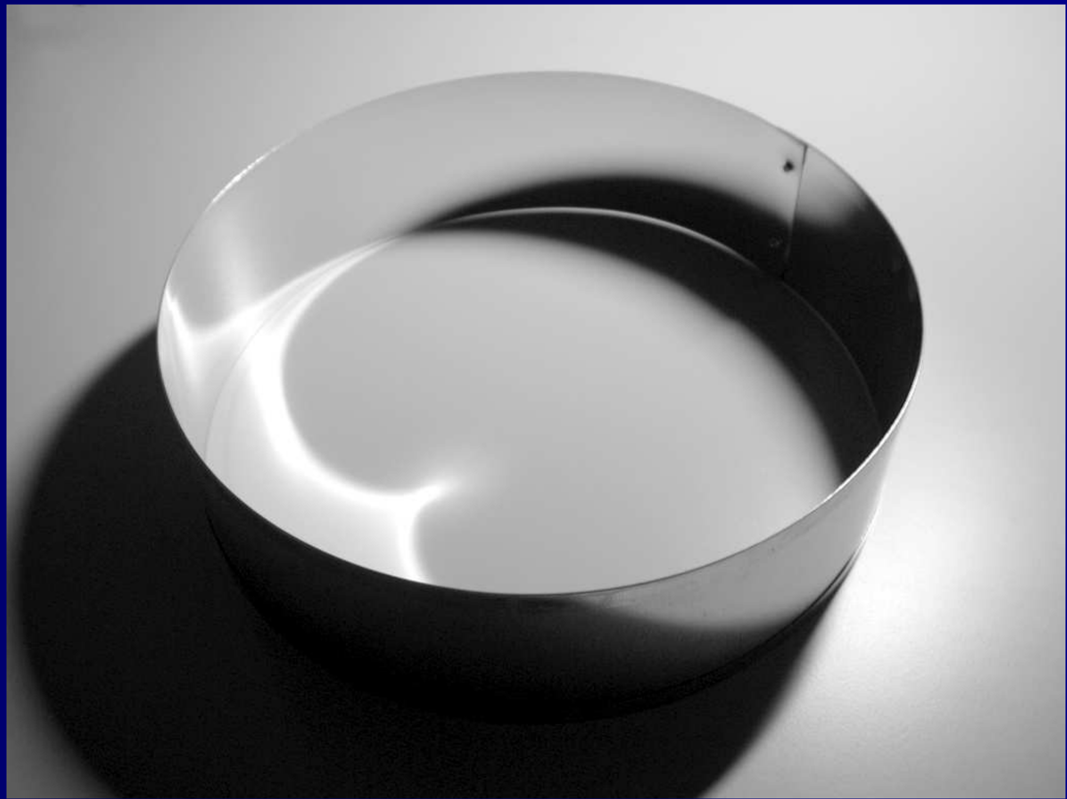
Introduction to Photon Mapping

RADIANCE Workshop 2010 – Course Advanced Fenestration

Roland Schregle

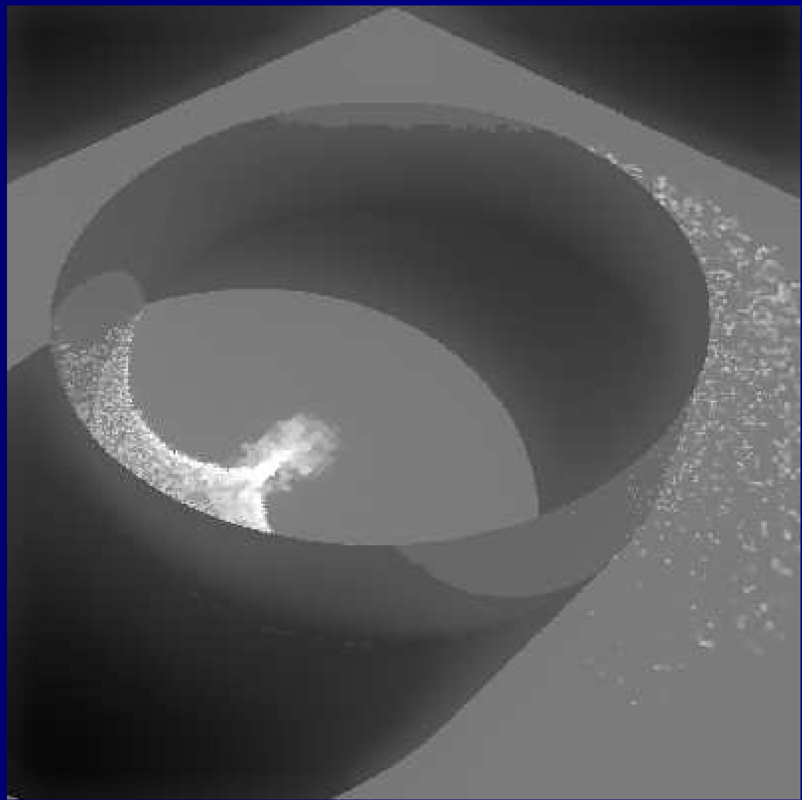
Motivation: Caustics

Light transport from specular surfaces gives rise to *caustics* on diffuse surfaces.



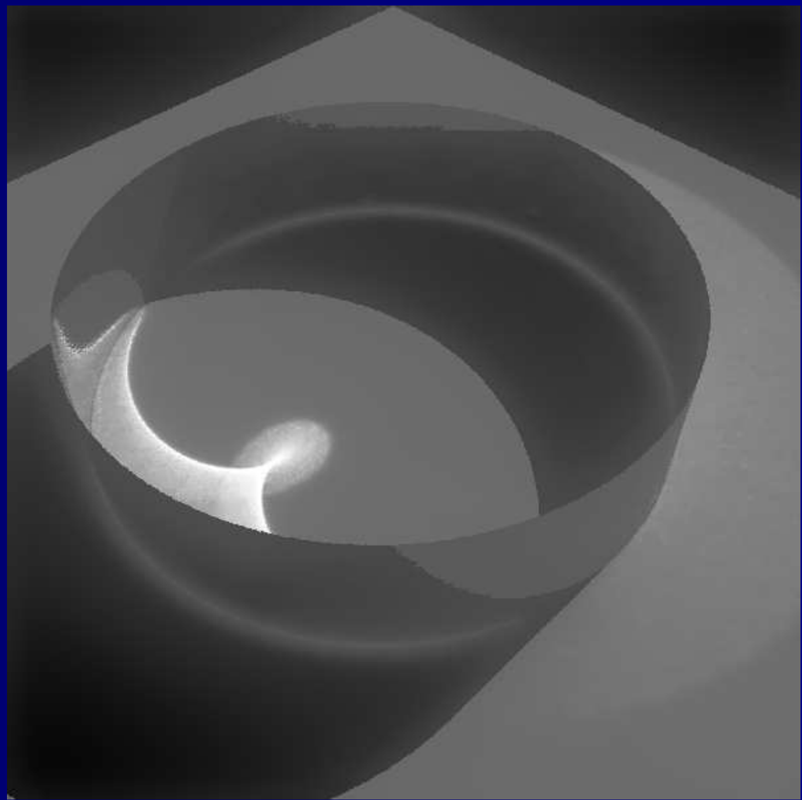
Motivation: Caustics

RADIANCE traces rays "backwards"
[viewer $\rightarrow$ objects $\rightarrow$ sources] $\Rightarrow$ inefficient and noisy.



Motivation: Caustics

Supplement RADIANCE with forward raytracer!
[sources $\longrightarrow$ objects $\longleftarrow$ viewer]



Overview

- Two-pass method based on Monte Carlo (light) particle transport simulation [Wann Jensen 1995]

  Forward pass emits photons from light sources, scatters/absorbs at objects, deposits on diffuse surfaces $\Rightarrow$ `mkpmap`.

  Backward pass evaluates irradiance from photons using RADIANCE’s ambient calculation $\Rightarrow$ `rpict/rtrace/rvu`.

