BSDFs, Matrices and Phases

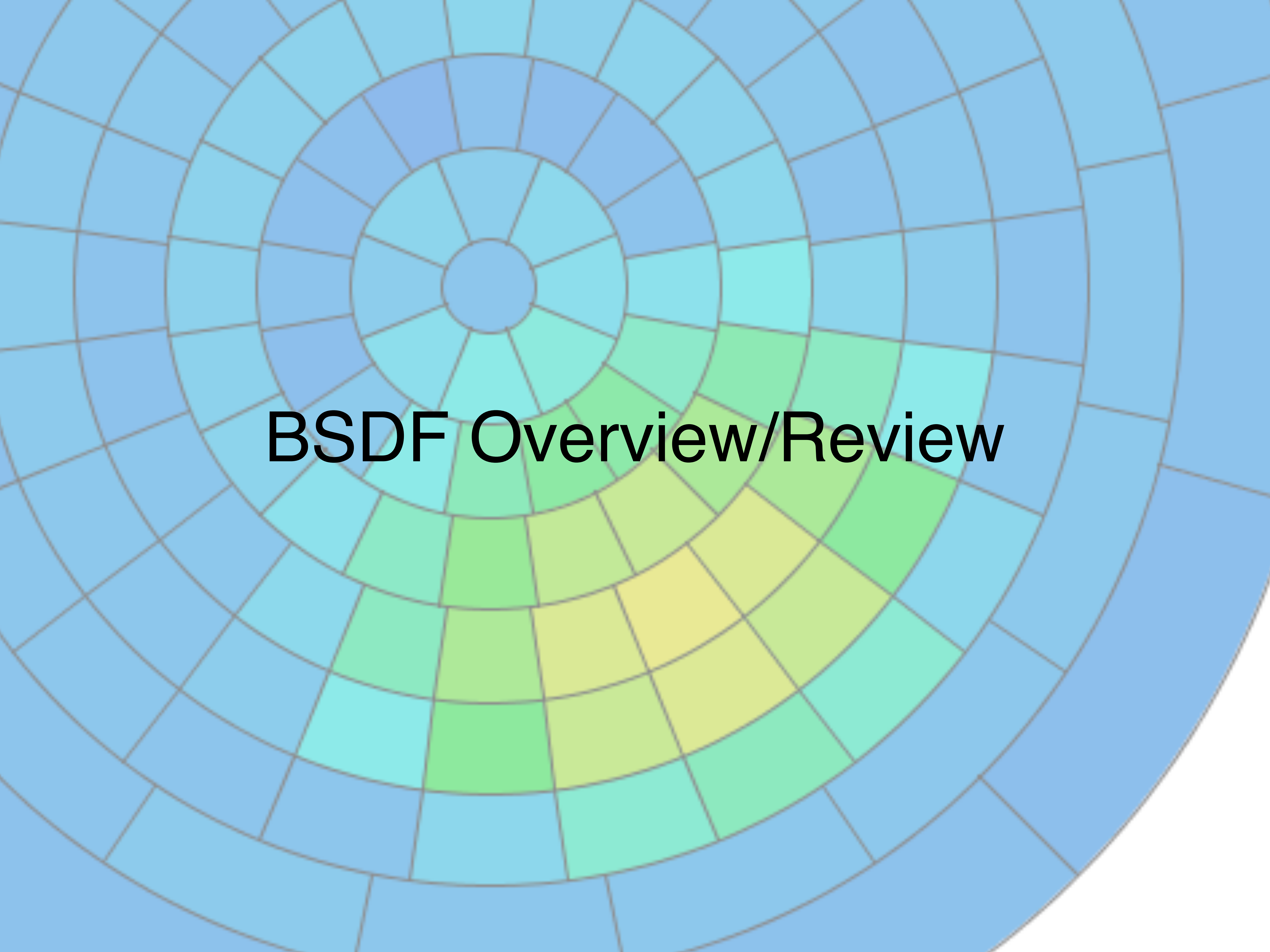
Andy McNeil, LBNL
September 1, 2014

BERKELEY LAB
Lawrence Berkeley National Laboratory

Are you ready for this?

- Brief overview/review of BSDFs
- Radiance BSDF material example
- Annual simulations (using new tools)
  - Daylight coefficient method
  - Three-phase method
    - Lightpipe example
  - Five-phase method
- What’s new at LBNL?

BSDF Overview/Review



Building design trends are moving towards new and innovative façades

We need the ability to simulate complex fenestration systems.

+

EnergyPlus

DAYSIM

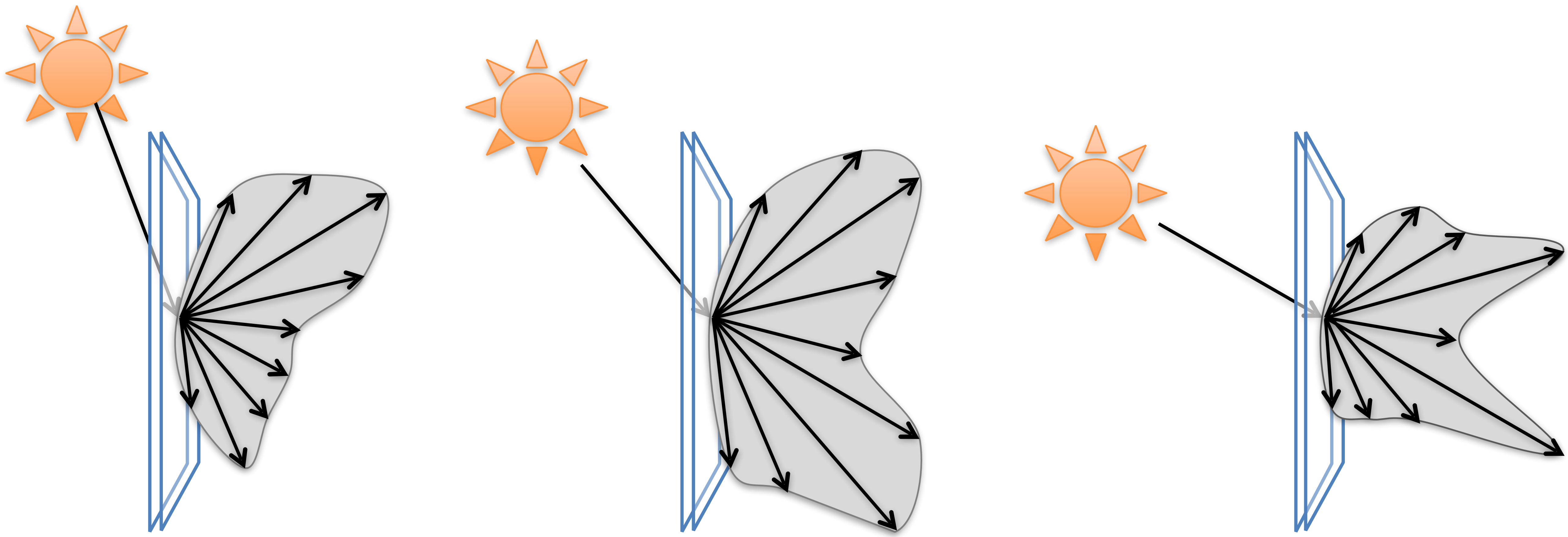
OpenStudio

Radiance
Synthetic Imaging System

=

?

A BSDF file is to a fenestration system what an IES file is to a luminaire



However, a BSDF file contains output distributions for many incident directions.

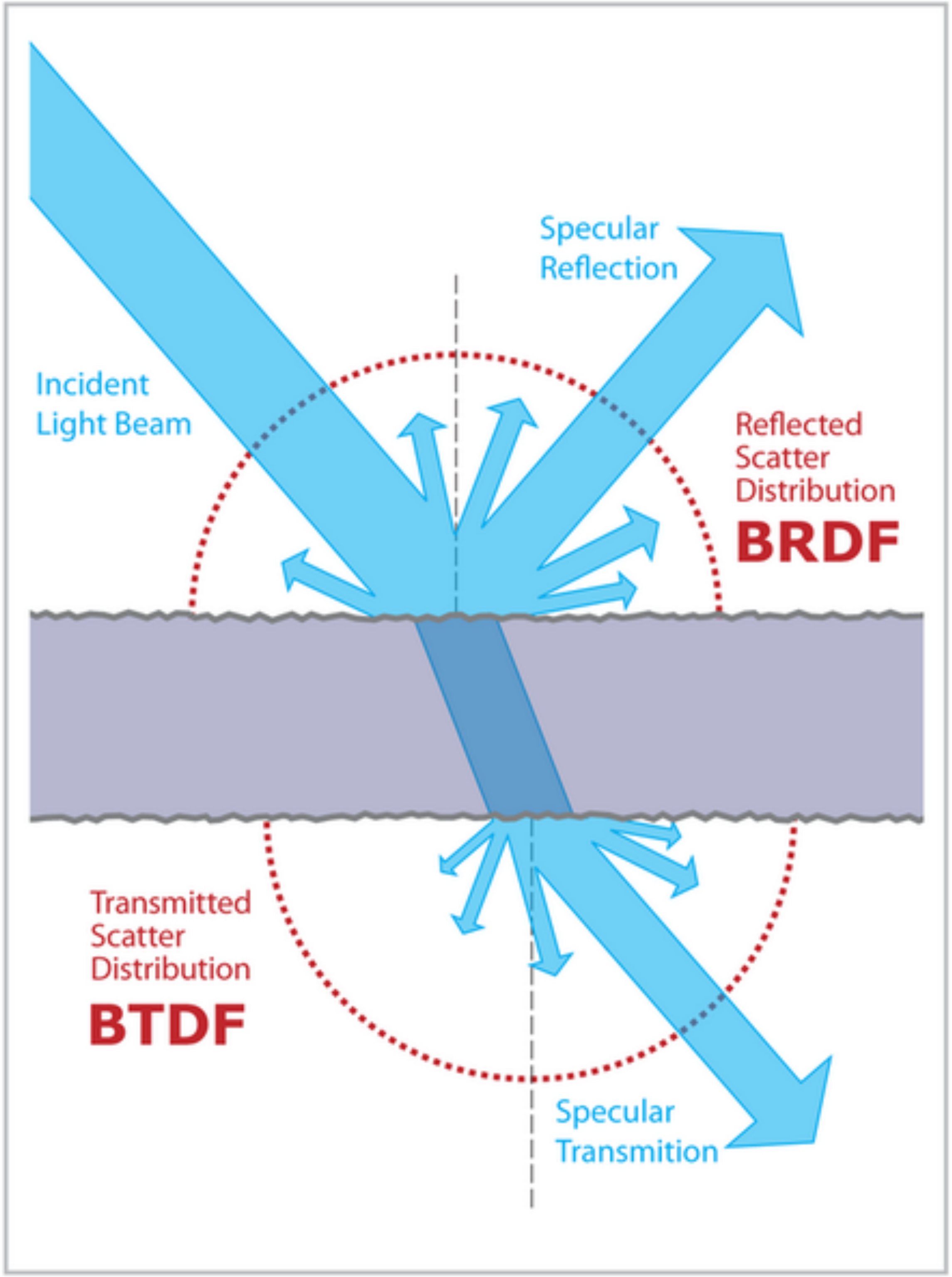
A BSDF characterizes light transmission, reflection and directional distribution of a surface or product.

BSDF = Bidirectional Scattering Distribution Function

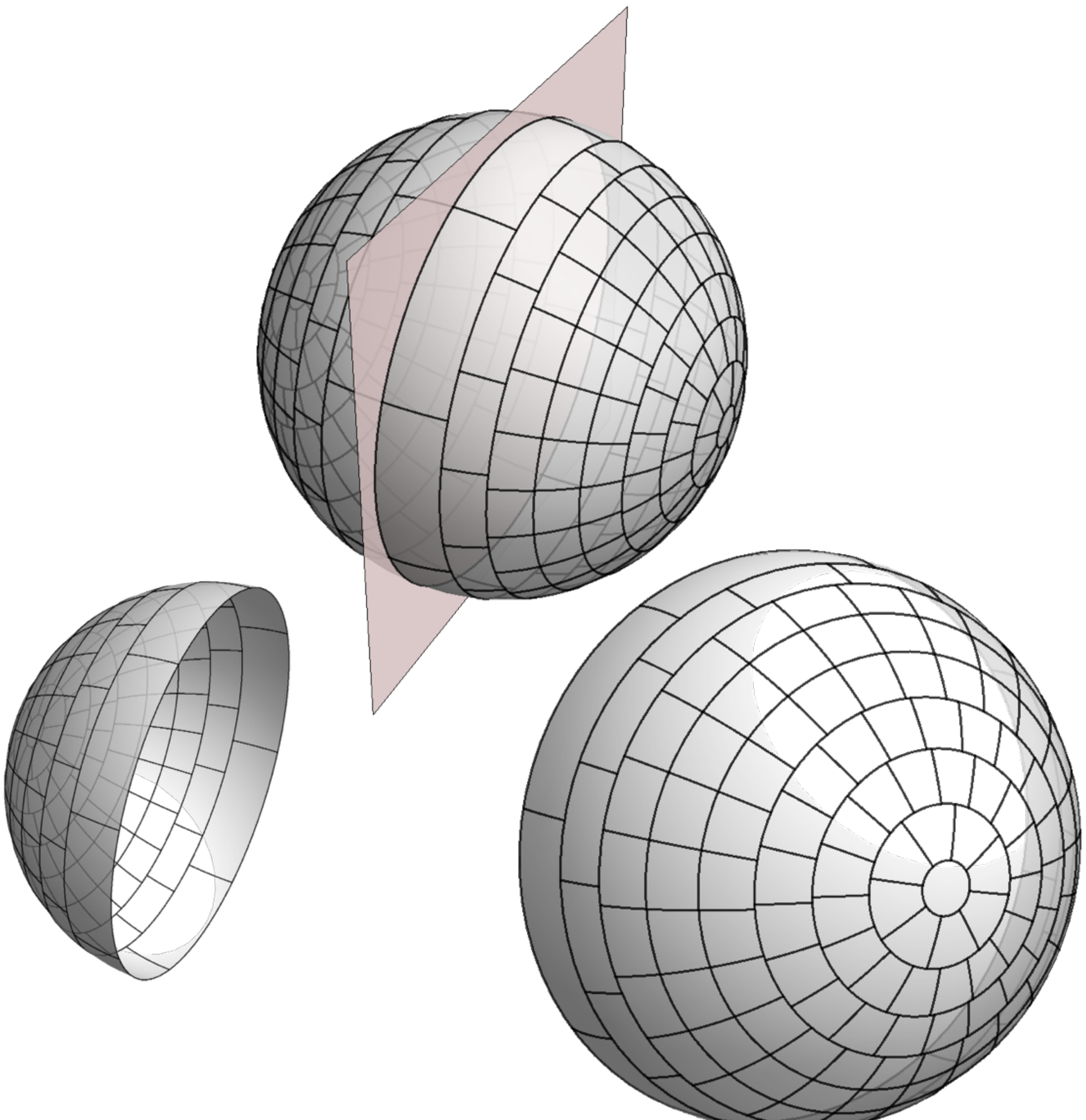
BRDF = Bidirectional Reflection Distribution Function

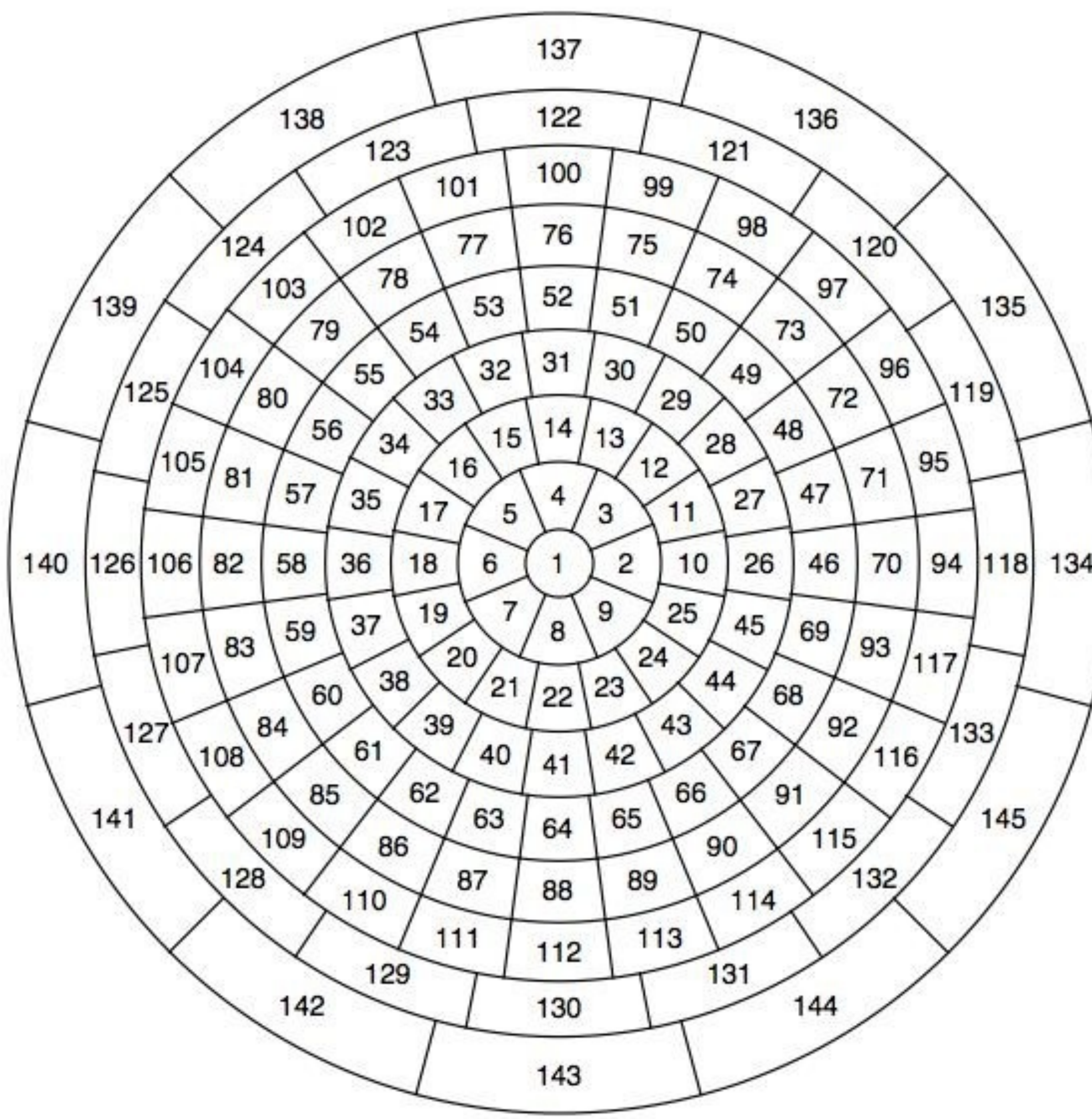
BTDF = Bidirectional Transmittance Distribution Function

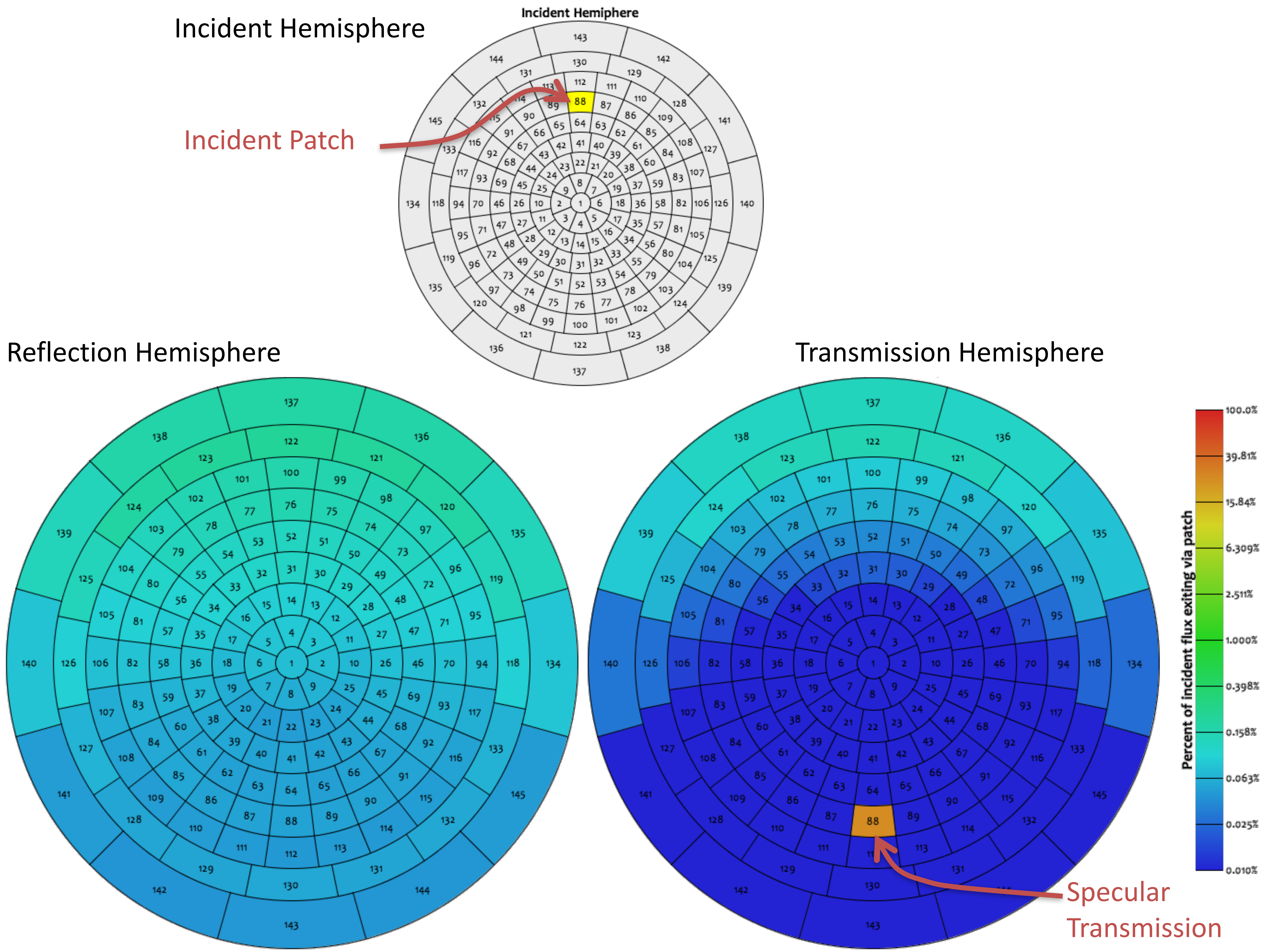
BSDF = BRDF + BTDF



LBNL Window BSDFs use the *Klems angle basis*, which divides reflection and transmission hemispheres into 145 patches

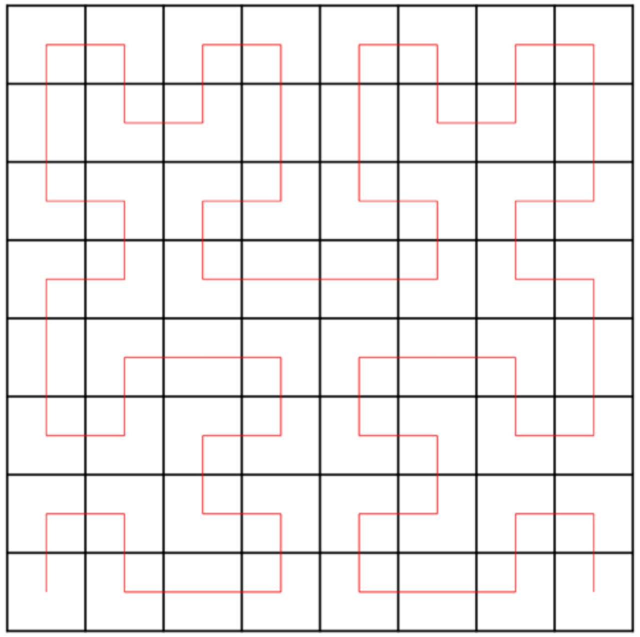




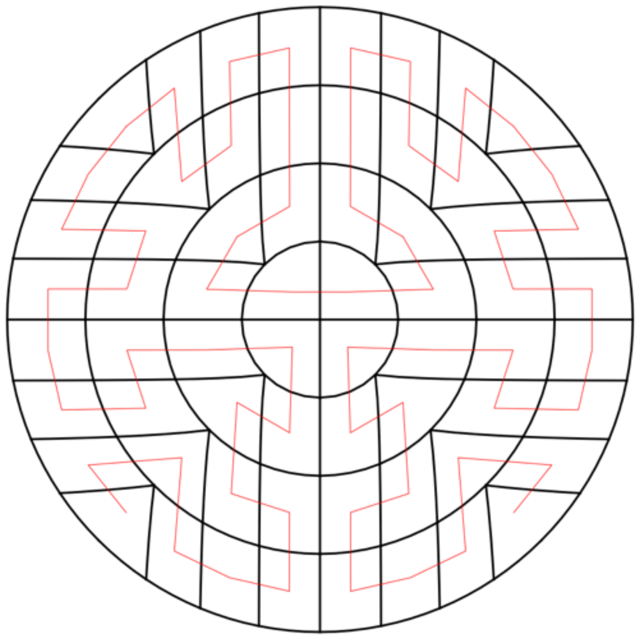


Incident Hemisphere
Incident Hemiphere
Incident Patch
Reflection Hemisphere
Transmission Hemisphere
Specular Transmission
Percent of incident flux exiting via patch
100.0%
39.81%
15.84%
6.309%
2.511%
1.000%
0.398%
0.158%
0.063%
0.025%
0.010%

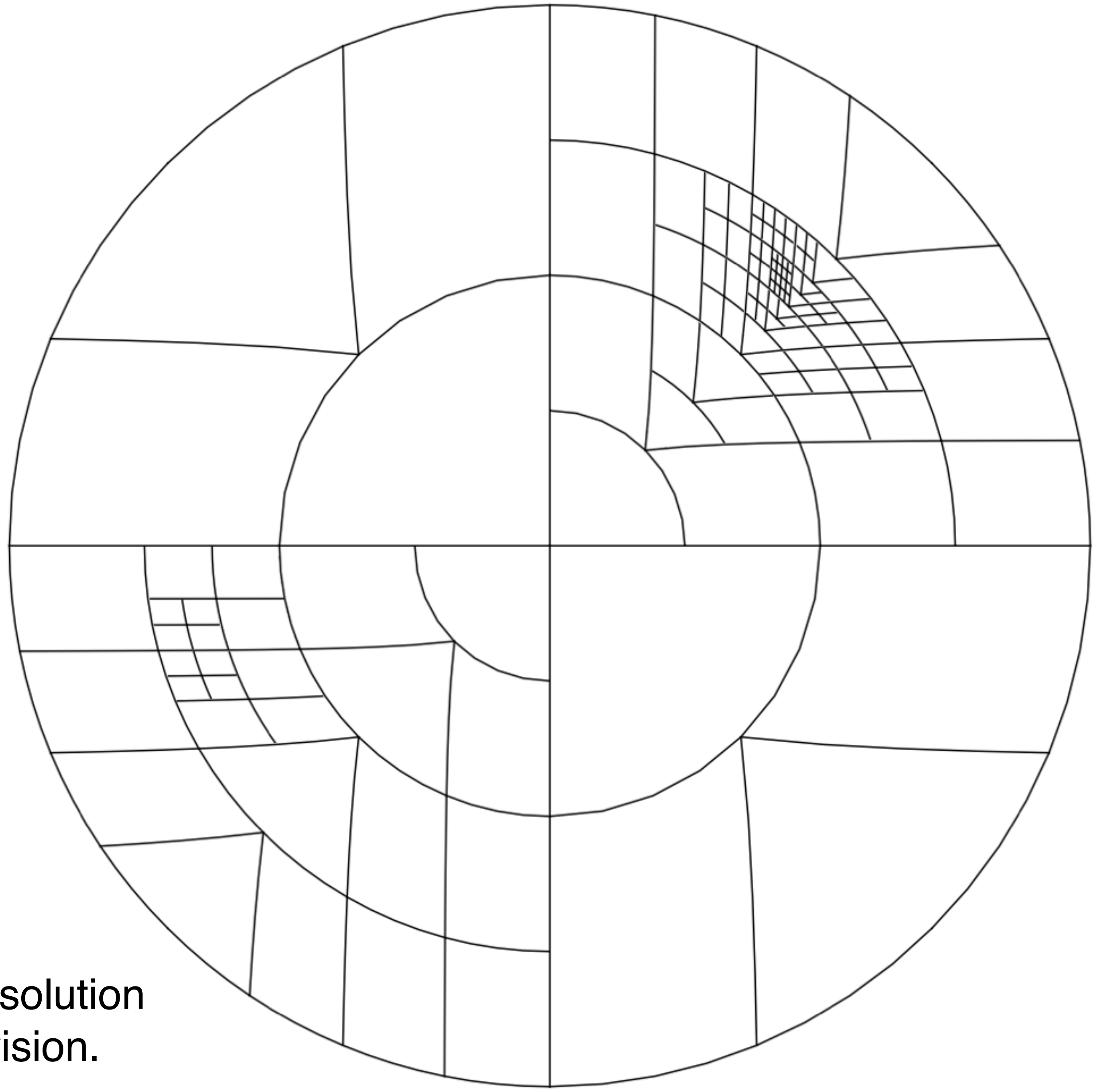
Radiance also supports variable resolution tensor tree BSDFs.



Hilbert Recursive Space Filling Curve

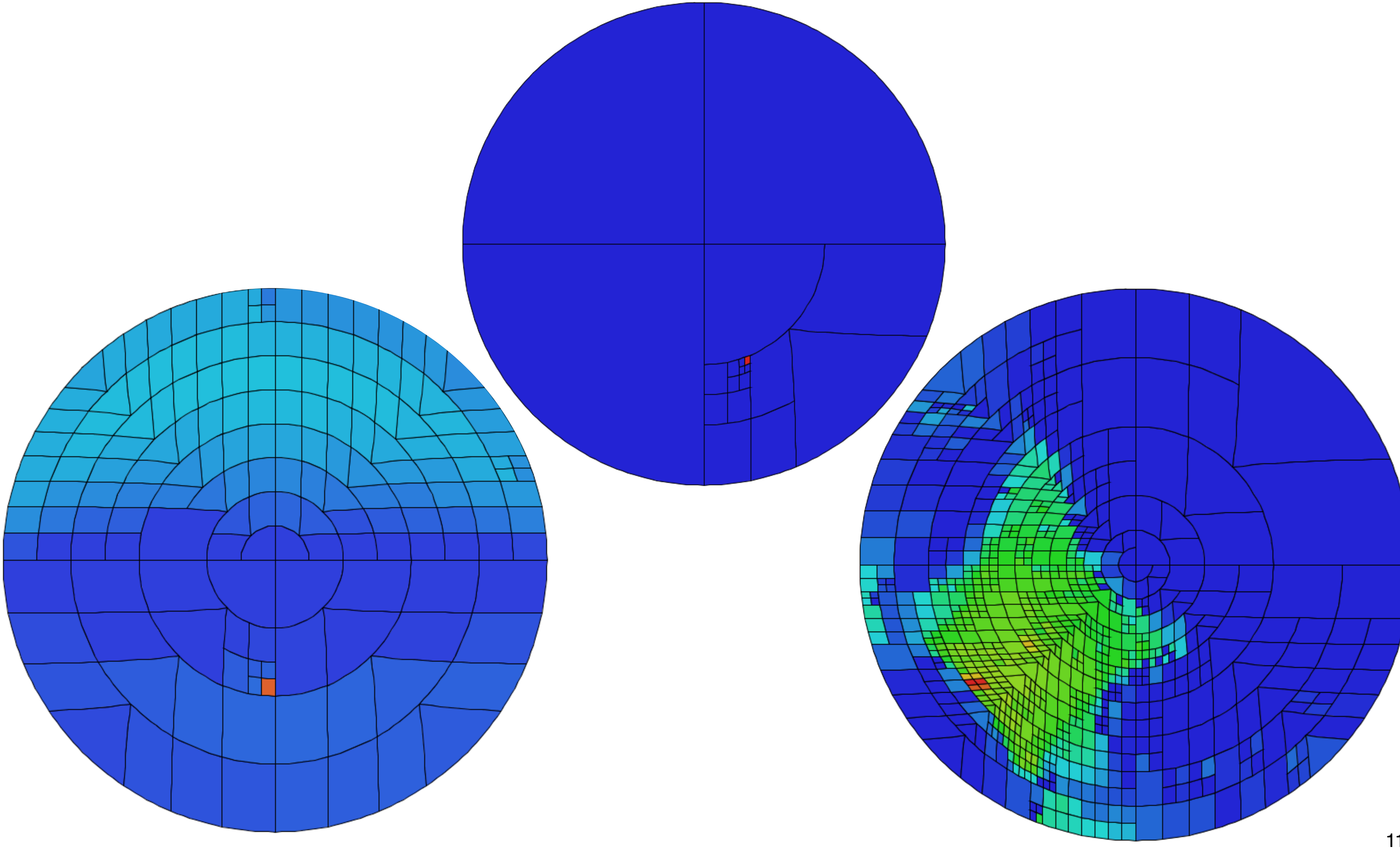


With Shirley-Chiu square to disk mapping



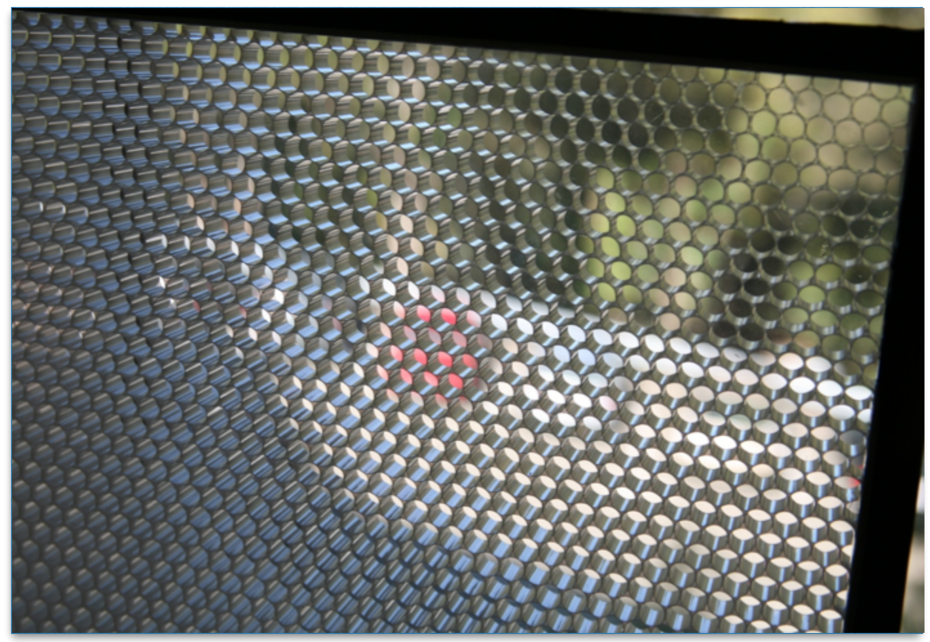
Permits variable resolution hemisphere subdivision.

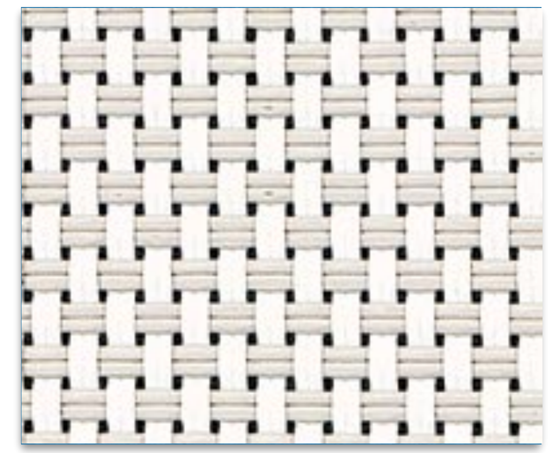
Radiance also supports variable resolution tensor tree BSDFs.



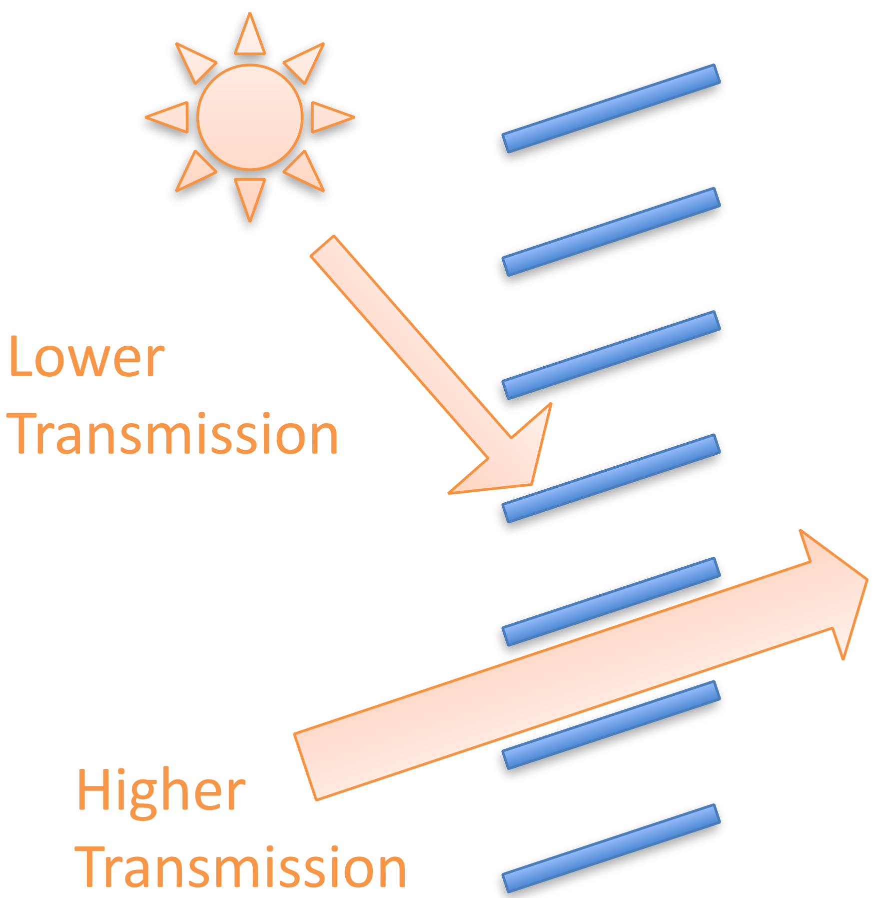
Use BSDFs to simulate performance of optically complex fenestration systems/elements.





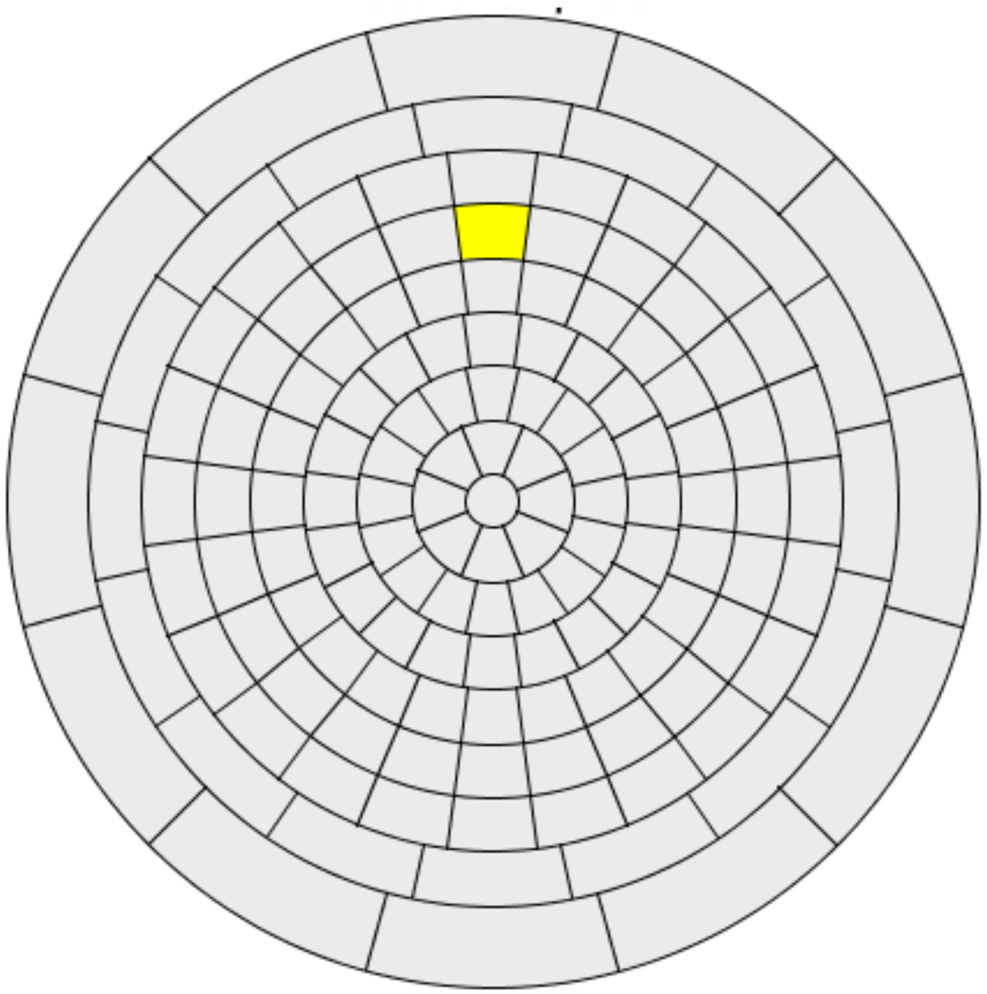



The slat angle of venetian blinds affects the distribution and quantity of light transmitted.

