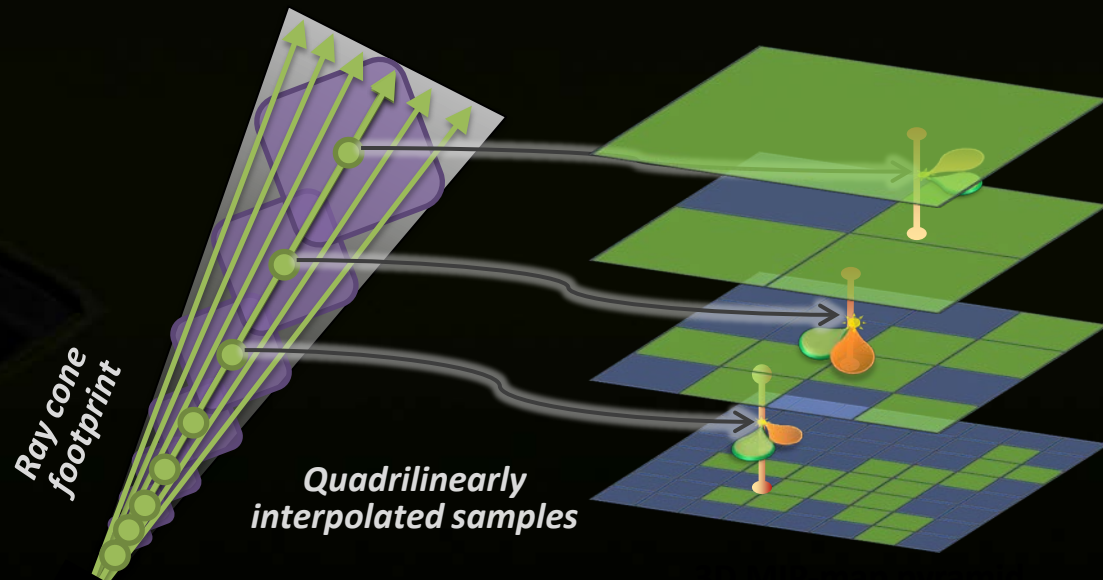
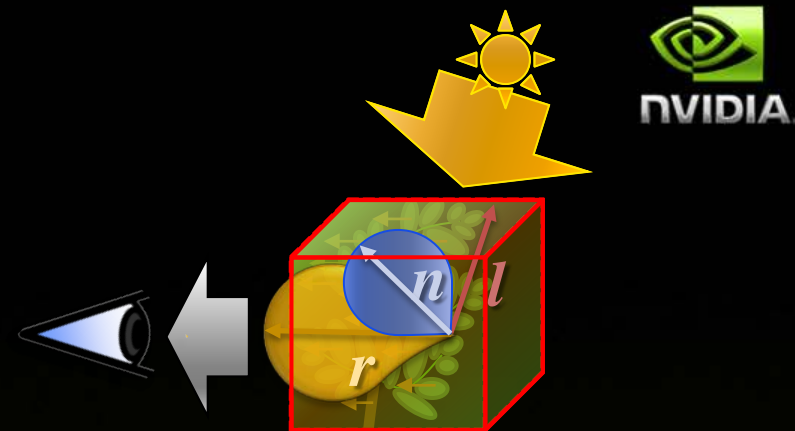
- **Hybrid rendering pipeline**
  - **Rasterized** primary rays
    - GPU pipeline optimized for direct visibility
  - **Cone-traced** secondary rays
    - Flexibility and scalability
- Forward or deferred rendering