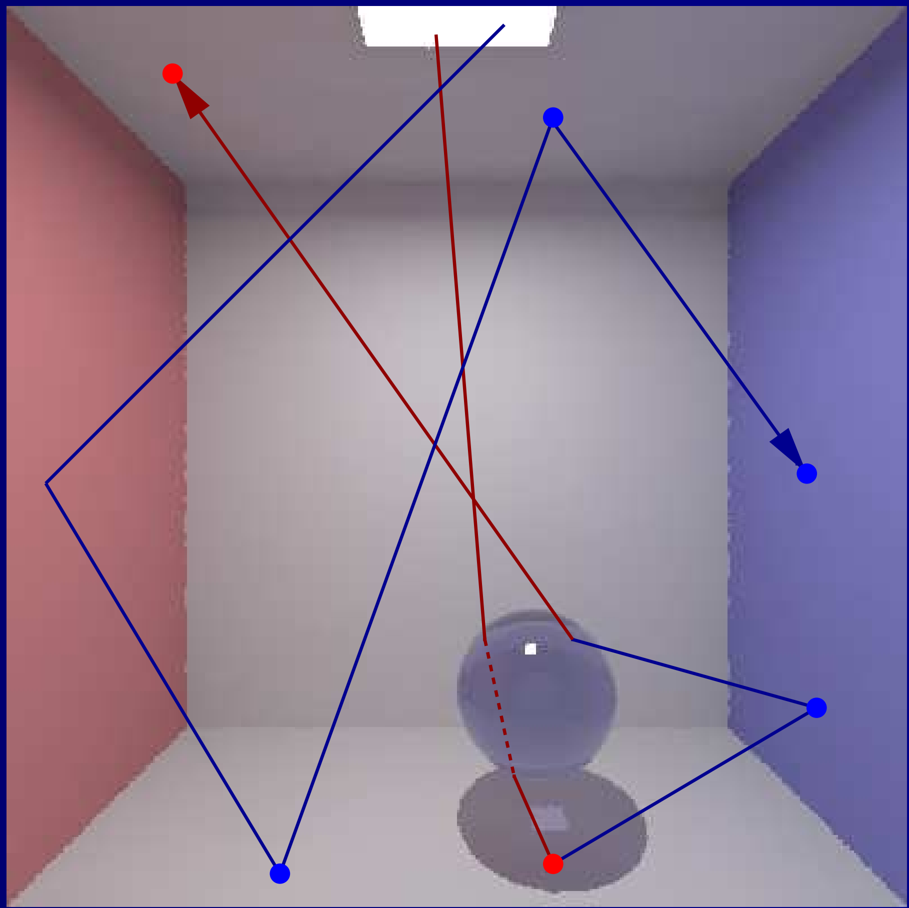
- Photometrically validated [Schregle/Wienold 2004]

Forward Pass

- Photons emitted from light sources, scattered at surfaces according to material properties until absorbed (*russian roulette*).
- Indirect photon hits on diffuse surfaces stored in kd-tree along with position $\vec{x}$, flux Φ, normal $\vec{N}$.
- global photons stored on every indirect $[$diffuse$|$specular $\longrightarrow$ diffuse$]$ hit.
- caustic photons stored on every indirect $[$specular $\longrightarrow$ diffuse$]$ hit.
- volume photons stored in participating media (`mist`).

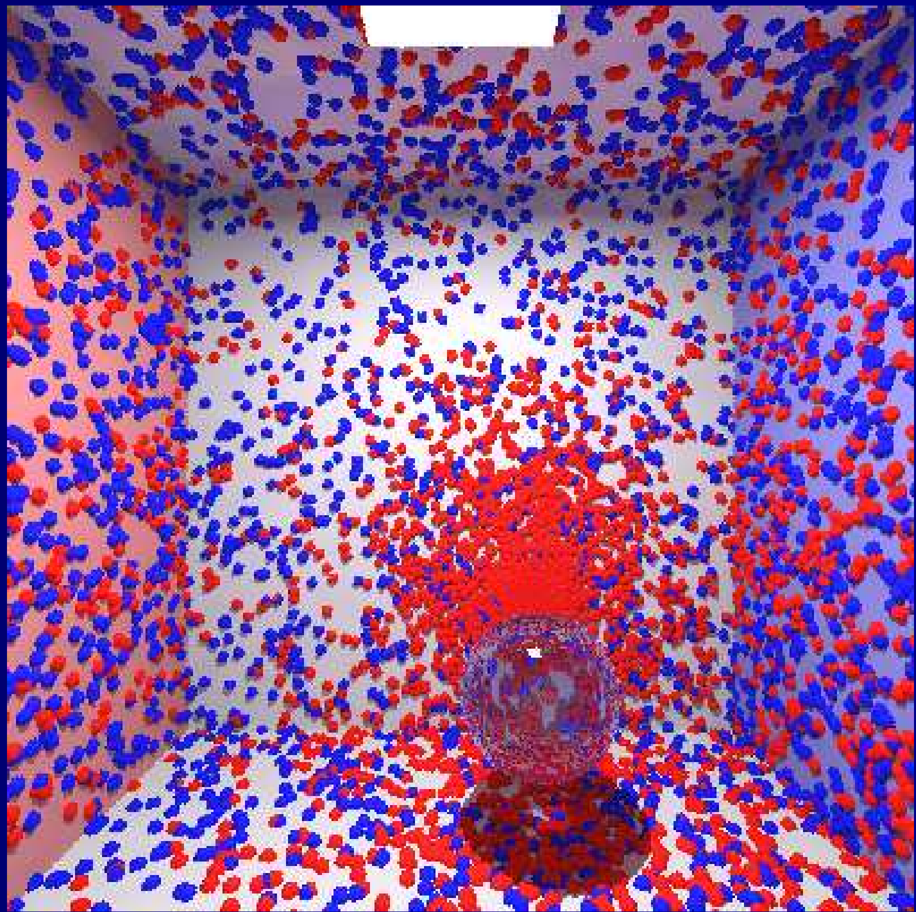
Forward Pass

→ Global → Caustic



Forward Pass

→ Global → Caustic



mkpmap

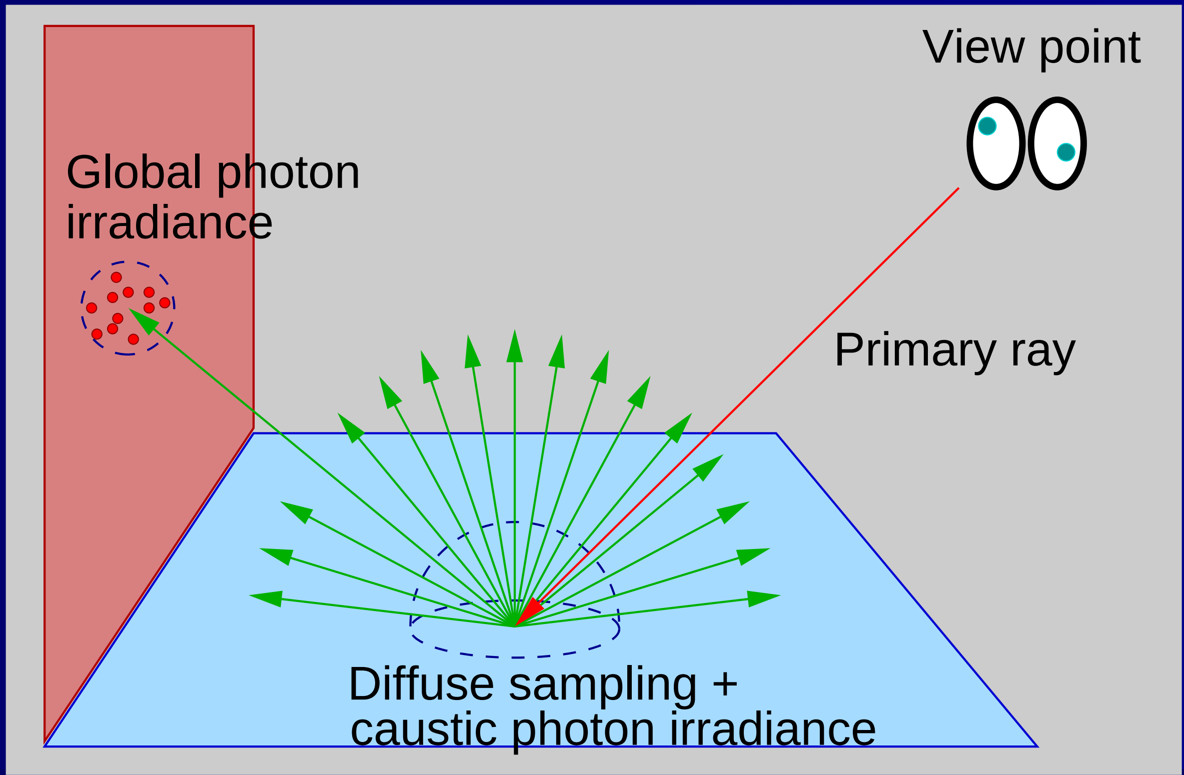
`mkpmap` performs forward pass and generates photon maps, e.g.

```
mkpmap -apg scene.gpm 10k -apc
scene.cpm 50k -apv scene.vpm 100k -t
60 scene.oct
```

generates global, caustic, and volume photon map files scene.{gpm,cpm,gpm} of approx. 10k, 50k, and 100k photons resp, with progress report every 60s.