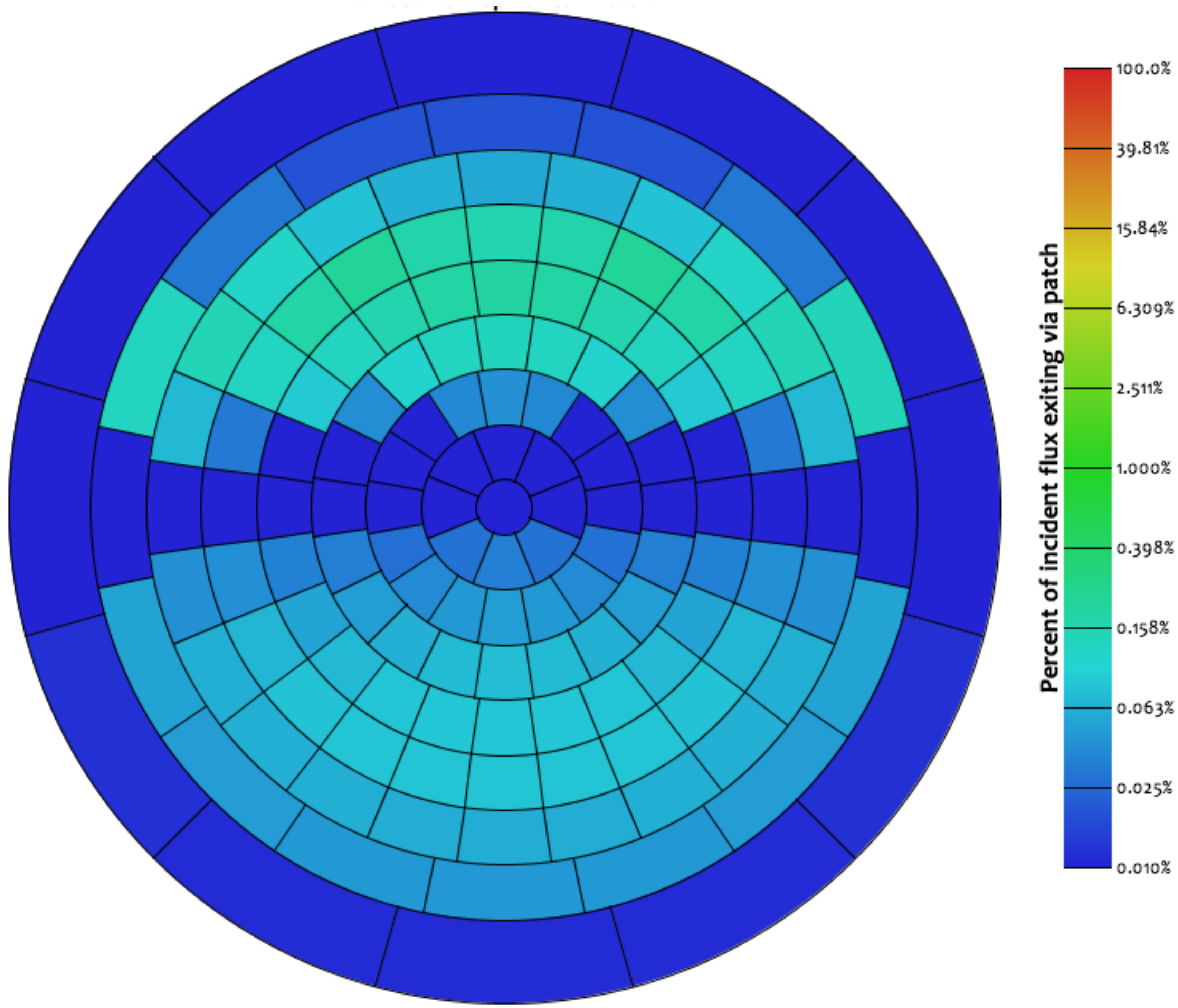


The slat angle of venetian blinds affects the distribution and quantity of light transmitted.

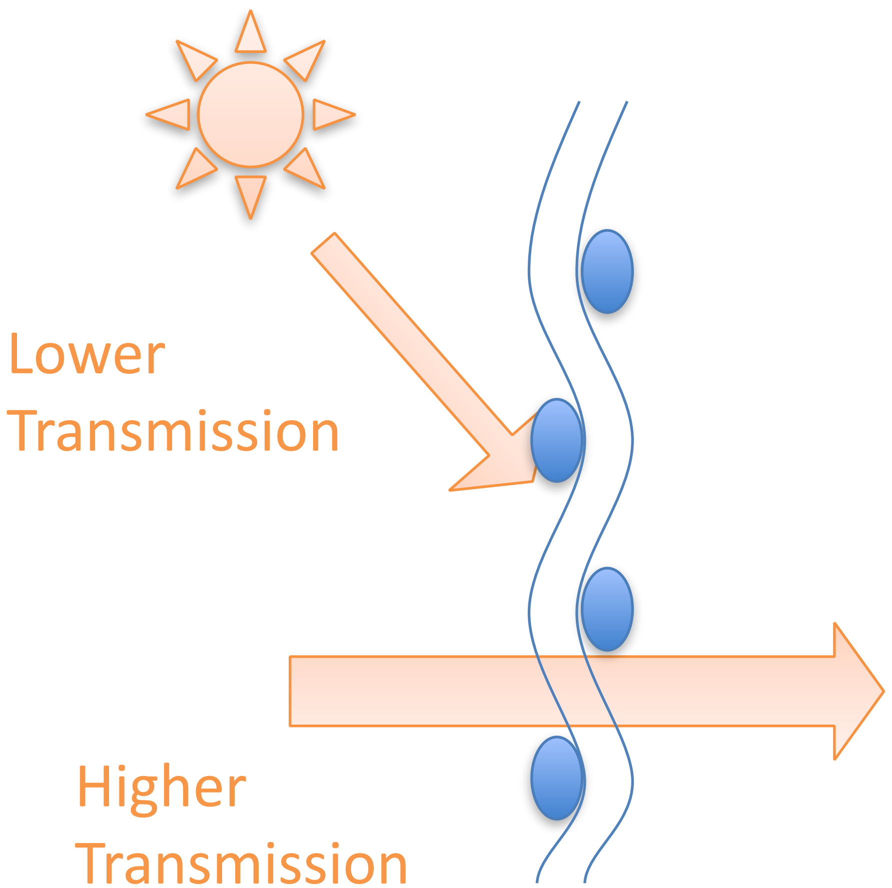
Incident Hemisphere



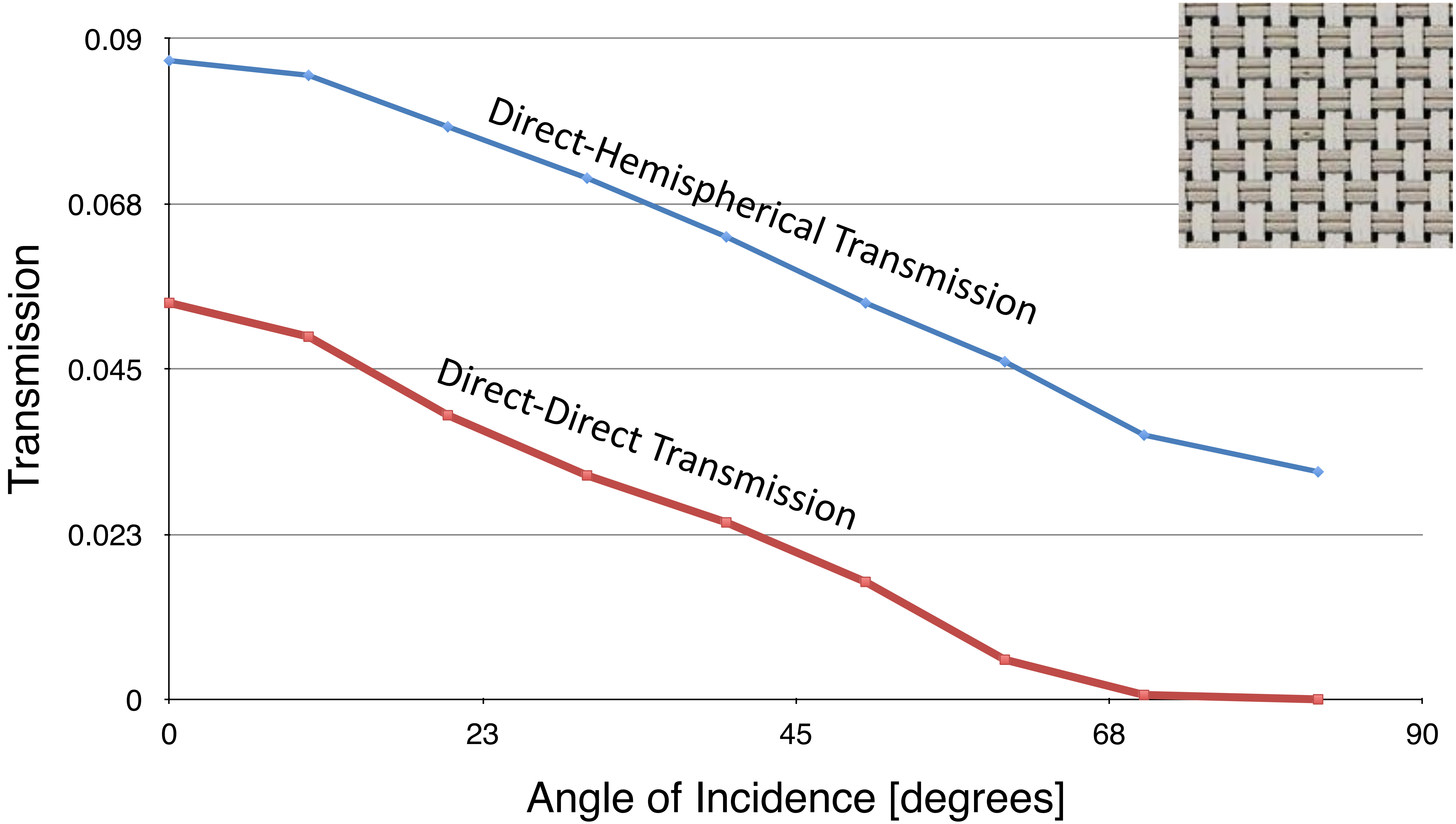
Transmission Hemisphere



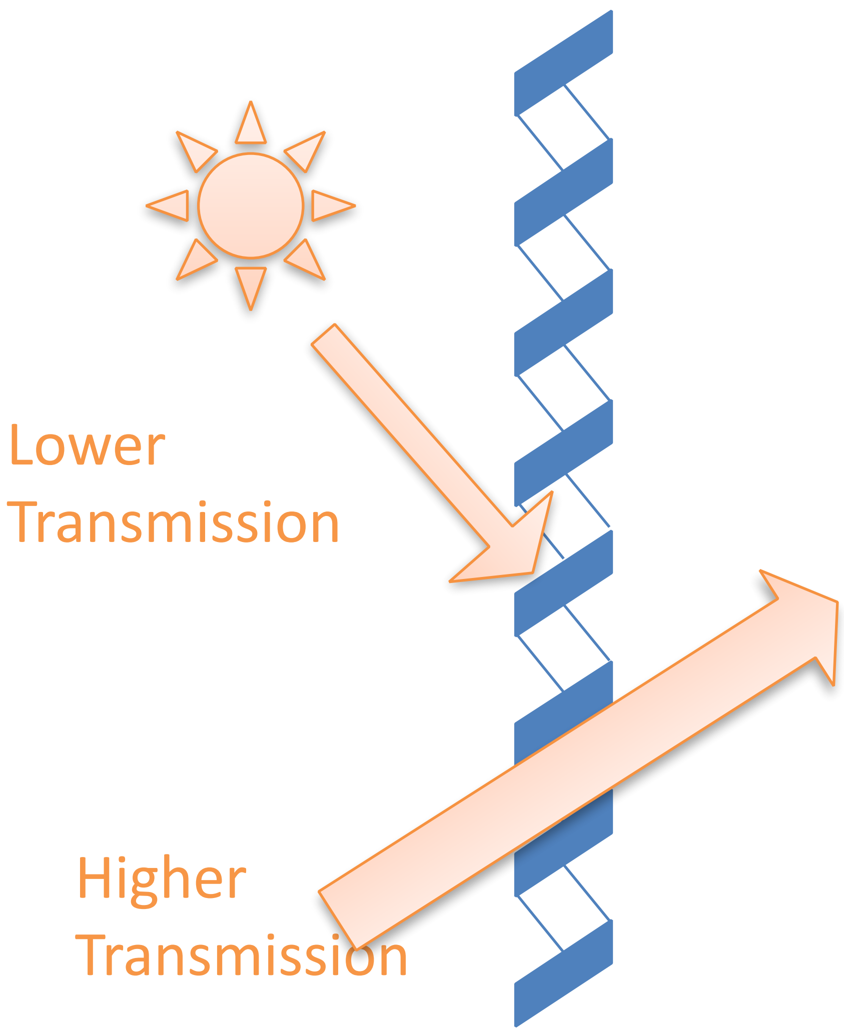
The thickness of threads in a woven shade cuts off direct transmission affecting directional transmission.



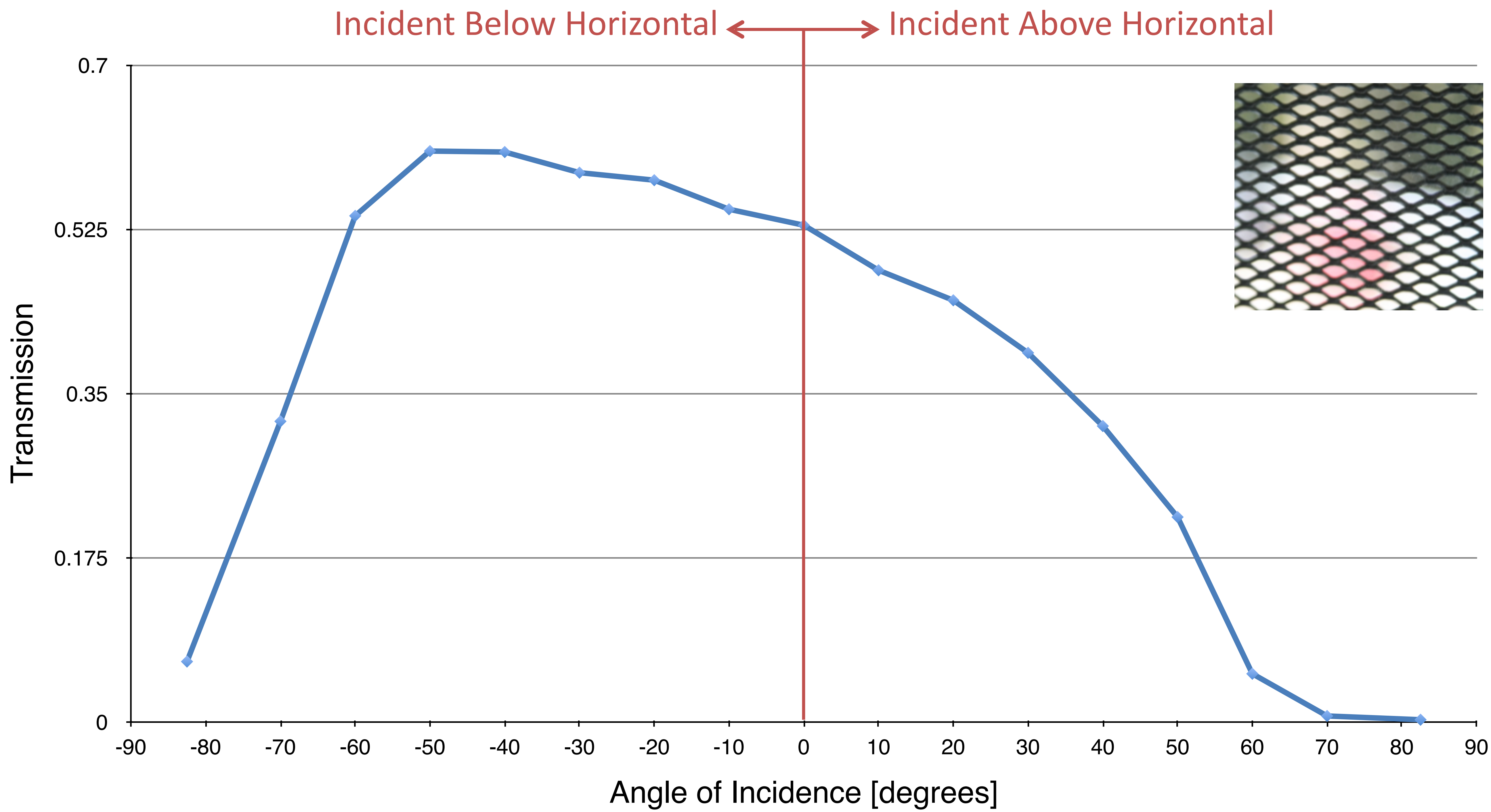
The thickness of threads in a woven shade cuts off direct transmission affecting directional transmission.



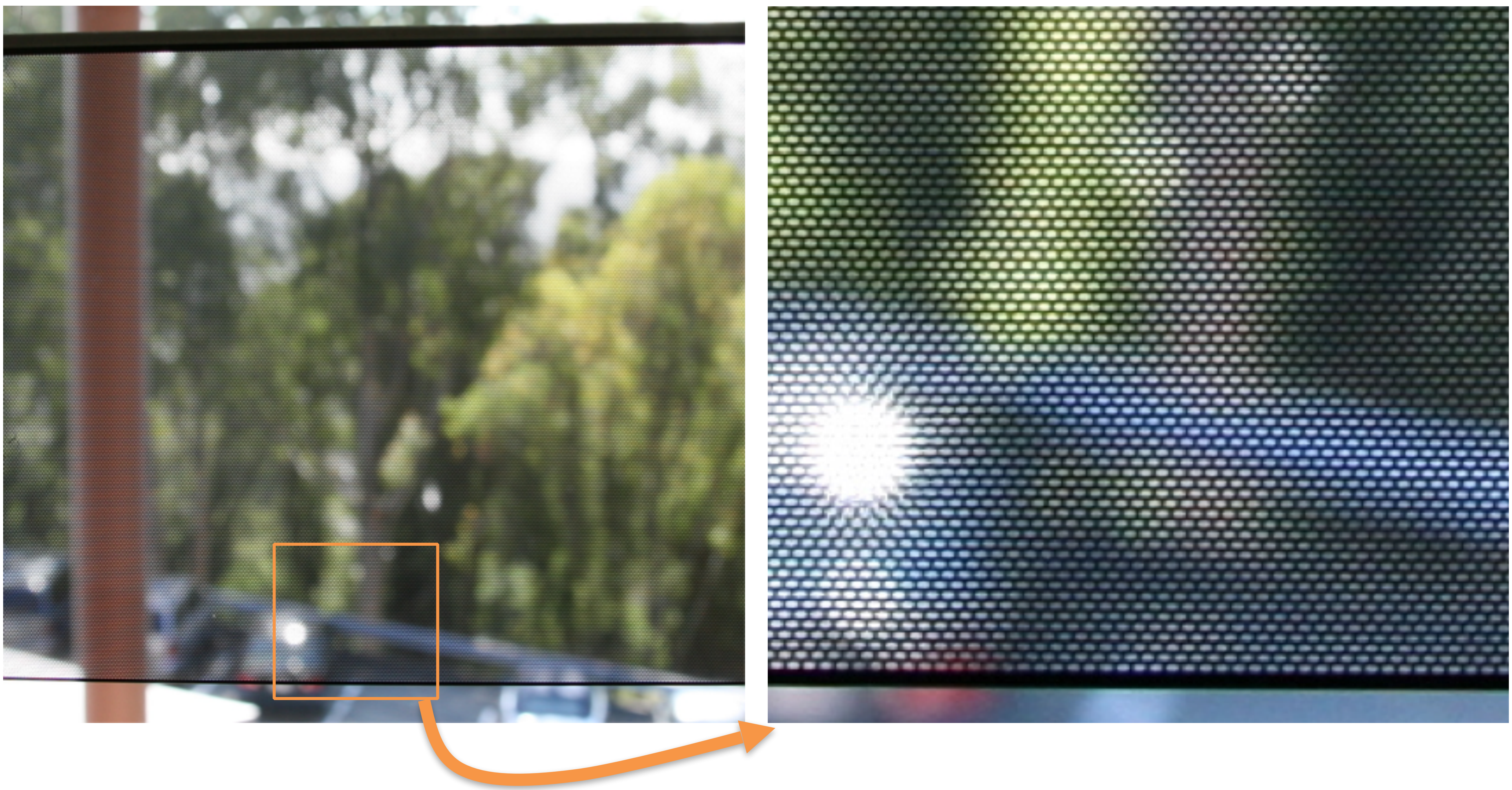
Expanded metal mesh has a higher transmission for light from below horizon than above.



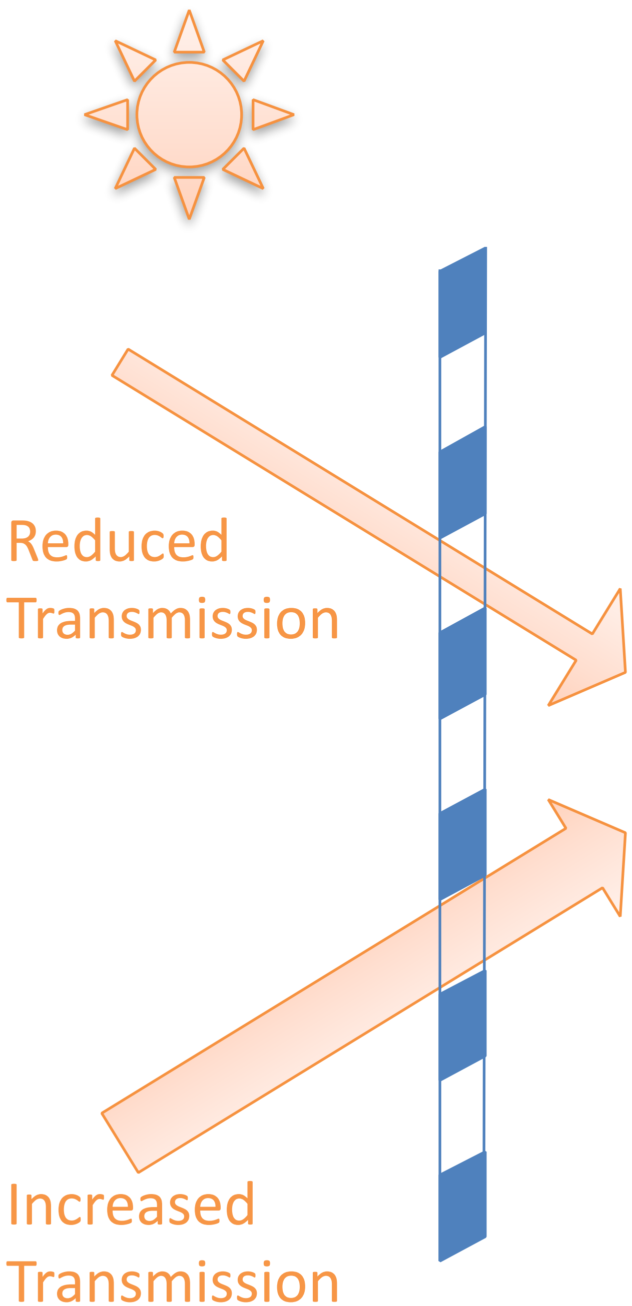
Expanded metal mesh has a higher transmission for light from below horizon than above.

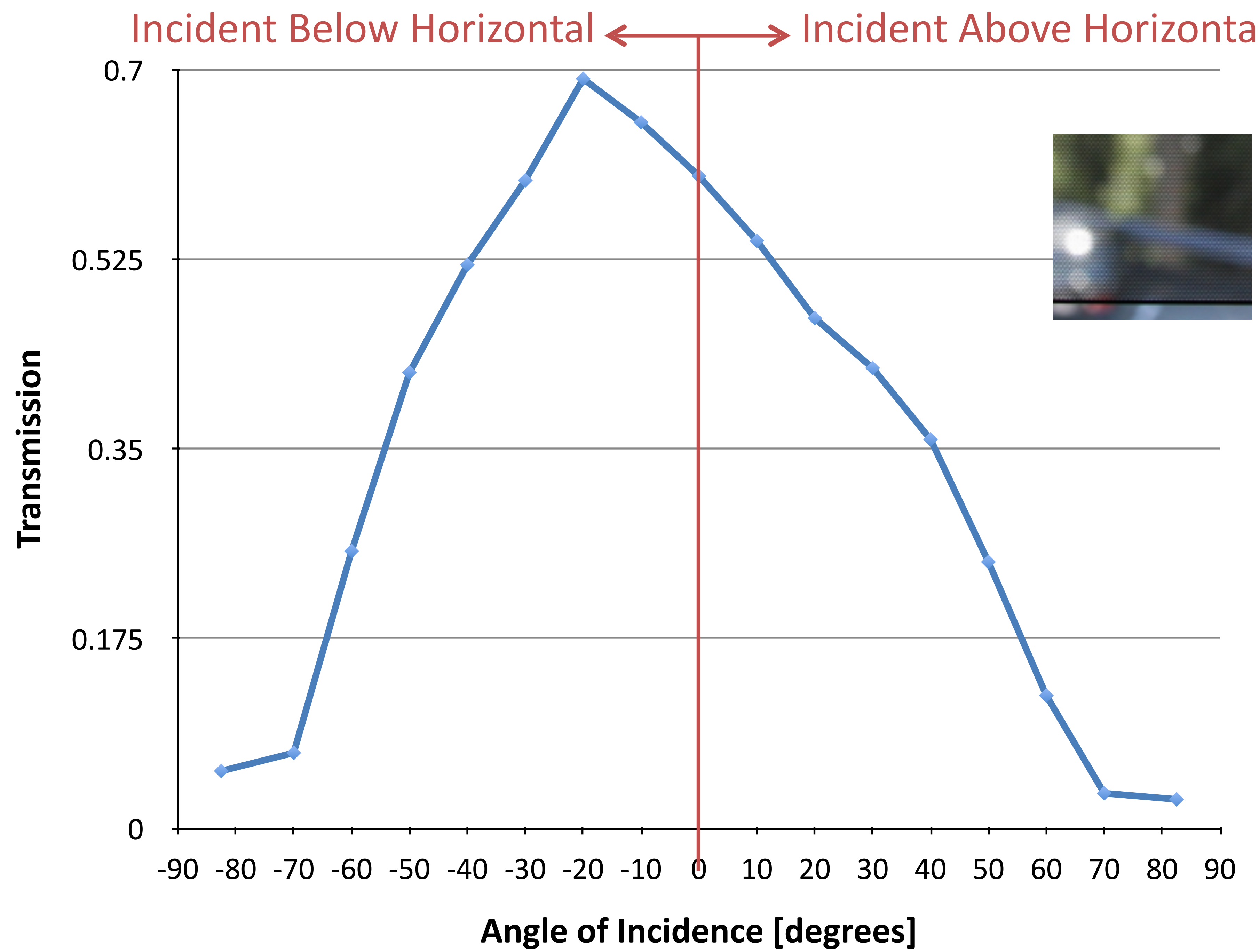


A micro perforated mesh cut at a downward angle transmits more light from below the horizon than above

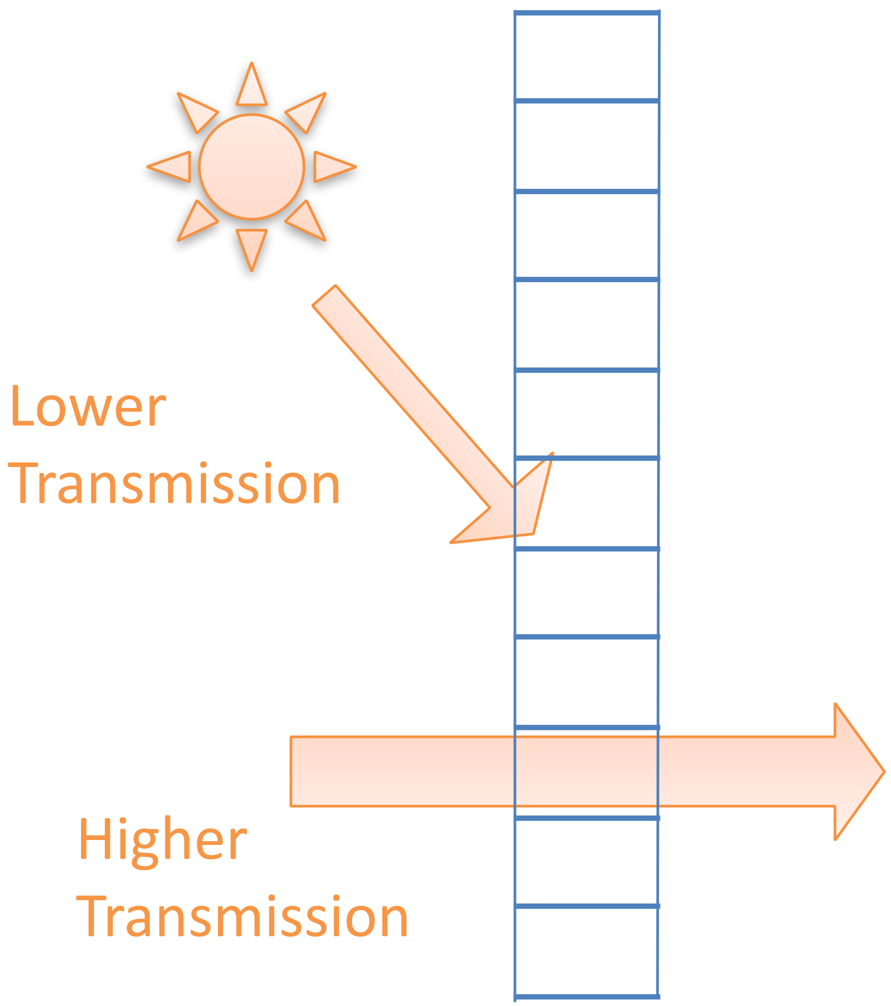


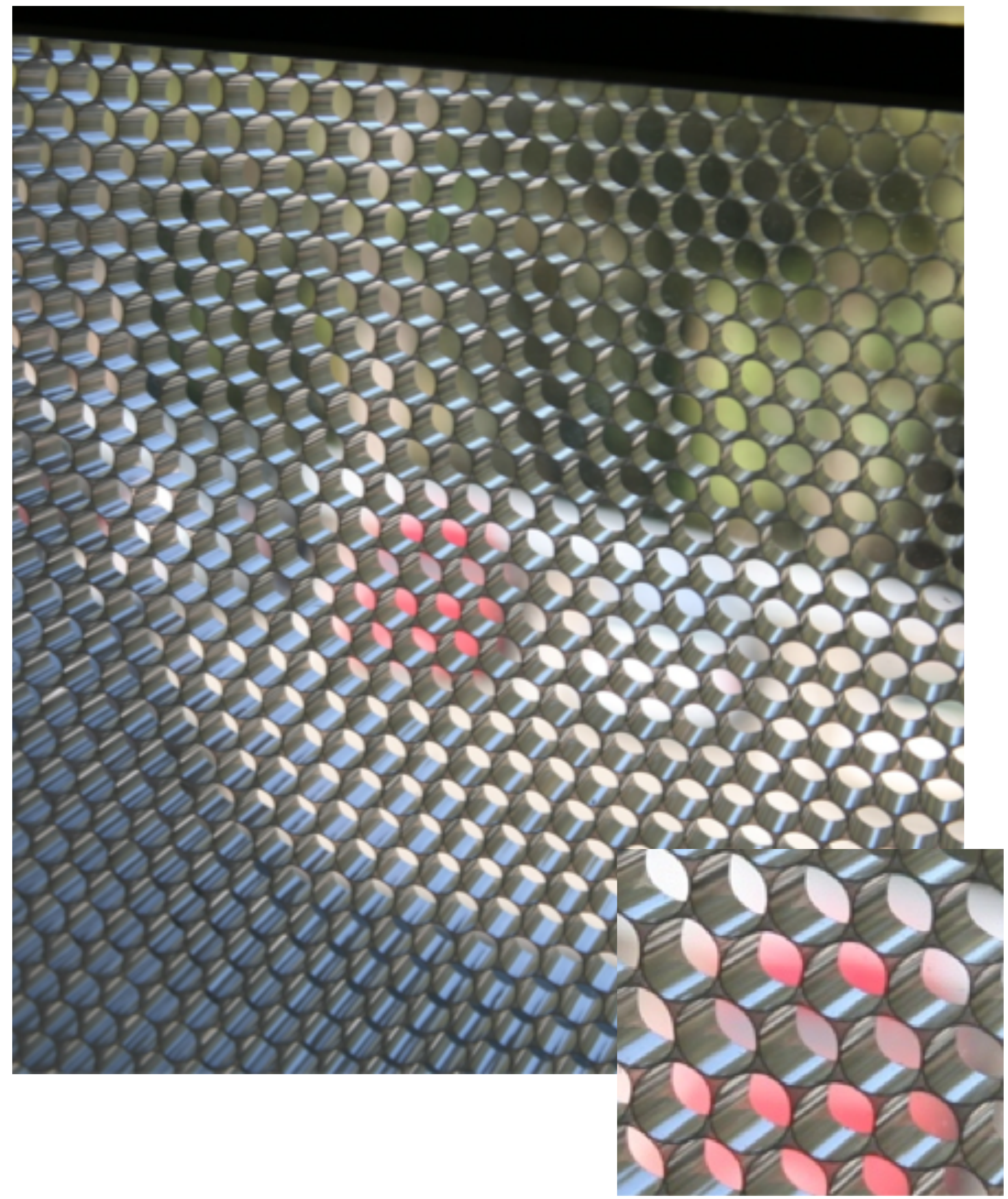
A micro perforated mesh cut at a downward angle transmits more light from below the horizon than above



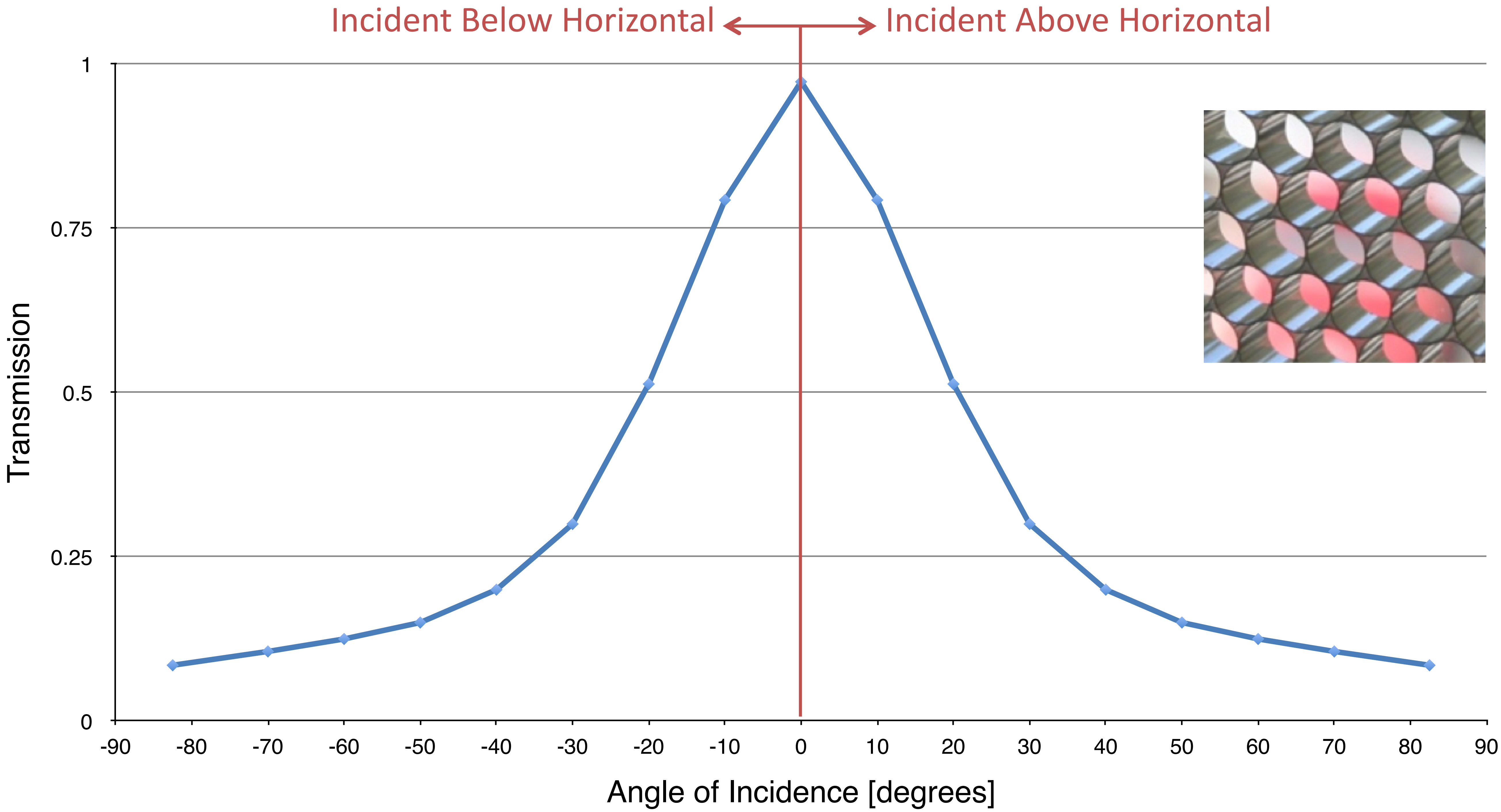


A perpendicular columnar structure admits more light at normal incidence than at off angles

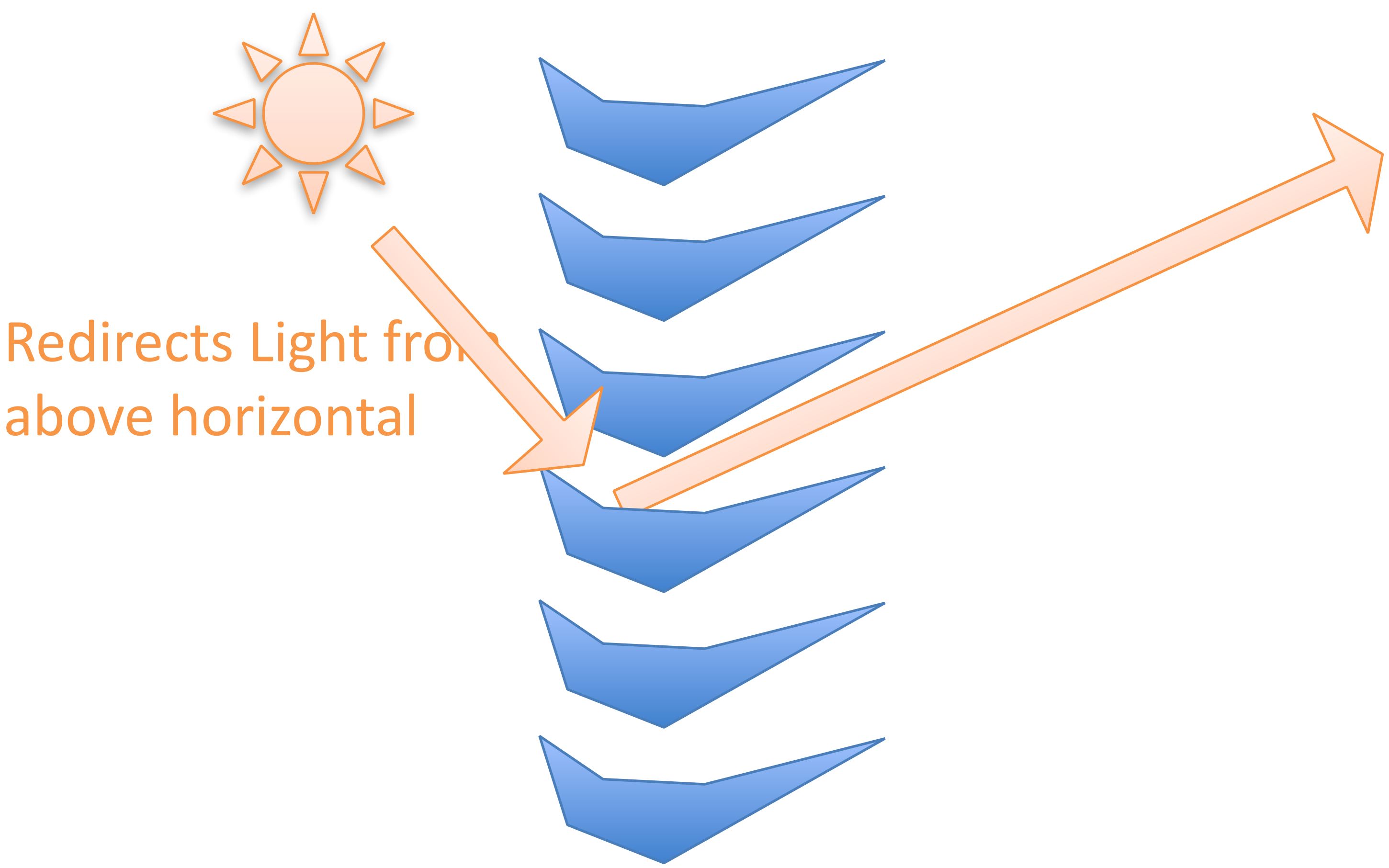




A perpendicular columnar structure admits more light at normal incidence than at off angles