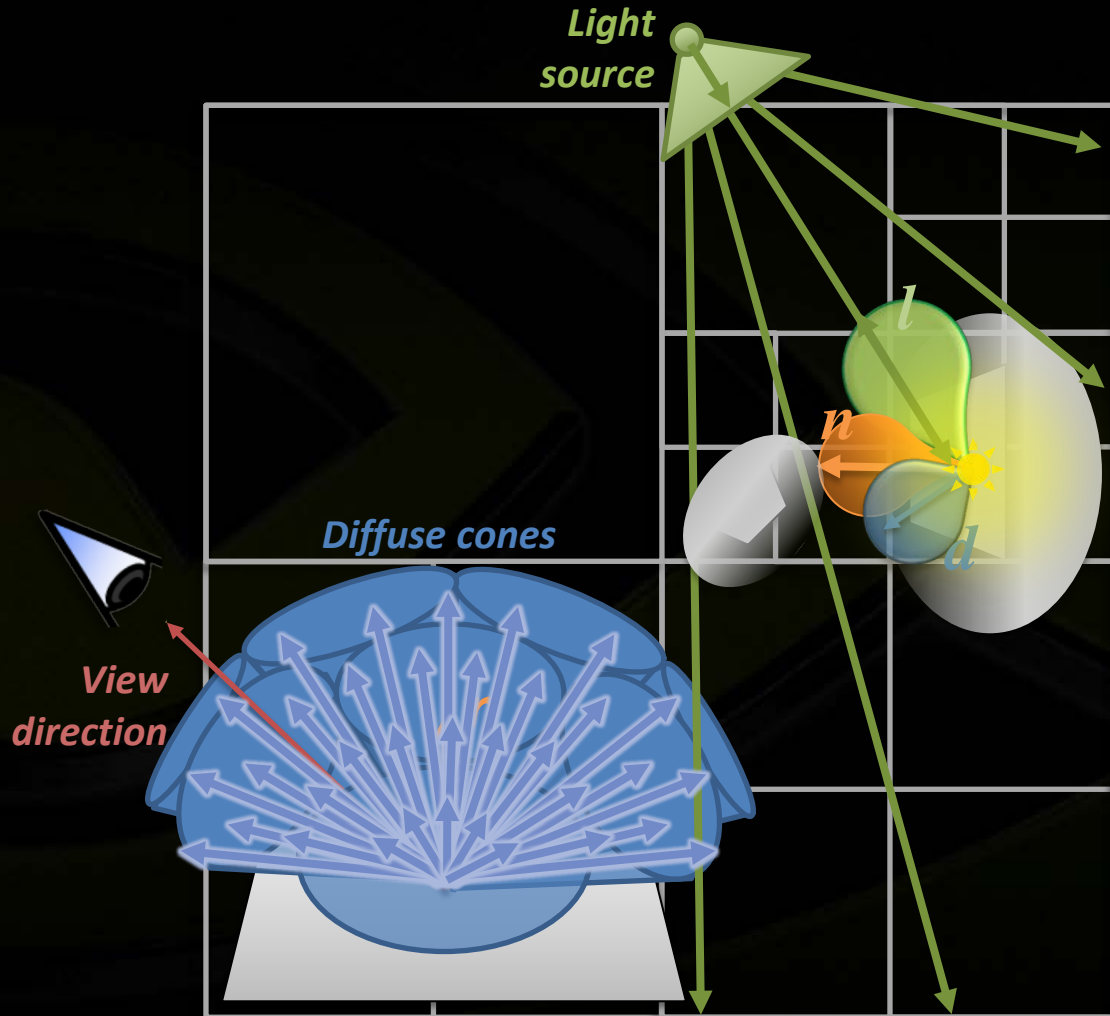
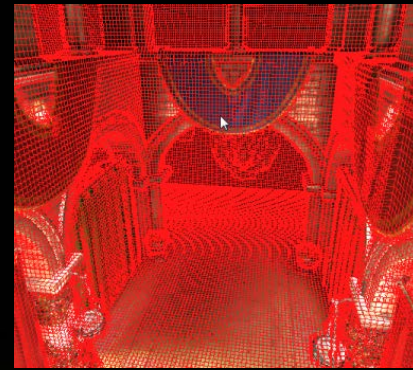
# Voxel cone tracing

- Geometry **pre-filtering**  
Traced like a participating media
  - Volume ray-casting
- Voxel representation  
Scene geometry : Opacity field  
+ **Incoming radiance**





# Rendering algorithm

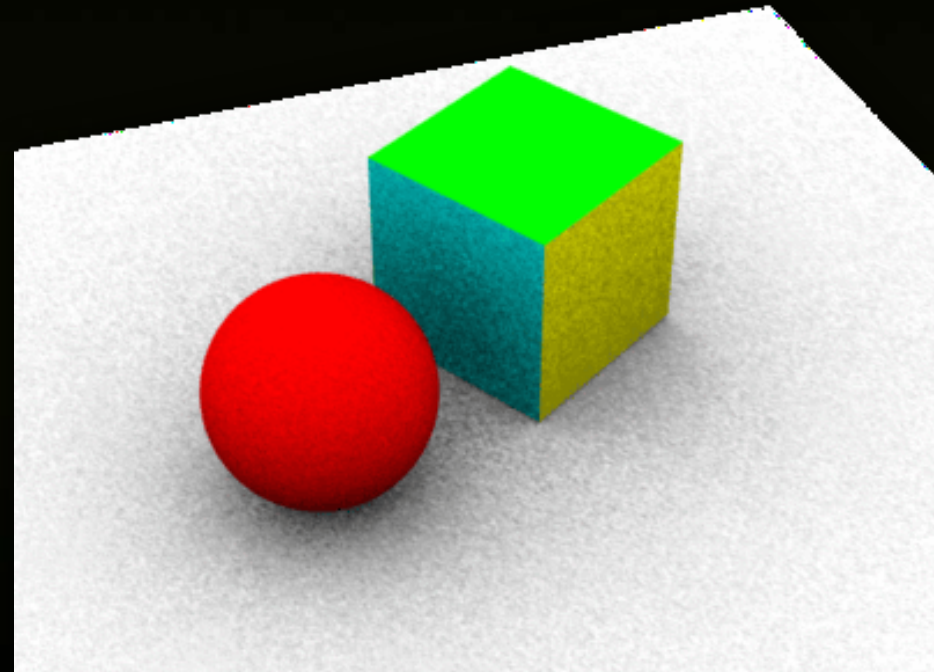


1. Light pass (es)
  - Bake irradiance (RSM)
2. Filtering pass
  - Down-sample radiance in the octree
3. Camera pass
  - For each visible fragment:  
Gather indirect radiance



# Discussion

- **Scalable** lighting rep. !
  - **Independent** of geometric complexity
  - **Control** over rendering time
- Large cones
  - Precision / Light leaking
- But **always smooth**
  - Never noisy !!