Backward Pass

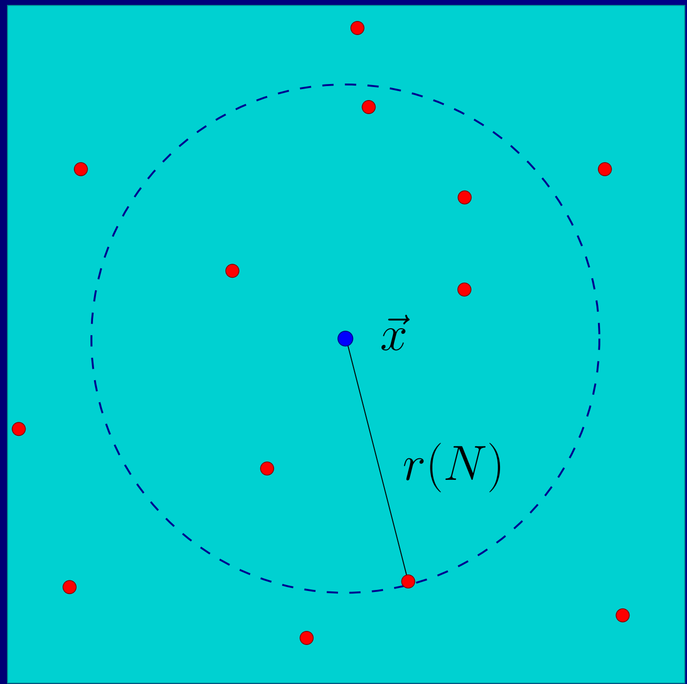
- Global photon irradiance after single ambient bounce
- Caustic photon irradiance at primary hitpoint



Density Estimation

Irradiance $\hat{f}$ is estimated from photon density:

- Find nearest N photons at $\vec{x}$ in kd-tree



Density Estimation

Irradiance $\hat{f}$ is estimated from photon density:

- Find nearest N photons at $\vec{x}$ in kd-tree
- Sum photon flux Φ_i weighted by normalised kernel K:

$$\hat{f}\left(\vec{x}\right) = \sum_{i=1}^{N} K\left(\|\vec{x} - \vec{x}_i\|\right) \Phi_i$$

rpict / rtrace / rvu

`rpict` et al. performs backward pass using photon maps, e.g.

```
rpict -ab 1 -apg scene.gpm 50 -apc
scene.cpm 50 -apv scene.vpm 50 ...
scene.oct
```

Loads global, caustic, and volume photon maps and computes irradiance using 50 photons per density estimate each.

Photon maps may be reused for multiple viewpoints!

rpict / rtrace / rvu

-ab has nonstandard behaviour:

-ab $>$ 0: global photon irradiance via *one* ambient bounce (regardless of actual -ab)

-ab $<$ 0: no ambient bounce, global photon irradiance at primary hitpoint

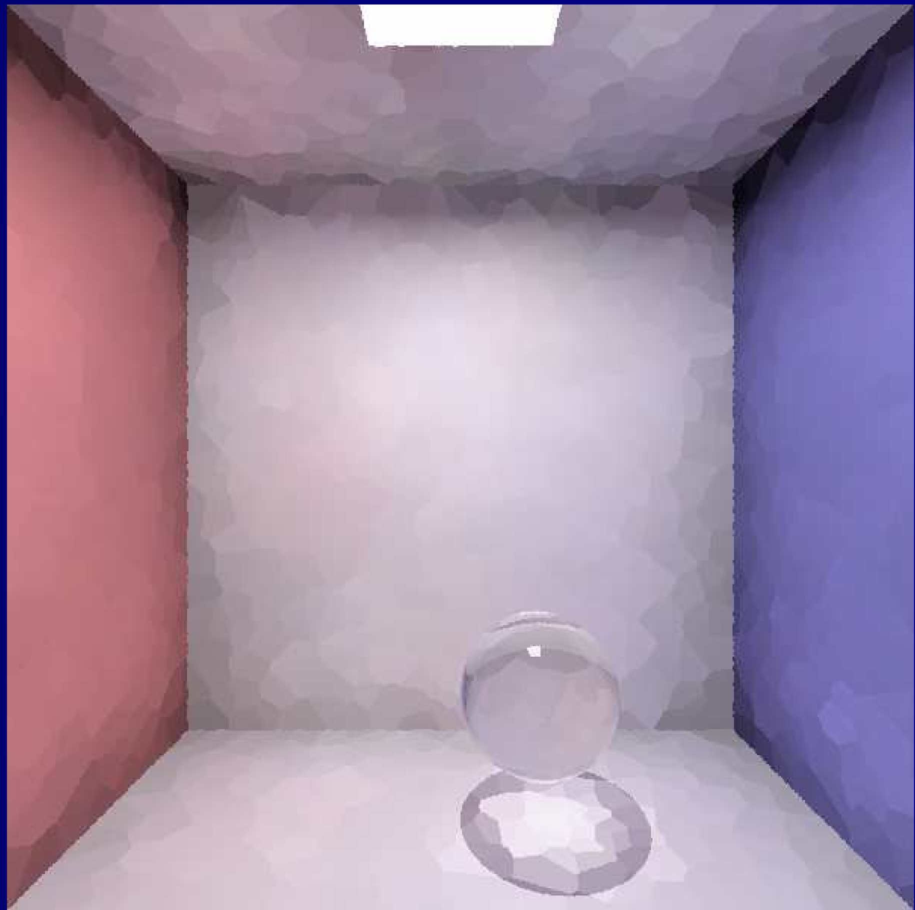
-ab $=$ 0: no ambient component at all, caustics at primary hitpoint

Optimisations: Final Gather

Precompute and store irradiance for a fraction of global photons after forward pass [Christensen 2000]

- PRO: only look up single closest photon during backward pass and use precomputed irradiance directly
- CON: inaccurate with nonuniform ambient illum

Optimisations: Final Gather



Precomp. global photons rendered with -ab < 0

Optimisations: Final Gather

```
mkpmap -app scene.pgpm 40k 50 -apf
0.25 scene.oct
```

distributes 40k global photons, precomputes irradiance for 0.25 $*$ 40k = 10k (discarding rest) using 50 photons / density estimate, and outputs them to photon map file.

```
rpict -ab 1 -app scene.pgpm ...
scene.oct
```