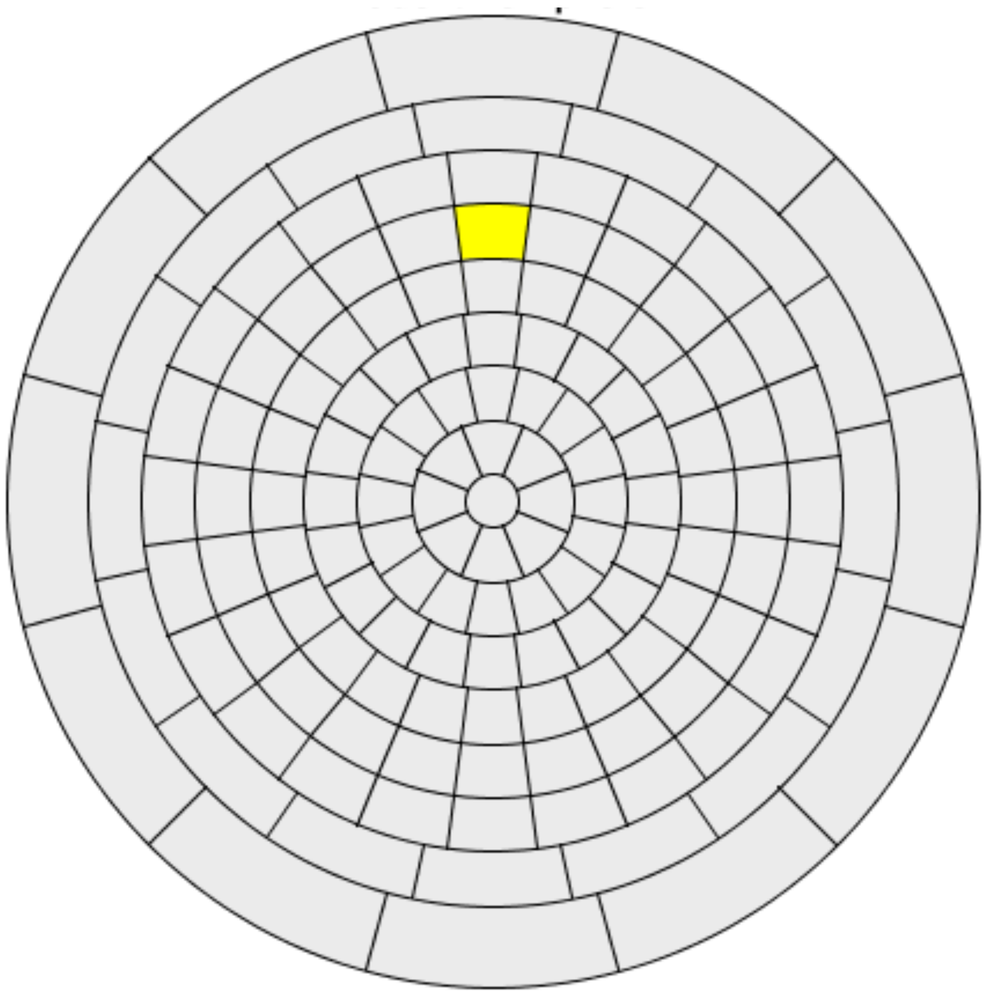


An optical light shelf redirects light towards the ceiling

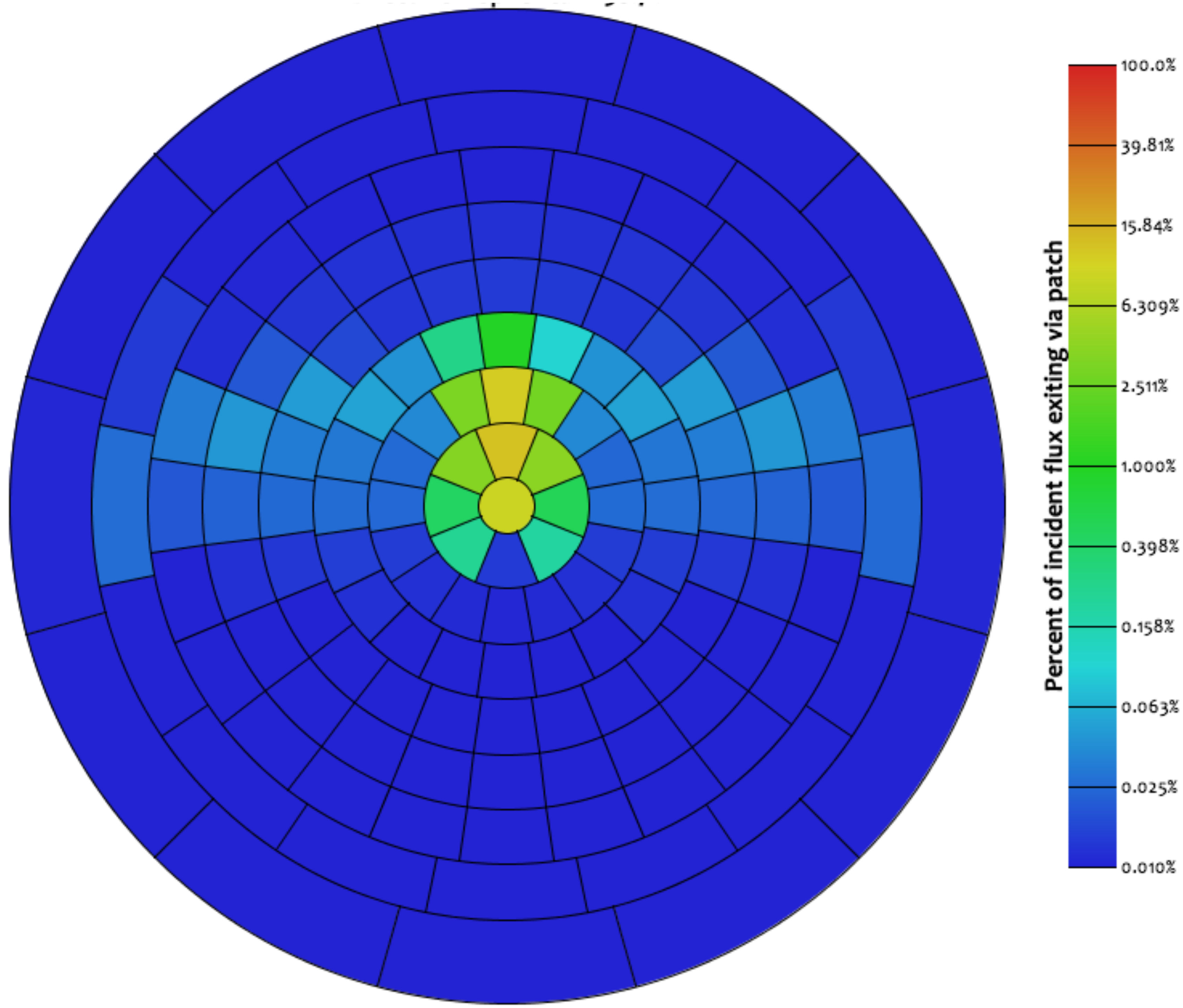


An optical light shelf redirects light towards the ceiling

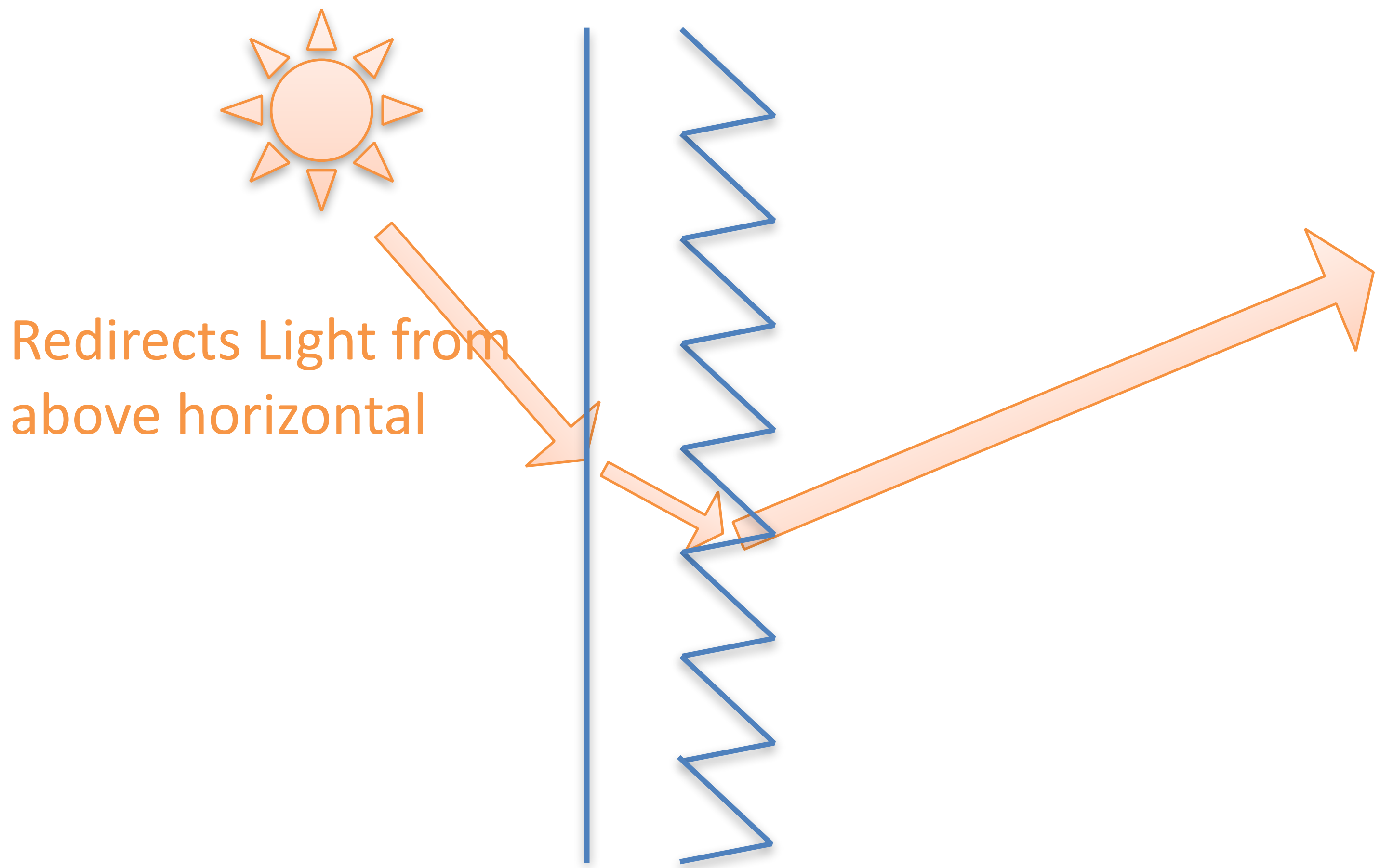
Incident Hemisphere

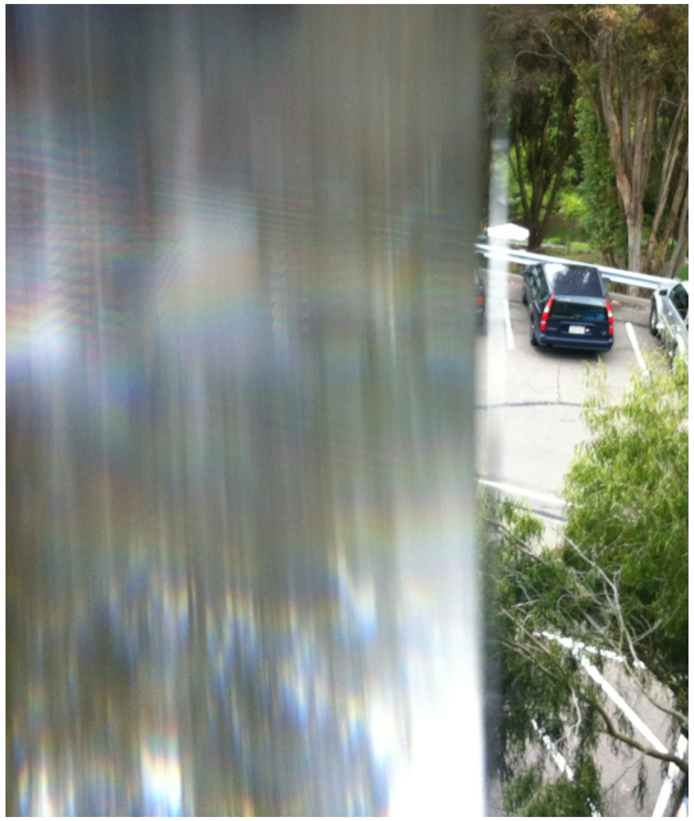


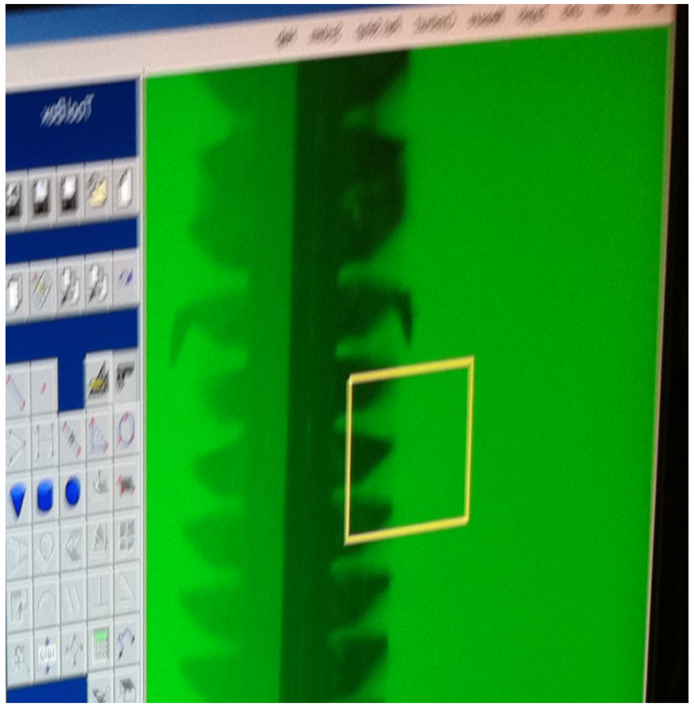
Transmission Hemisphere



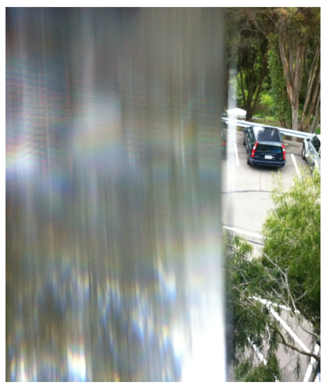
A micro-structured film refracts light towards the ceiling



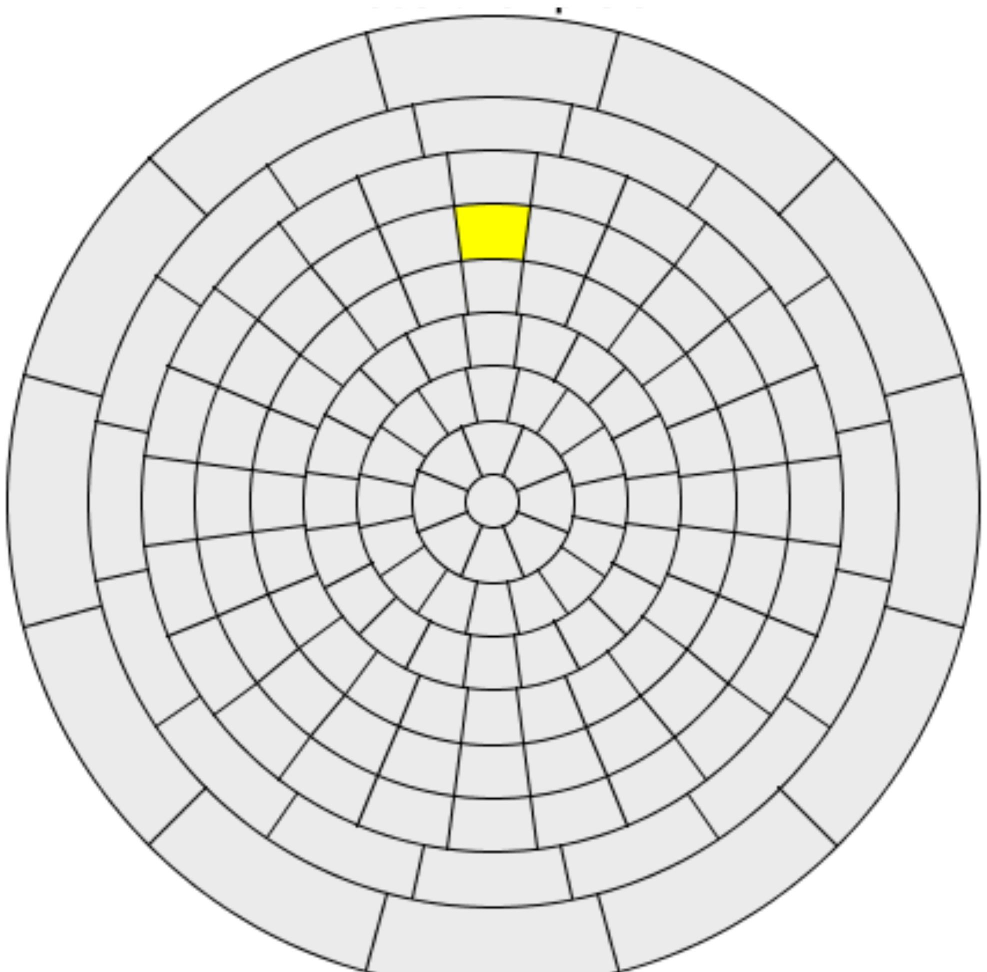




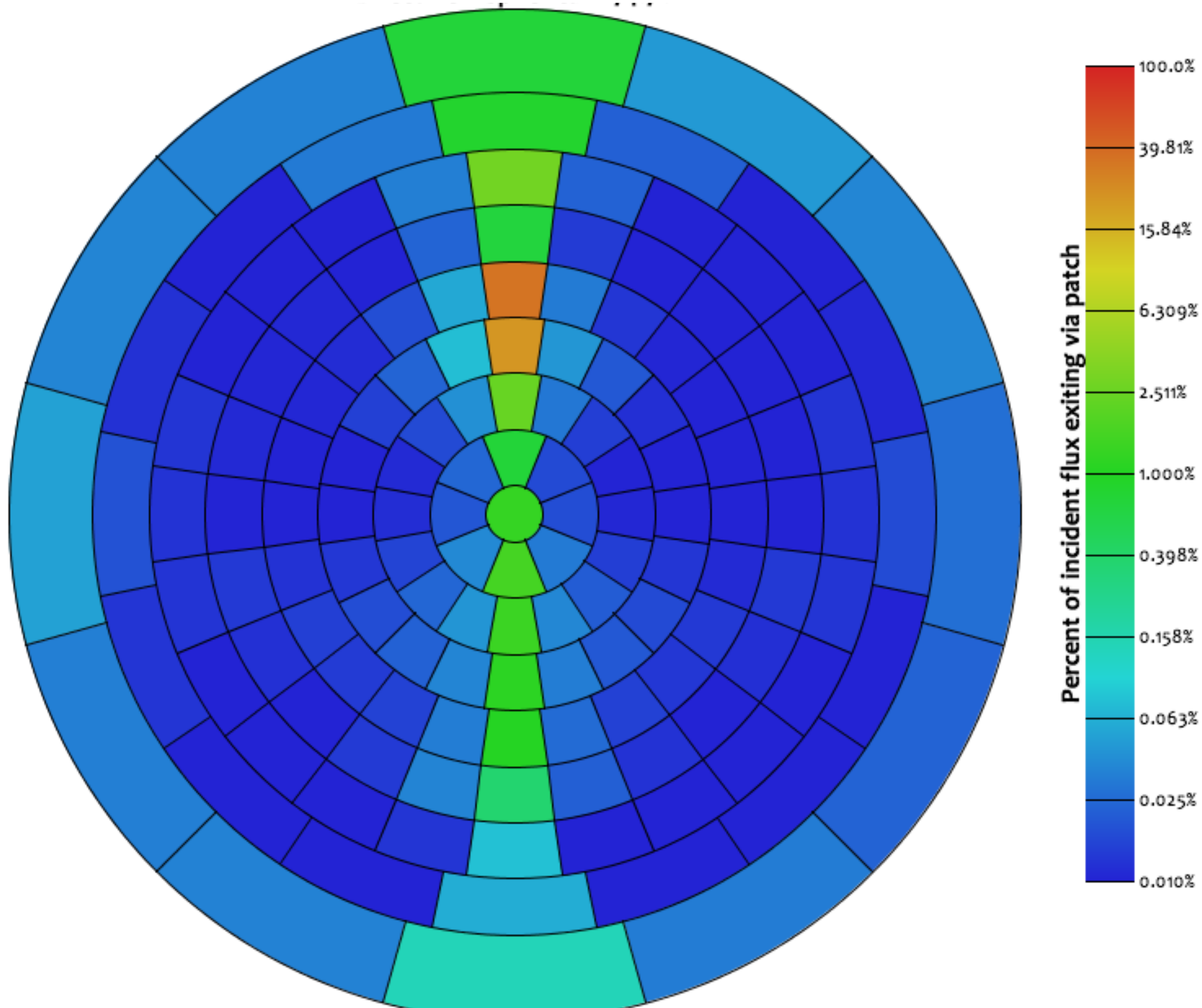
A micro-structured film refracts light towards the ceiling



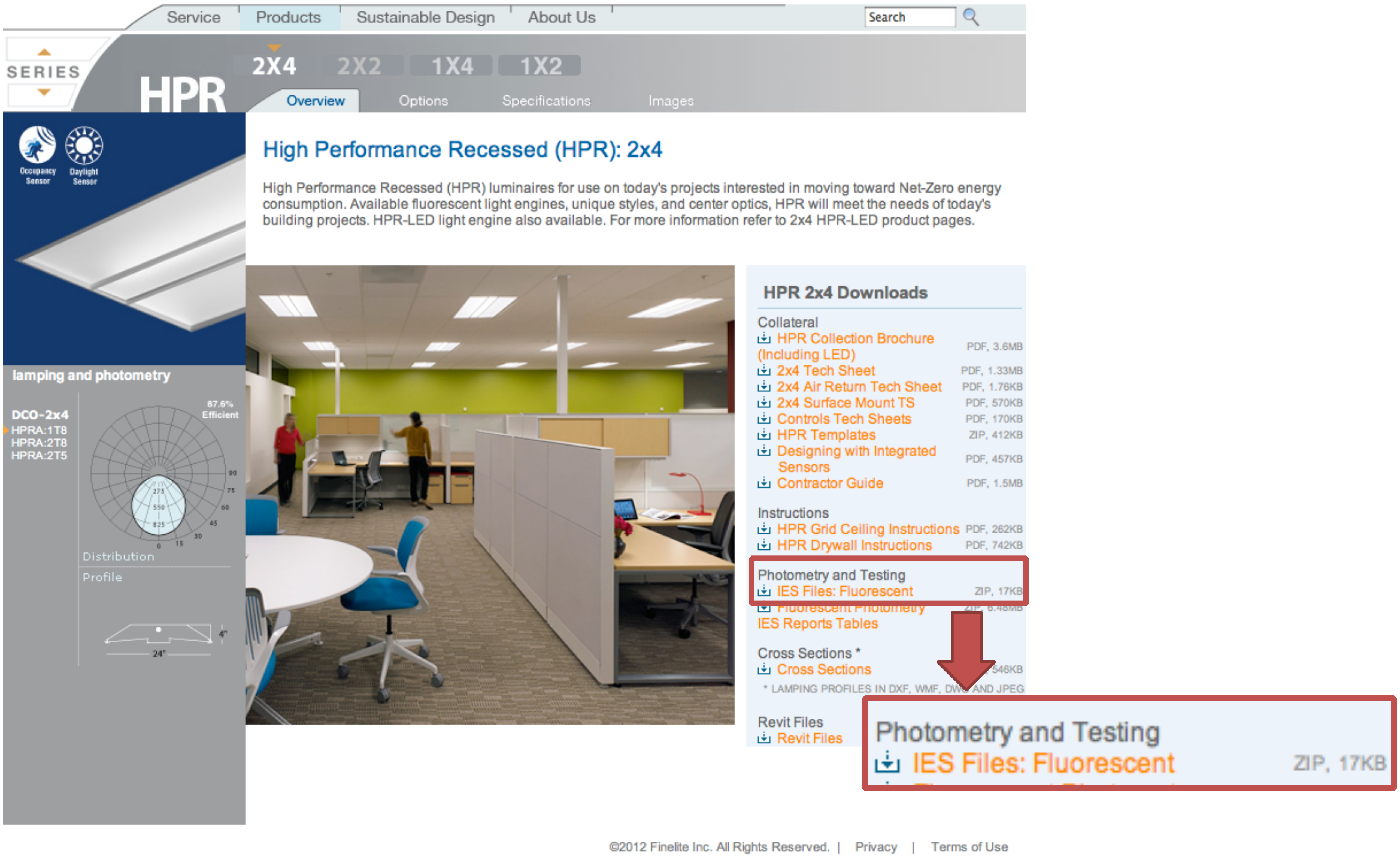
Incident Hemisphere



Transmission Hemisphere



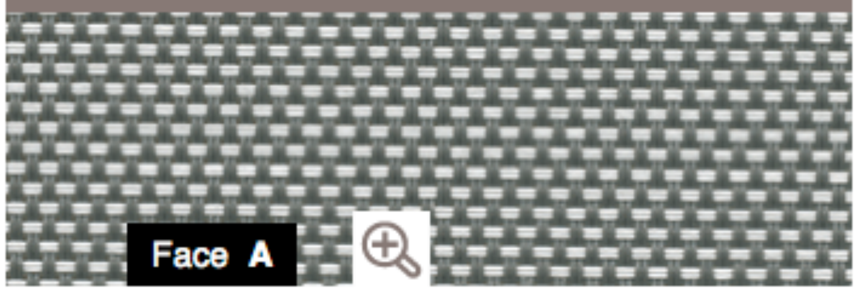
IES files enhanced our ability to simulate electric lighting, we hope BSDFs will do the same for daylight



BSDF files can help manufacturers, sales people, designers, owners and software developers

◀ Previous sheet | Imprimer la fiche | sélectionner ce coloris | Download the data PDF | Next sheet ▶

0102 Grey White



> Characteristics

FIRE RETARDANT
5 YEAR WARRANTY
M1 B1 BS F3

> Labels

CONFIDENCE IN TEXTILES Tested for harmful substances
GREENGUARD Children & Schools Indoor Air Quality Certified
Enduris™ GLASS CORE

RETOUR À LA LISTE

SV 3%

9 colours

> Width(s) :
250 cm

> Les plus produits :

+ Basket weave fabric to combine visual comfort with transparency
+ Outstanding glare control: up to 96% of light rays filtered (Tv = 4%)

> Applications :
Internal - Printable

Decorative panels / Roller blinds / Roman shades / Roof windows blinds / Skylight blinds / Tensile structures / Velums

> Technical datas :

Composition	36% Fibreglass - 64% PVC
Weight per m²	375 g/m² ± 5%
Thickness	0.48 mm ± 5%
Visible transmittance	6 %
Internal solar factor (gv=0.59)	gtot : 0.47

Using a BSDF in Radiance is often better than modeling the geometry a complex fenestration system

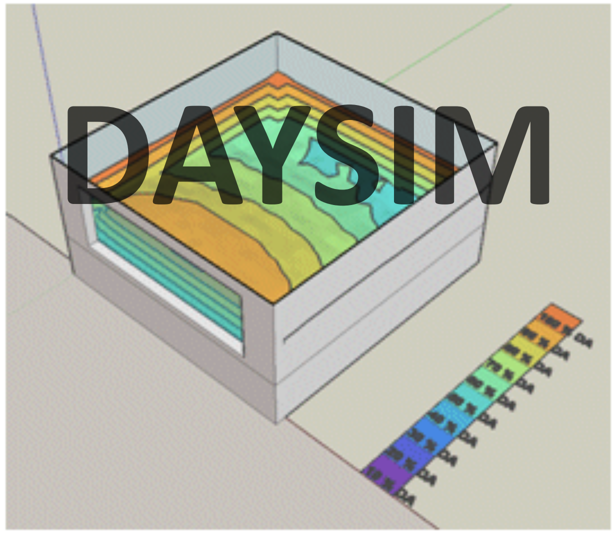
- Reduces simulation load & saves time
  - Avoids many additional polygons
  - Reduces inter-reflections required
- Some systems can’t be modeled conventionally.
  - i.e. systems with microscopic geometry
- Some systems can’t be simulated conventionally.
  - i.e. systems that exhibit diffraction

Simulation programs that accommodate BSDFs

Radiance
Synthetic Imaging System

EnergyPlus

OpenStudio



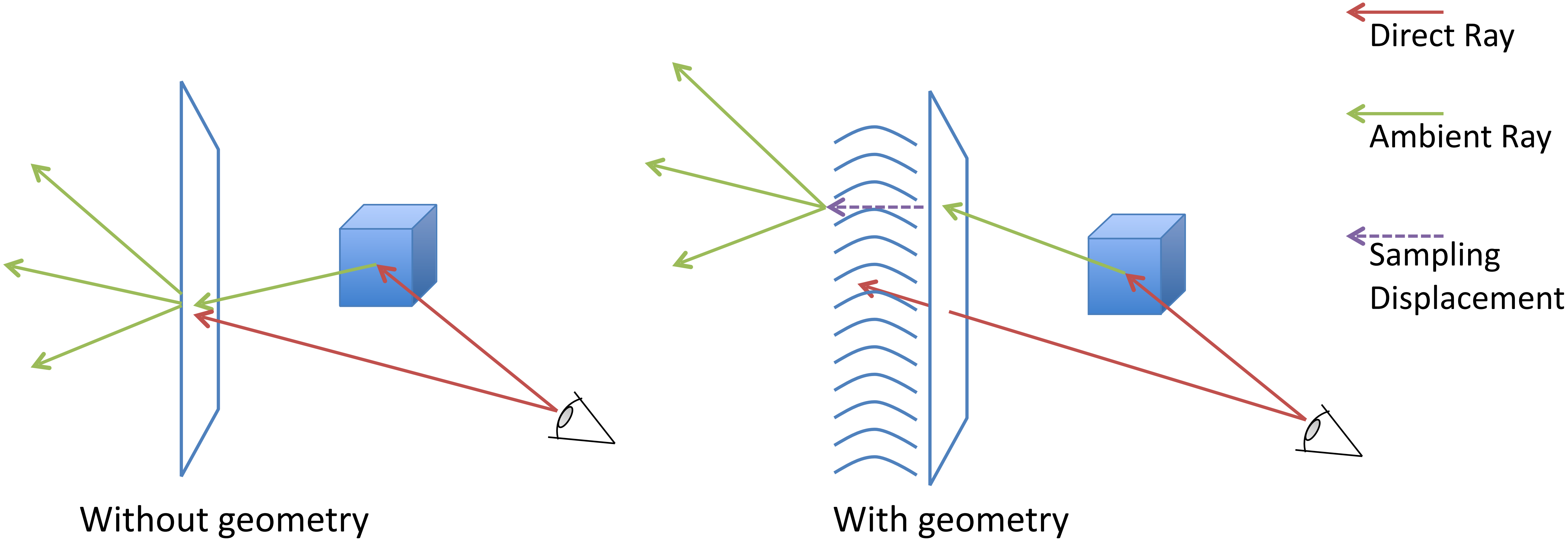
(nearly completed implementation)

Radiance BSDF Capabilities

- BSDF based material primitive allows BSDFs to be used in any simulation
- “Quick” annual daylight simulations are possible with BSDFs, including for dynamic systems.
- Users can create BSDFs (with genBSDF) for use in simulations

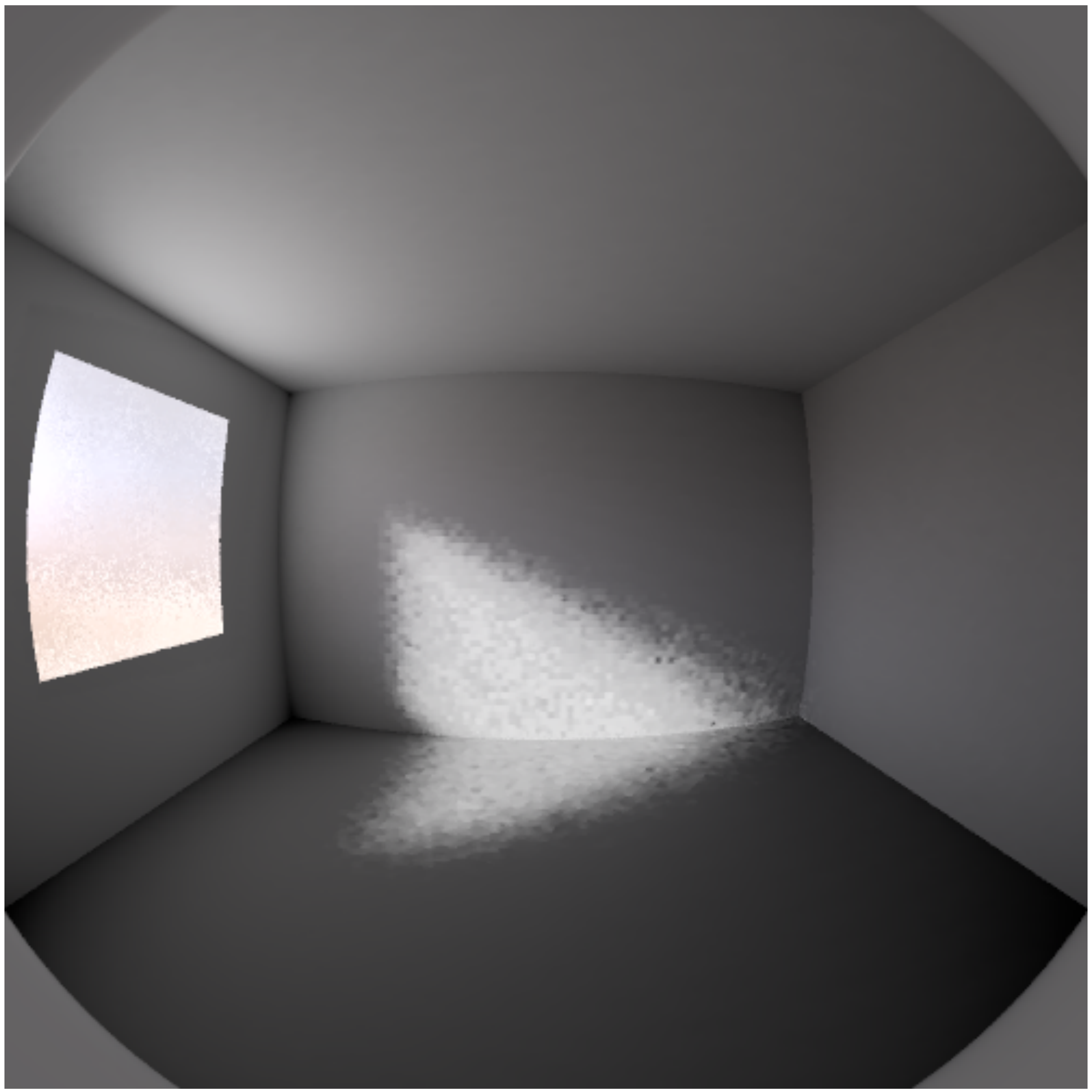
Radiance BSDF Material

- A material primitive using BSDF data
- Proxied geometry can be included for better rendering.