Indirect diffuse

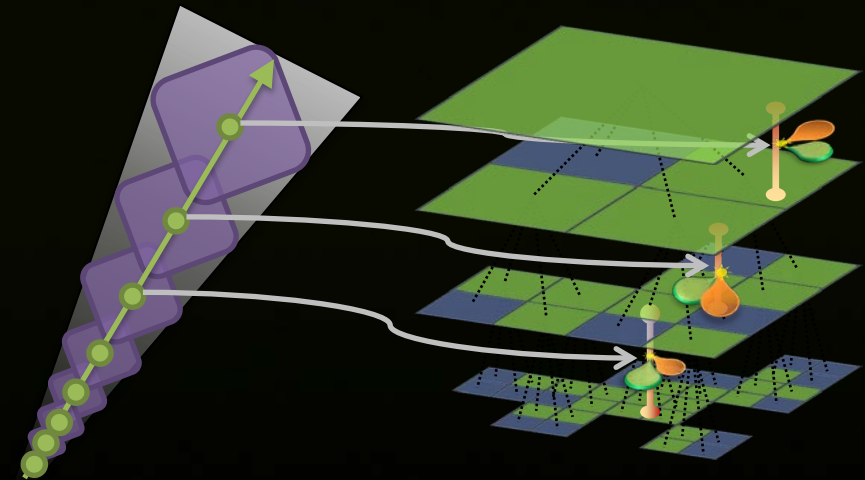
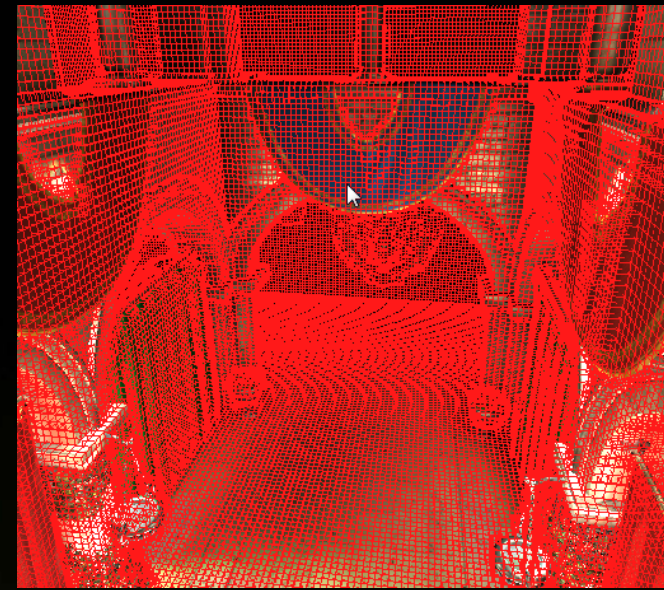
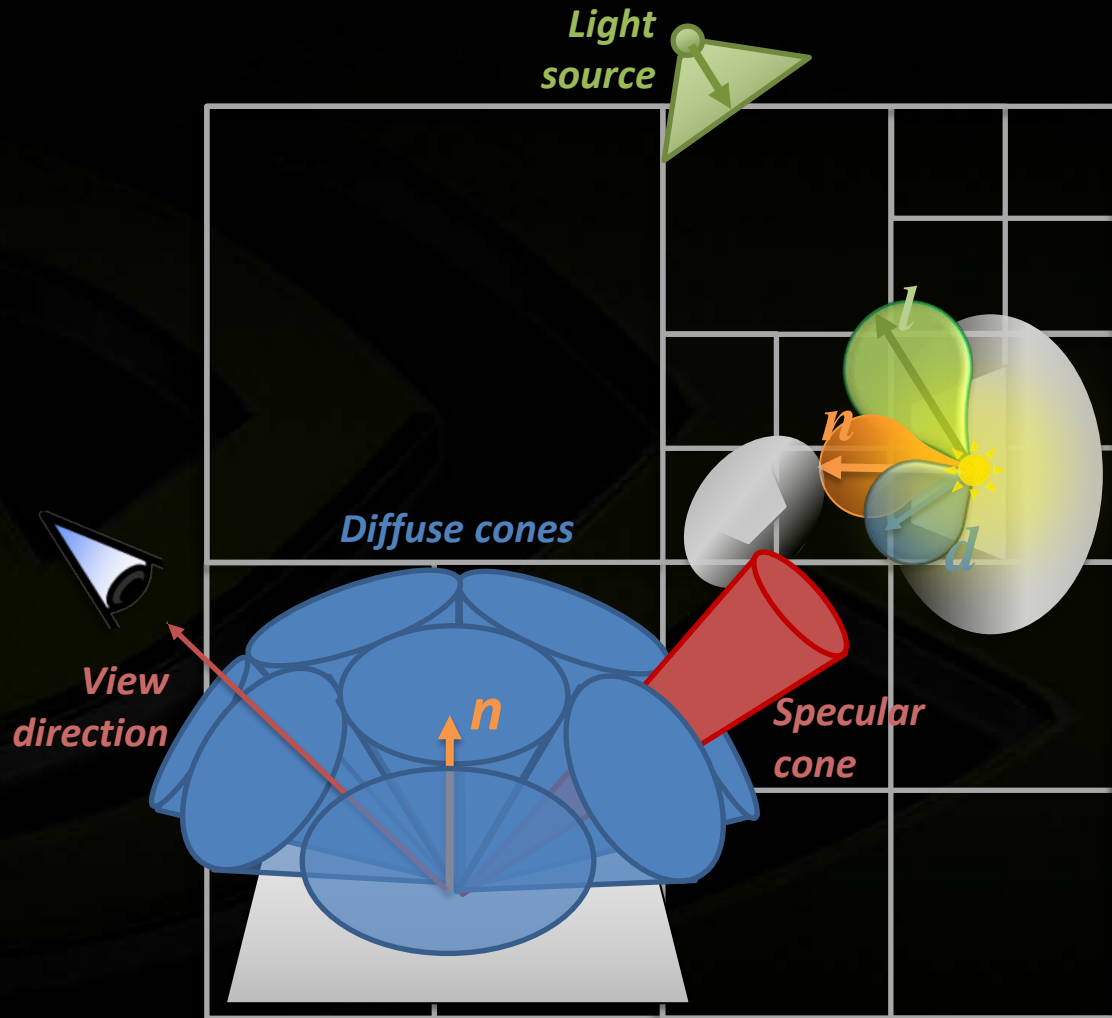