renders using closest precomputed global photon

Optimisations: Bias Compensation

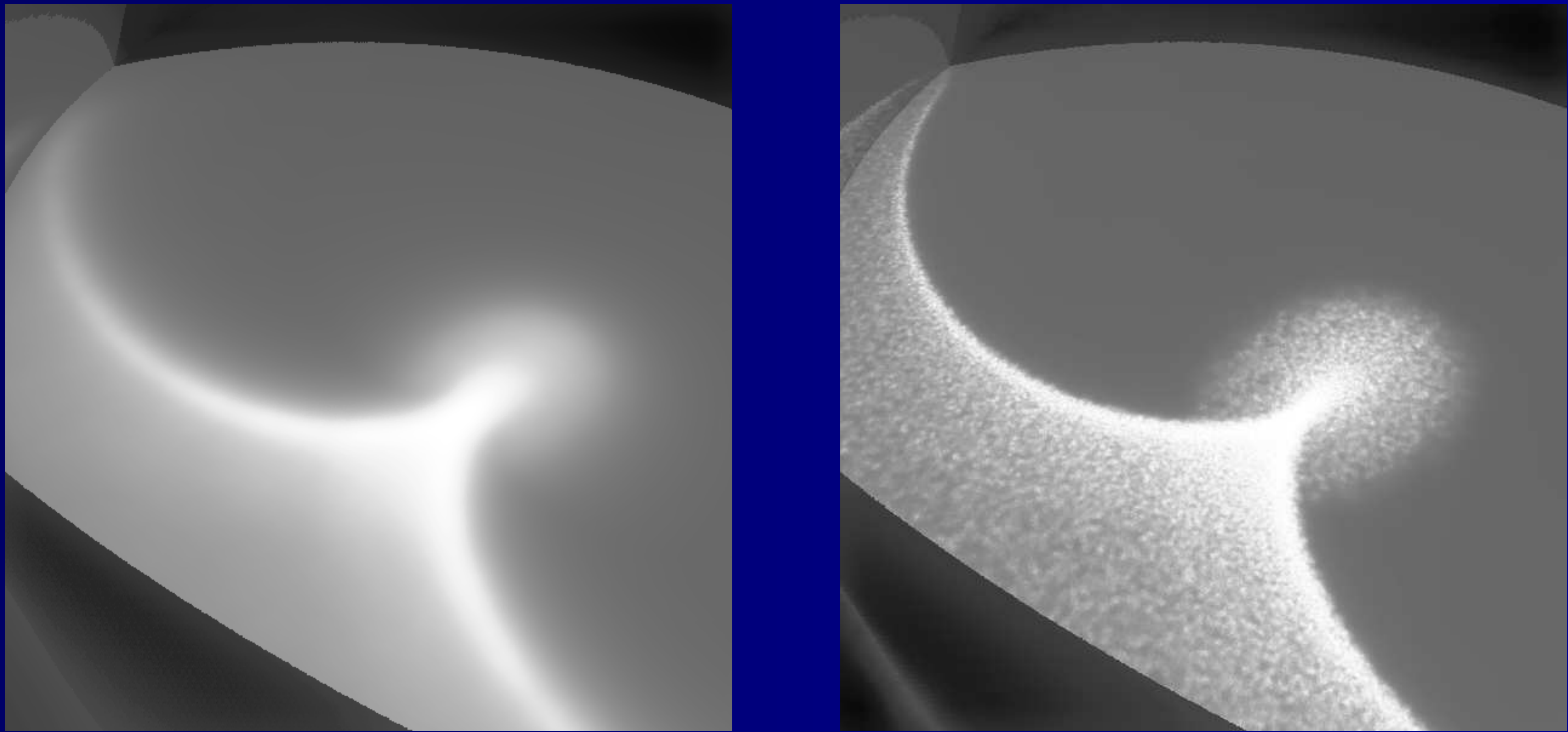
Density estimation is inherently biased; photon irradiance $\hat{f}$ converges to actual irradiance f <u>convolved with K</u> rather than f proper:

$$E\left[\hat{f}\left(\vec{x}\right)\right] = \int_{\{\vec{y}:\|\vec{x}-\vec{y}\|\leq r(N)\}} K\left(\|\vec{x}-\vec{y}\|\right) f\left(\vec{y}\right) d\vec{y}$$

Bias visible as blur, esp. in caustics and with large N

BUT... reducing N increases noise!

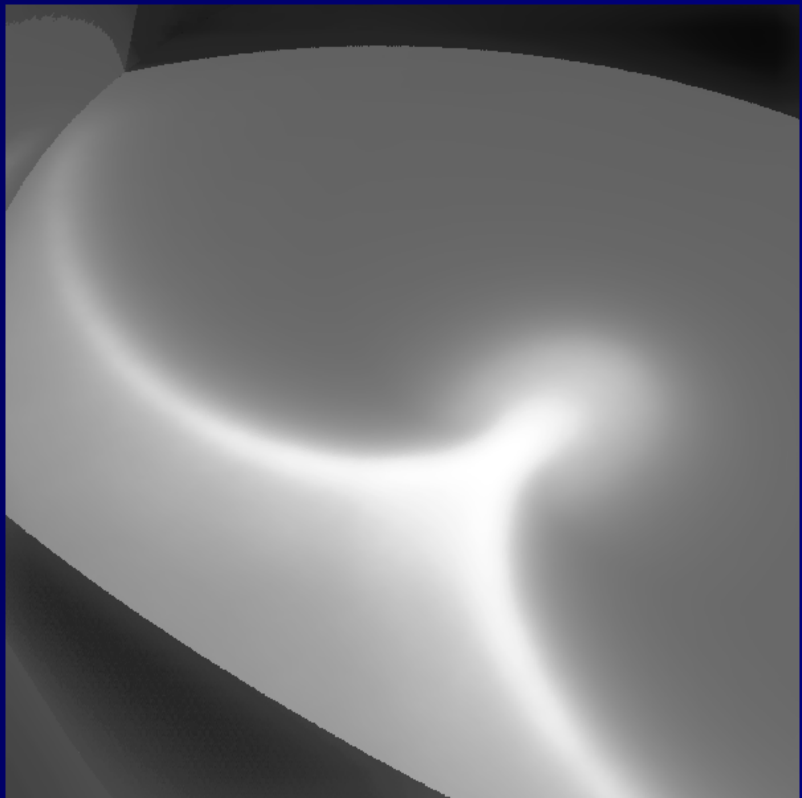
Optimisations: Bias Compensation

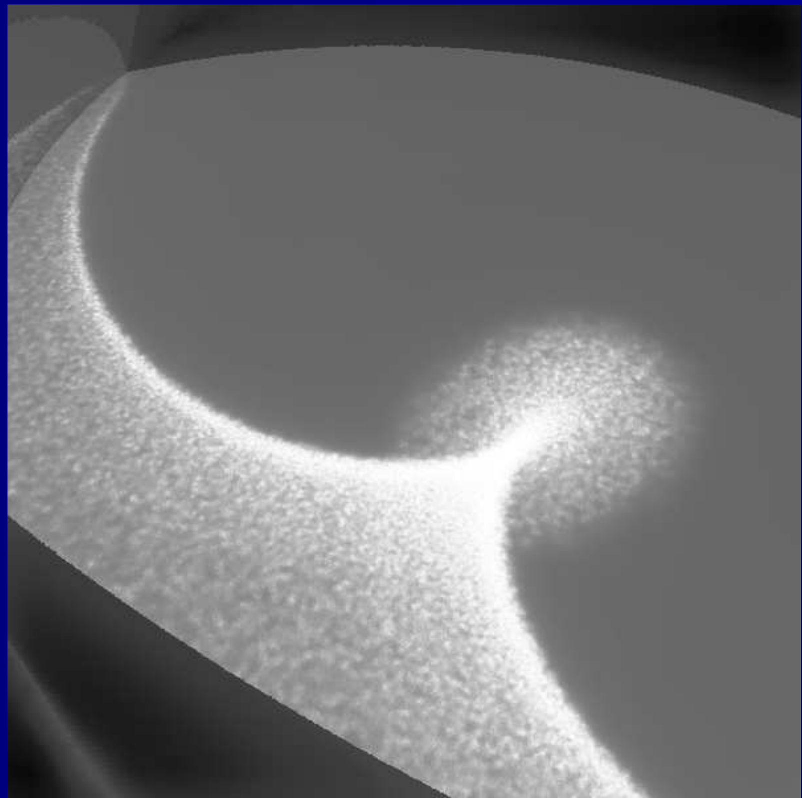


N=2000 N=20

Bias/noise tradeoff depends on photons/estimate N

Optimisations: Bias Compensation





⇒Dynamically adapt N according to estimated bias in given interval $[N_{min}, N_{max}]$!