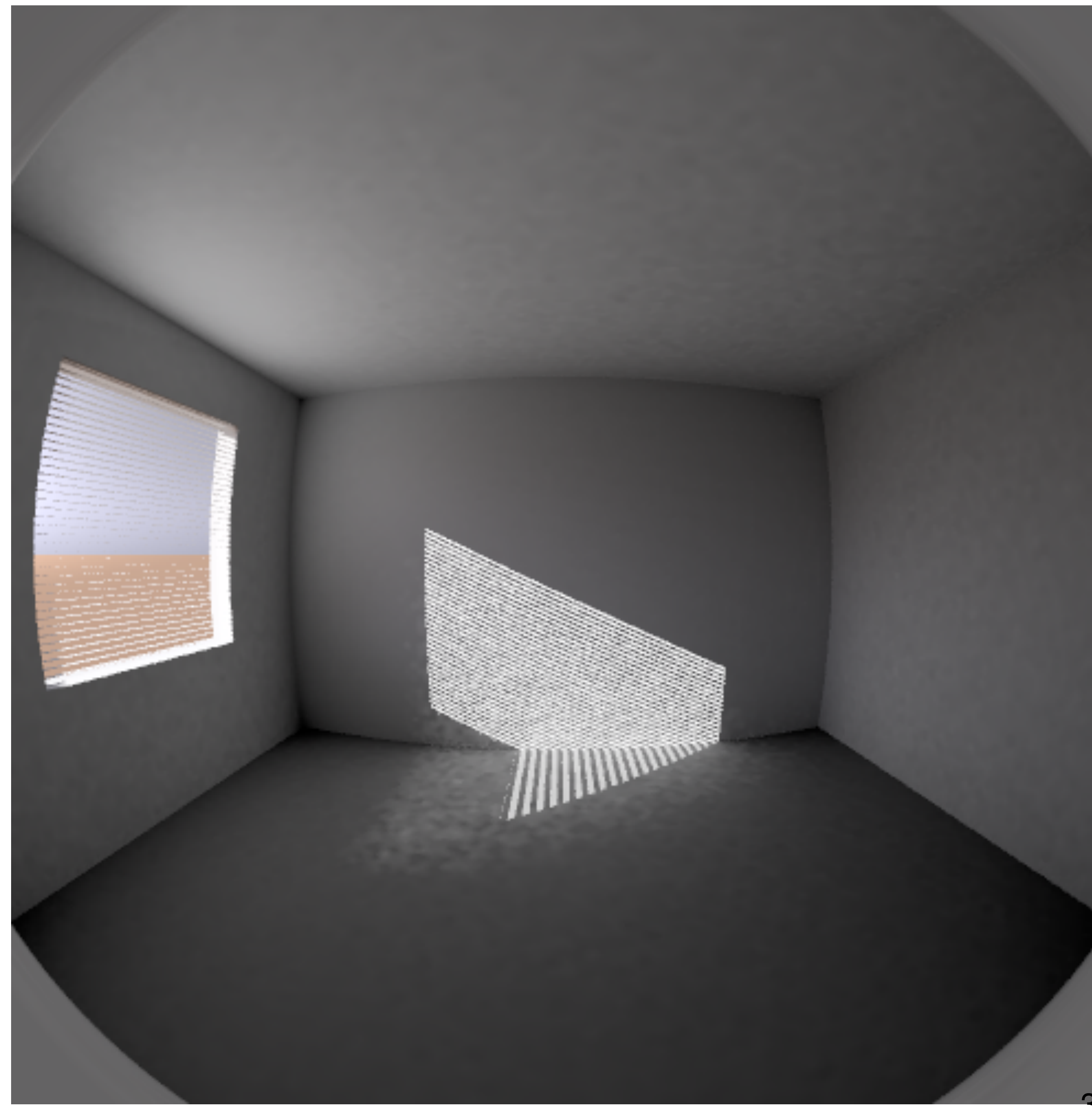


Rendering w/ Venetian Blinds

Without geometry

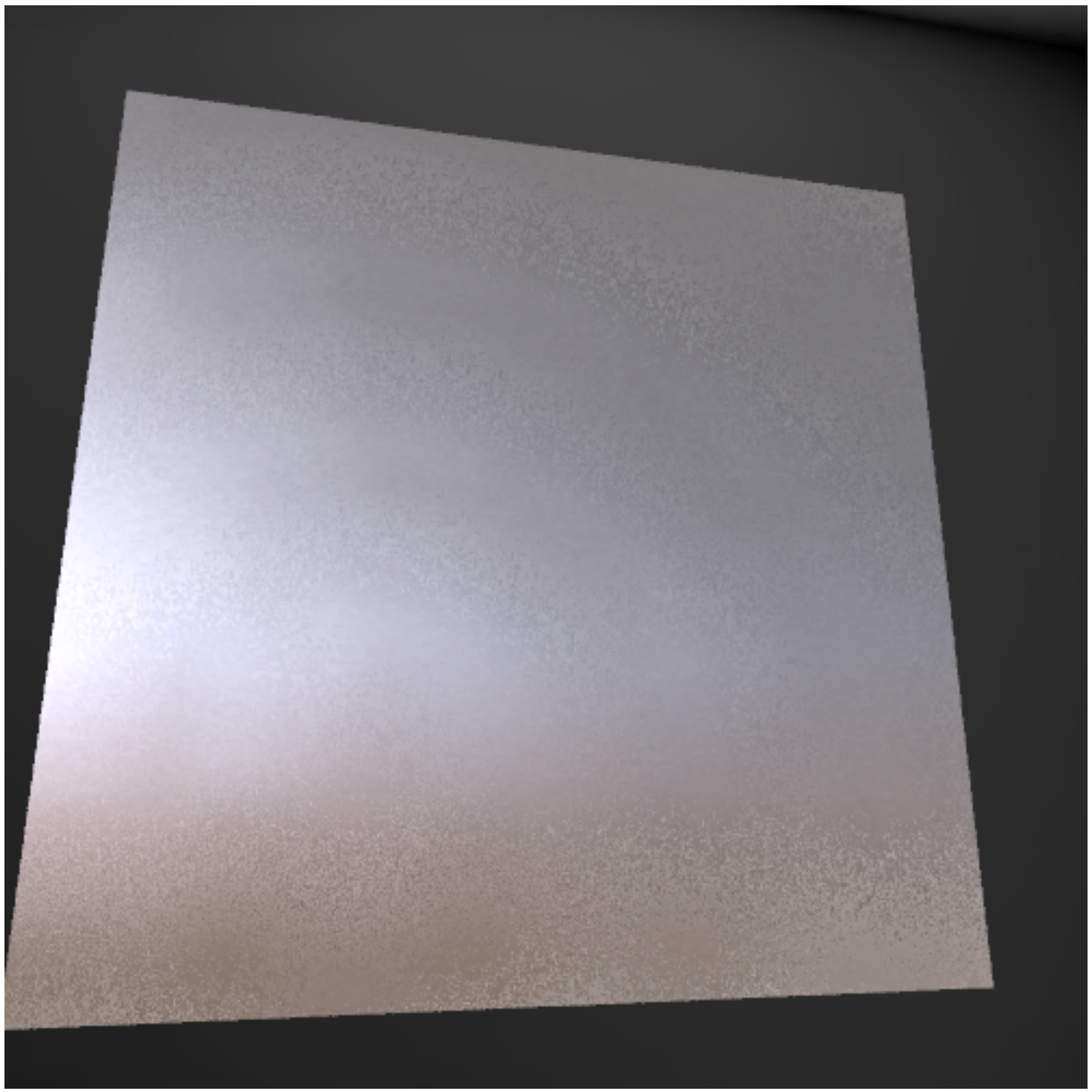


With geometry

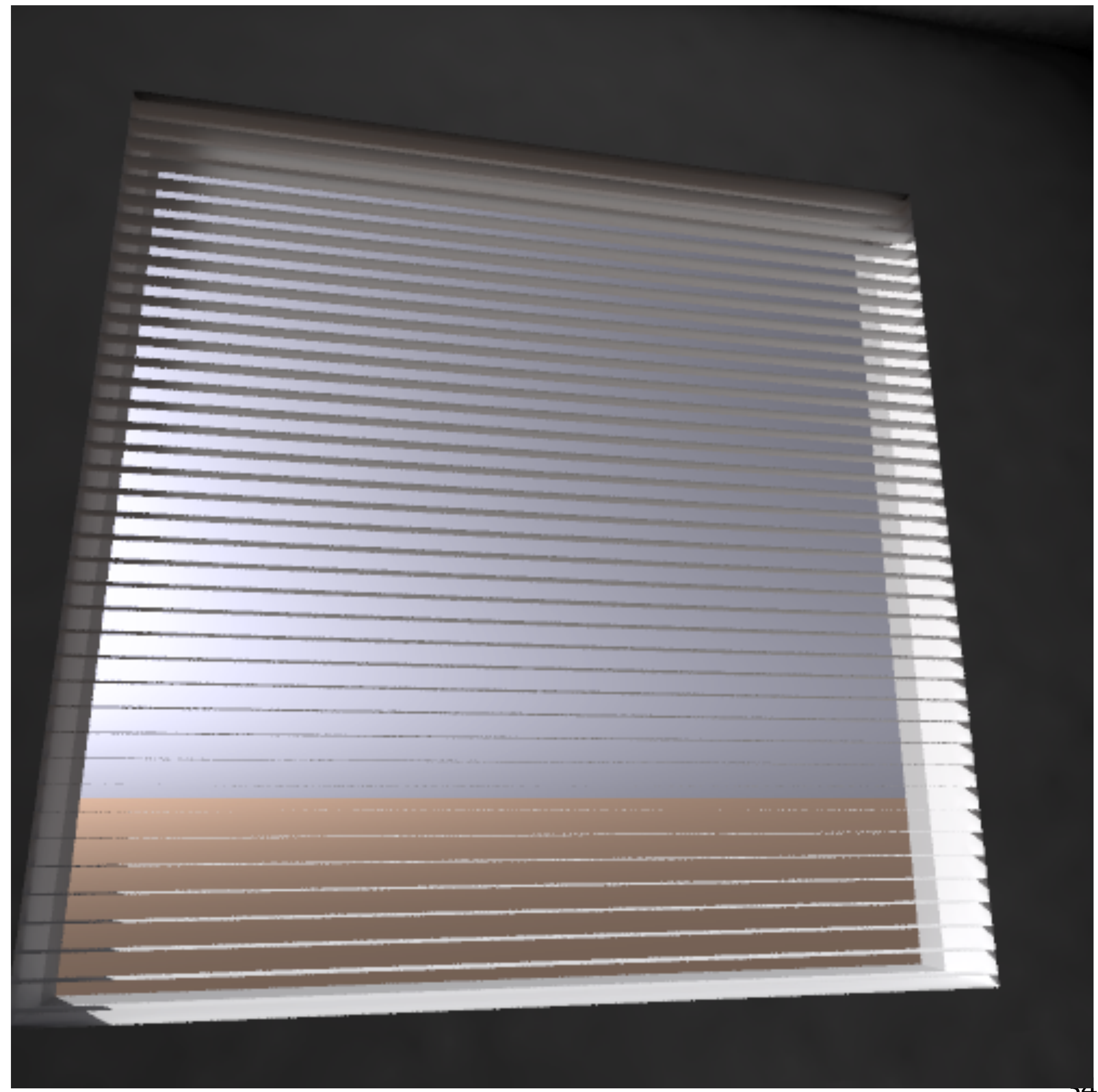


Rendering w/ Venetian Blinds

Without geometry

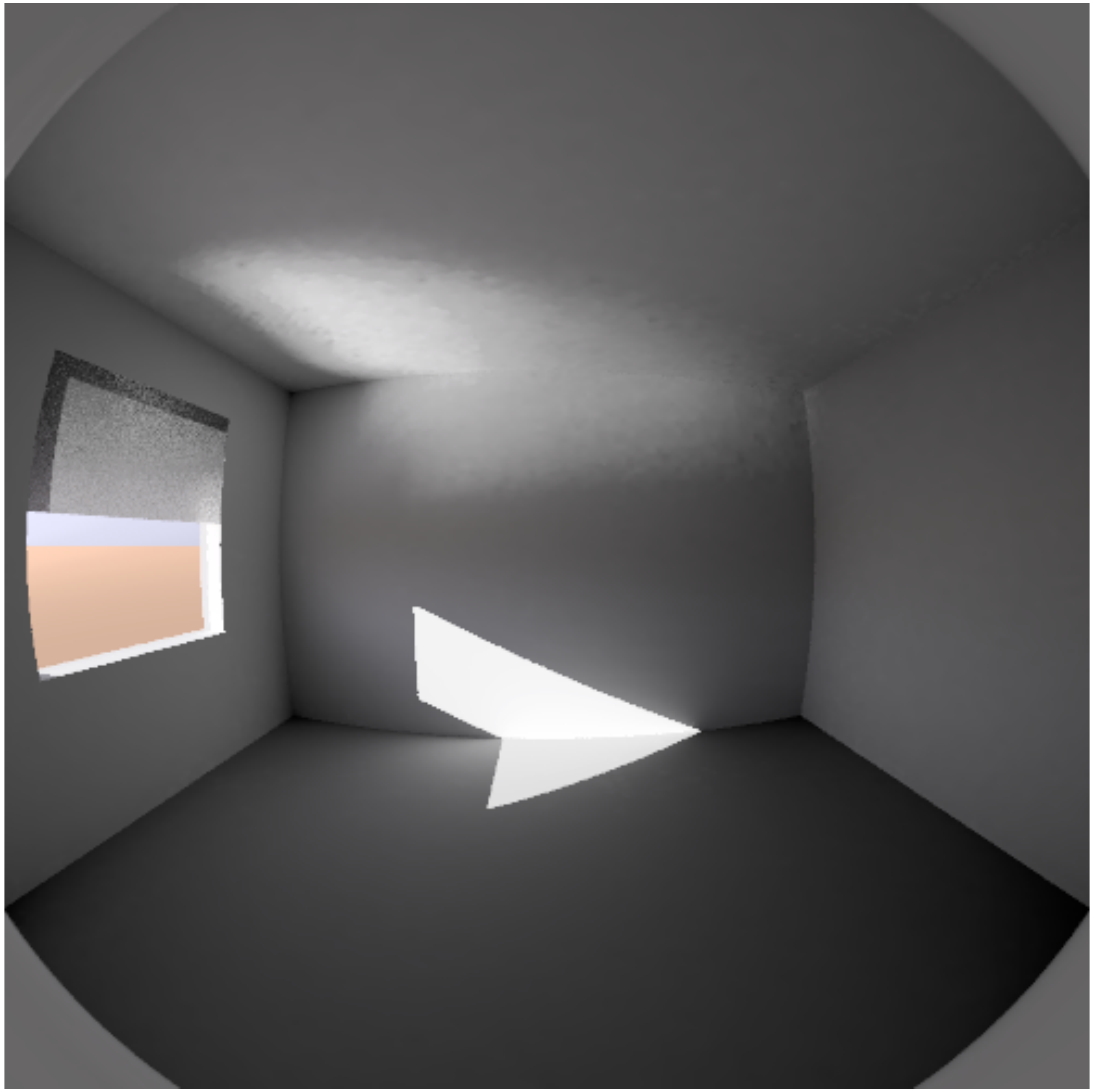


With geometry

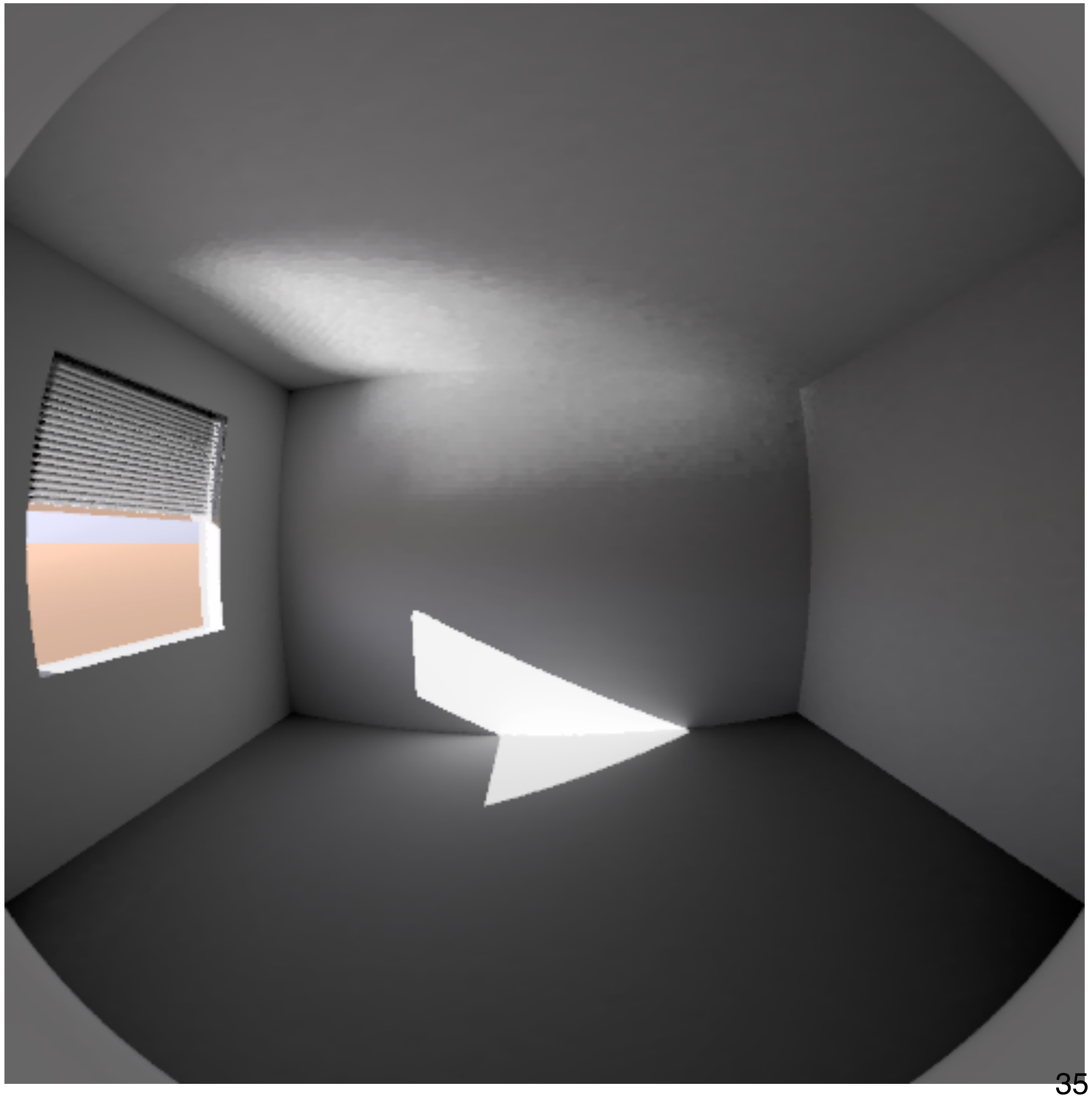


Rendering w/ Optical Light Shelf

Without geometry

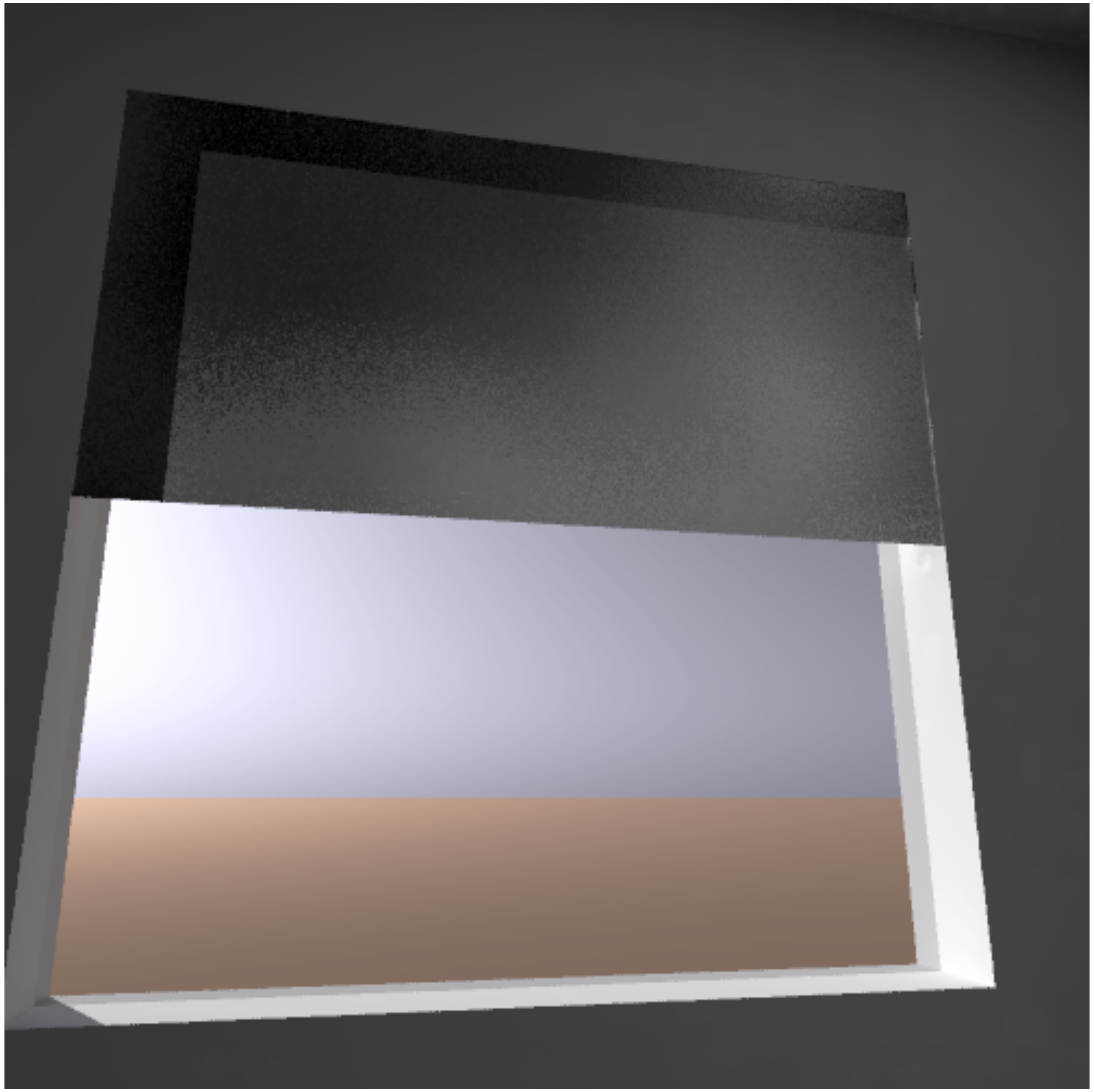


With geometry

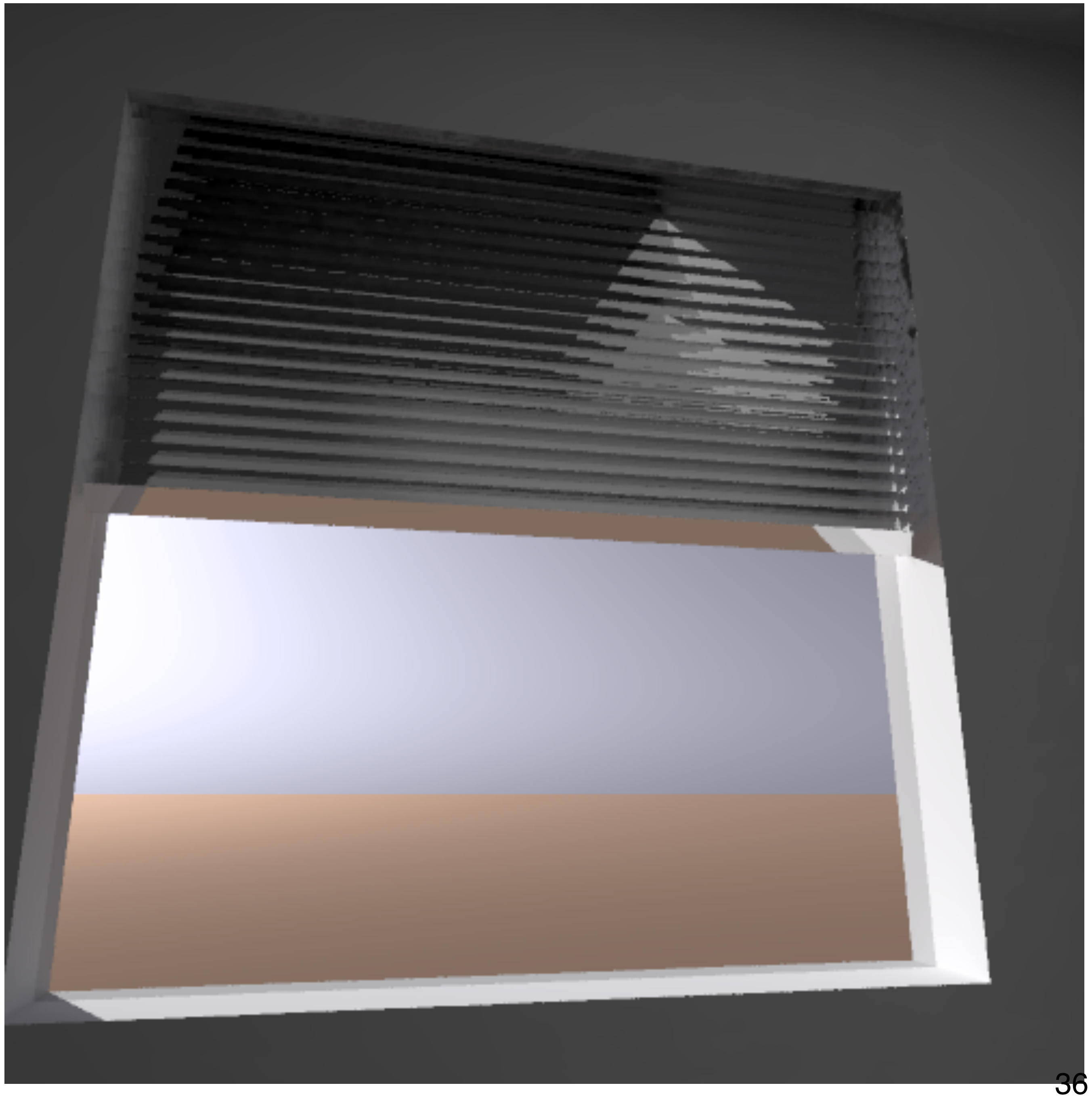


Rendering w/ Optical Light Shelf

Without geometry



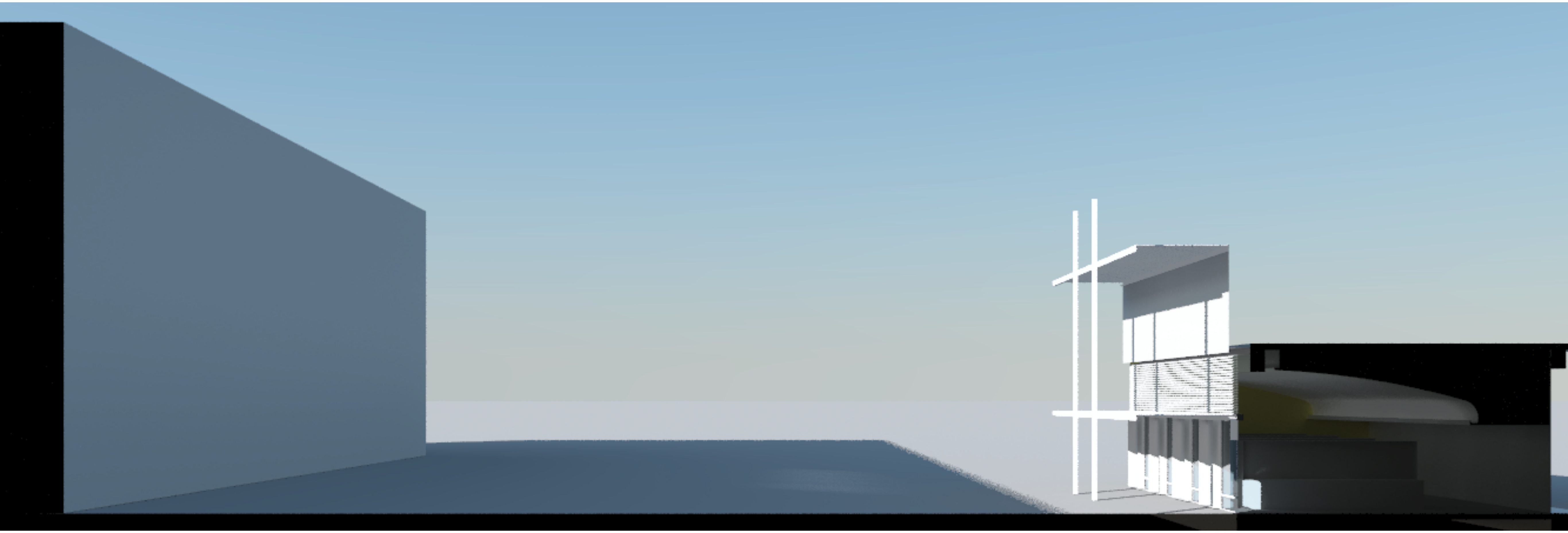
With geometry



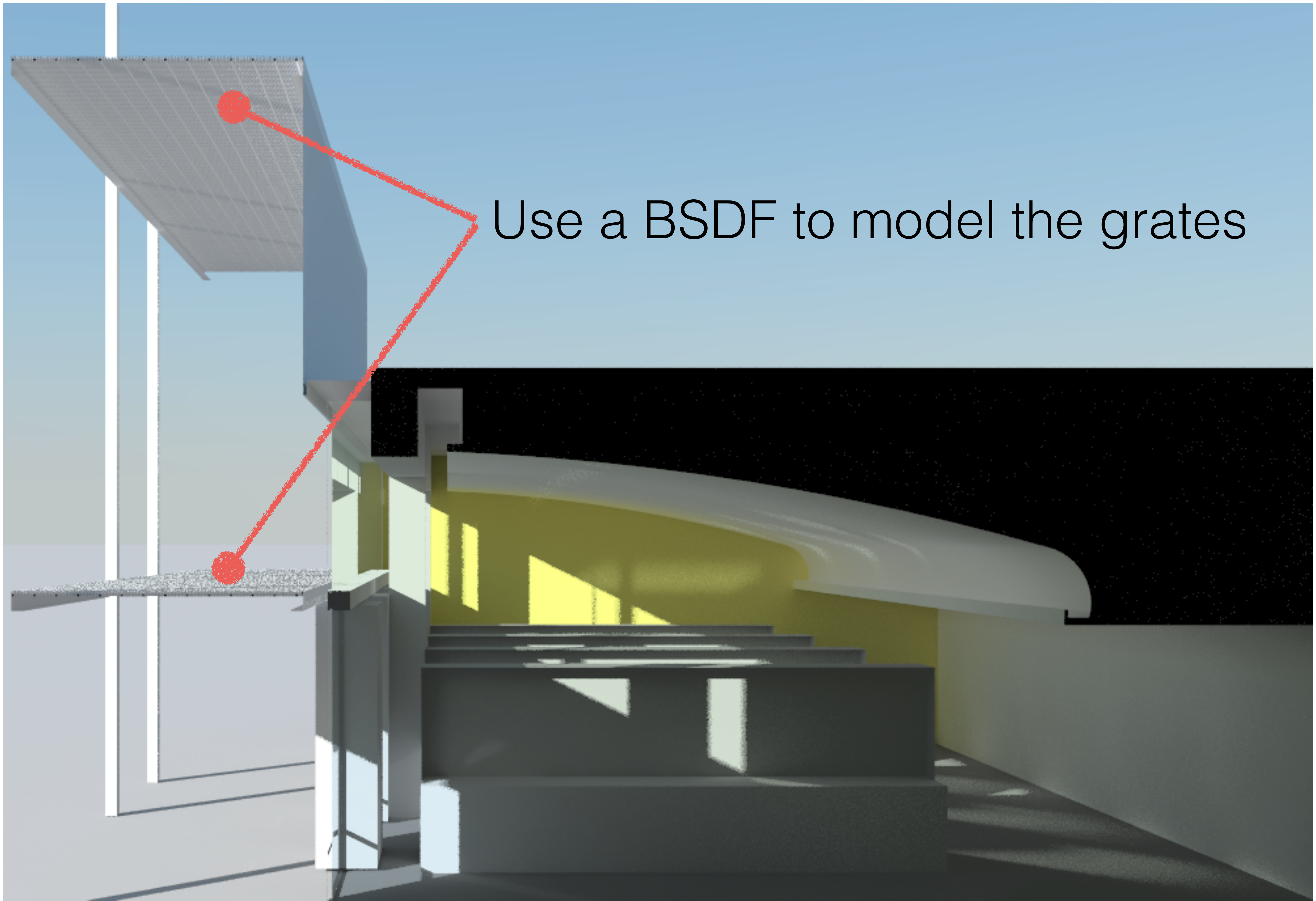
BSDF Example


EDCO

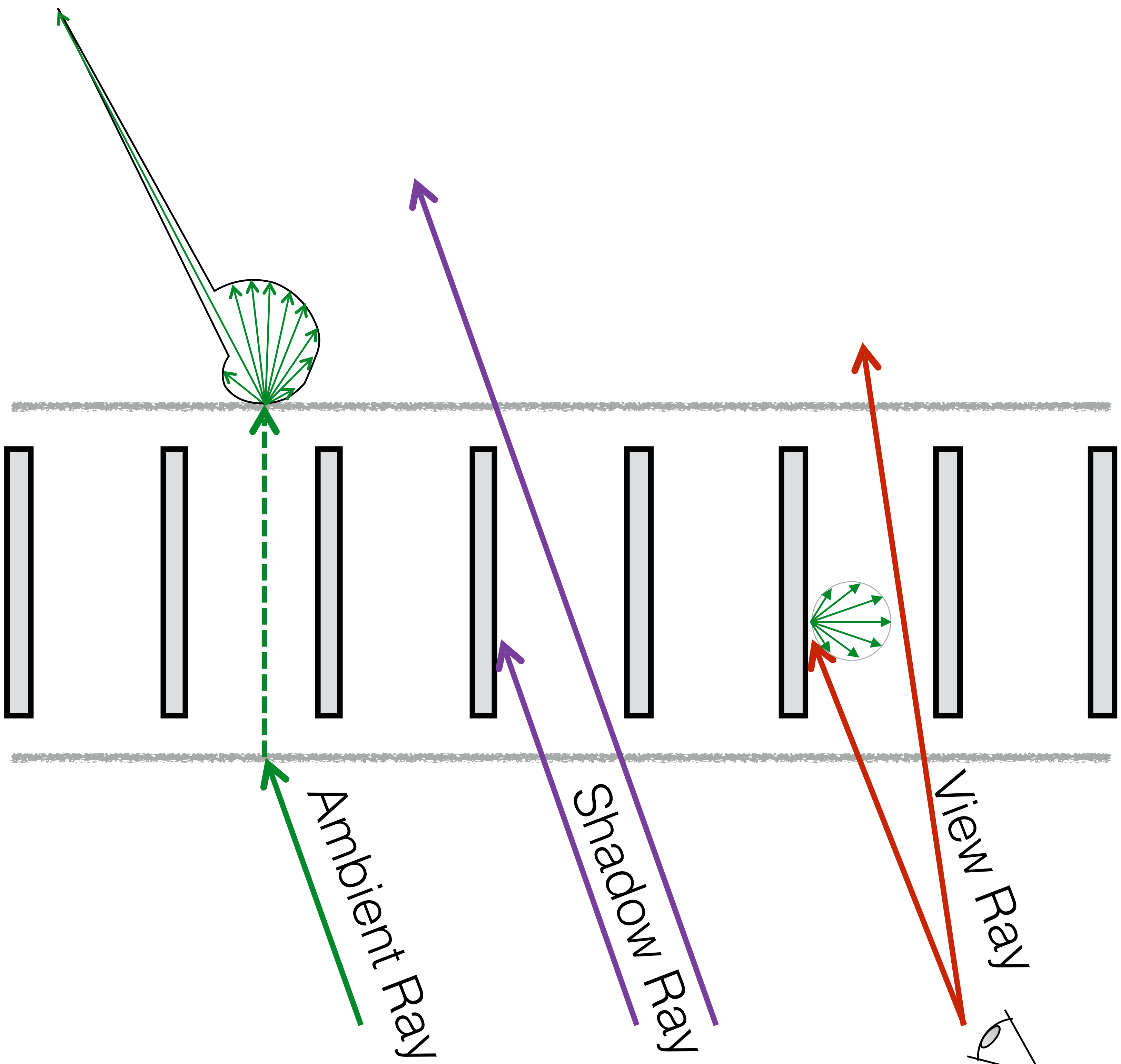
Radiance Model



Radiance Model

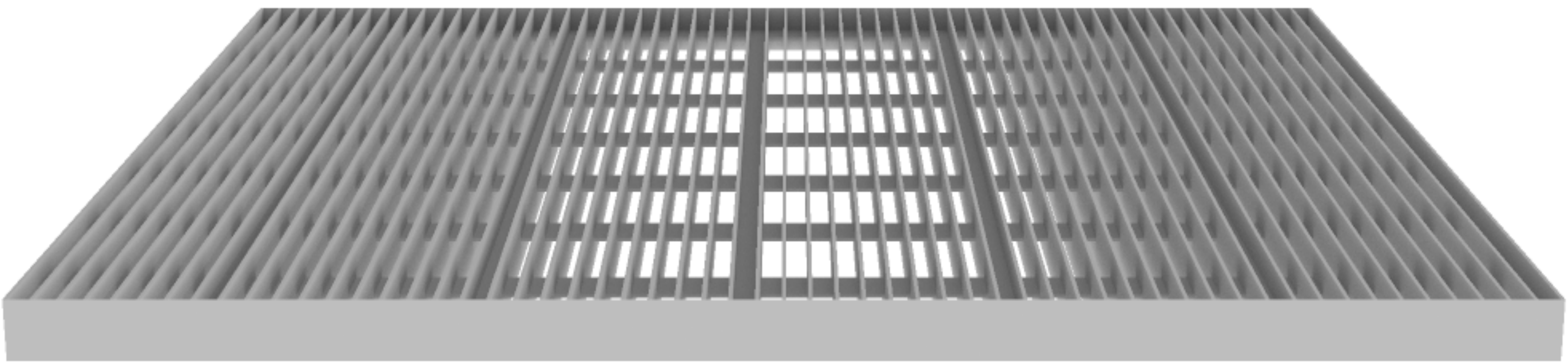


Sandwich the grate with impostor BSDF surfaces.



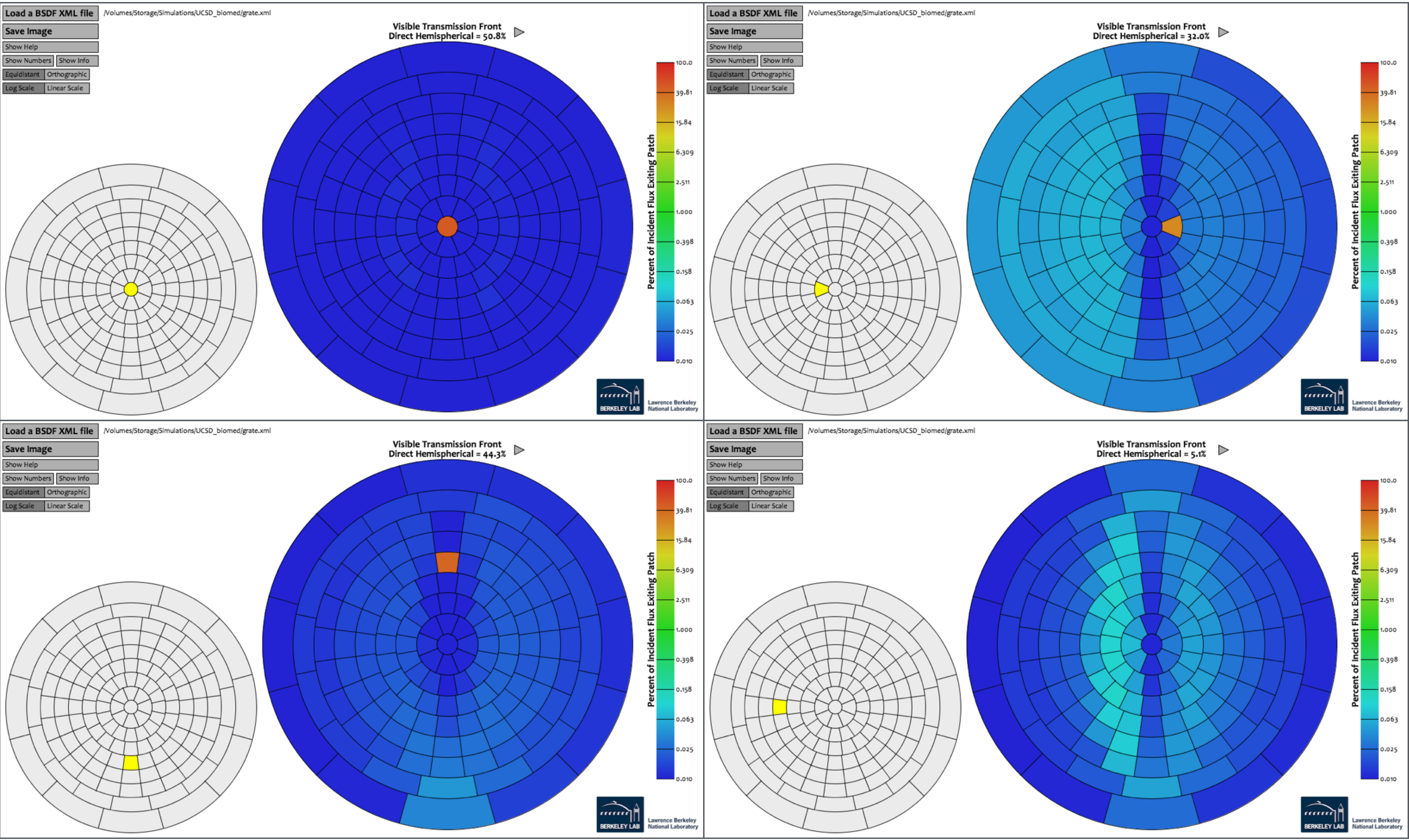
1. Create a BSDF of the grate

- Create a smaller representative model.
  - Model should reside in -Z half space
  - +Z is front / outside / opposite normal
  - enclose edges



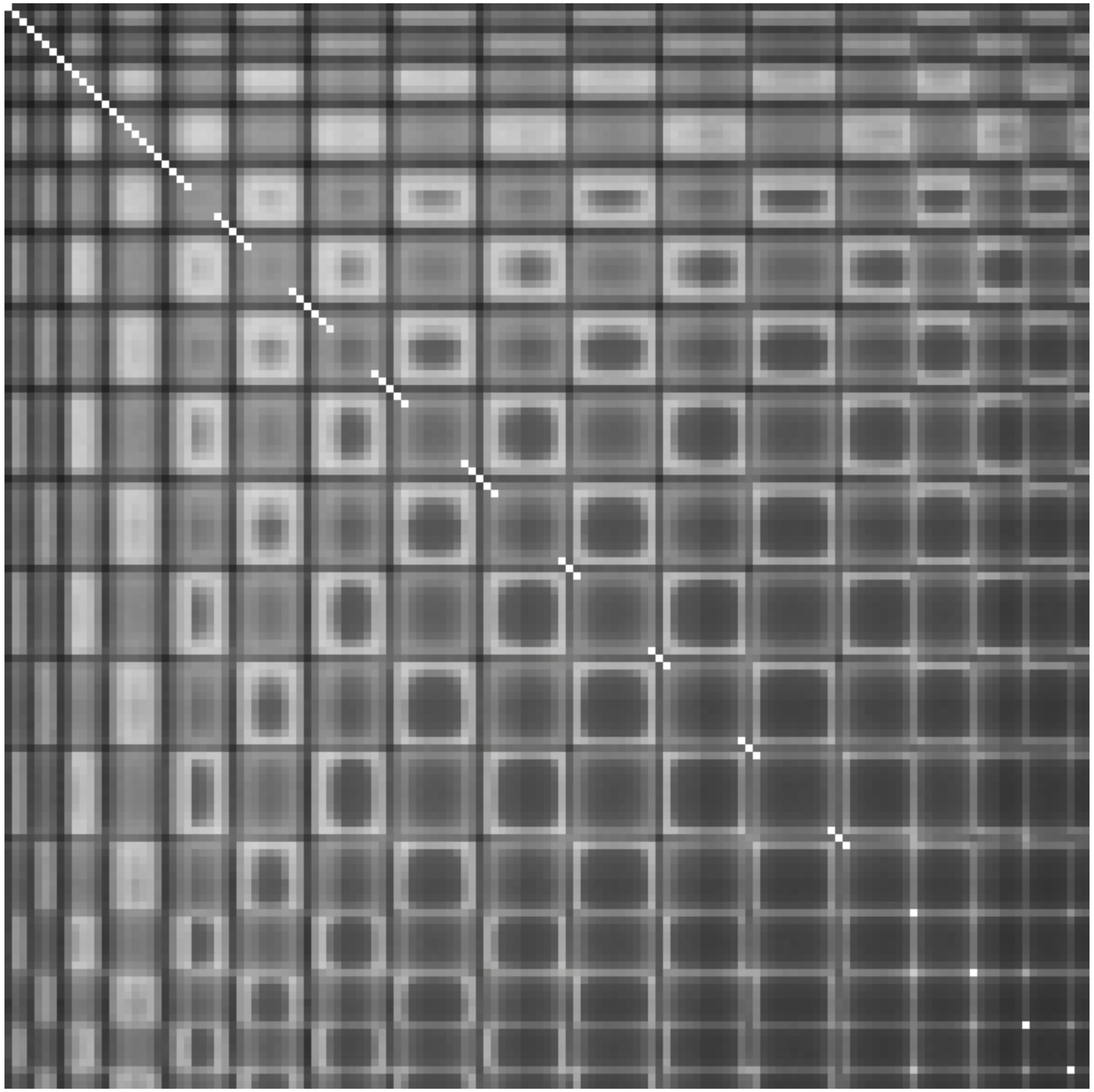
- genBSDF +b +f -n 16 -c 2000 -geom inch grate.rad