# Indirect diffuse

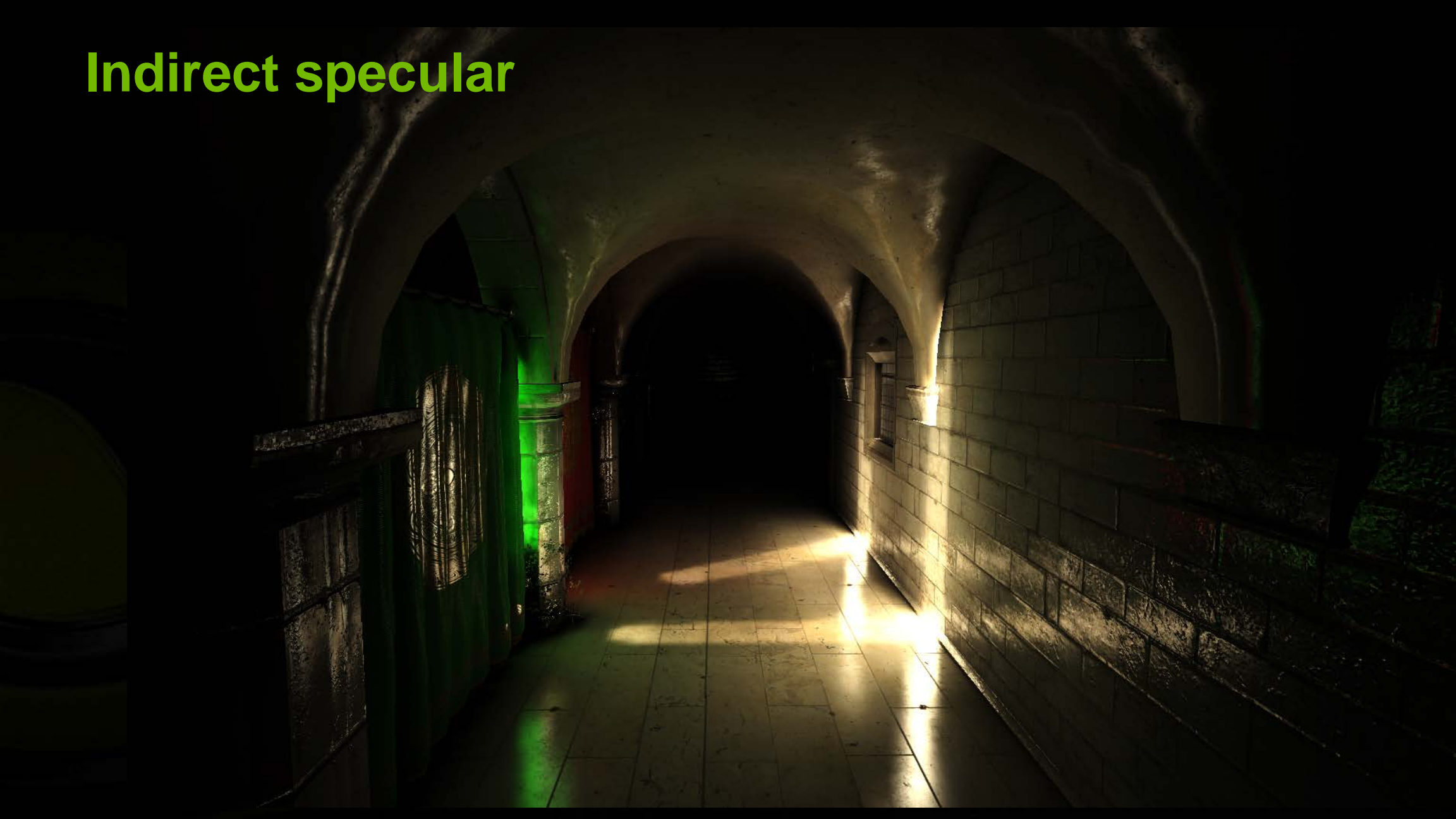




# Specular tracing



# Indirect specular





Indirect specular





Settings:

Indirect specular config ▾

Cone half-angle: 0.000

Radiance multiplier: 1.136





Settings:

Off



# Multiple-bounces

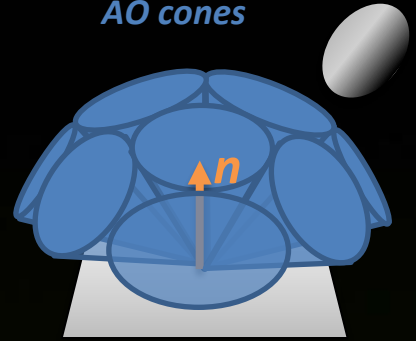




# Voxel Ambient Occlusion



AO cones



**+ Distant +  
off-screen  
occlusions**

**- Resolution**

**5.5ms @ 1280x720**

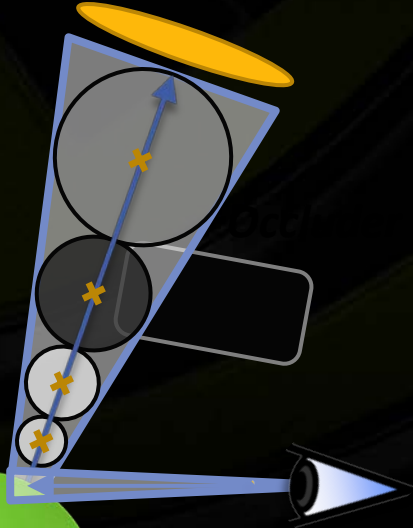
Scene model courtesy of Guillermo M. Leal Llaguno