Optimisations: Bias Compensation

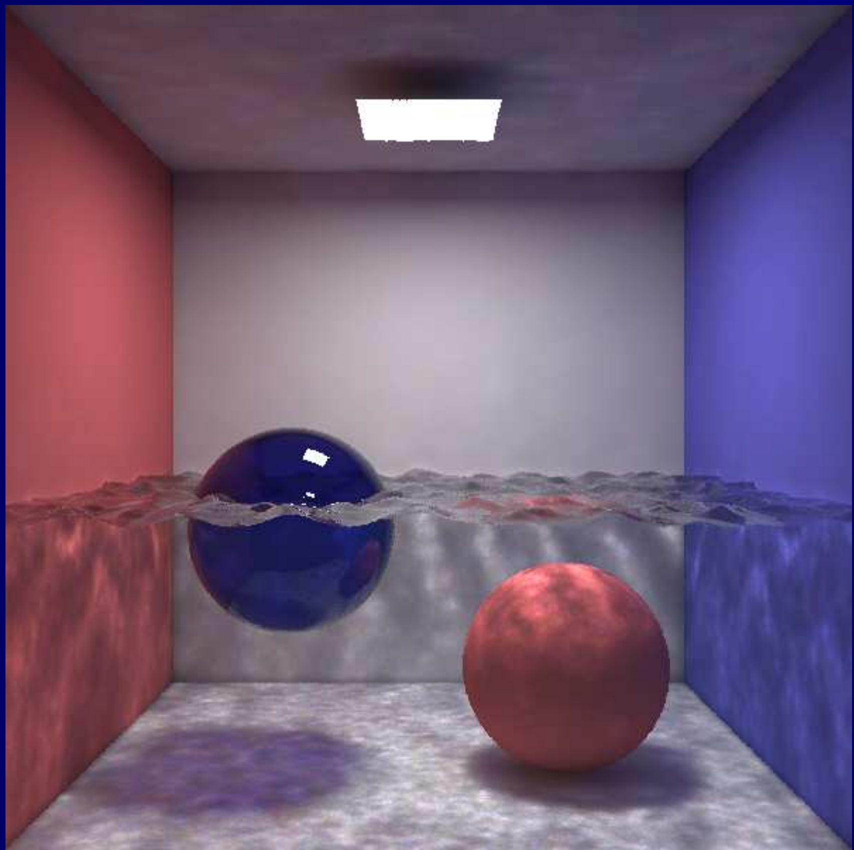
Bias Compensation adapts photons/density estimate N to minimise bias *and* noise [Schregle 2003]

- Maintains running mean and variance of density estimates to estimate bias
- Increases N until *probable* bias is detected

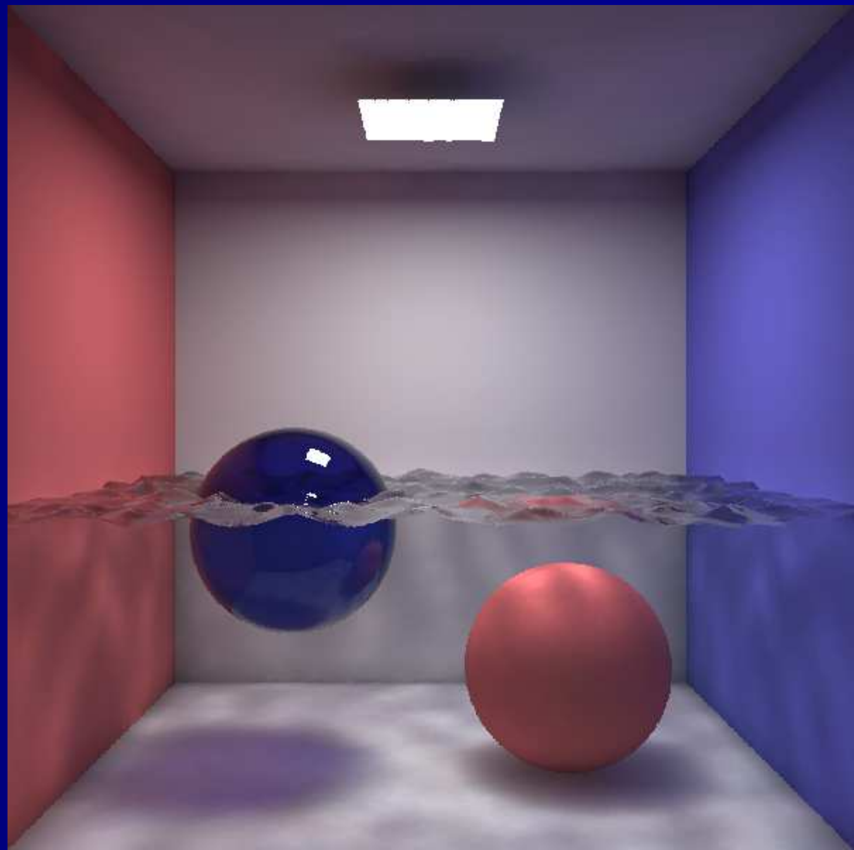
```
rpict ...  -apcb scene.cpm 50 500
...  scene.oct
```

renders caustics with bias compensation using 50..500 photons/density estimate.