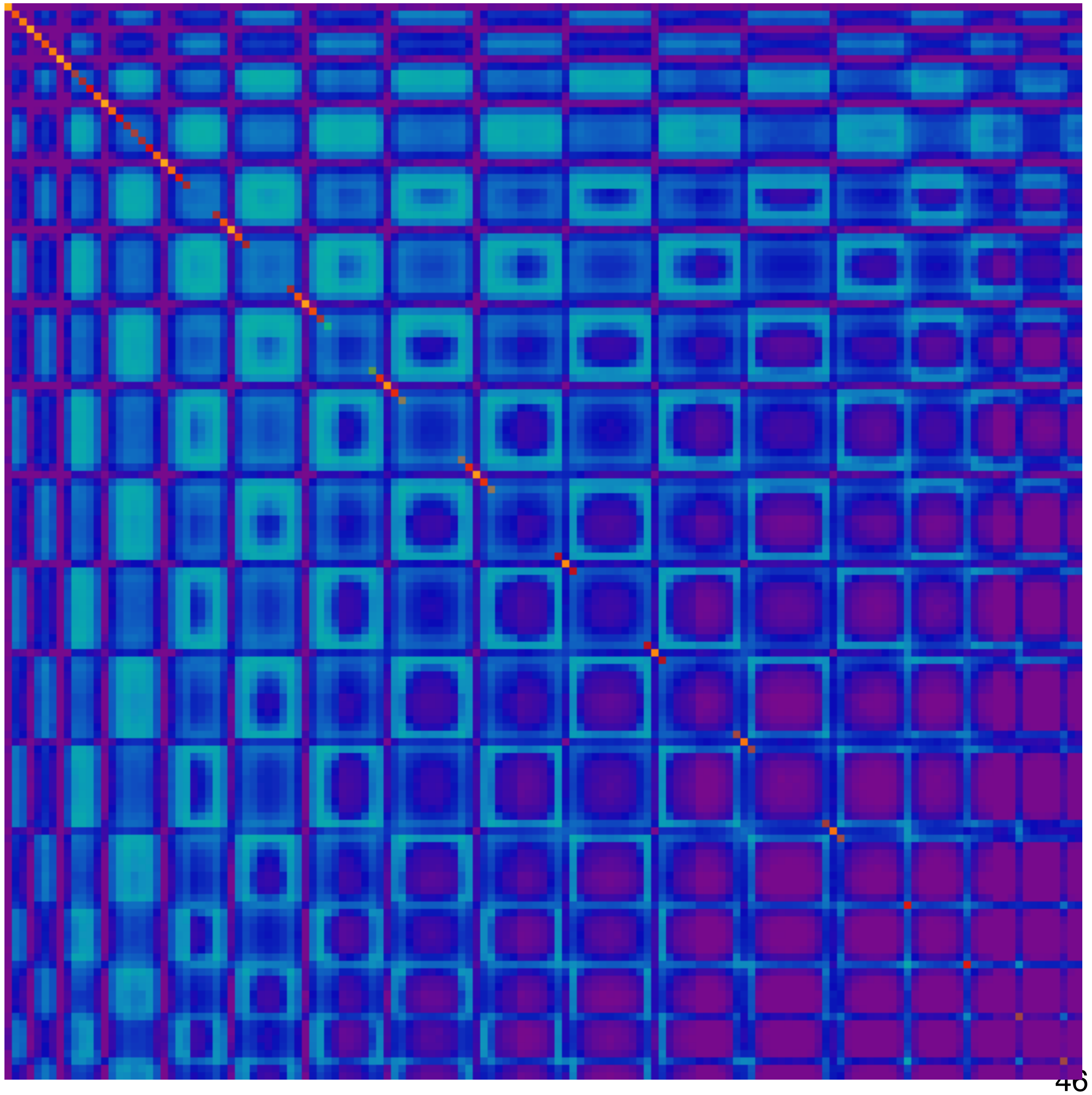
1b. Make sure you have a sensible BSDF

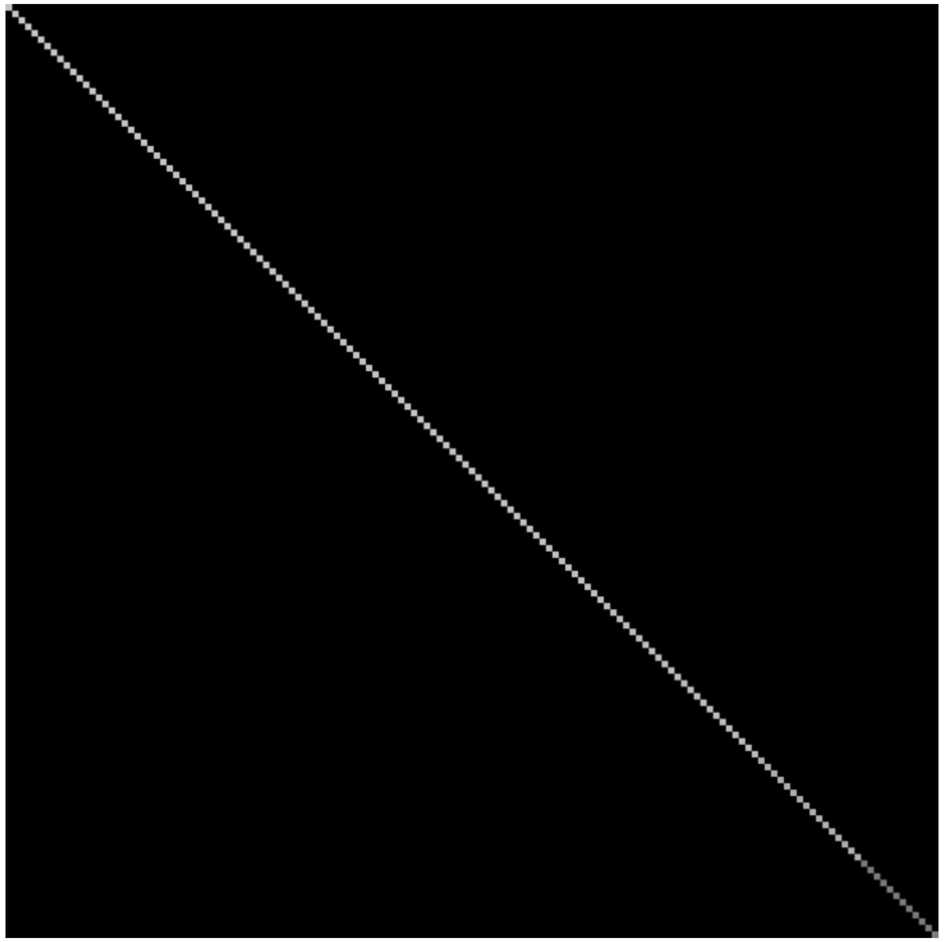


1b. Make sure you have a sensible BSDF

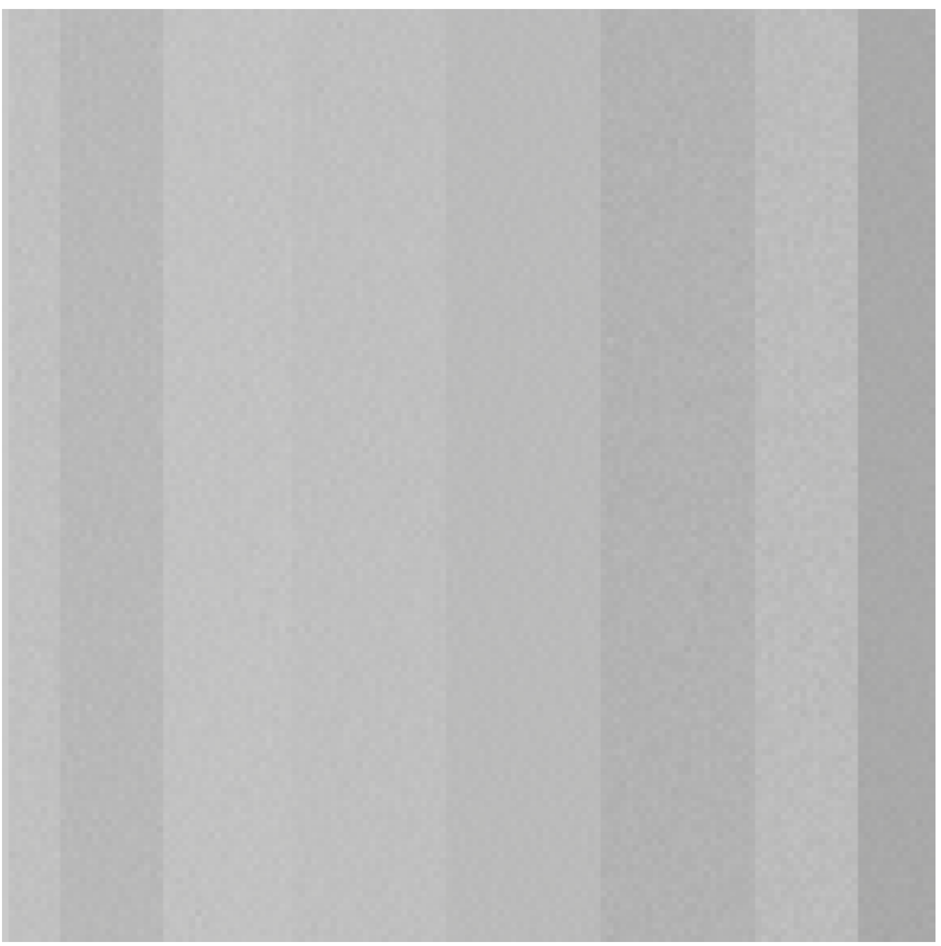
rmtxop -fc grate.xml | pfilt -x 725 -y 725 > grate_bsdf.hdr



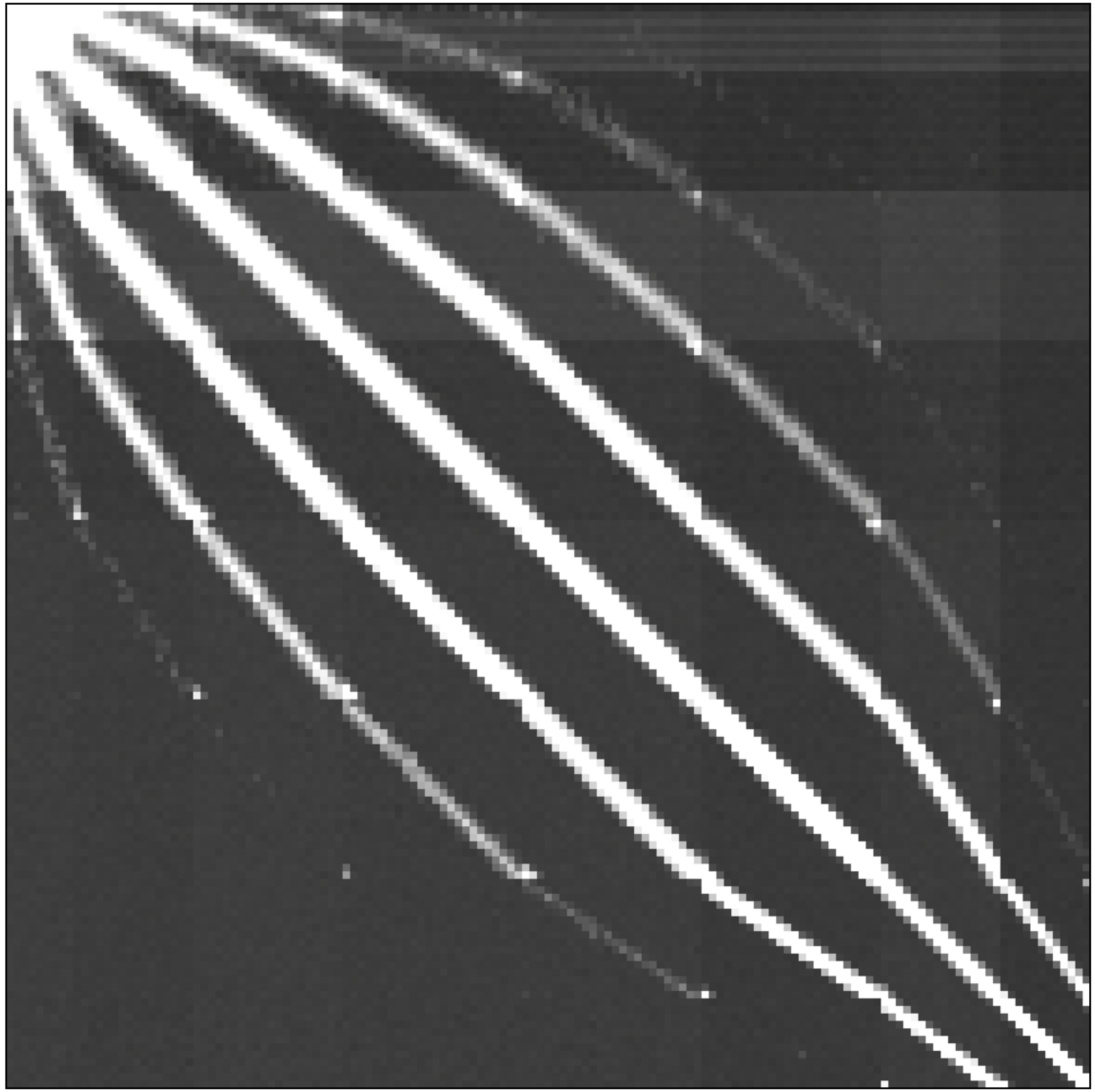




Specular BSDF



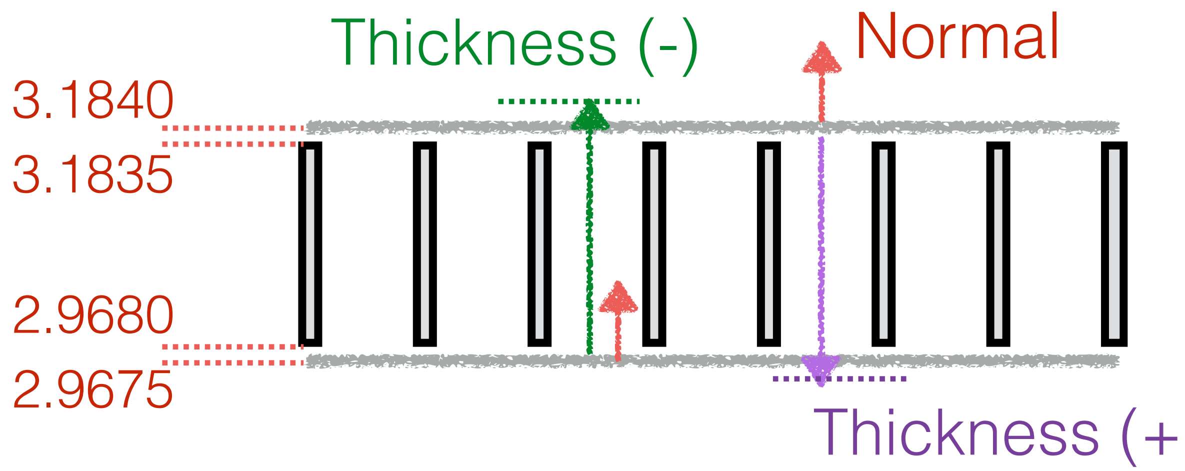
Diffuse BSDF



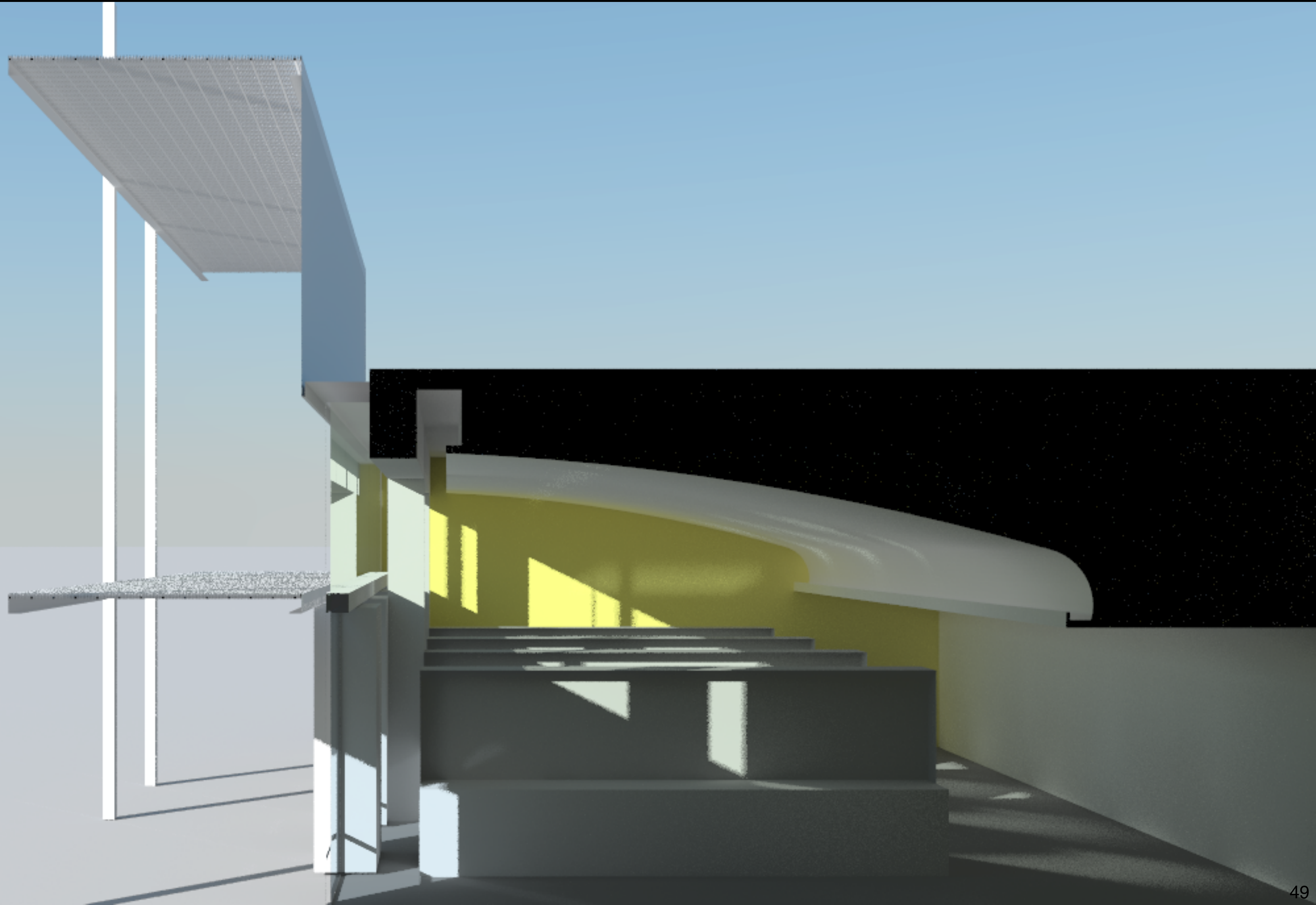
Semi-Diffuse

2. Sandwich the geometry

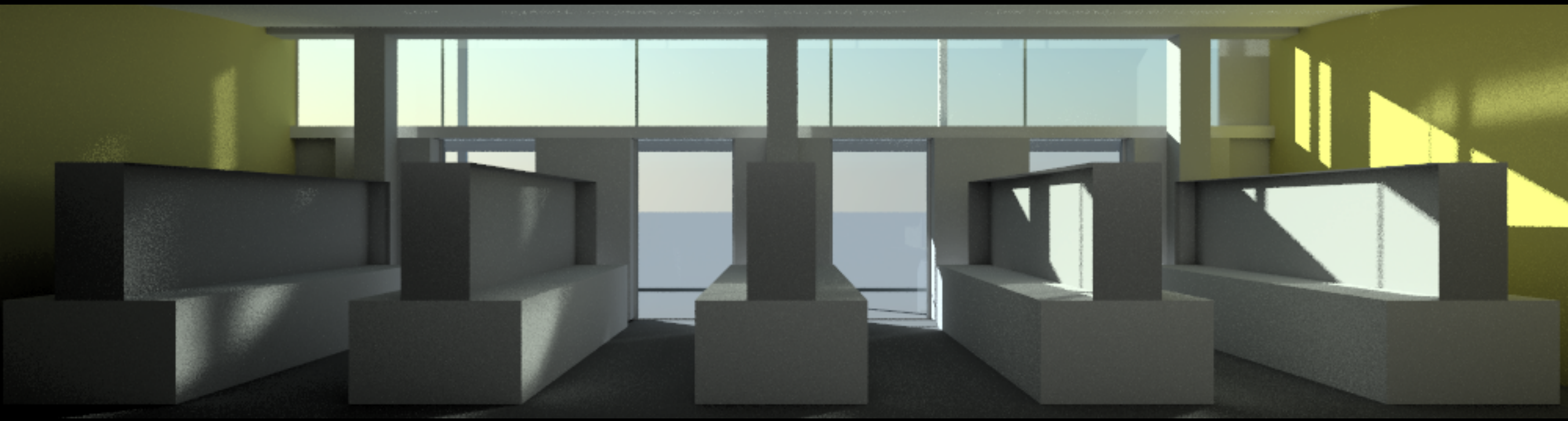
```
void BSDF grate_top
6 0.05091 grate.xml 0 1 0 .
0
0
grate_top polygon zone23.rad04614
0 0 12      -7.22613  0.25400  3.0184
            -4.40417  0.25400  3.0184
            -4.40417  20.94617  3.0184
            -7.22613  20.94617  3.0184

void BSDF grate_bottom
6 -0.05091 grate.xml 0 1 0 .
0
0
grate_bottom polygon zone23.rad04614
0 0 12      -7.22613  0.25400  2.96750
            -4.40417  0.25400  2.96750
            -4.40417  20.94617  2.96750
            -7.22613  20.94617  2.96750

!xform -s .0254 -t -7.22613 0.25400 3.01835 shadegrate.rad
```

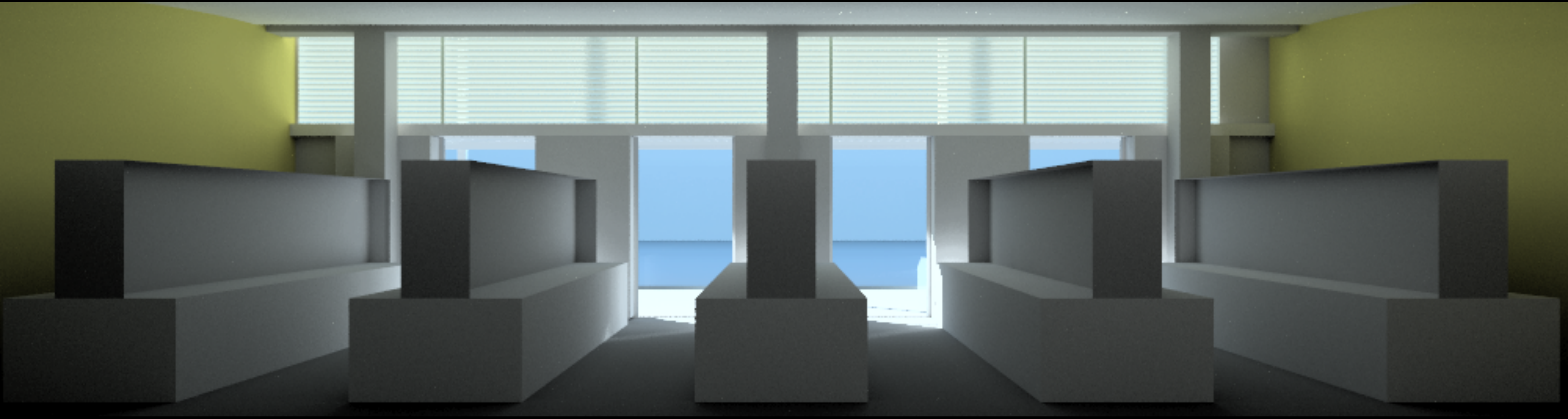


Good To Go!



Good To Go!





New and Improved

Annual Simulations

now with rfluxmtx and rmtxop!

Daylight Coefficient Method

one/two phase method?

The Equations

Daylight Coefficient Equation:

$$E_{DC} = C_{dc}S$$

Three-Phase Equation:

$$E_{3ph} = VTDS$$

Five-Phase Equation:

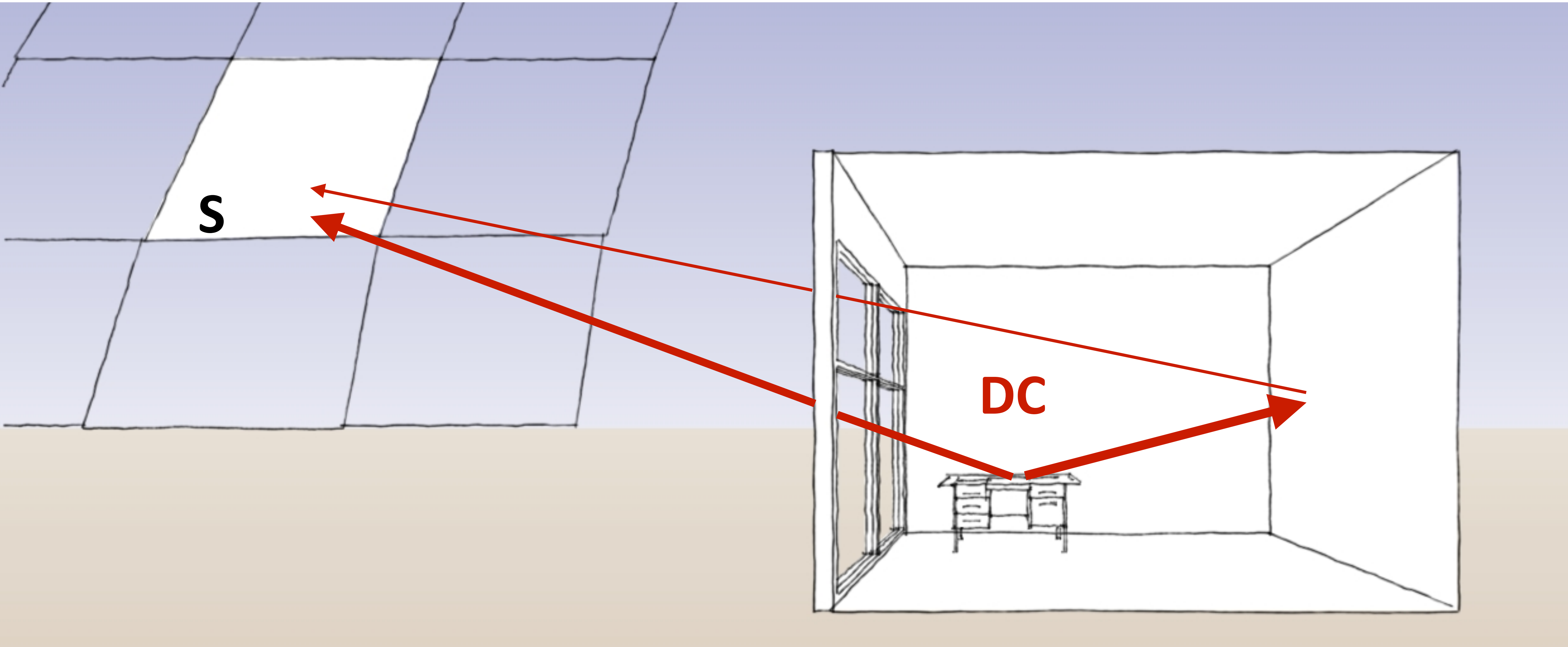
$$E_{5ph} = VTDS - V_dTD_dS_d + C_{ds}S_{sun}$$

$E_{DC} = C_{dc}S$

Daylight Coefficient Matrix (D_{dc})

Coefficients that describe the contribution of each sky patch to scene illumination.

Includes inter-reflection

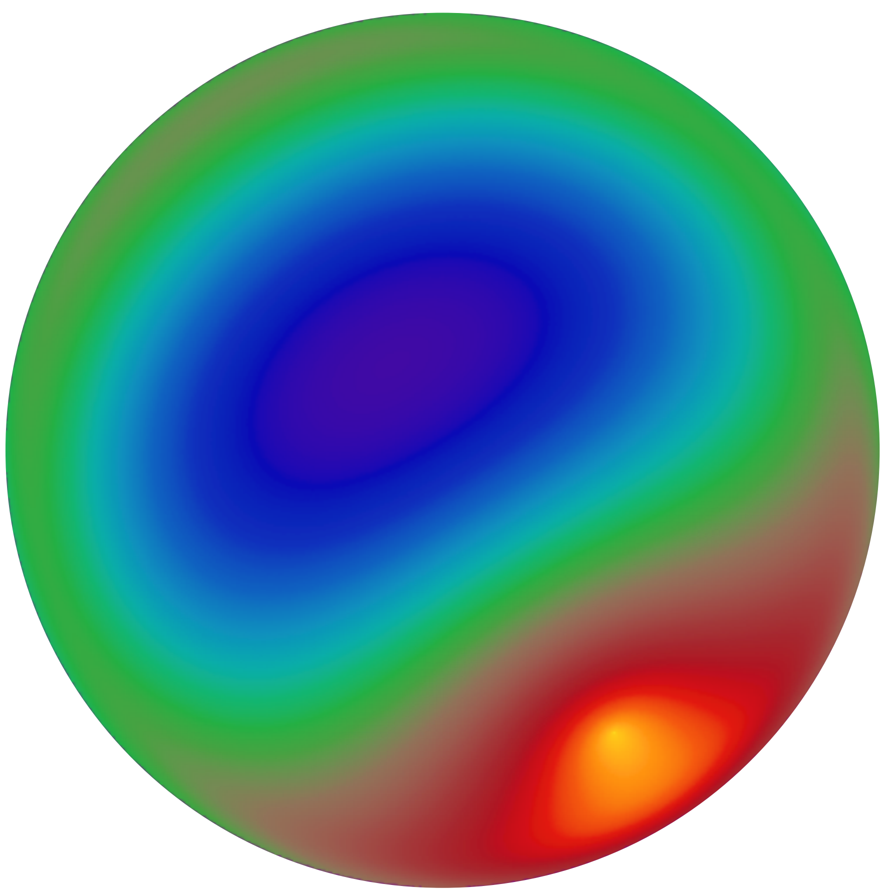


$E_{DC} = C_{dc}S$

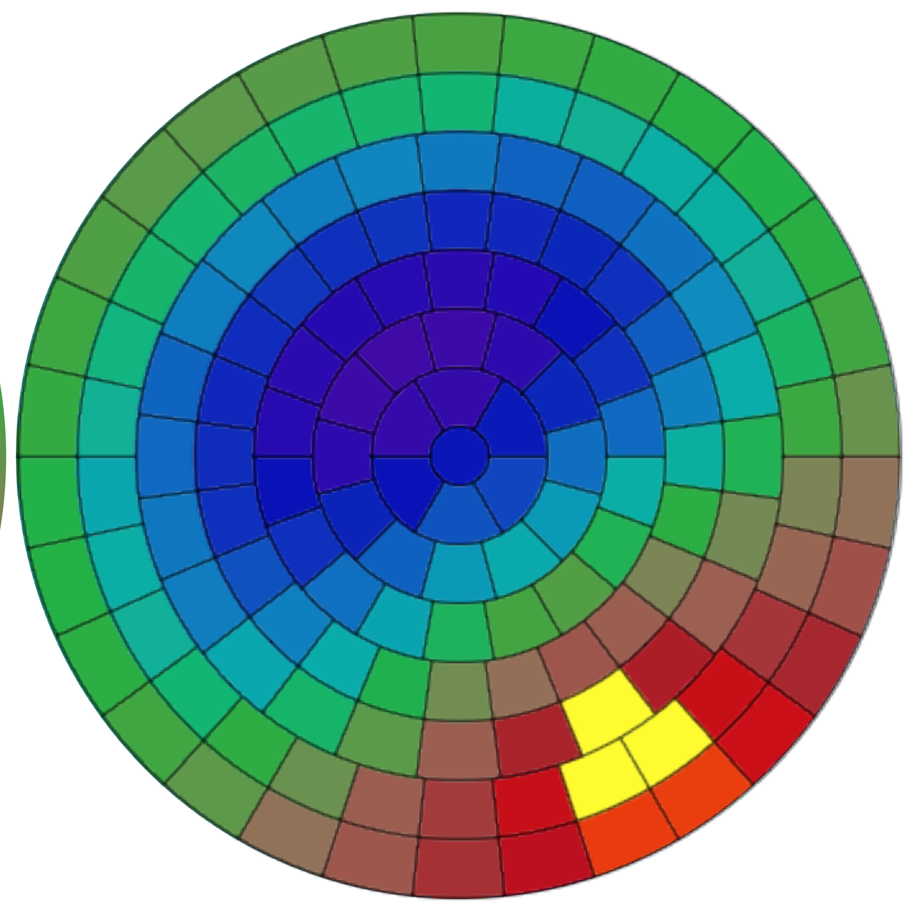
Sky matrix (S)

A sky vector contains average sky luminance in a discretized sky patch for a specific sky luminance pattern (i.e. clear sky at 15:00 on December 21) . (created using genskyvec)

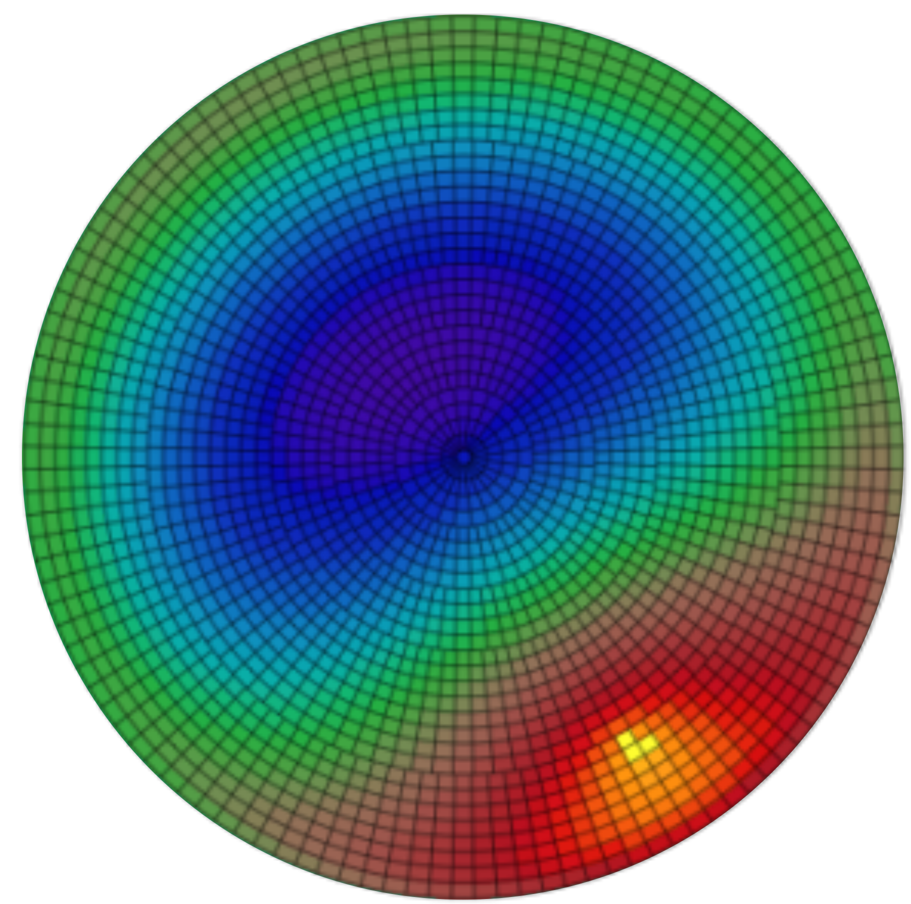
A sky matrix is a series of sky vectors encompassing many time steps. (created from a weather data file using gendaylit)



Sky Luminance Gradient



Discretized Sky Luminance (Tregenza)



Discretized Sky Luminance (Reinhart MF:4)

Creating the Daylight Coefficient Matrix

```
rfluxmtx -n 8 -I+ -ab 12 -ad 10000 -lw 1e-4 \
        < points.txt  -  skies/sky.rad \
        materials/testroom.mat objects/testroom.rad \
        > DC.dmx
```

Old:

```
rcontrib -n 8 -I+ -ab 12 -ad 10000 -lw 1e-4 \
    < points.txt -e MF:1 -f reinhart.cal -b rbin -bn Nrbins \
    -m sky_glow model.oct > DC.mtx
```

In the receiver file (sensor points):

```
#@rfluxmtx h=u u=Y
void glow ground_glow
0
0
4     1 1 1 0

ground_glow source ground
0
0
4     0 0 -1  180

#@rfluxmtx h=r1 u=Y
void glow sky_glow
0
0
4     1 1 1 0

sky_glow source sky
0
0
4     0 0 1  180
```

- Order Matters! - put the ground first, so that it is first in the matrix file.
- For the ground use a uniform hemisphere sampling basis.
- For the sky use a Reinhart (Tregenza) sampling basis

Creating a Rendered DC Matrix

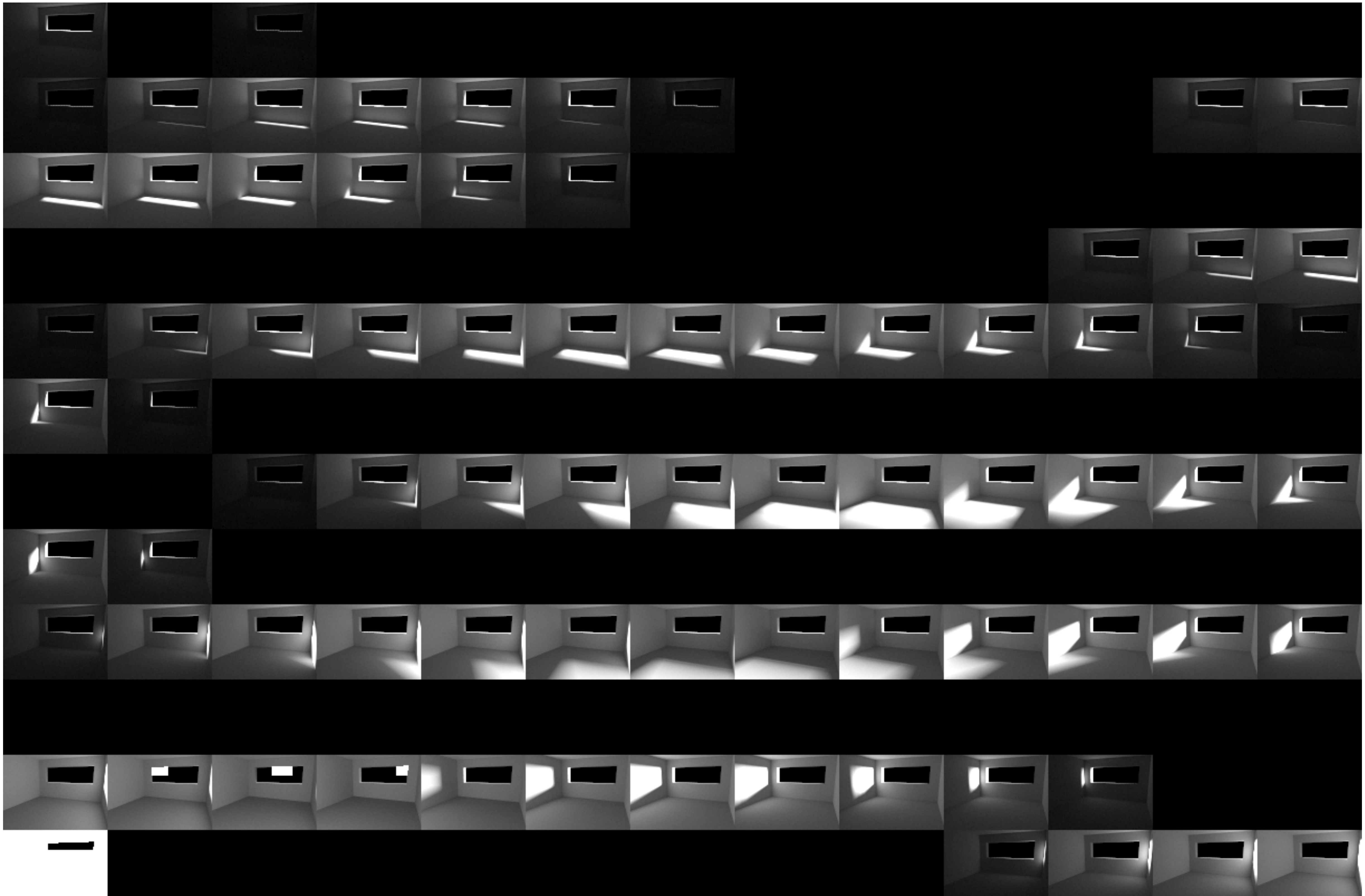
```
vwrays -ff -vf views/back.vf -x 600 -y 600  \
   | rfluxmtx `vwrays -vf views/back.vf -x 600 -y 600 -d` \
   -ffc -ab 12 -ad 50000 -lw 2e-5   \
   -  skies/sky.rad \
   materials/testroom.mat objects/testroom.rad
```