# Voxel soft shadows

- One cone per pixel

*The **smoother**, the **faster** to compute !*

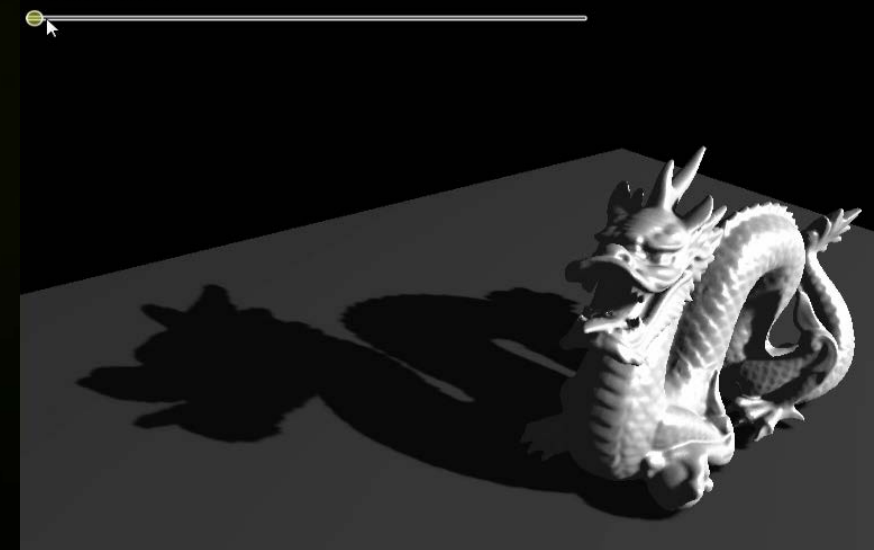
*Area light source*



*3-9ms @ 1280x720*

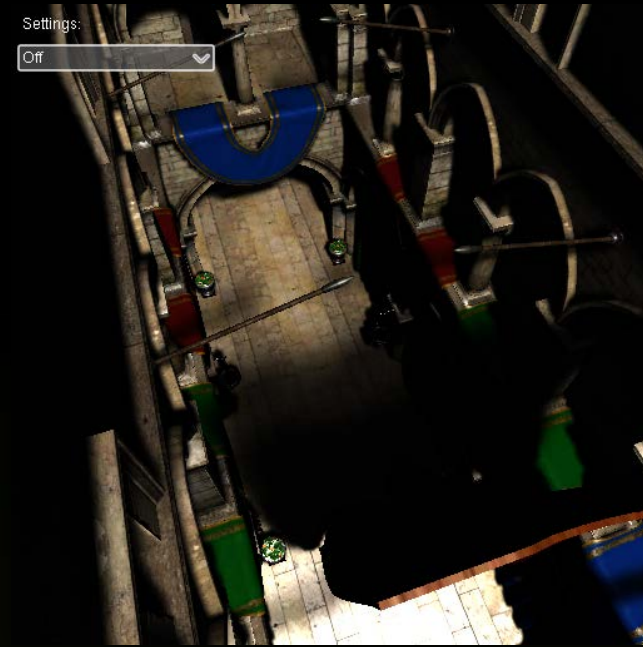


Cone half-angle: 0.000



Settings:

Off





# Publications

## Interactive indirect illumination using voxel cone tracing

C. Crassin, F. Neyret, M. Sainz, S. Green, E. Eisemann

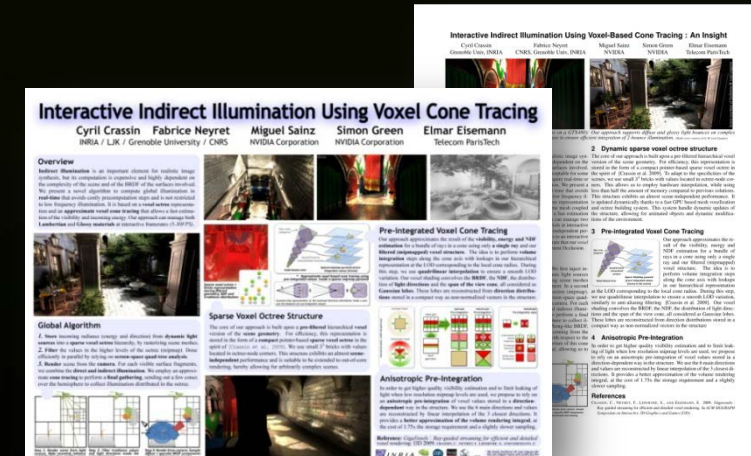
- Computer Graphics Forum (Proc. of Pacific Graphics 2011)
- <http://research.nvidia.com/publication/interactive-indirect-illumination-using-voxel-cone-tracing>