Optimisations: Bias Compensation

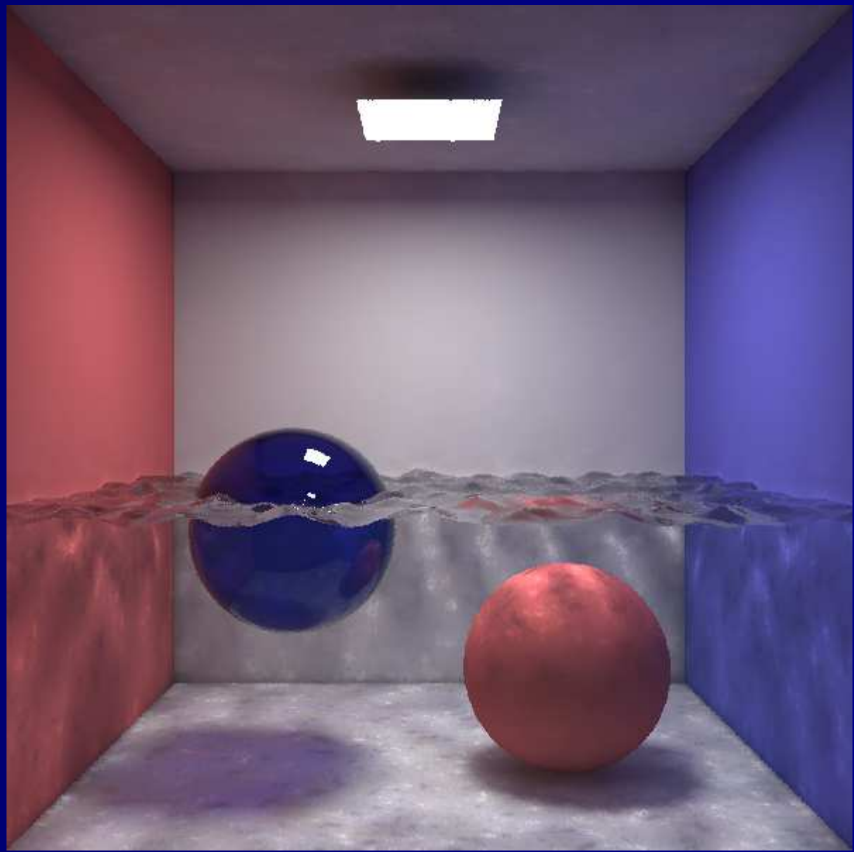


N=50 caustic/estimate



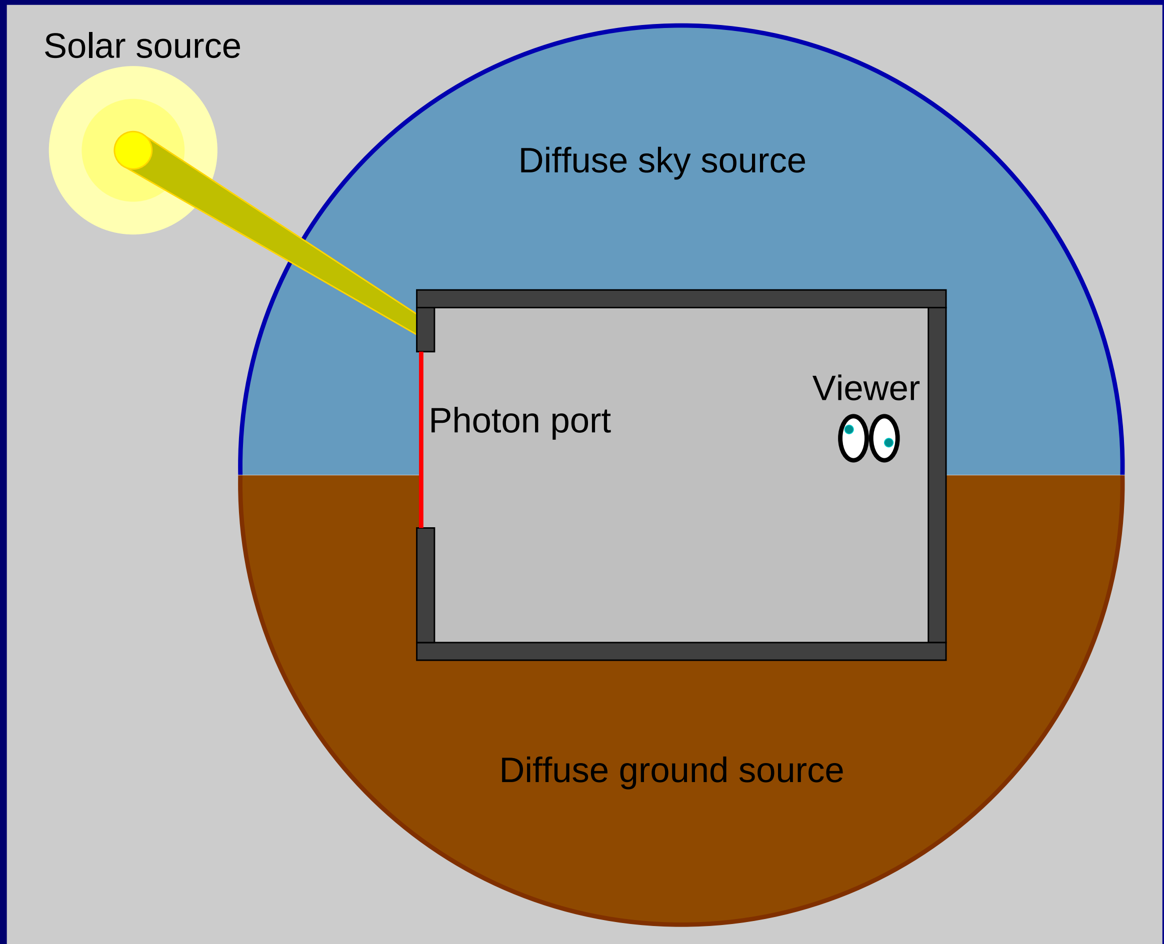
N=500 caustic/estimate

Optimisations: Bias Compensation



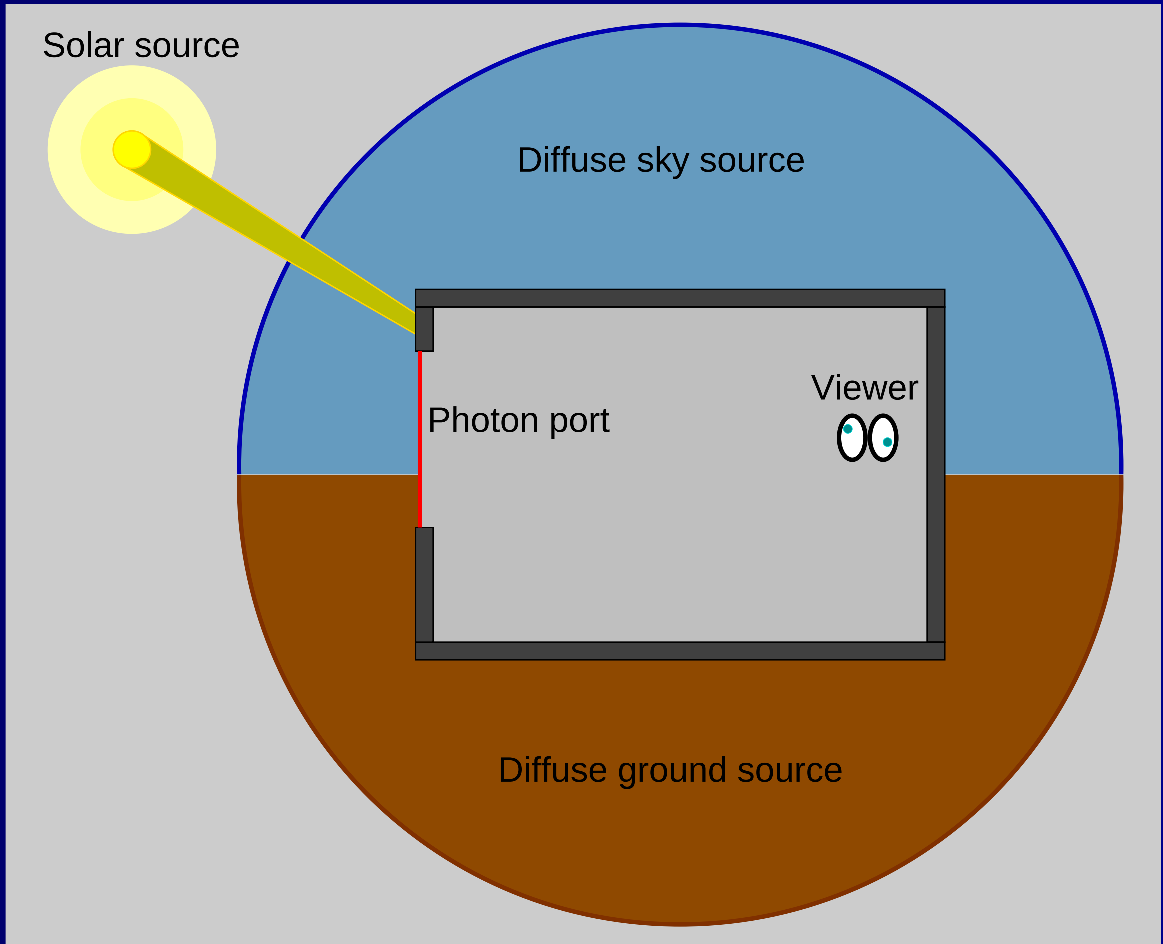
Bias compensated, 50..500 caustic/estimate

Optimisations: Photon Ports



Few photons emitted from source reach viewer (slow)

Optimisations: Photon Ports



⇒ Emit photons directly from window!

Optimisations: Photon Ports

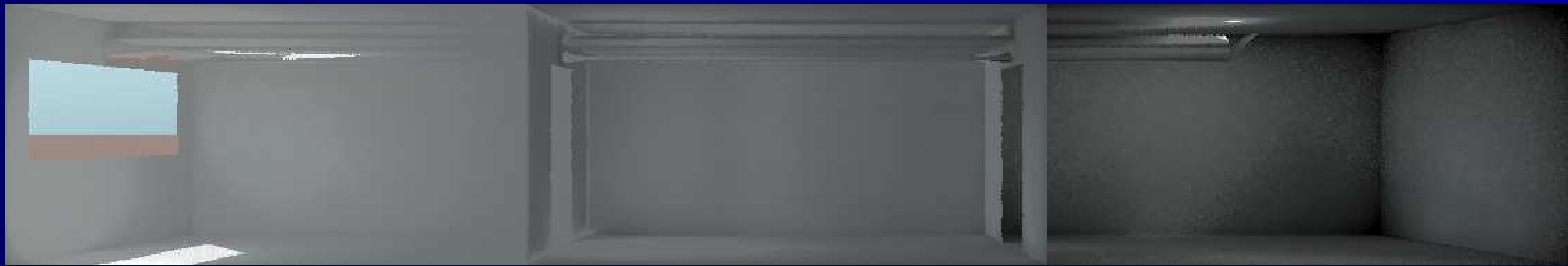
- Ports define apertures as part of regular scene geometry for photon emission from sources
- Flux incident from sources and occlusion taken into account
- Photons emitted directly from port object

Ports are specified by modifier, e.g.

```
mkpmap -apg scene.gpm 100k -apo
windowMat scene.oct
```

distributes photons from all objects using the windowMat material in scene.oct.

Optimisations: Photon Ports



62k precomp. global
250k caustic
Port at pipe aperture
(antimatter material)

Utilities: pmapinfo

`pmapinfo <pmapFile>` gives following info on a photon map:

- photon map type (global, caustic, volume, etc)
- `mkpmap` command line used in forward pass
- Number of photons in map
- Average spectral photon flux

Utilities: pmapdump

`pmapdump <radius> <pmapFile>...<pmapFile>`
generates scene description of photons as spheres for visualisation of distribution in original scene, e.g.