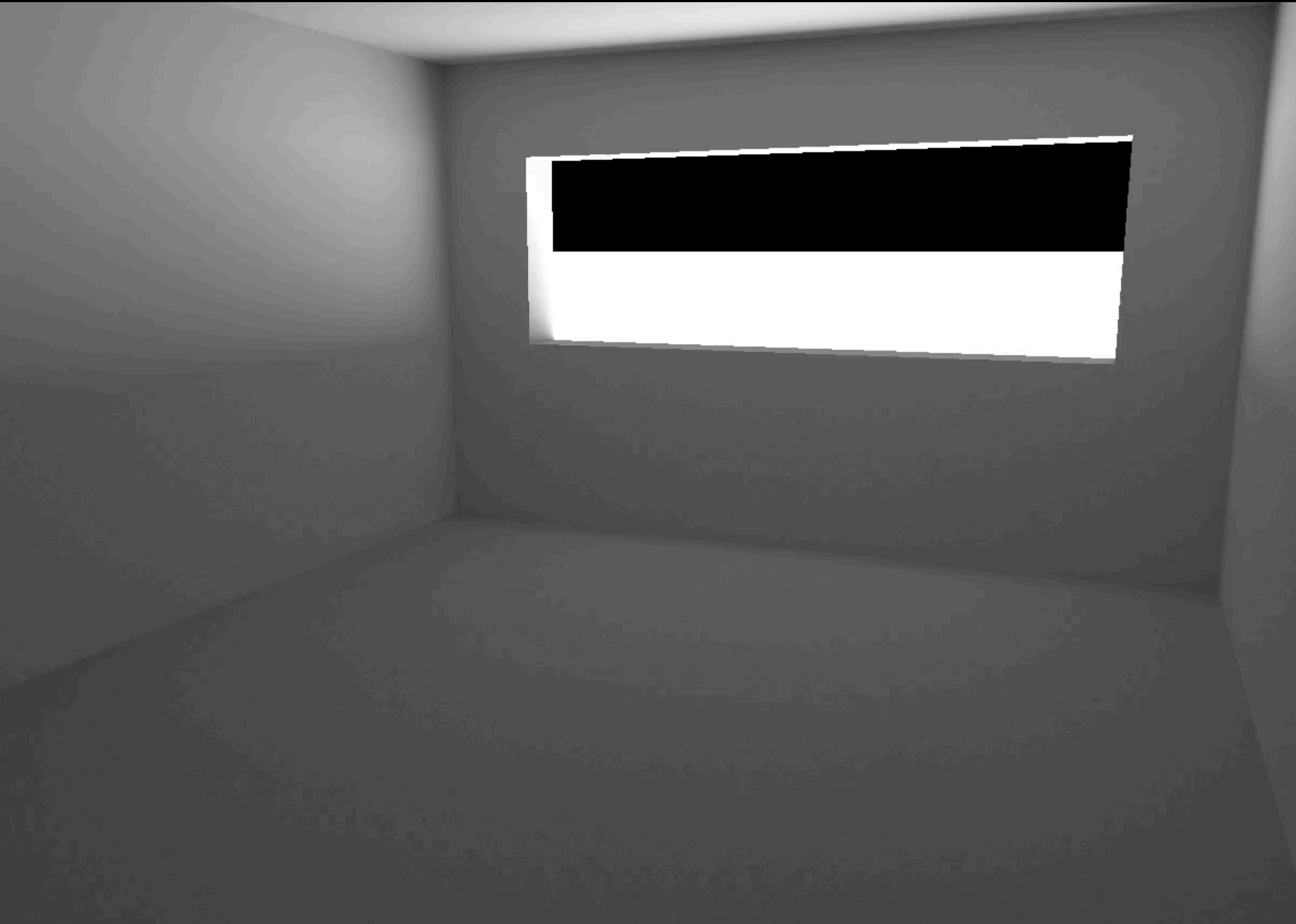
Sender Receiver Model

In the receiver file (renderings):

```
#@rfluxmtx h=u u=Y o=dcx/g%03d.hdr
void glow ground_glow
0
0
4      1 1 1 0

ground_glow source ground
0
0
4      0 0 -1  180

#@rfluxmtx h=r1 u=Y o=dcx/s%03d.hdr
void glow sky_glow
0
0
4      1 1 1 0

sky_glow source sky
0
0
4      0 0 1  180
```

Rendered DC matrix





Generating the sky vector/matrix

- gendaymtx - sky matrix for multiple time steps
  - **epw2wea** USA_Ca_Oakland.epw USA_CA_Oakland.wea
  - **gendaymtx** USA_CA_Oakland.wea > Oakland.smx

- genskyvec - sky vector for single time step
  - **genskyvec** -m 1 sky_12-21-15.rad > skyvec_12-21-15.skv
  - gensky 12 21 15 | **genskyvec** -m 1 > skyvec_12-21-15.skv

gendaymtx

epw2wea USA_Ca_Oakland.epw \
USA_CA_Oakland.wea

```
place Oakland Metropolitan Arpt_USA
latitude 37.72
longitude 122.22
time_zone 120
site_elevation 2.0
weather_data_file_units 1
1 1 0.500 0 0
1 1 1.500 0 0
1 1 2.500 0 0
1 1 3.500 0 0
1 1 4.500 0 0
1 1 5.500 0 0
1 1 6.500 0 0
1 1 7.500 71 1
1 1 8.500 463 41
1 1 9.500 573 81
1 1 10.500 712 95
1 1 11.500 655 129
1 1 12.500 684 121
1 1 13.500 717 110
…
```

gendaymtx USA_CA_Oakland.wea \
> skies/Oakland.smx

```
#?RADIANCE
gendaymtx USA_CA_Oakland.Intl.AP.724930_TMY3.wea
LATLONG= 37.72000000 -122.22000000
NROWS=146
NCOLS=8760
NCOMP=3
FORMAT=ascii

0 0 0
0 0 0
0 0 0
0 0 0
0 0 0
0 0 0
0 0 0
0.0678 0.0678 0.0678
4.31 4.31 4.31
9.92 9.92 9.92
15 15 15
17.2 17.2 17.2
17.7 17.7 17.7
16.6 16.6 16.6
…
```

Header

genskyvec

```
gendaylit 1 1 9.500 -W 573 81 -g .25 \
    | genskyvec -m 1 -c 1 1 1
```

9.93429571	9.93429571	9.93429571
26.8912634	26.8912634	26.8912634
25.0276164	25.0276164	25.0276164
24.1410850	24.1410850	24.1410850
23.7024527	23.7024527	23.7024527
23.9114161	23.9114161	23.9114161
25.3207076	25.3207076	25.3207076
28.4104614	28.4104614	28.4104614
34.8112246	34.8112246	34.8112246
45.5503249	45.5503249	45.5503249
65.3156192	65.3156192	65.3156192
100.950139	100.950139	100.950139
145.021263	145.021263	145.021263
1248.24006	1248.24006	1248.24006
120.810790	120.810790	120.810790
77.4222078	77.4222078	77.4222078
54.1575689	54.1575689	54.1575689
39.5426404	39.5426404	39.5426404
32.0955405	32.0955405	32.0955405
26.4445438	26.4445438	26.4445438
24.9248813	24.9248813	24.9248813
24.1096202	24.1096202	24.1096202
…		

No Header
(will likely change)

Putting it all together (Sensor Points)

```
rmtxop DC.dmx skies/Oakland.smx \
    | rmtxop -fa -c 47.4 119.9 11.6 -t - > E_dcx.txt
```

Transpose Matrix

Convert from Irradiance to Illuminance

Putting it all together (Sensor Points)

#?RADIANCE
rmtxop -fa -c 47.4 199.9 11.6 -t -
NROWS=8760
NCOLS=6
NCOMP=1
FORMAT=ascii

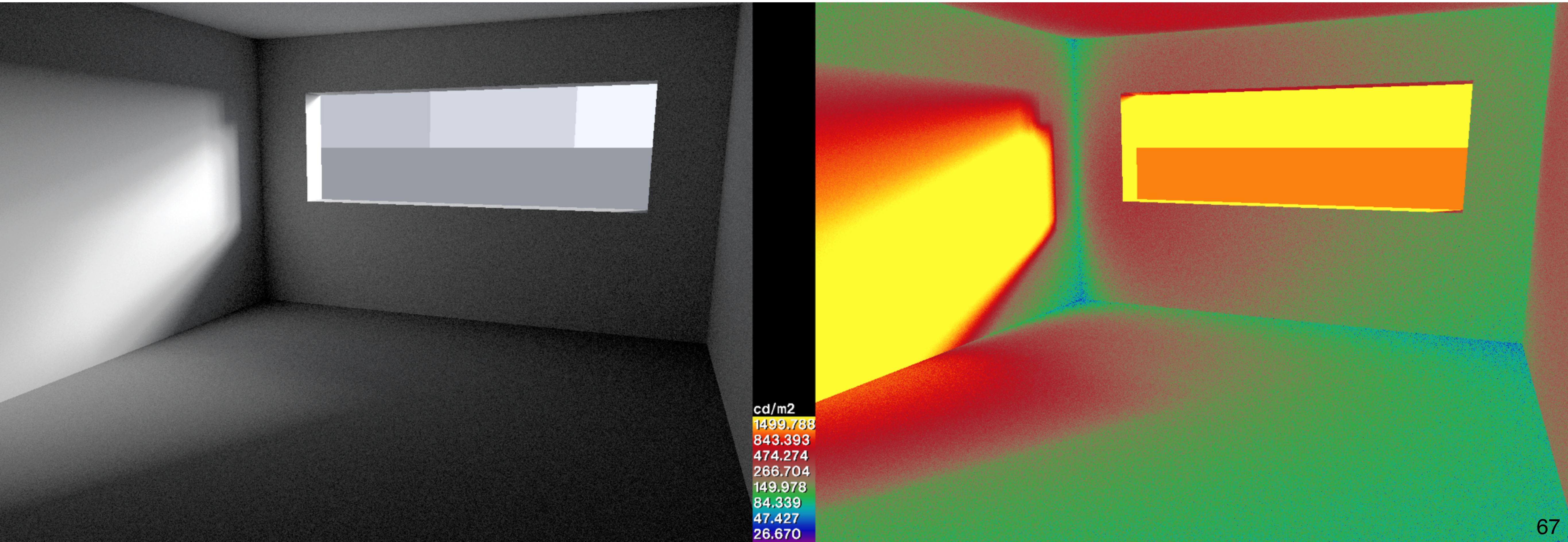
0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
8.2342E+02	1.7052E+02	9.0510E+01	5.6598E+01	4.6165E+01	3.2211E+01
1.7411E+04	5.2983E+03	2.0582E+03	1.2842E+03	9.3520E+02	6.8428E+02
3.9953E+04	2.3970E+04	7.7398E+03	2.6302E+03	1.8669E+03	1.5224E+03
6.0483E+04	4.4820E+04	2.1390E+04	6.8013E+03	2.8950E+03	2.0802E+03
7.0504E+04	5.1113E+04	2.3234E+04	6.7426E+03	4.0815E+03	2.6280E+03
6.8177E+04	5.1835E+04	2.7949E+04	8.9567E+03	5.2072E+03	2.8571E+03
6.5962E+04	4.9738E+04	2.5599E+04	9.4266E+03	4.2354E+03	2.2747E+03
5.2922E+04	3.4003E+04	1.3659E+04	4.6648E+03	2.4278E+03	1.6722E+03
3.0453E+04	1.2224E+04	3.9111E+03	2.0176E+03	1.2847E+03	9.4587E+02
6.0474E+03	1.1789E+03	7.1055E+02	4.9880E+02	2.8636E+02	2.4174E+02
0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00

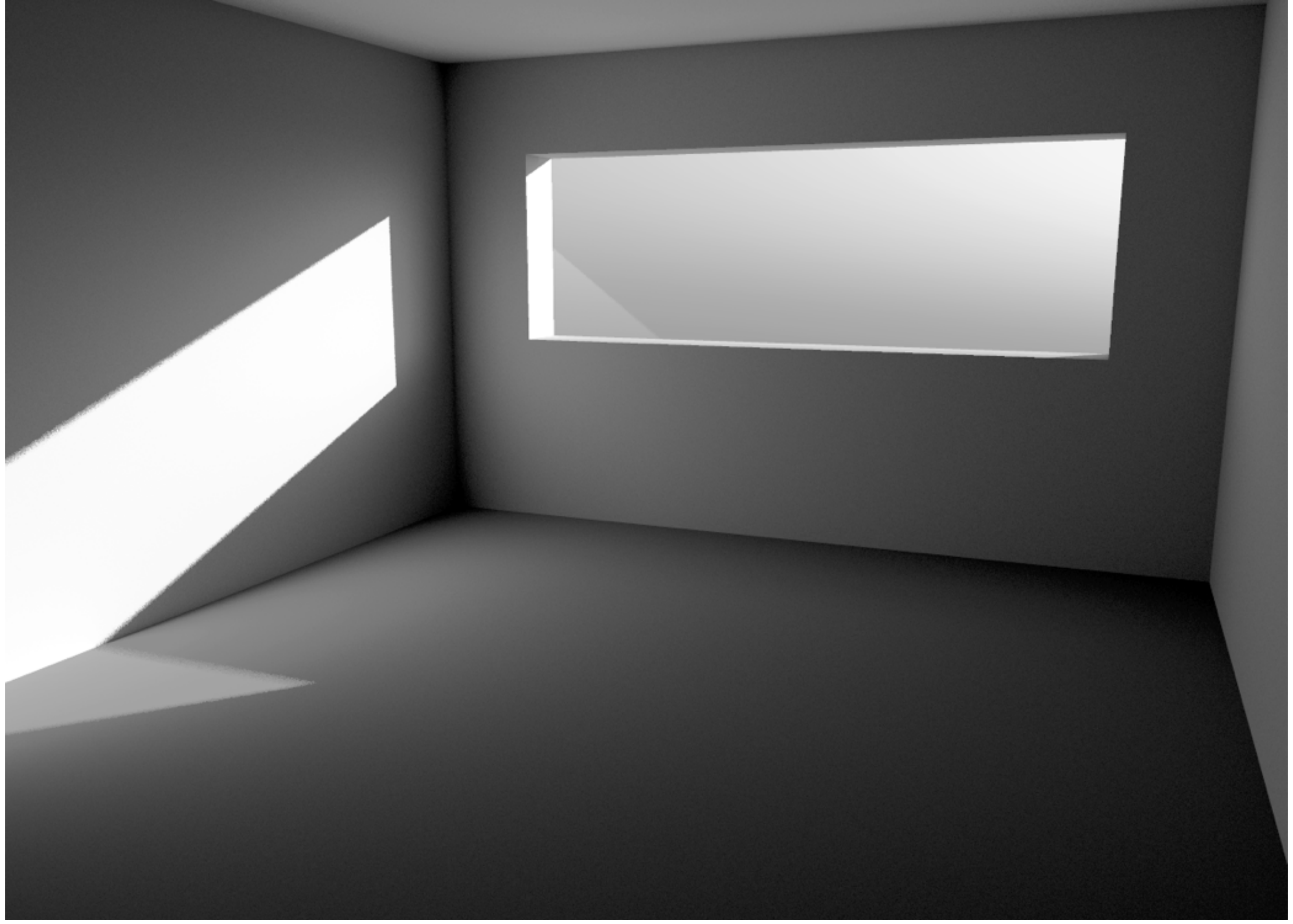
…

Putting it all together (Renderings)

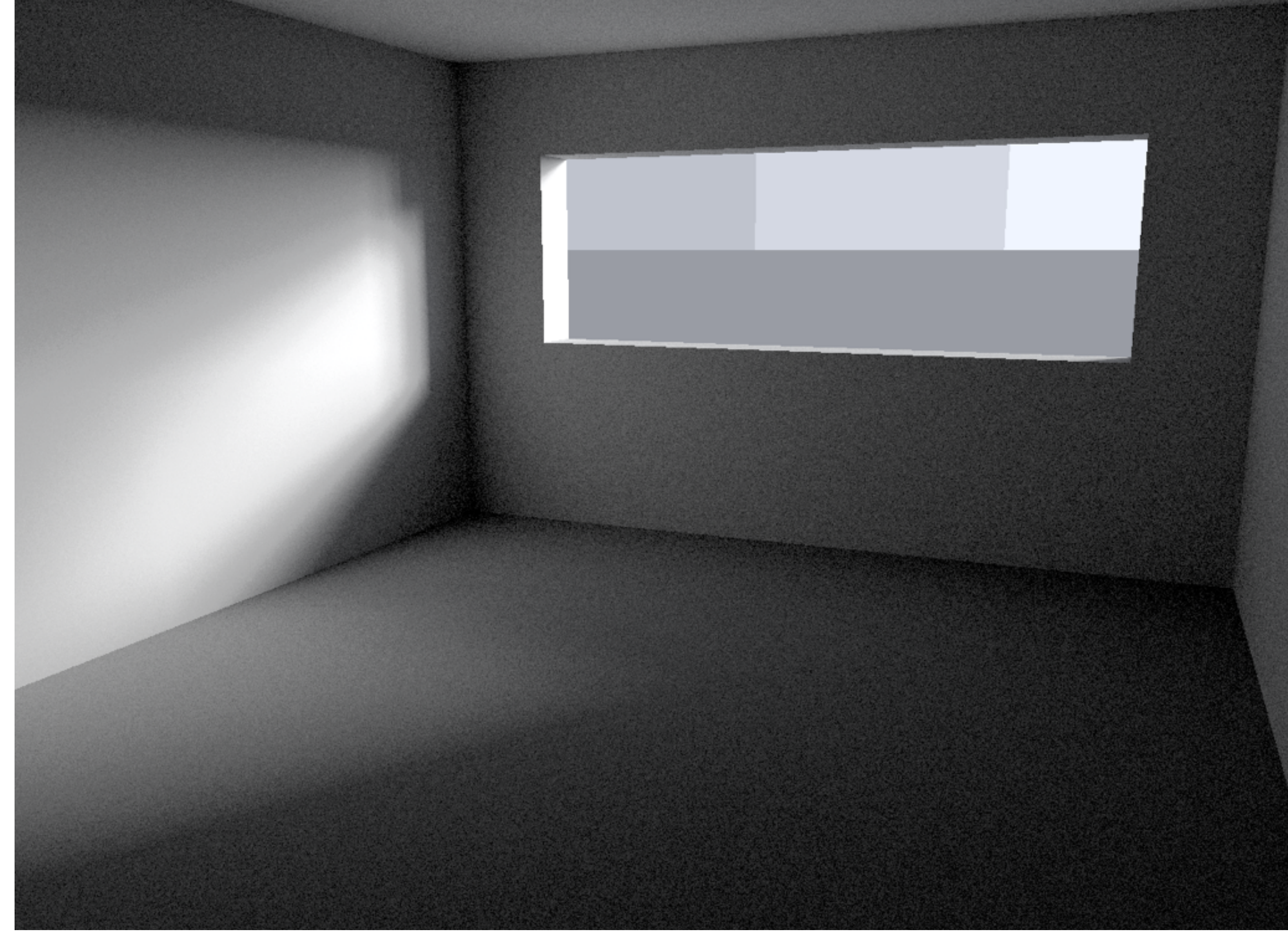
dctimestep dcx/g_%03d.hdr skies/12_21_15.skv \
> images/122115_clear_dcx.hdr

```
pcond images/122115_clear_dcx.hdr | \
    pcompos -a 2 - '!falsecolor -s 2000 -log 2 -i images/122115_clear_dcx.hdr’ | \
    ra_tiff -z - images/122115_clear_dcx.tif
```





Clear Glazing, ground truth simulation



Clear Glazing, daylight coefficient simulation

Clear Glazing

Clear Glazing

dctimestep

- Usage:
  - Daylight Coefficients
    - **dctimestep** DC.mtx sky.mtx (2 arguments)
    - cat sky.mtx | **dctimestep** DC.mtx (1 argument)
  - Three-phase
    - **dctimestep** V.mtx T.xml D.mtx sky.mtx (4 arguments)
    - cat sky.mtx | **dctimestep** V.mtx T.xml D.mtx (3 arguments)
- Can handle a rendered DC or View matrix
- When giving 3 or 4 arguments the second argument must be a BSDF xml file.
- dctimestep does not expect a header on sky vector/matrix file (will likely change).

rmtxopt

- Usage:
  - **rmtxopt** DC.mtx S.mtx > E.mtx
  - **rmtxopt** V.mtx T.xml D.mtx S.mtx > E.mtx
  - **rmtxopt** V.mtx T1.xml T2.xml D.mtx S.mtx > E.mtx
  - **rmtxopt** V.mtx T1.xml LP.mtx T2.xml D.mtx S.mtx > E.mtx
  - and more…
- Performs multiplication, addition, scaling and component operations.
- Any number of matrices can be passed to rmtxopt.
- Matrix format can be a coefficient matrix or BSDF XML file. (Can not handle image matrix)

Three-Phase Method

The Equations

Daylight Coefficient Equation:

$$E_{DC} = C_{dc}S$$

$$C_{dc} \approx VTD$$

Three-Phase Equation:

$$E_{3ph} = VTDS$$

Five-Phase Equation:

$$E_{5ph} = VTDS - V_dTD_dS_d + C_{ds}S_{sun}$$

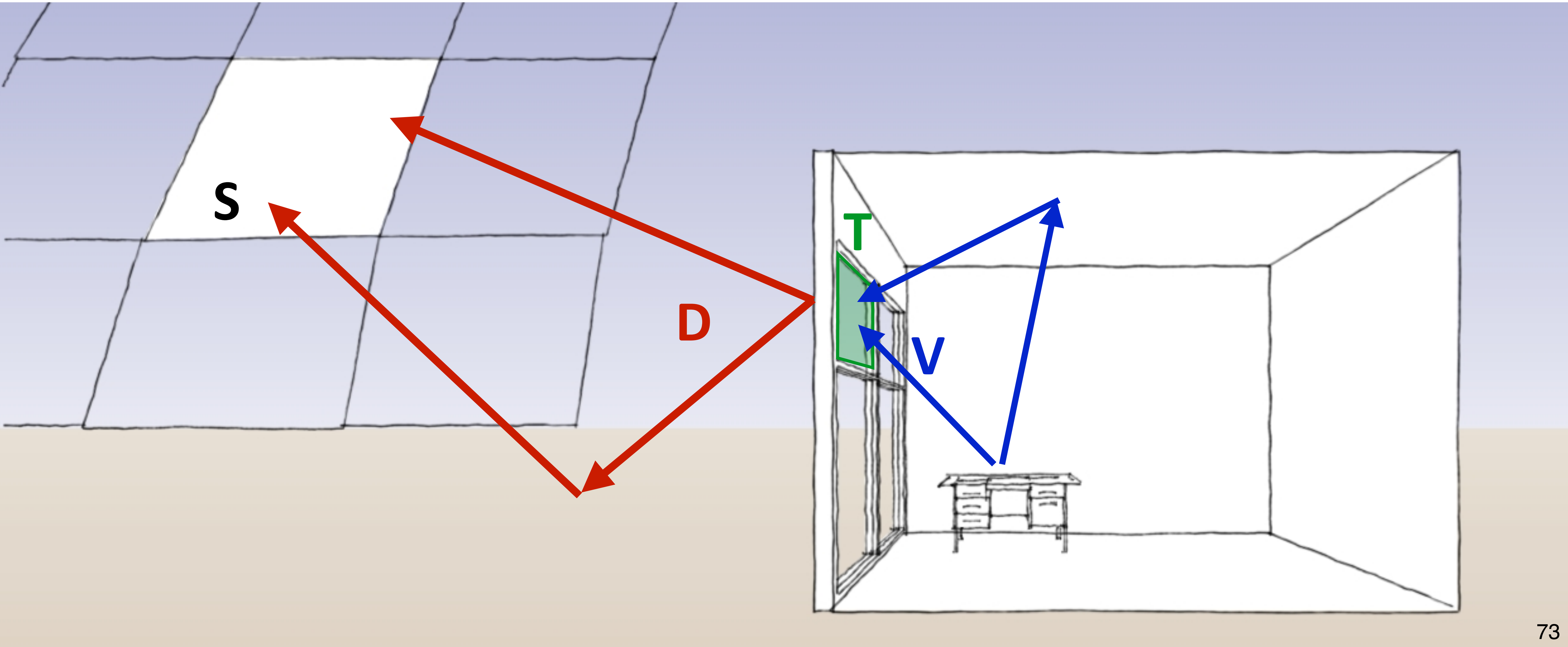
Three-Phase Method

V = View Matrix (interior)

T = Transmission Matrix (BSDF)

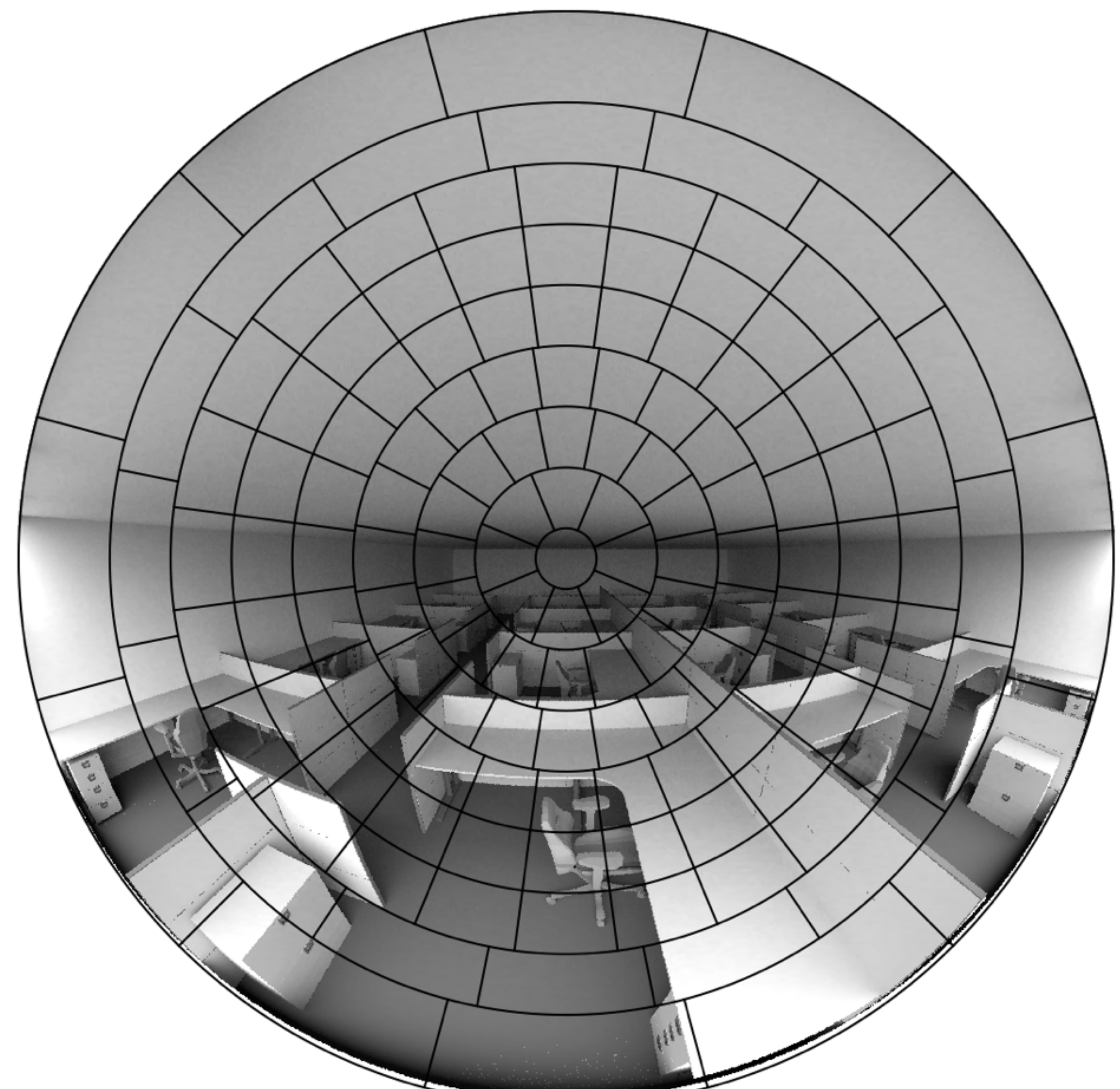
D = Daylight Matrix (exterior)

S = Sky Matrix



View matrix (V)

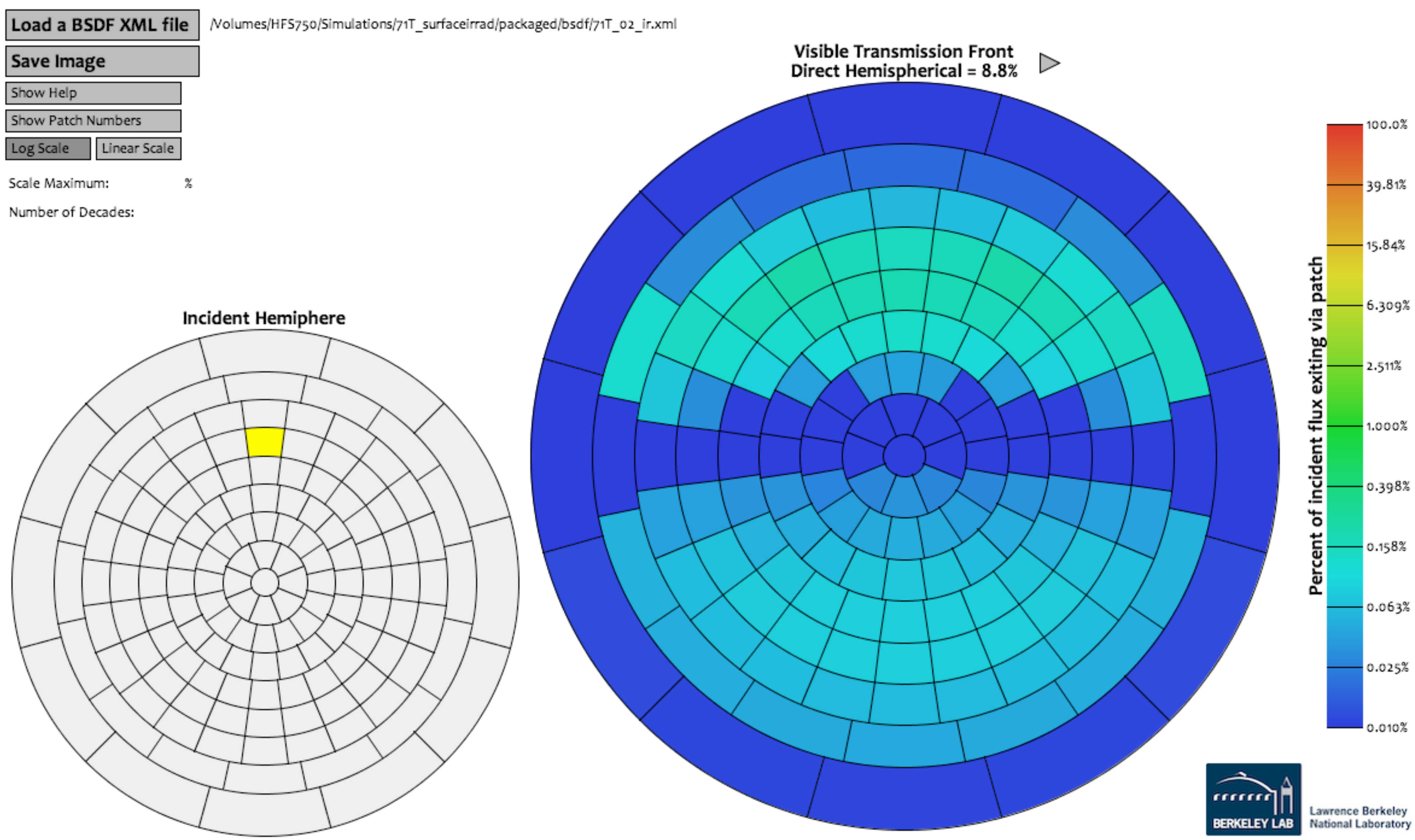
The view matrix contains coefficients relating energy leaving a window in klems direction bins energy incident at a sensor point or image pixel.



E_{3ph} = VTDS

Transmission matrix (T) / BSDF

The transmission matrix contains coefficients relating energy incident on a window and energy leaving a window in Klems directional bins.

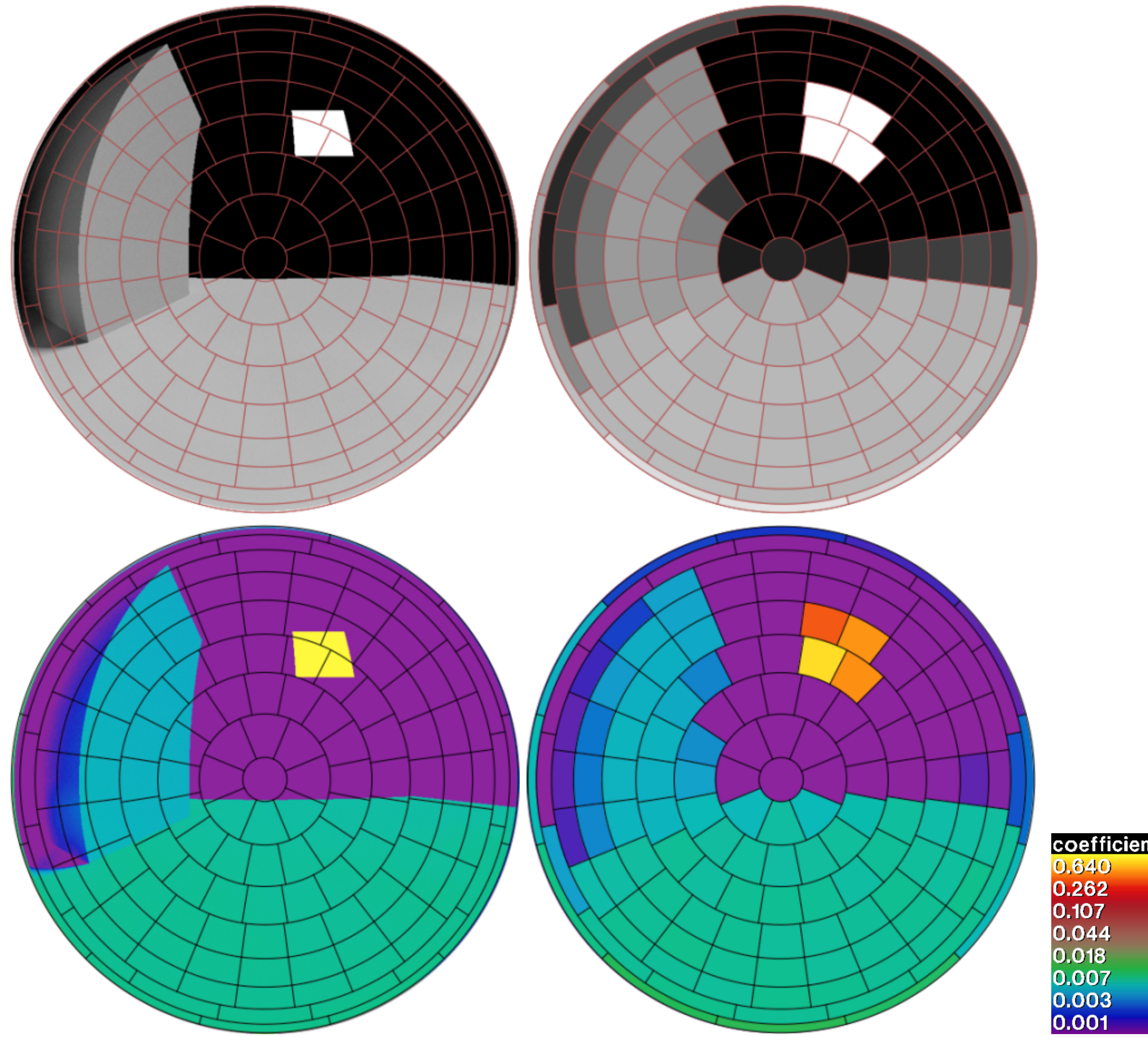


BERKELEY LAB Lawrence Berkeley National Laboratory

Daylight matrix (D)

The daylight matrix contains coefficients relating energy leaving sky patches with energy incident on a window in a klems directional bin.

An building with nearby obstruction



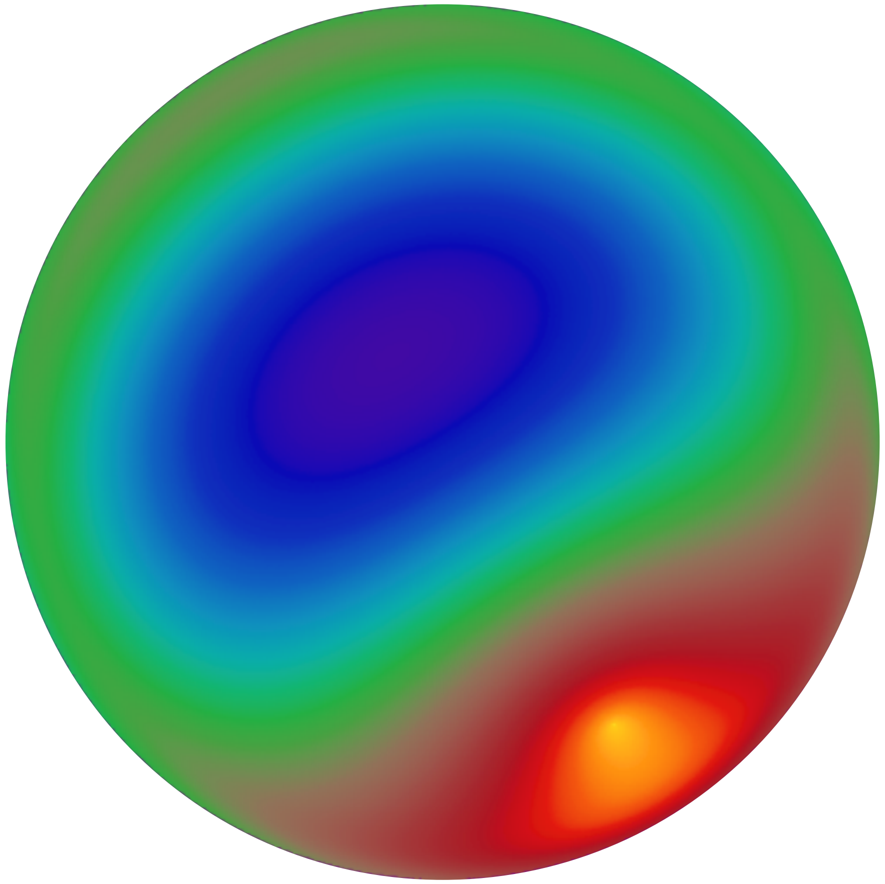
View through Klems screen

Depiction of Daylight matrix values for one sky patch.

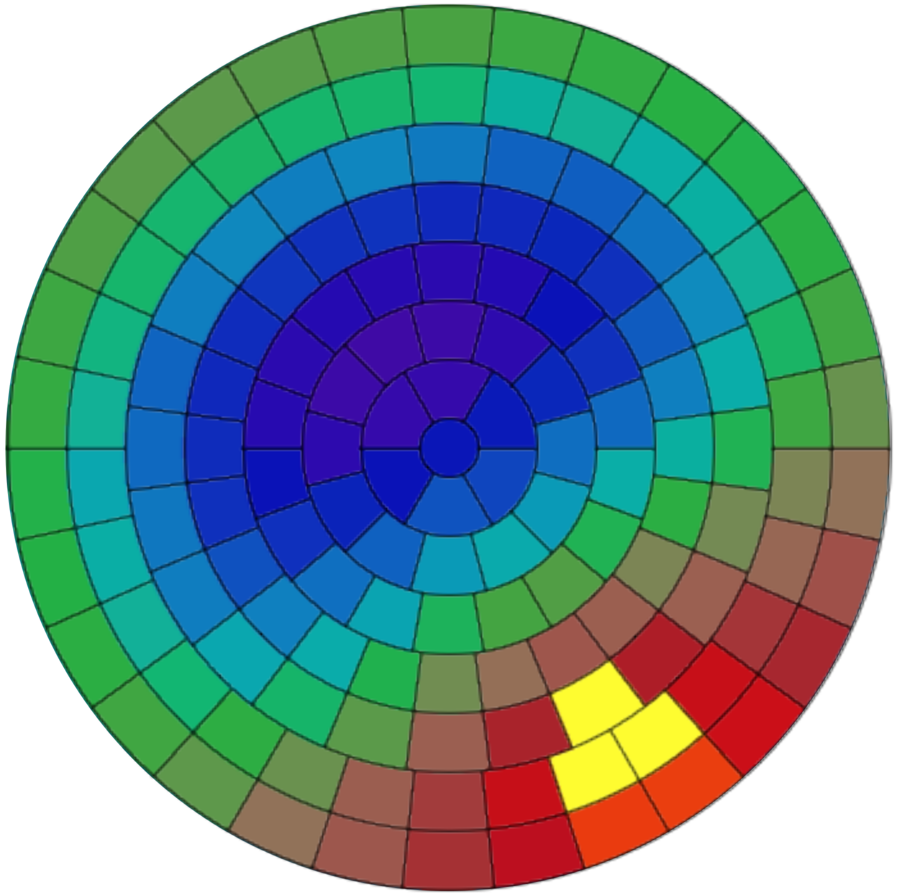
Sky matrix (S)

A sky vector contains average sky luminance in a discretized patch. A sky matrix is a series of sky vectors encompassing many time steps.