## I3D 2011 Poster

<http://maverick.inria.fr/Publications/2011/CNSGE11/>

## Siggraph 2011 Talk

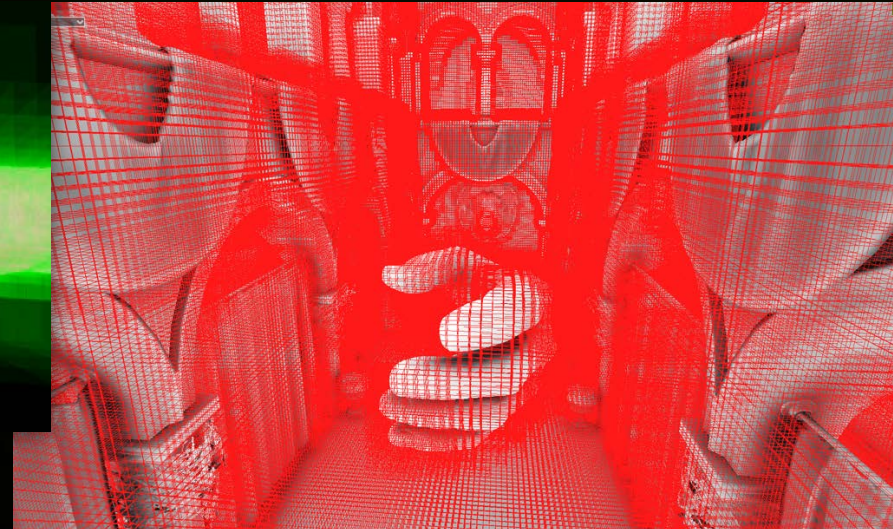
<http://maverick.inria.fr/Publications/2011/CNSGE11a/>



# Dynamic Voxelization



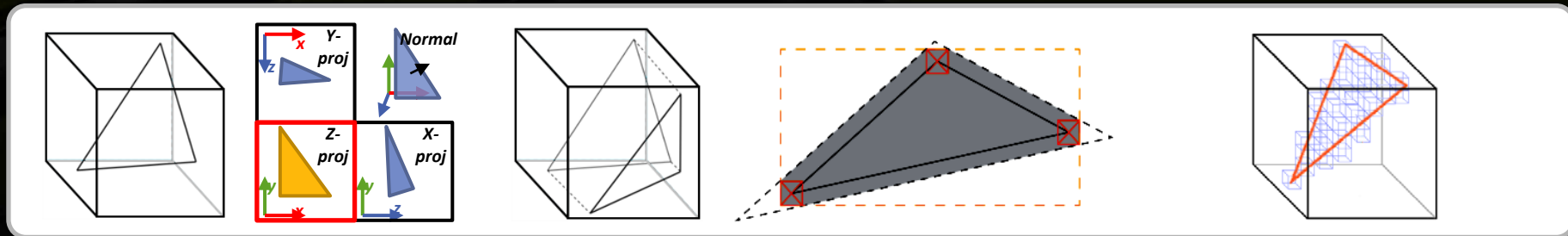
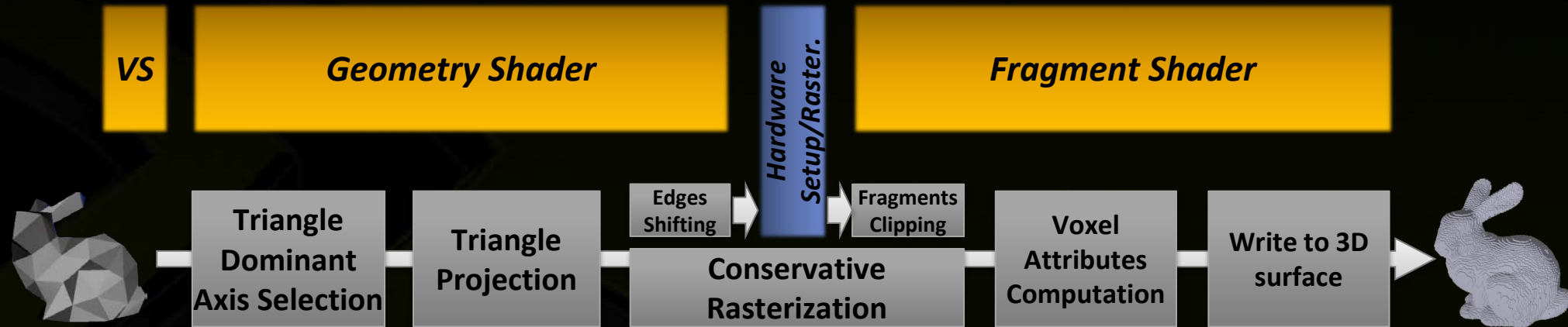
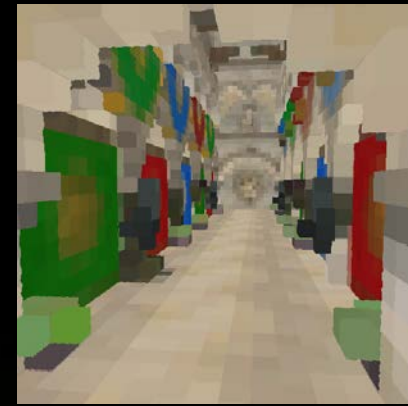
- Entirely done using the GPU graphics pipeline
  - Sparse (No plain grid allocation)
- Two modes :
  - Static environment
    - Pre-voxelized (~20ms)
  - Dynamic objects
    - Added to the structure at runtime (~4-5ms)