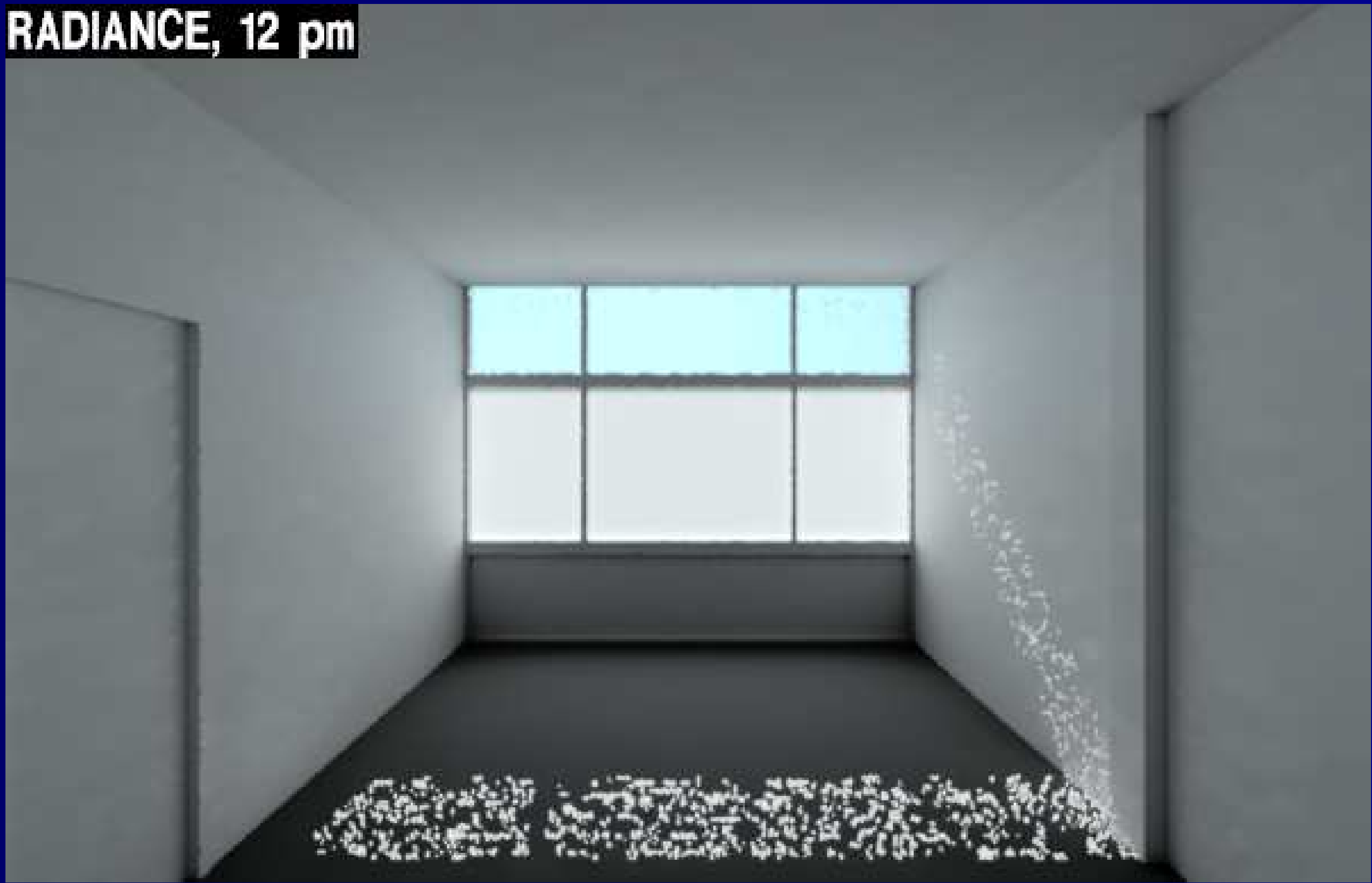
```
% oconv scene.rad > scene.oct
% mkpmap -apg pmap.gpm -apc pmap.cpm
scene.oct
% pmapdump 0.01 pmap.gpm pmap.cpm |
oconv -n 16 - scene.rad >
scene-distr.oct
```

Useful for debugging with *small* number of photons.