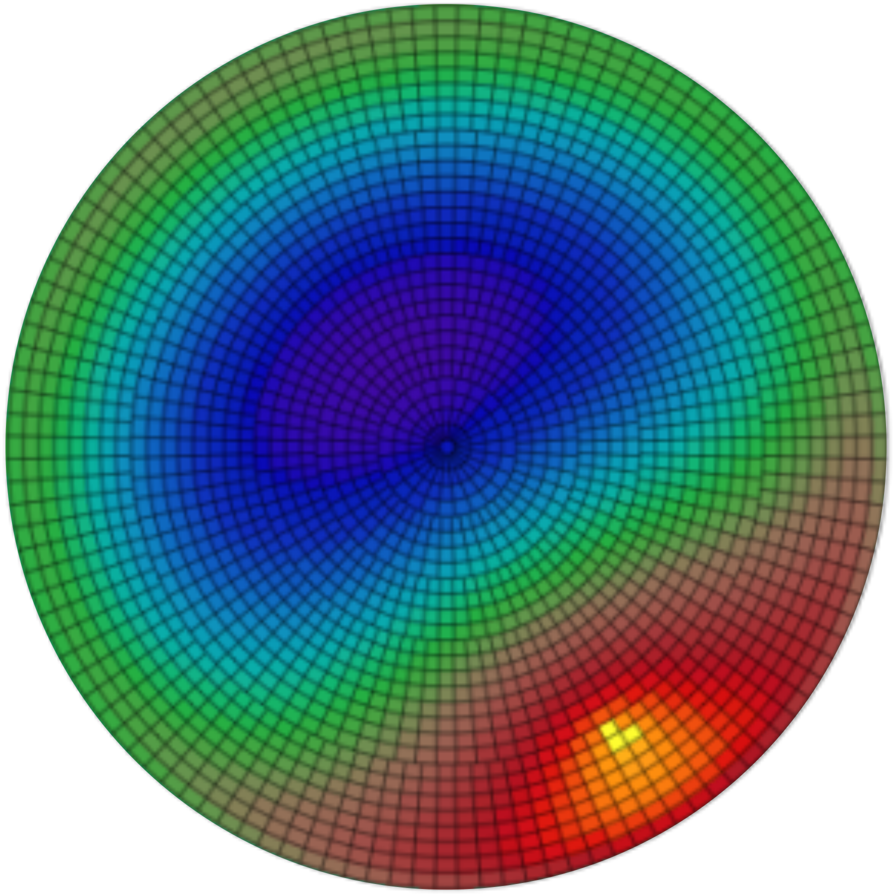
gendaymtx was recently added to Radiance to create a sky matrix from a *.wea weather data file.



Sky Luminance Gradient



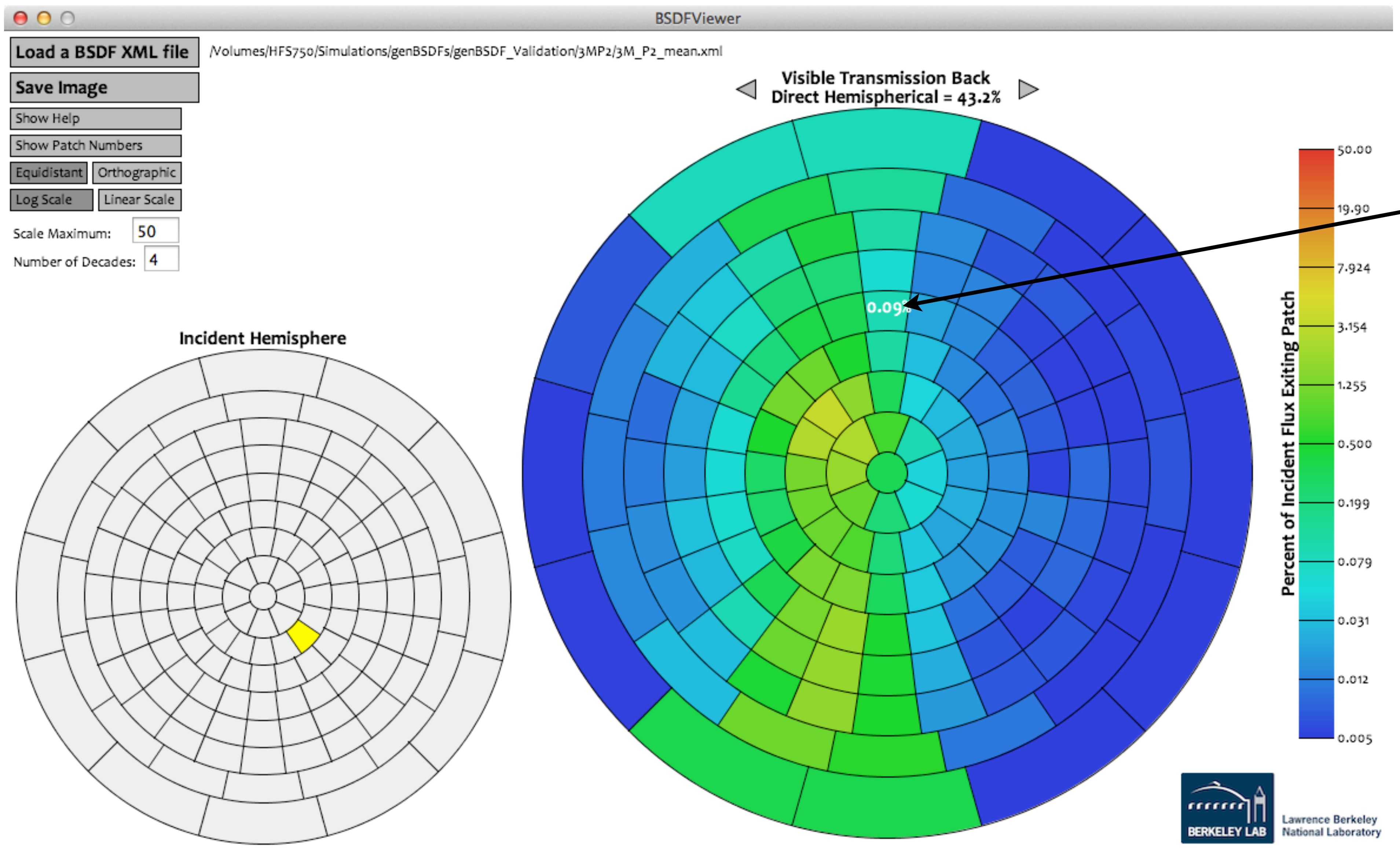
Discretized Sky Luminance (Tregenza)



Discretized Sky Luminance (Reinhart MF:4)

Why can't we do a daylight coefficient simulation with CFS?

- Sky is a glow material - stochastically sampled
- BSDF rays are emitted based on importance
- No deterministic rays for sun



- ~ 14 Reinhart patches to 1 Klems patch
- 0.09% / 43.2% from BSDF
- 6,720 ambient rays ~ 1 sample/ Reinhart patch.
- 0.09% of 100,000 lux = 90 lux

Generating a rendered view matrix:

Old:

```
$ vwrays -ff -vf views/back.vf -x 600 -y 600 | \
    rcontrib -n 8 `vwrays -vf views/back.vf -x 600 -y 600 -d` \
        -ffc -ab 12 -ad 50000 -lw 2e-5 \
        -o images/vmx/window_%03d.hdr  -f klems_int.cal \
        -b kbinS -bn Nkbins -m windowglow  model.oct
```

New:

```
$ vwrays -ff -vf views/back.vf -x 600 -y 600 | \
    rfluxmtx `vwrays -vf views/back.vf -x 600 -y 600 -d` \
    -ffc -ab 12 -ad 50000 -lw 2e-5   \
    -  objects/window.rad \
    materials/testroom.mat objects/testroom.rad
```

Generating a rendered view matrix:

Old:

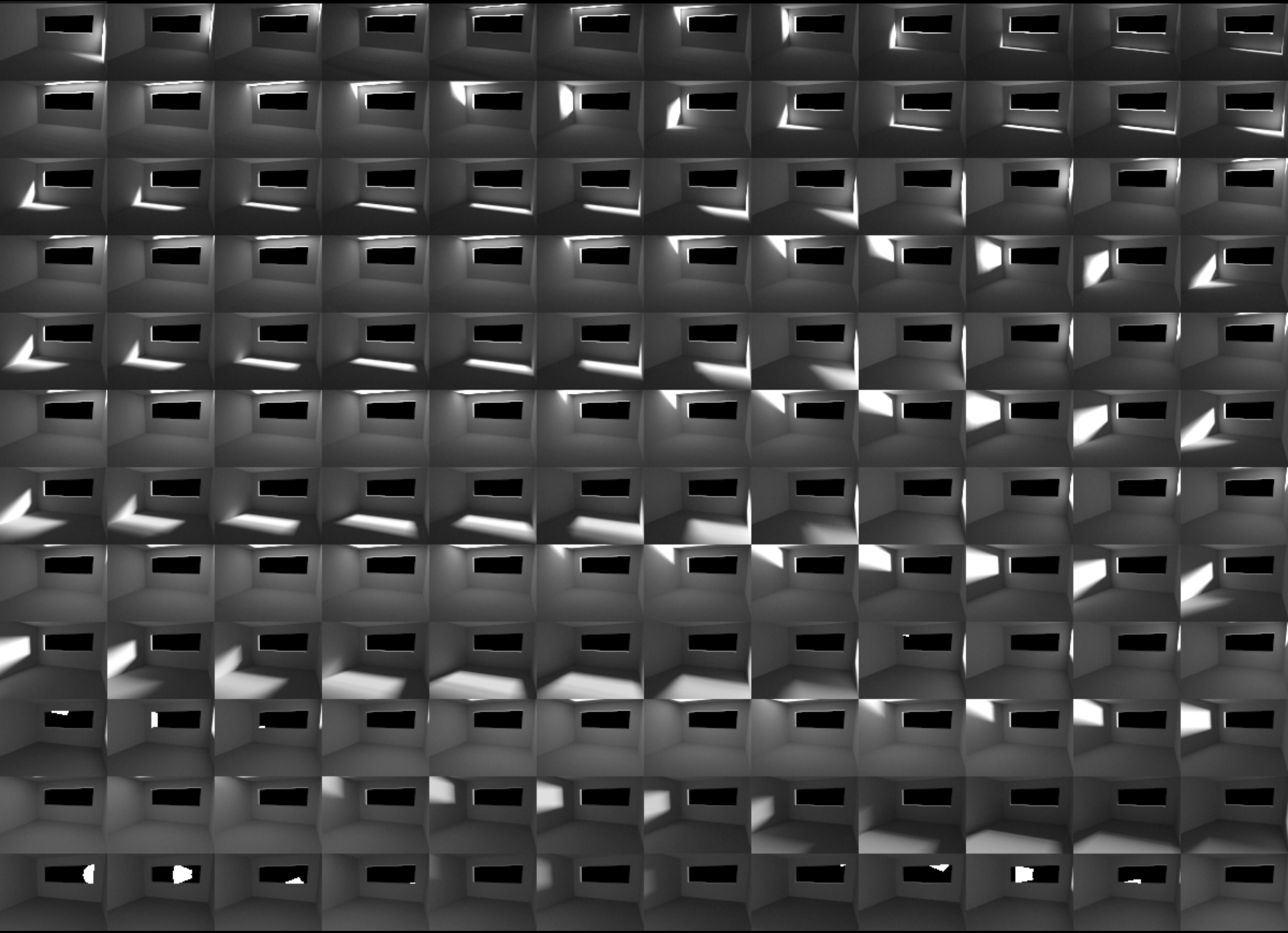
```
$ vwrays -ff -vf views/back.vf -x 600 -y 600 | \
    rcontrib -n 8 `vwrays -vf views/back.vf -x 600 -y 600 -d` \
        -ffc -ab 12 -ad 50000 -lw 2e-5 \
        -o images/vmx/window_%03d.hdr  -f klems_int.cal \
        -b kbinS -bn Nkbins -m windowglow  model.oct
```

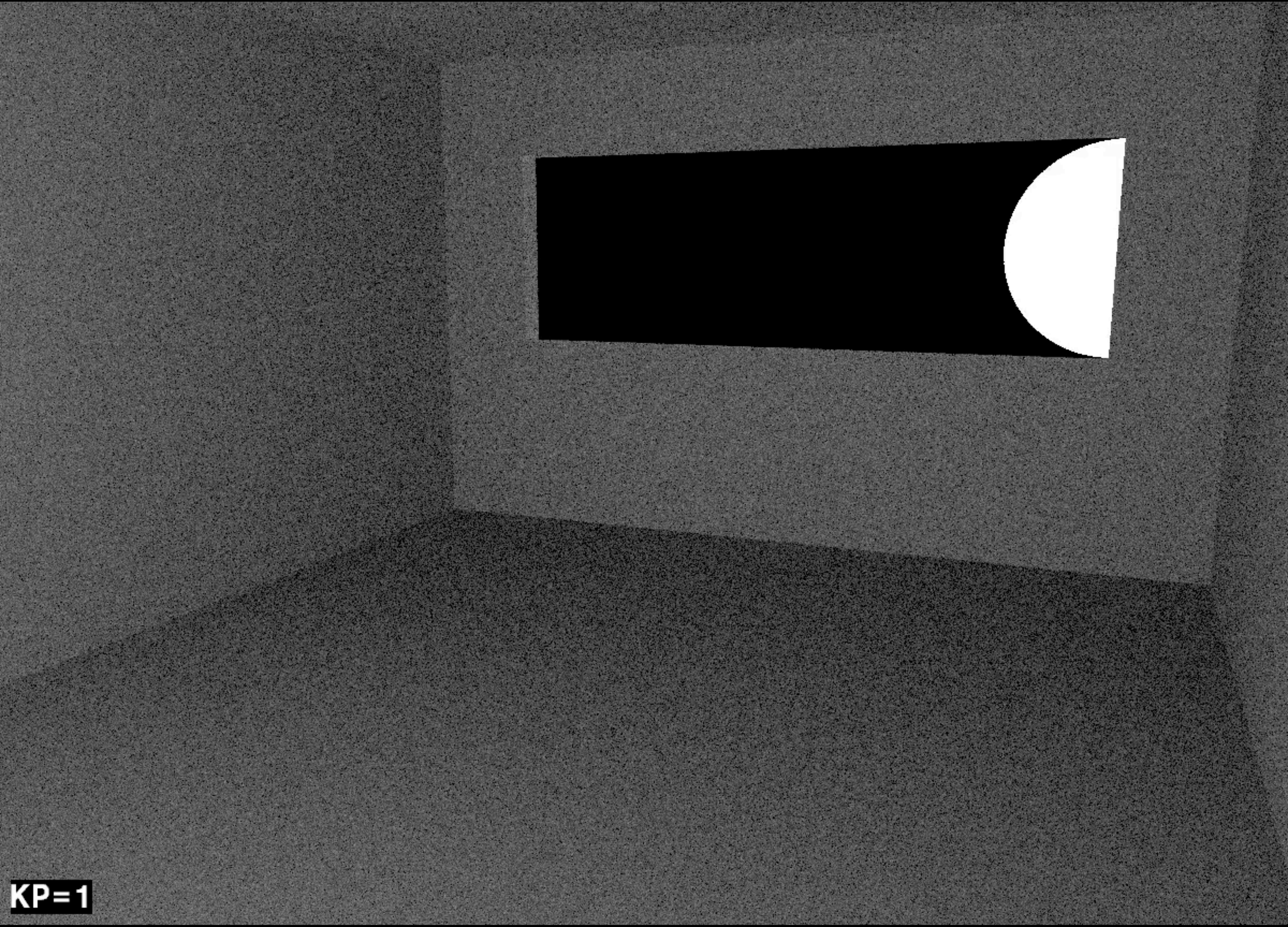
New:

```
$ vwrays -ff -vf views/back.vf -x 600 -y 600 | \
    rfluxmtx -n 8 `vwrays -vf views/back.vf -x 600 -y 600 -d` \
        -ffc -ab 12 -ad 50000 -lw 2e-5   \
        -  objects/window.rad \
        materials/testroom.mat objects/testroom.rad
```

View Matrix: In the window.rad file

```
#@rfluxmtx h=kf u=Z o=images/vmx/window_%03d.hdr

void glow windowglow
0
0
4   1    1    1  0

windowglow polygon window
0
0
12    0.5    -.15    1
      0.5    -.15    2
      3.5    -.15    2
      3.5    -.15    1
```




KP=1

Recommendations for View matrix

- Use glow material for window geometry - improves sampling of large / nearby windows.
- Increase -ad substantially
- Set -lw to ad^{-1}

parameter	default setting	recommended*
ab	1	**9**
ad	350	**16,384**
lw	2E-03	**6.10e-5 (ad**

* recommendations based on BRE validation space

Generating a daylight matrix using rfluxmtx:

Old:

```
genklemsamp -vd 0 -1 0 objects/window.rad | \
    rcontrib -e MF:4 -f reinhart.cal -b rbin -bn Nrbins \
        -m sky_glow -faf model_dmx.oct > south.dmx
```

New:

```
 rfluxmtx objects/window.rad skies/sky.rad \
         materials/testroom.mat objects/testroom.rad \
         > south.dmx
```

Generating a daylight matrix using rfluxmtx:

Old:

```
genklemsamp -vd 0 -1 0 objects/window.rad | \
    rcontrib -e MF:4 -f reinhart.cal -b rbin -bn Nrbins \
        -m sky_glow -faf model_dmx.oct > south.dmx
```

New:

```
 rfluxmtx objects/window.rad skies/sky.rad
        materials/testroom.mat objects/testroom.rad
        > south.dmx
```

In the sender file:

```
#@rfluxmtx h=kf u=Z o=images/vmx/window_%03d.hdr

void glow windowglow
0
0
4   1   1   1   0

windowglow polygon window
0
0
12   0.5   -.15   1
      0.5   -.15   2
      3.5   -.15   2
      3.5   -.15   1
```

- This is the same as the view matrix receiver file!
- Output specification is ignored in a sender file
  (Rejoice! You don't have to have a separate sender and receiver file for the same geometry)

In the receiver file:

```
#@rfluxmtx h=u u=Y
void glow ground_glow
0
0
4    1 1 1 0

ground_glow source ground
0
0
4    0 0 -1  180

#@rfluxmtx h=r1 u=Y
void glow sky_glow
0
0
4    1 1 1 0

sky_glow source sky
0
0
4    0 0 1  180
```

- Order Matters! - put the ground first, so that it is first in the daylight matrix file.
- For the ground use a uniform hemisphere sampling basis.
- For the sky use a Reinhart (Tregenza) sampling basis

Putting it all together

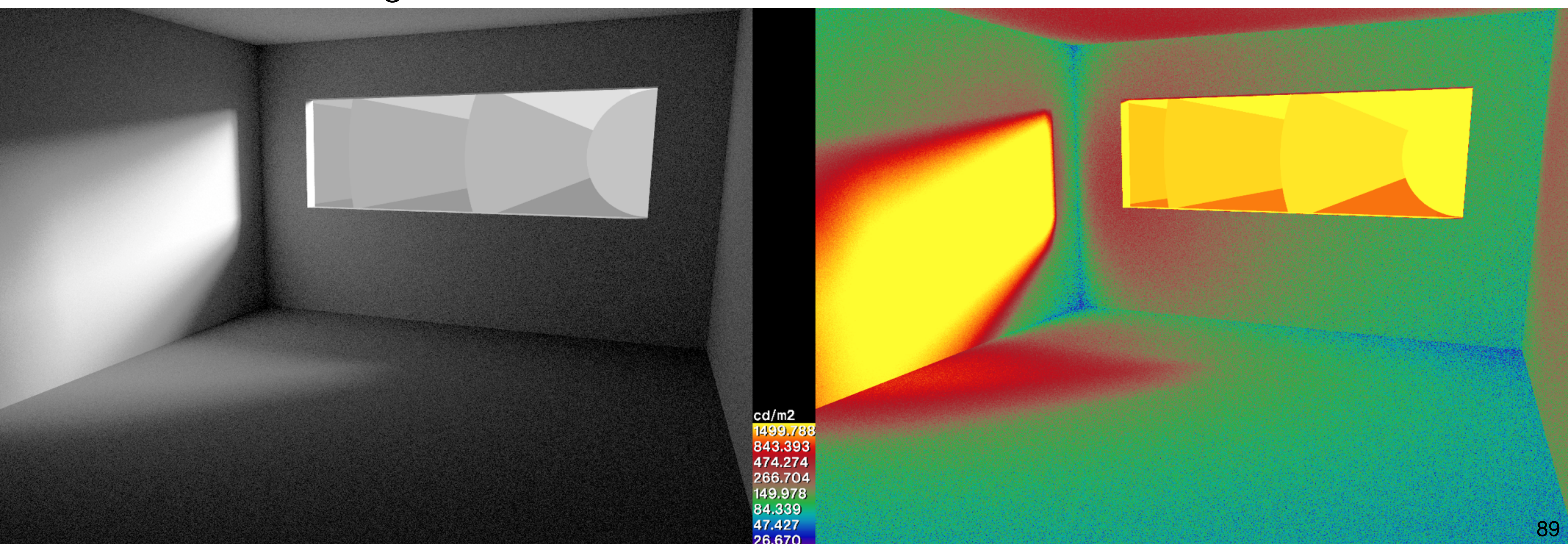
V

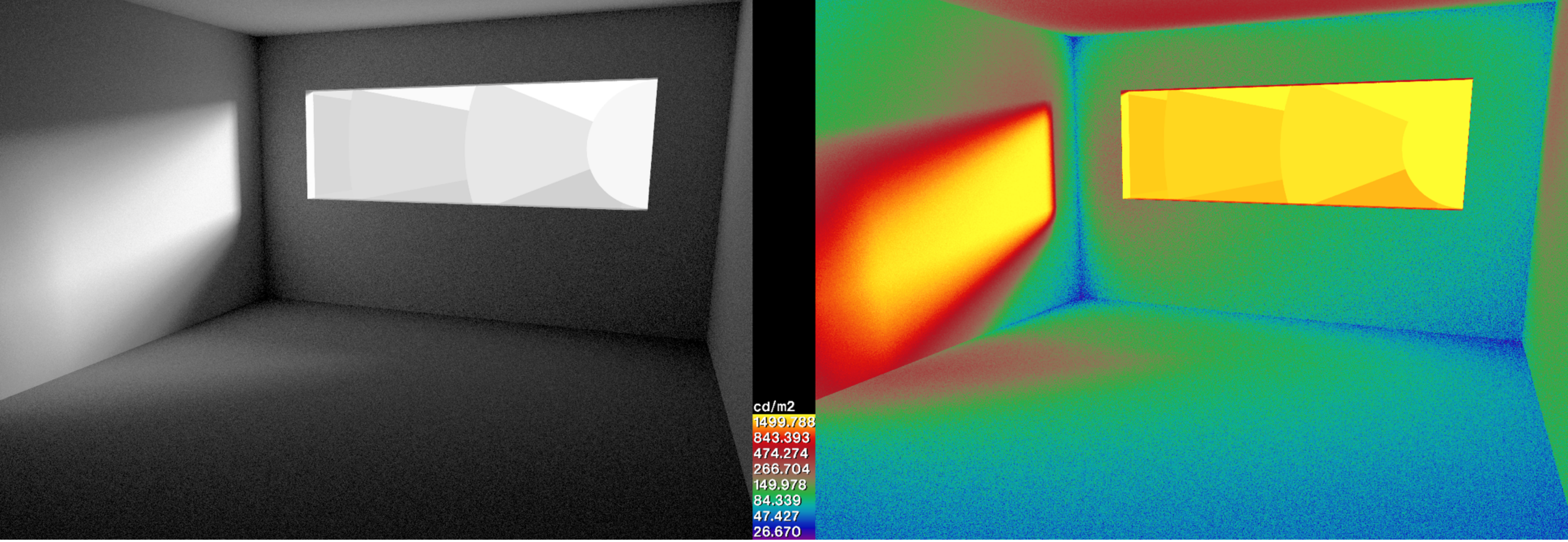
dctimestep images/vmx/window_%03d.hdr \

T data/singleclear.xml results/south.dmx \ D

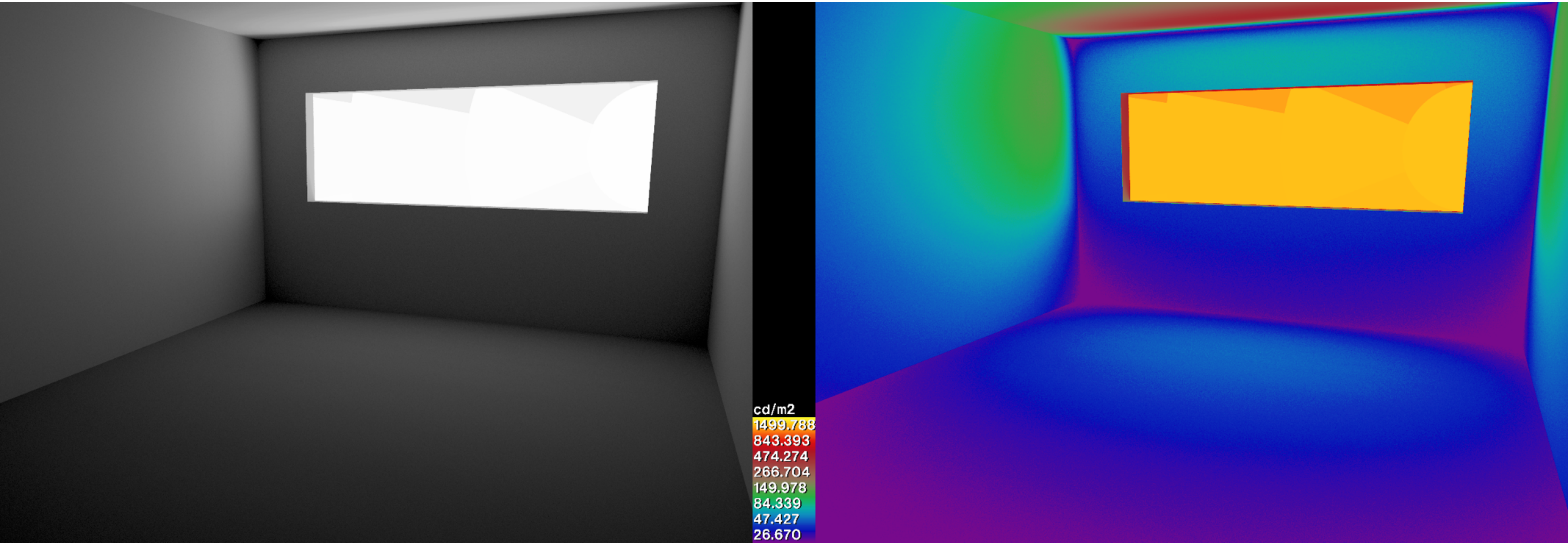
s skies/12_21_15.skv > images/122115_clear.hdr

```
pcond images/122115_clear.hdr | \
    pcompos -a 2 - '!falsecolor -s 2000 -log 2 -i images/122115_clear.hdr' | \
    ra_tiff -z - images/122115_clear.tif
```

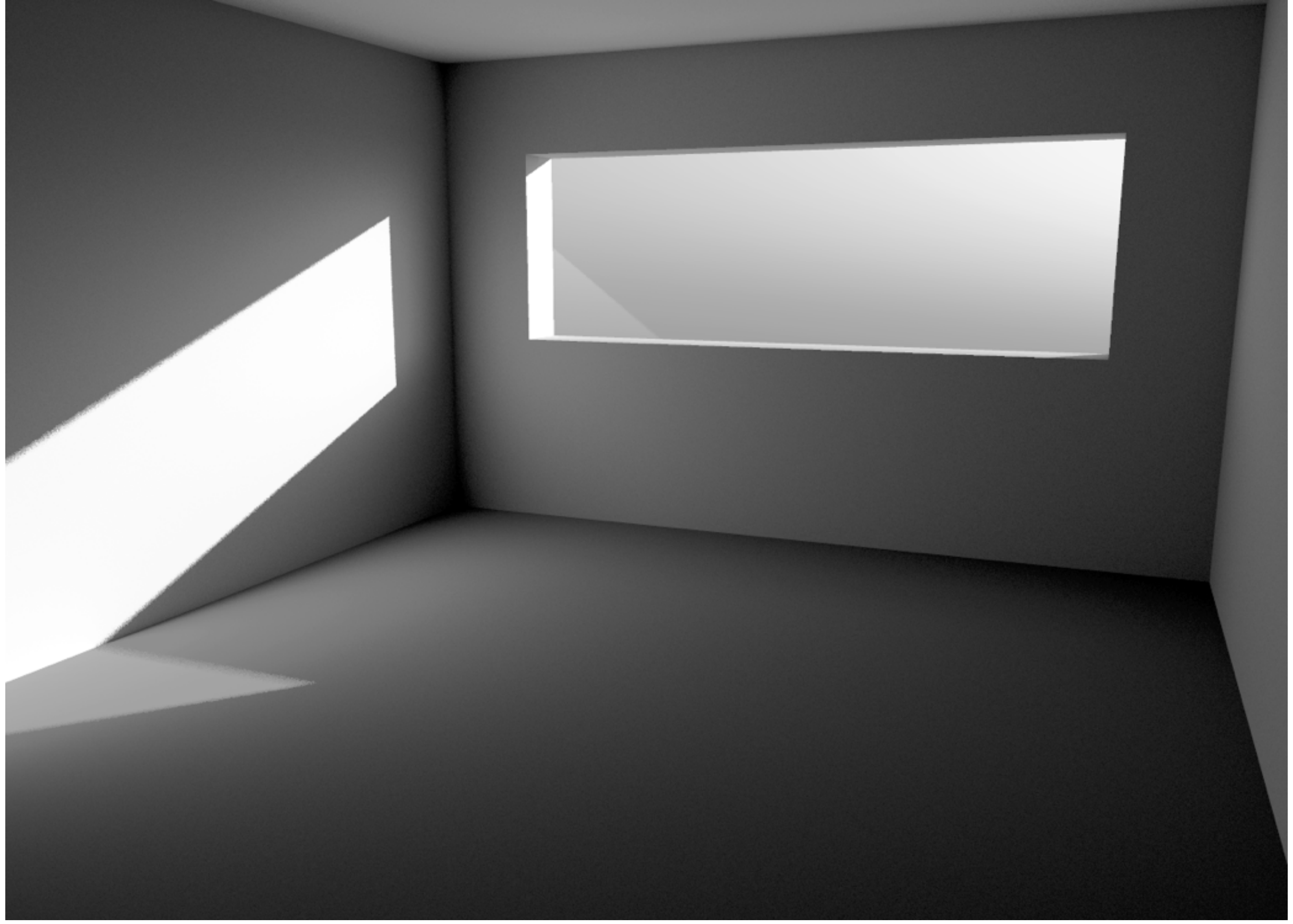




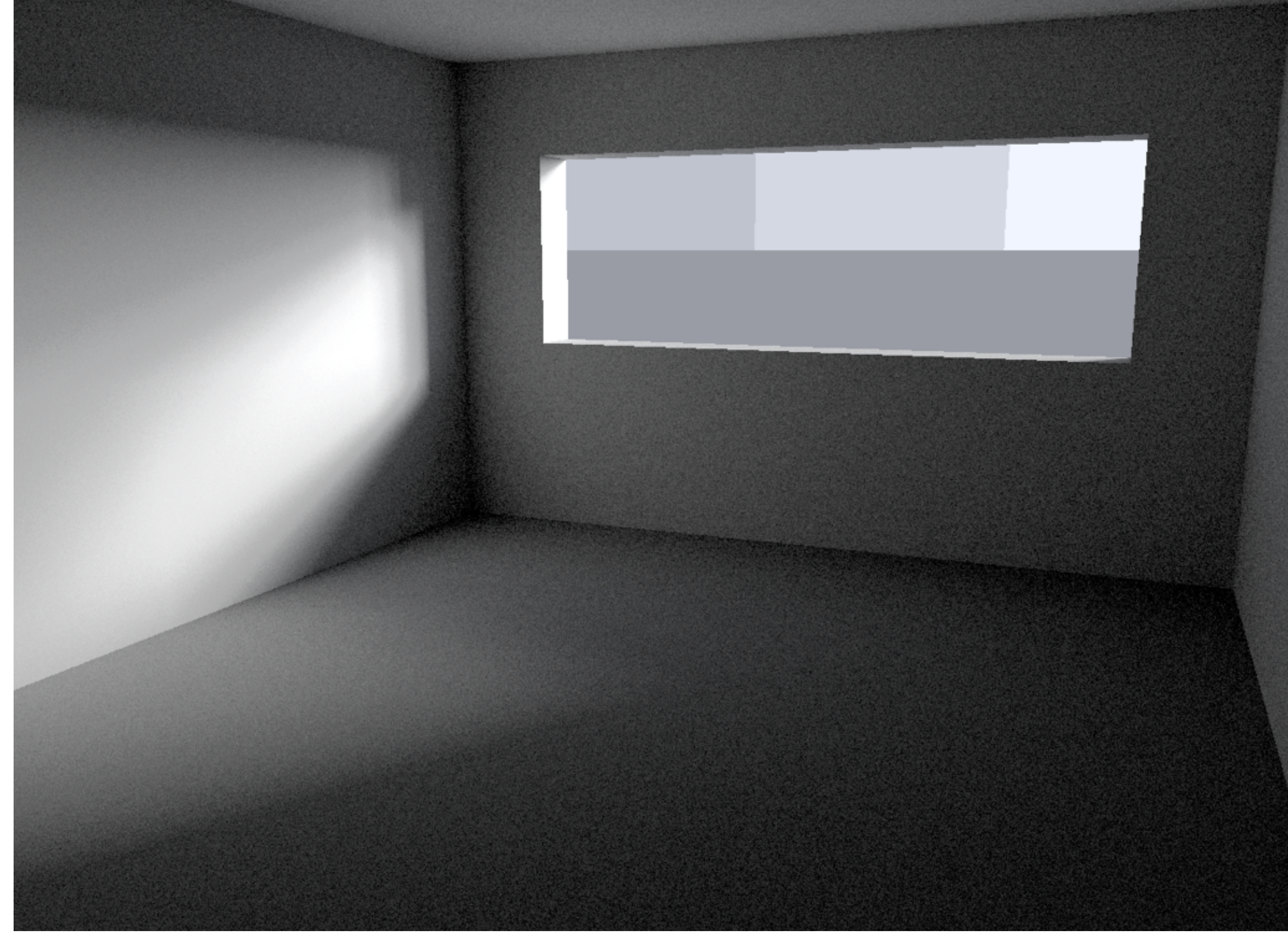
Venetian Blind 0° tilt, three-phase simulation



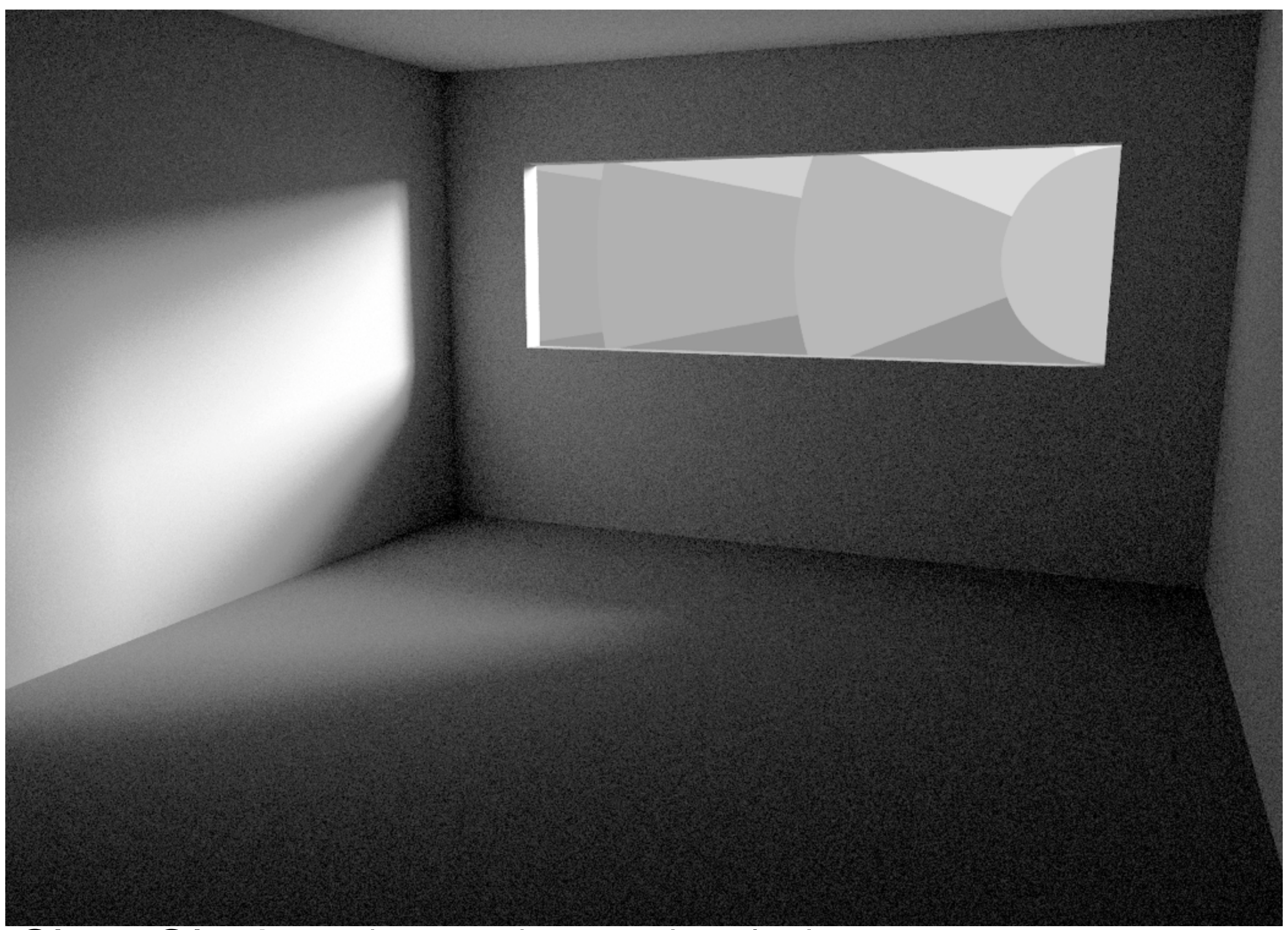
Venetian Blind 45° tilt, three-phase simulation



Clear Glazing, ground truth simulation

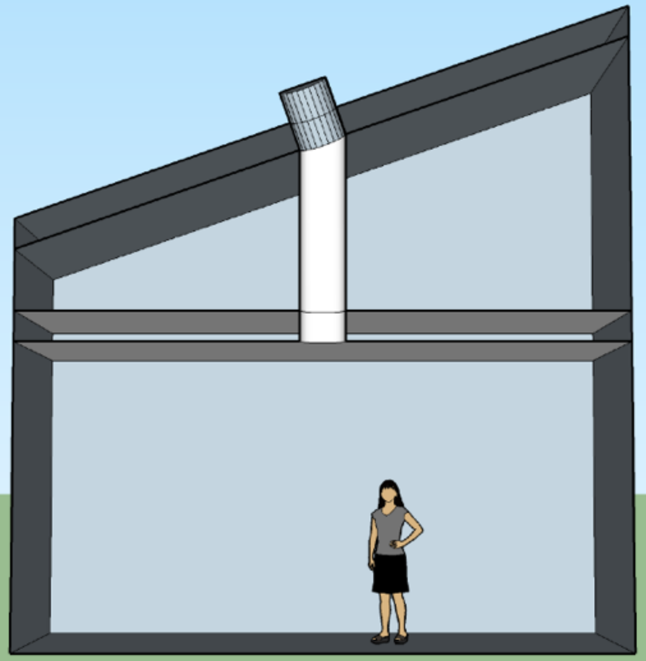


Clear Glazing, daylight coefficient simulation



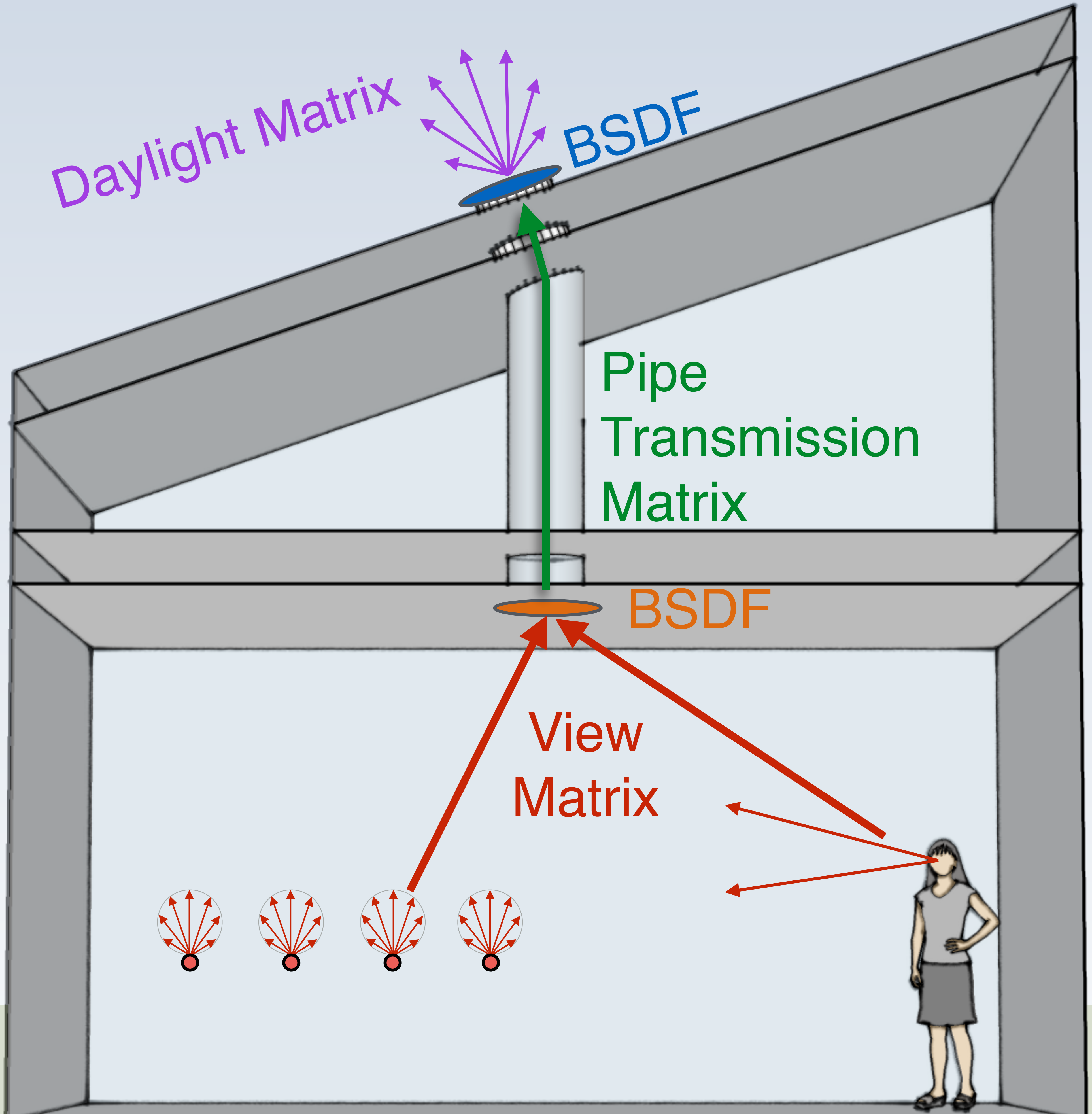
Clear Glazing, three-phase simulation

Clear Glazing



Light Pipe Example

New capability enabled by rfluxmtx & rmtxop


Daylight Matrix
BSDF
Pipe
Transmission
Matrix
BSDF
View
Matrix

View Matrix (sensor points)