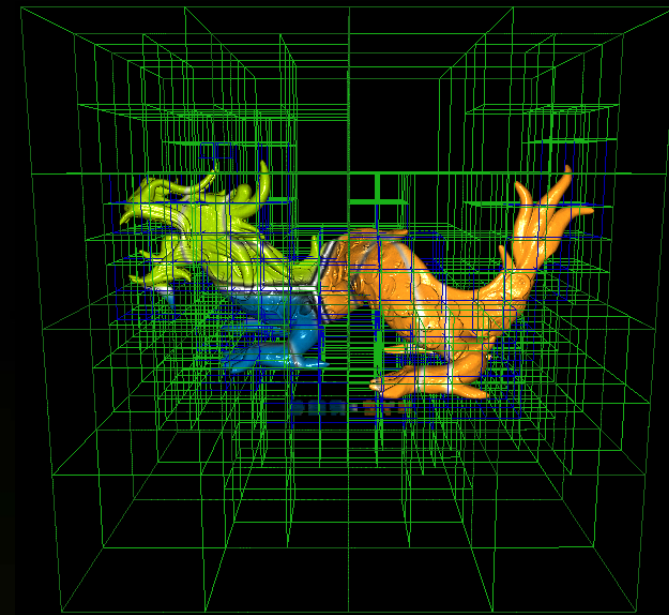
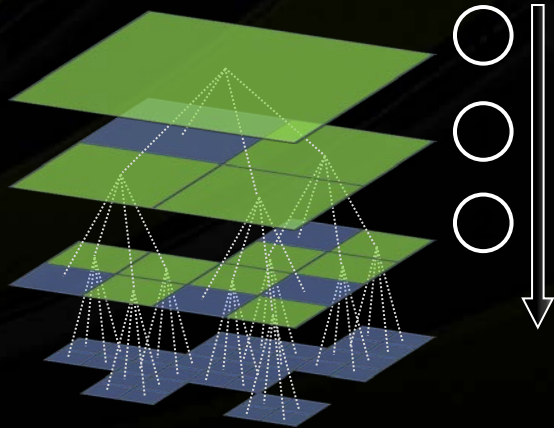
# One pass voxelization pipeline

- Thin surface voxelization

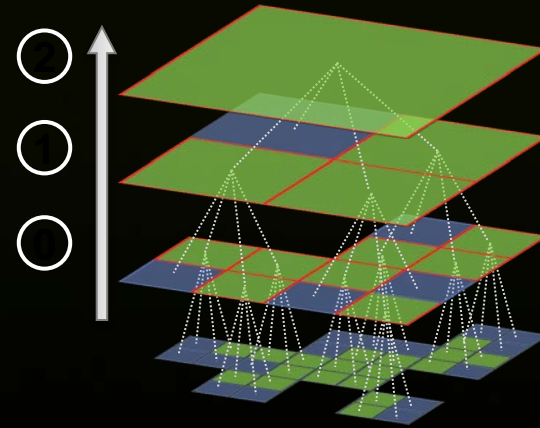


# Sparse Octree construction

- **Sparse voxelization**
  - No plain grid allocation
- **Two steps:**
  1. Octree subdivision



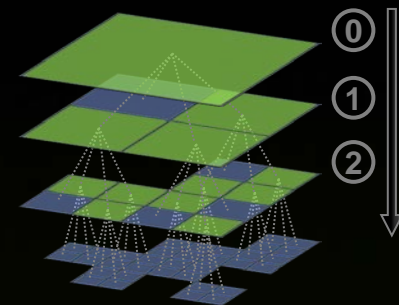
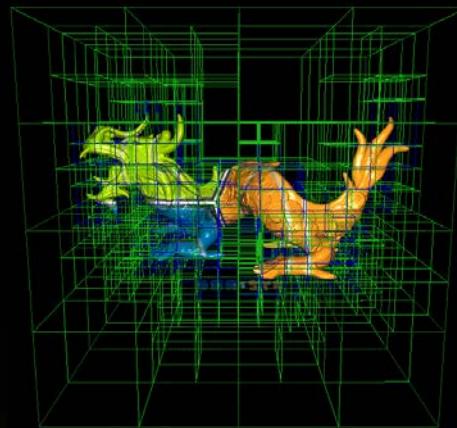
2. Values MIP-mapping





# Octree construction

- **Top-down** octree construction
  - Compute + Graphics



Voxelize Mesh at level  
resolution

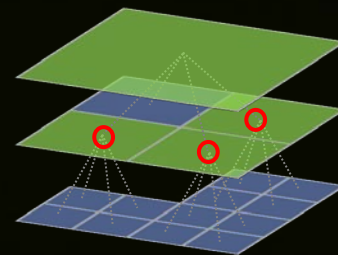
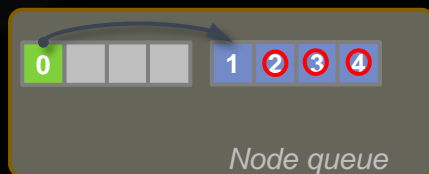
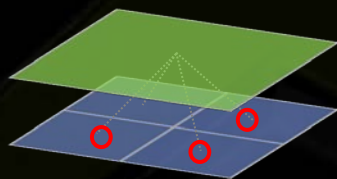
Tag octree nodes