Utilities: pmapdump

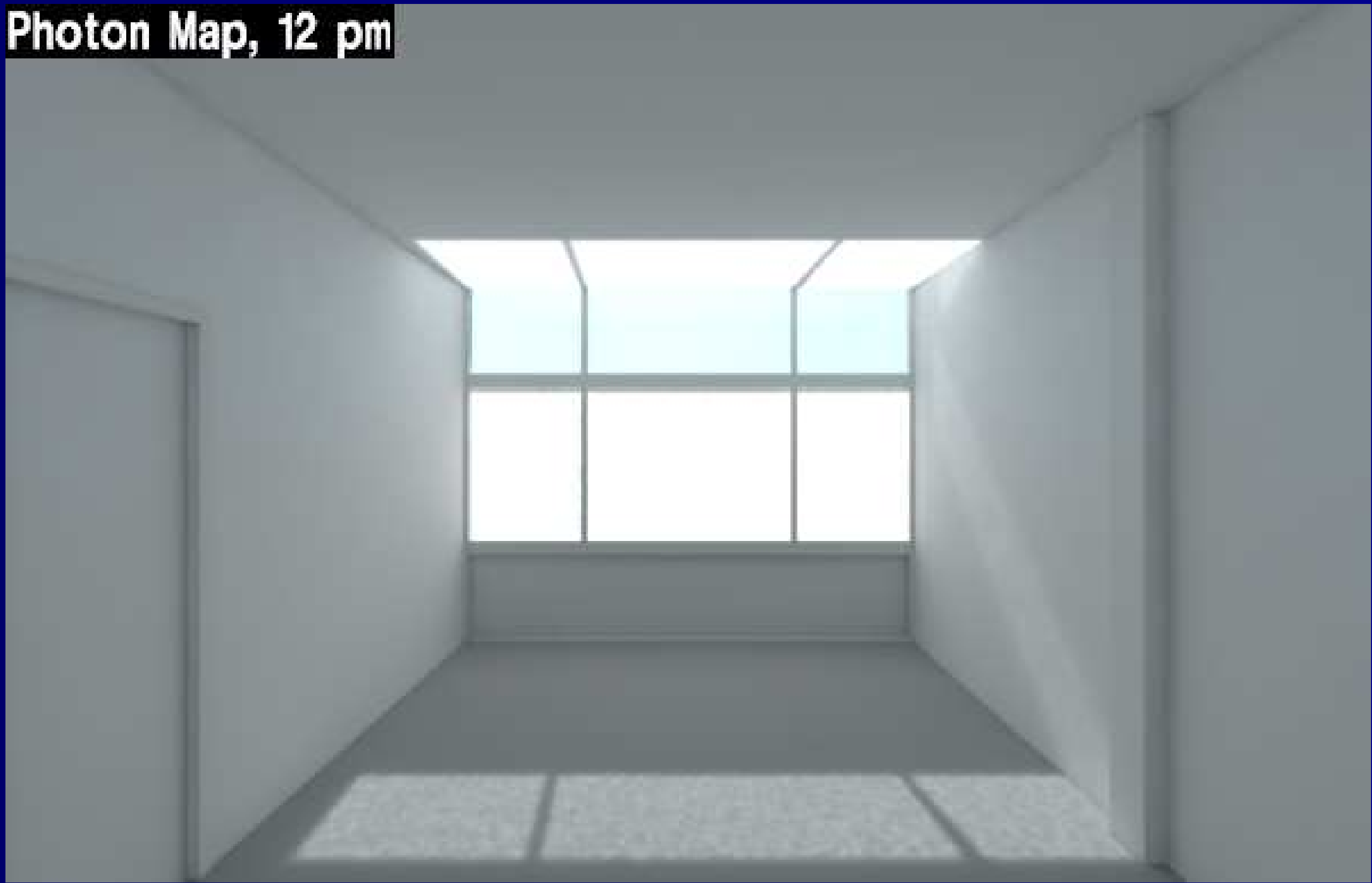
5k global, 10k caustic dump

Example: Y-Glass



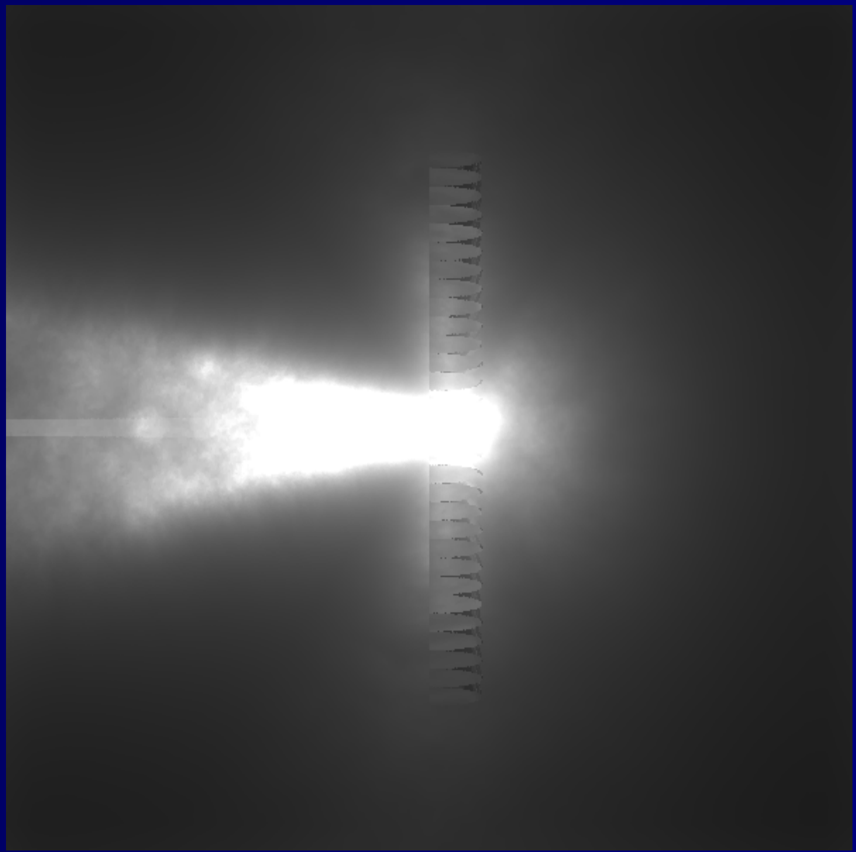
-ab 10 -ad 4096 -aa 0.05 -ar 64

Example: Y-Glass

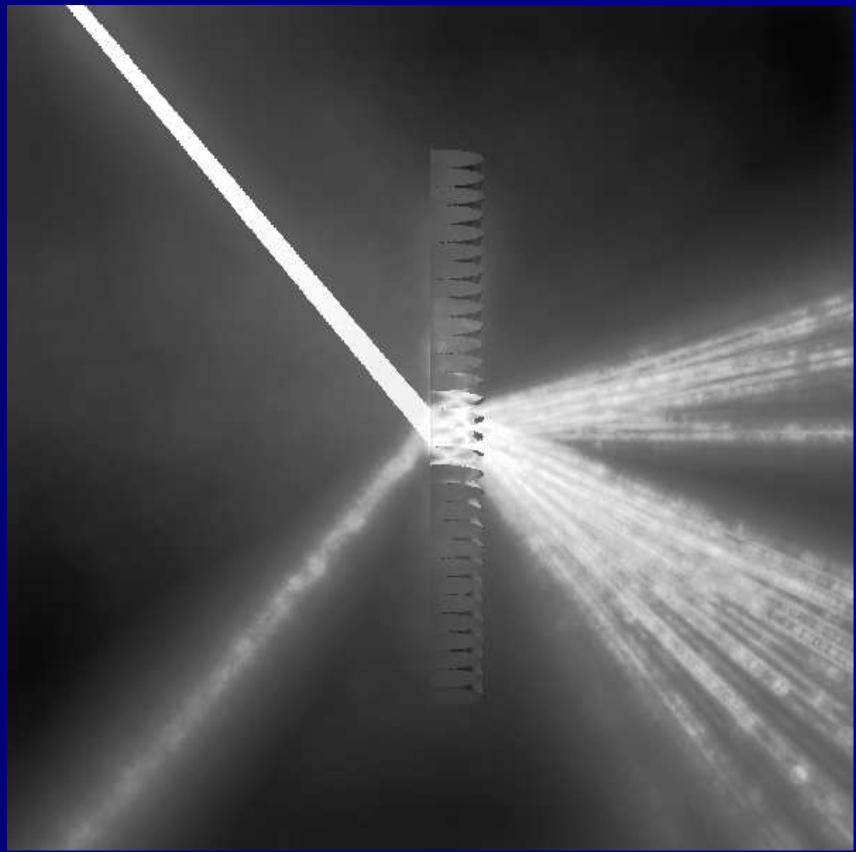


25k precomp. global, 100k caustic, ports at window

Example: CPC

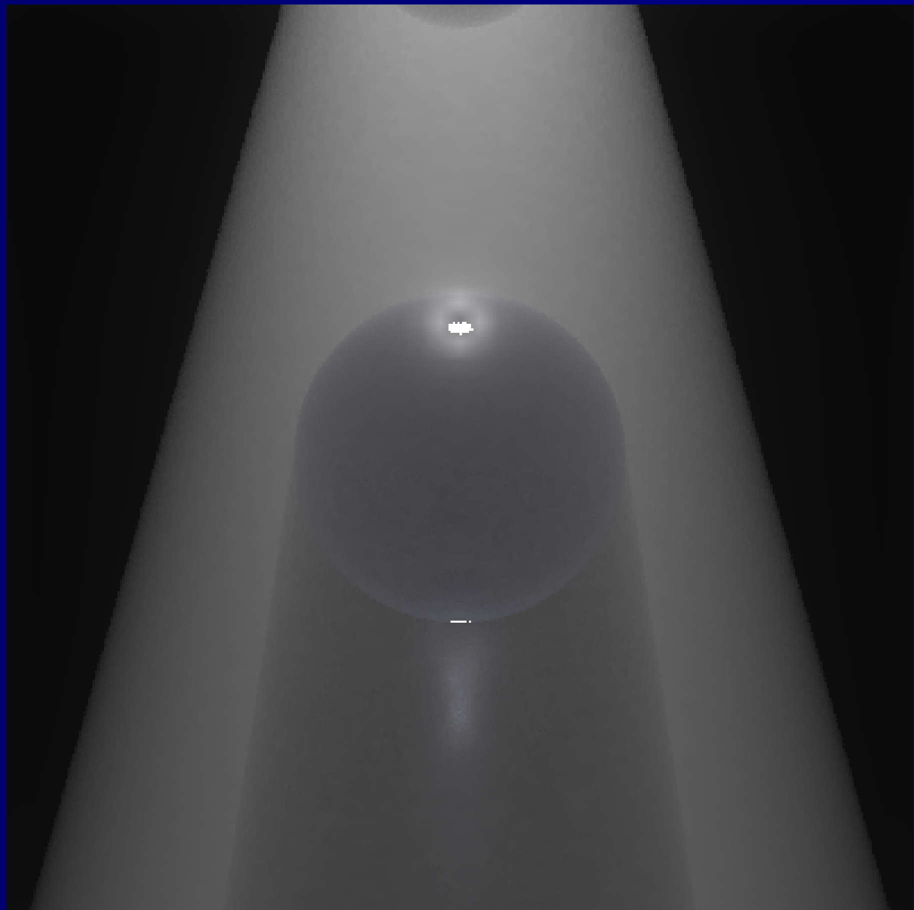


0° incidence



45° incidence

Example: Volume Caustics



97k caustic photons

Limitations

- Not part of official distribution – requires patching and recompiling stock RADIANCE code
- (Still) no support for user defined BRDFs (`brtdfunct`, `transfunc`, `transdata`, etc)
- Suitable parametrisation requires some experience
- Slow with complex or pathological (e.g. “leaky” or highly absorbant) scene geometry
- Inaccurate density estimates on curved surfaces or outside plane
- Bias/noise tradeoff

Acknowledgements

Contributors: Peter Apian-Bennewitz, Jan Wienold, Christian Reetz, Carsten Bauer

Daylight Simulation with Photon Maps:
`www.ganjatron.net/pmap/schregle-daylight_simulation_with_photon_maps.pdf`

Official website: `www.ise.fhg.de/radiance`