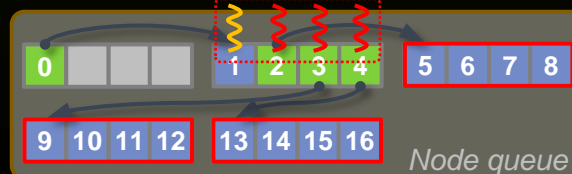
Create New Node Tiles



1 thread per  
voxel-fragment

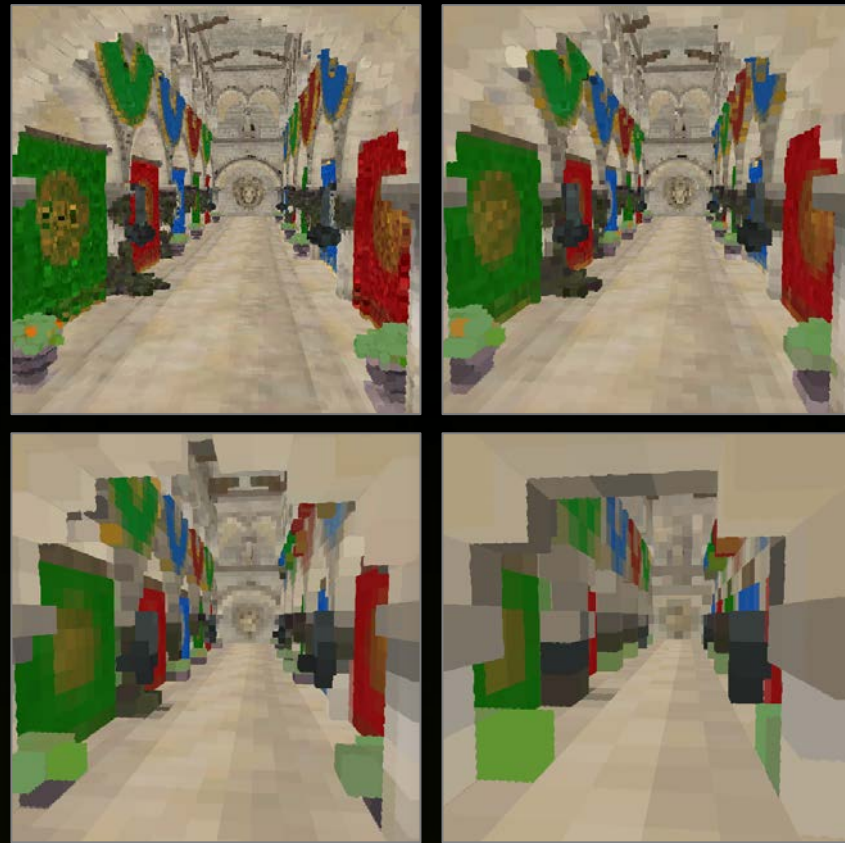


1 thread per node



# Results

- 9 levels octree (512^3)
  - RGBA32F
- Kepler GK104 performance
  - 30% - 58% faster than Fermi GF100
  - Atomic merging up to 80% faster.



<i>Times in ms</i>	Frag list	Octree construction				Write	MIP map	Total
Scene		Flag	Create	Init	Total			
Sponza	2.07	5.65	0.37	1.32	7.34	3.94	2.09	15.44



# OpenGL Insights

- *Octree-Based Sparse Voxelization Using The GPU Hardware Rasterizer*

Cyril Crassin and Simon Green

- *Just released at Siggraph 2012*

Patrick Cozzi & Christophe Riccio



**THANKS !**





# GTC 2013 | March 18-21 | San Jose, CA

The Smartest People. The Best Ideas. The Biggest Opportunities.

## Opportunities for Participation:

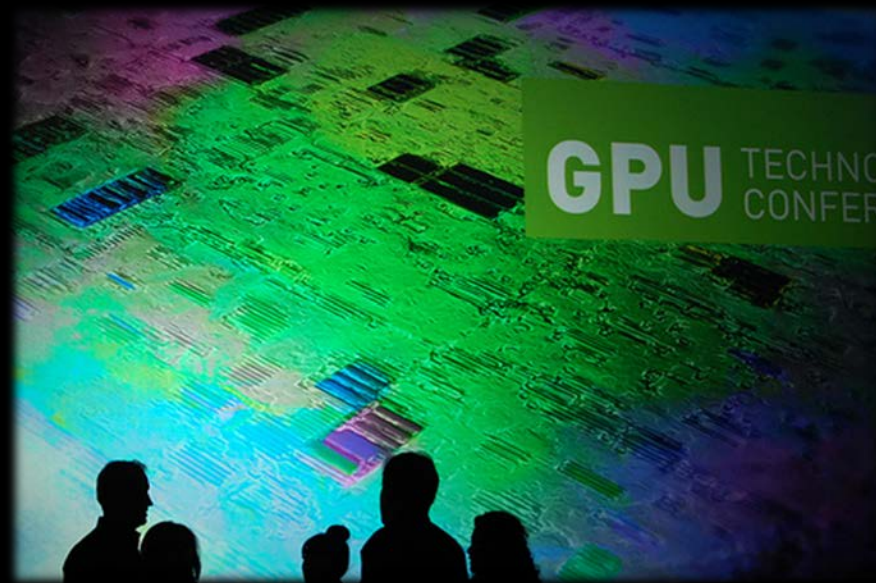
**SPEAK** - Showcase your work among the elite of graphics computing

- Call for Sessions: August 2012
- Call for Posters: October 2012

**REGISTER** - learn from the experts and network with your peers

- Use promo code **GM10SIGG** for a 10% discount

**SPONSOR** - Reach influential IT decision-makers



Learn more at [www.gputechconf.com](http://www.gputechconf.com)