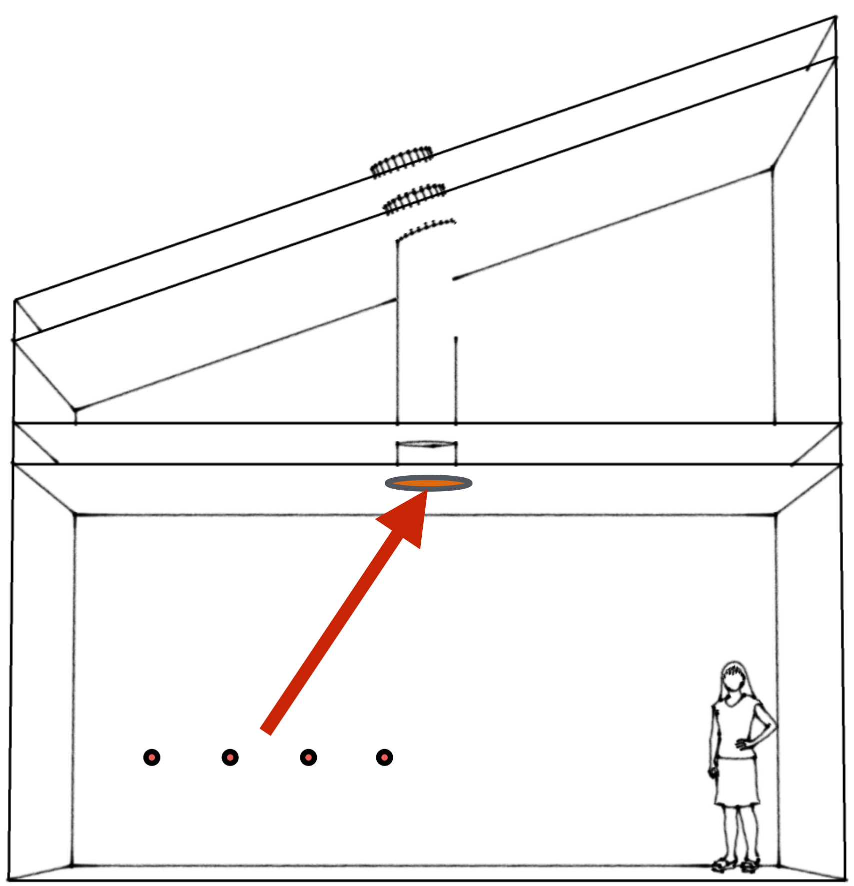
of Points

```
rfluxmtx -n 8 -faa -l+ -ab 12 -ad 50000 -lw 2e-5 -y 6 \
    <points.txt - objects/LP_bottom.rad \
    materials_light.rad model.rad > results/points.vmx
```

Sender

Model

Receiver



```
#@rfluxmtx h=kf u=+Y

LP_bottom polygon f_119_0
0
0
72
    3.107166  2.827189  3.048000
    3.048000  2.819400  3.048000
    2.988834  2.827189  3.048000 ...
```

View Matrix (rendering)

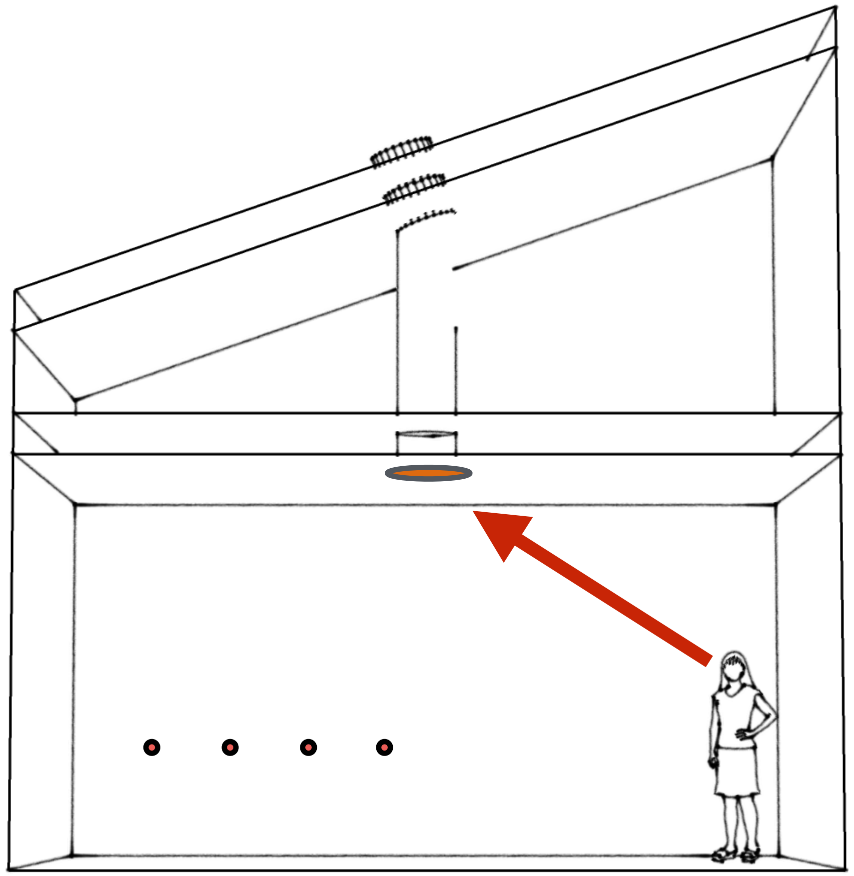
```
vwrays -fa -vf views/v2.vf -x 600 -y 600 | \
    rfluxmtx -n 8 `vwrays -vf views/v2.vf -x 600 -y 600 -d` \
    -fac -ab 8 -ad 5000 -lw 2e-4  - objects/LP_bottom.rad \
    materials_light.rad model.rad
```

Model

Sender

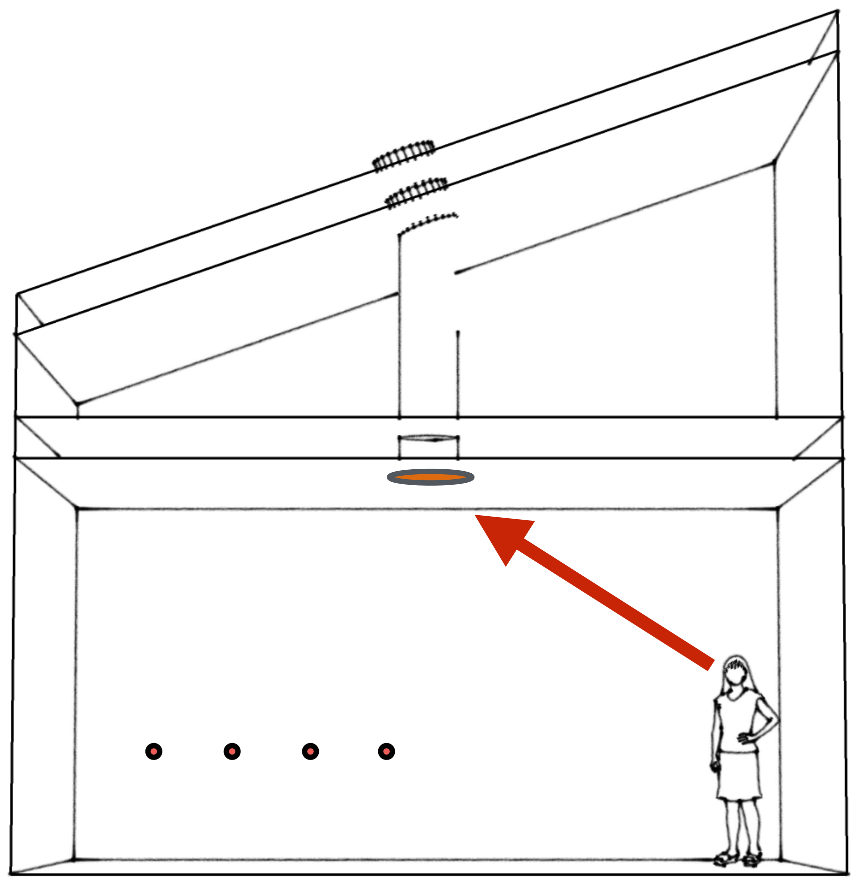
Receiver

```
#@rfluxmtx h=kf u=+Y o=results/v2_vmx/v2_%03d.hdr

LP_bottom polygon f_119_0
0
0
72
    3.107166  2.827189  3.048000
    3.048000  2.819400  3.048000
    2.988834  2.827189  3.048000 …
```



Why do we use the light material?

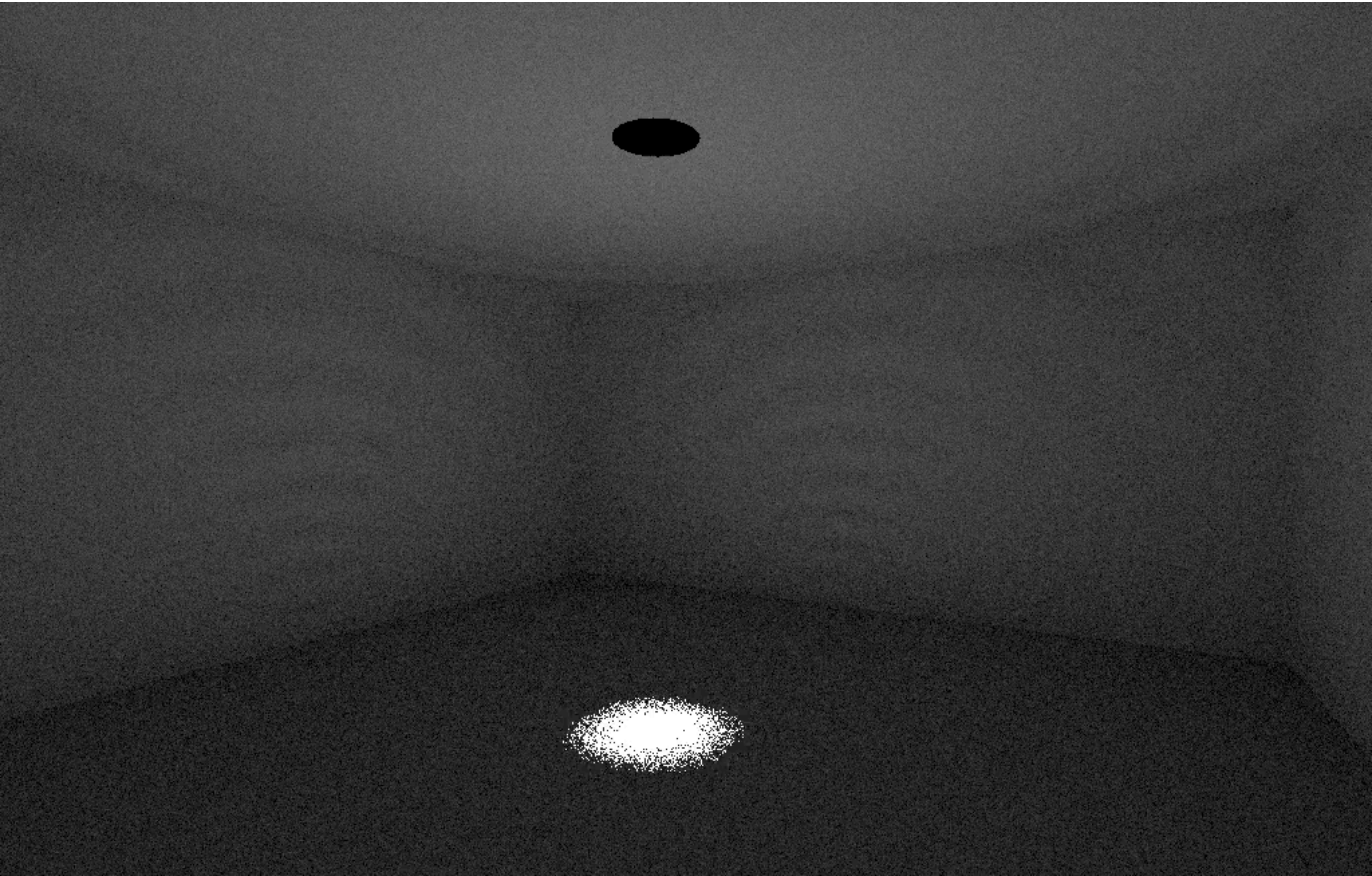
```
vwrays -fa -vf views/v2.vf -x 600 -y 600 | \
    rfluxmtx -n 8 `vwrays -vf views/v2.vf -x 600 -y 600 -d` \
    -fac -ab 6 -ad 1000 -lw 1e-3  - objects/LP_bottom.rad \
    materials_light.rad model.rad
```



light materials are sampled deterministicly, which is desirable for the light pipe because:

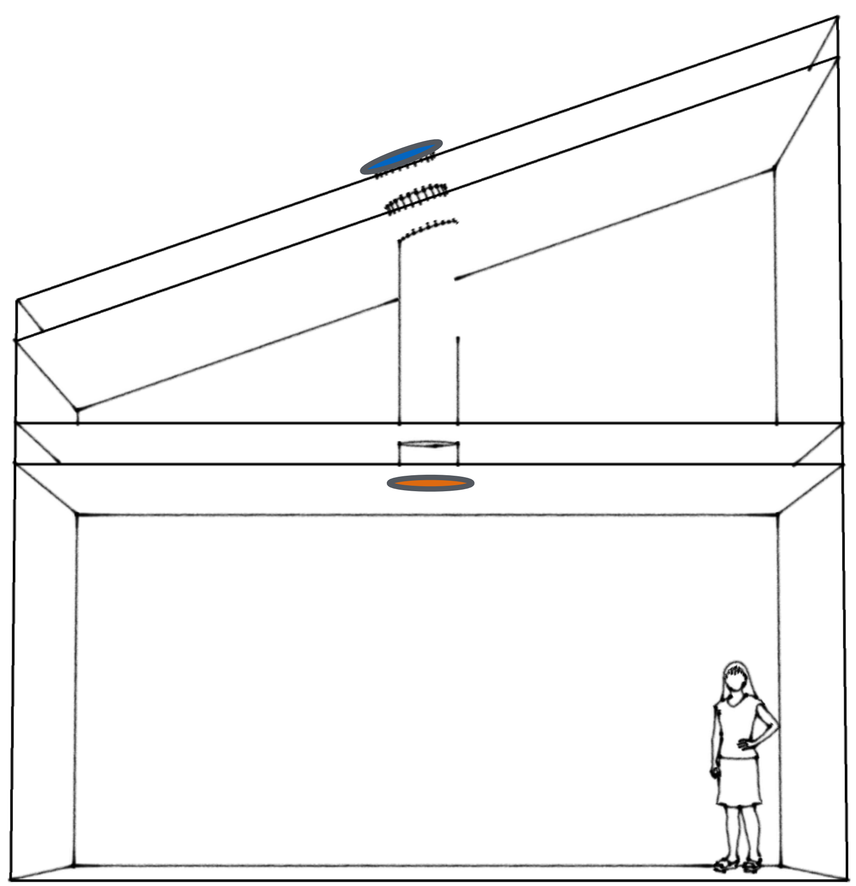
- small aperture
- no nearby points

View Matrix Renderings



Characterizing the lenses:

- Roof lens = glass (88% transmission)
- Room lens = trans (Td=0.05 Ts=0.55 Rd=0.3 Rs=0.03 rough=0.25) Also, importantly no fresnel effects w/trans)

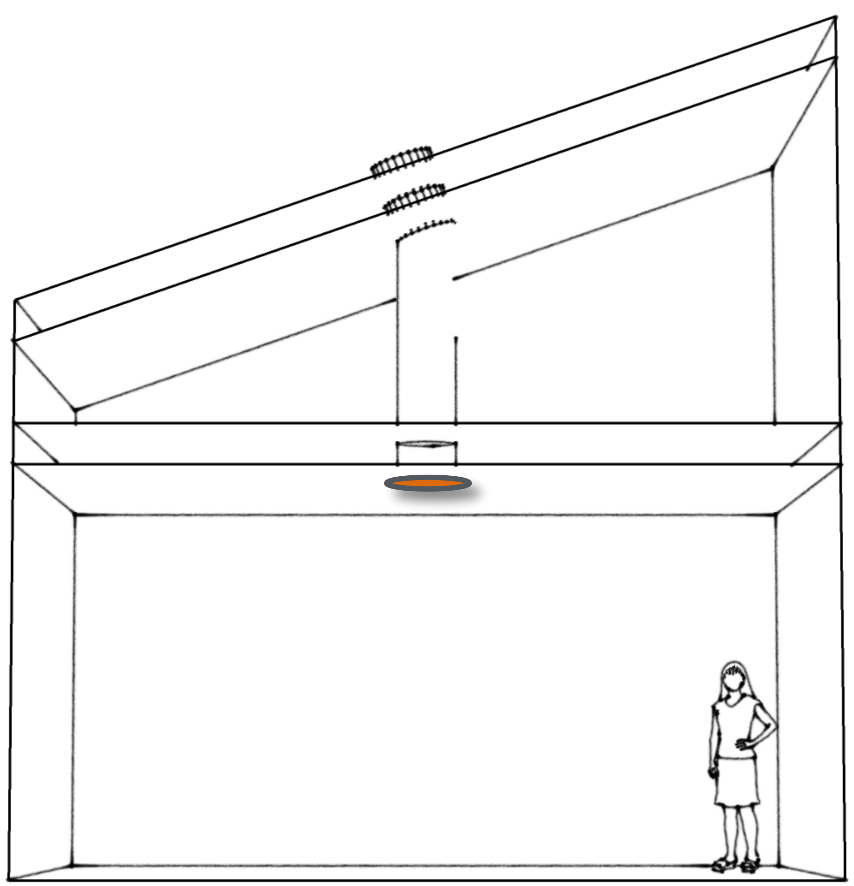


- You can model a much more complex lens!
- We'll use genBSDF with a rectangular surface. This assumes that the lens is thin without edge effects.

Characterizing the lenses:

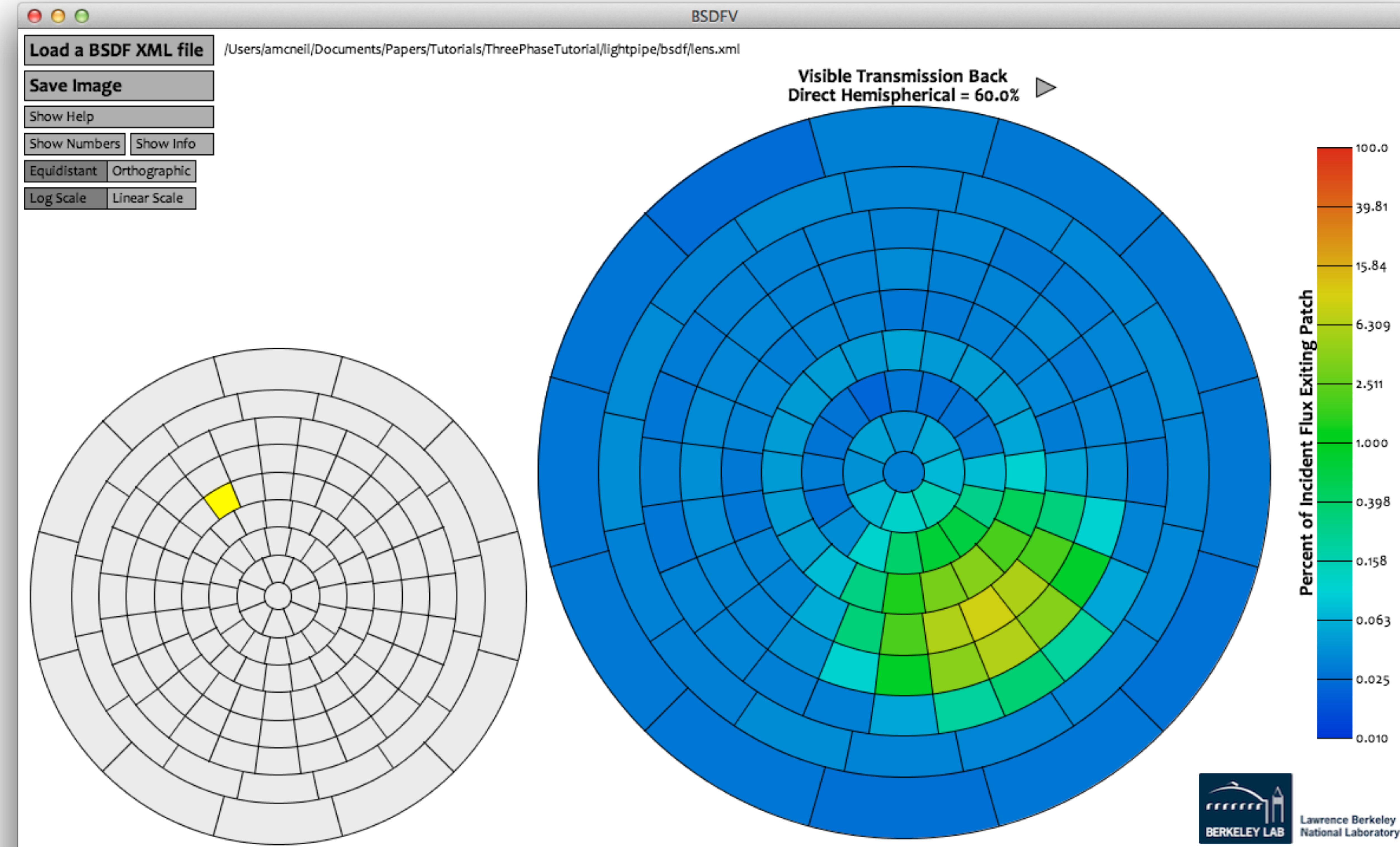
genBSDF -n 8 +f +b -r ‘-ab 2 -ad 1000 -lw 1e-3’ -c 10000 \
lens.rad > lens.xml

Model

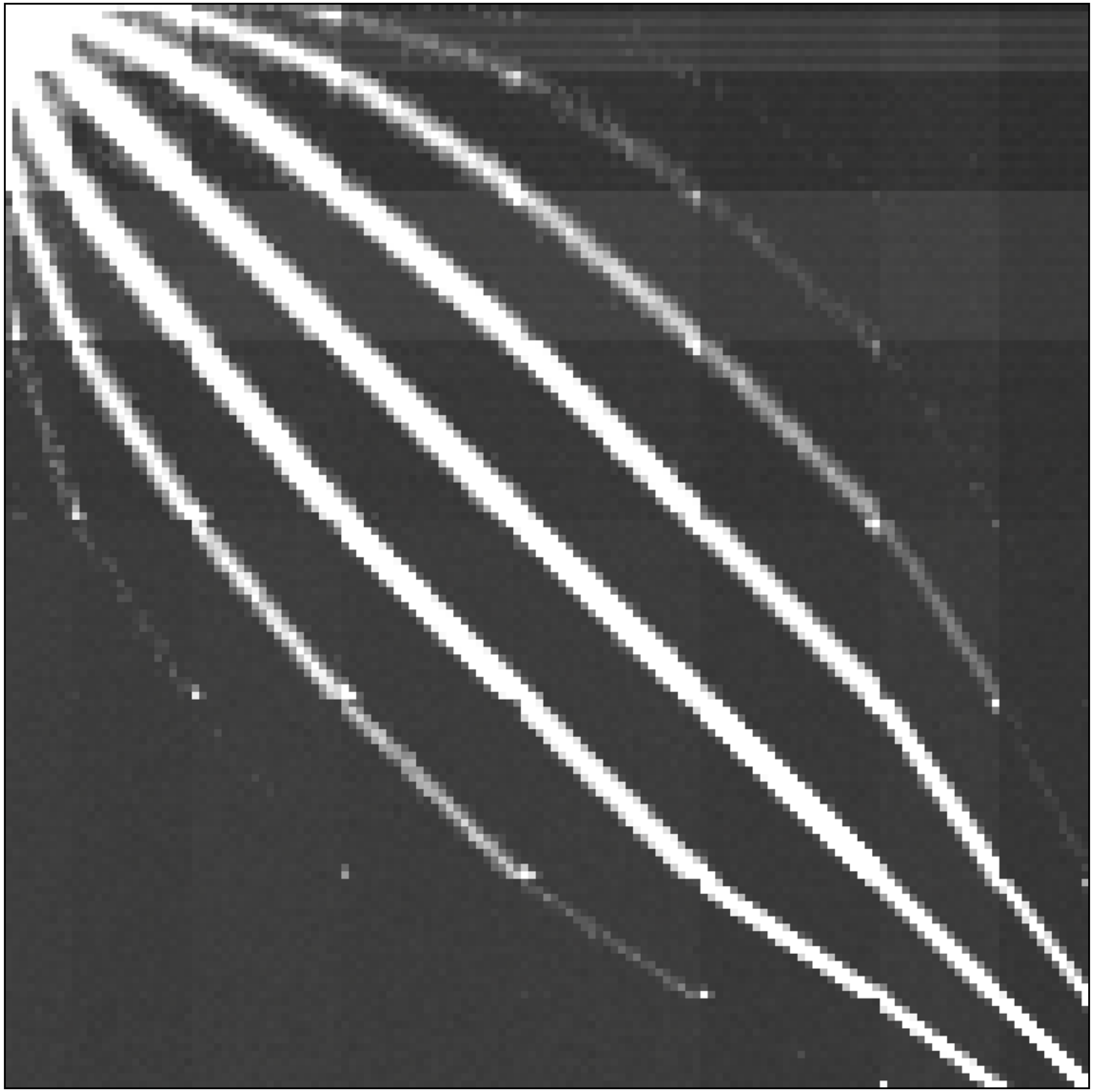


```
#Td=0.05 Ts=0.55 Rd=0.3 Rs=0.03
void trans LP_bottom
0
0
7 0.928 0.928 0.928 0.03 0.25 0.667 0.917

LP_bottom polygon poly4BSDF
0
0
12   0 0 0
     1 0 0
     1 1 0
     0 1 0
```

Generated BSDF



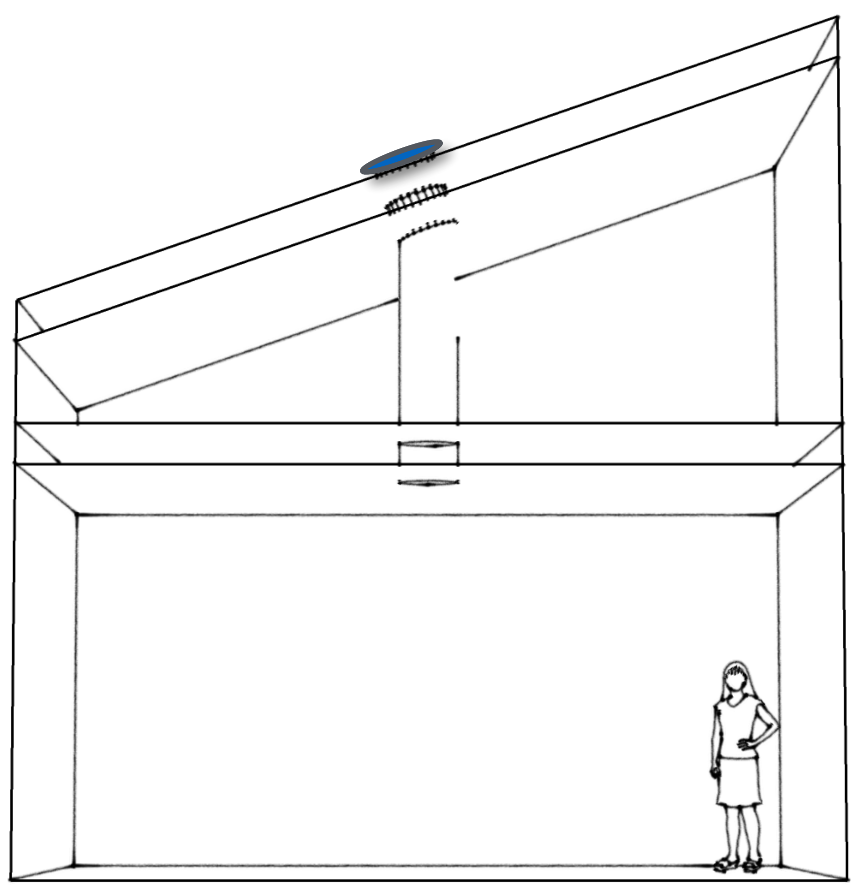
Generated BSDF



Characterizing the lenses:

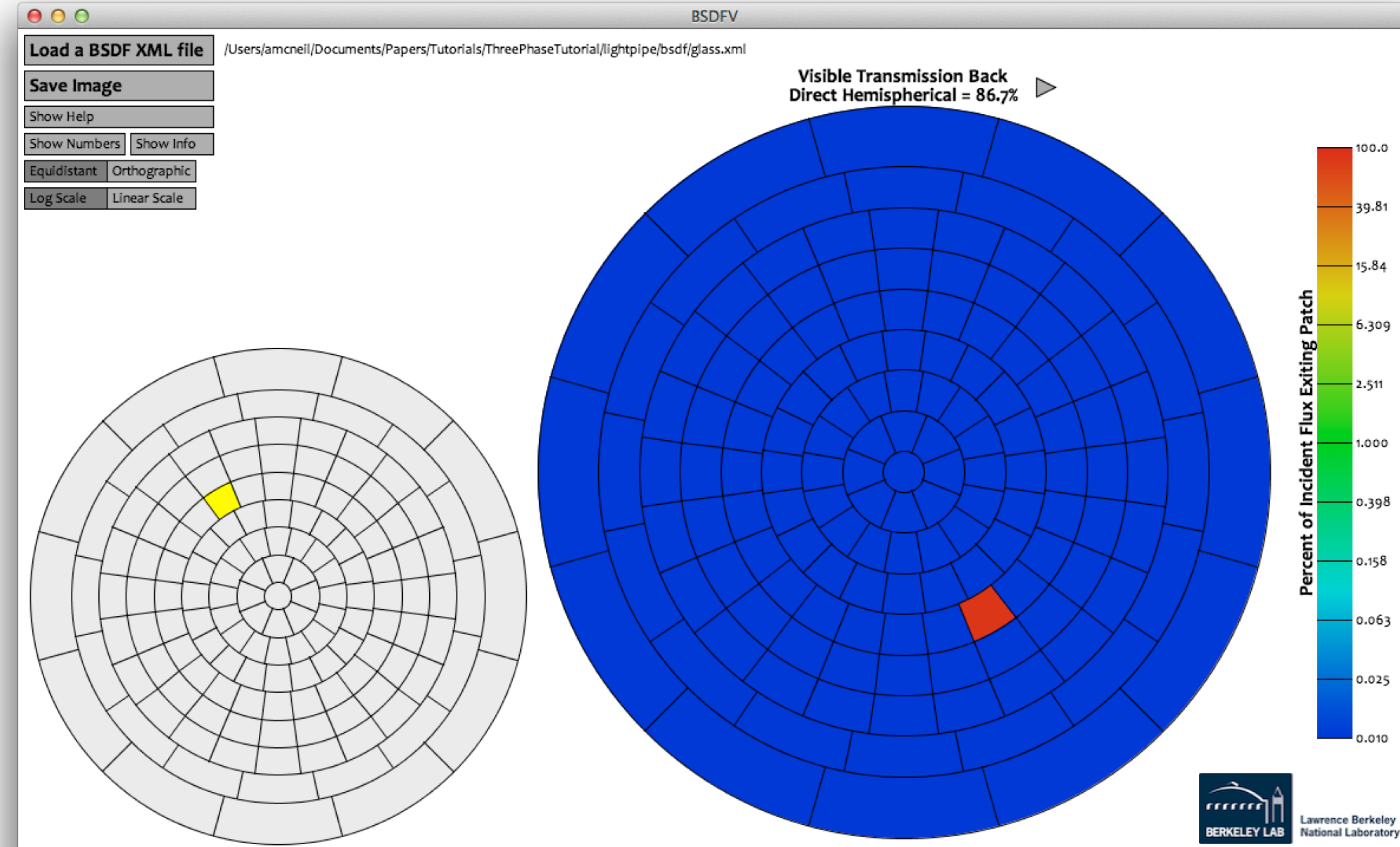
genBSDF -n 8 +f +b -r '-ab 2 -ad 10 -lw 1e-3' -c 100 \
glass.rad > glass.xml

Model

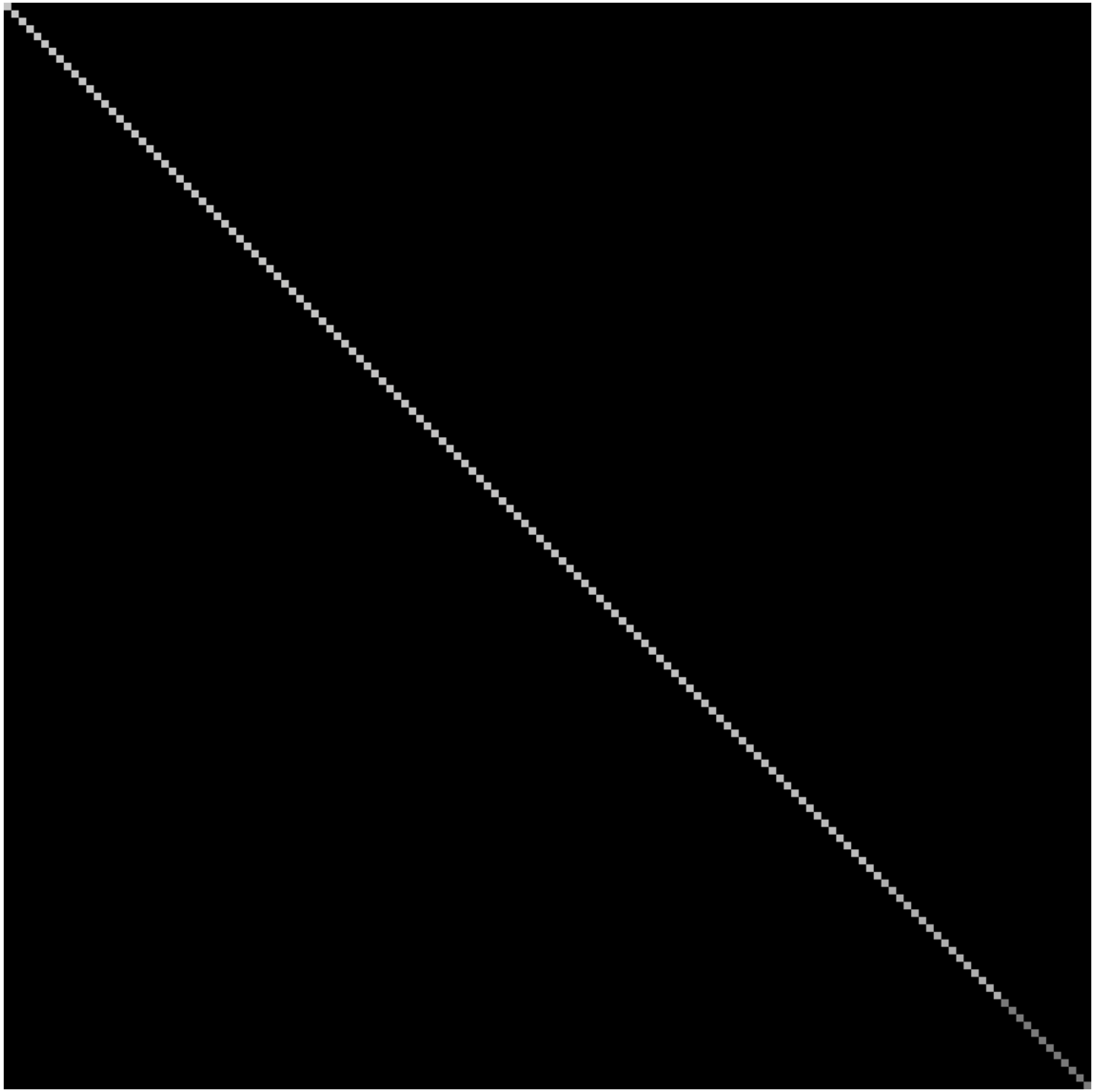


```
#Tn=0.88
void glass LP_top
0
0
3 0.9584 0.9584 0.9584

LP_top polygon poly4BSDF
0
0
12   0 0 0
     1 0 0
     1 1 0
     0 1 0
```

Glass BSDF

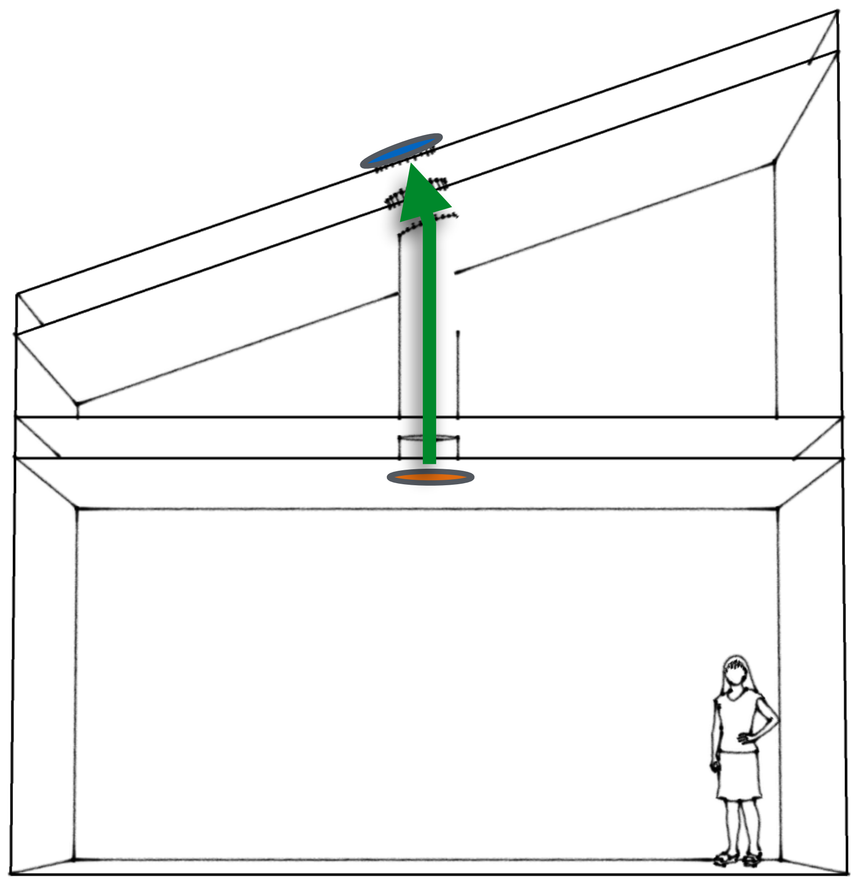

BSDFV
Load a BSDF XML file
/Users/amcneil/Documents/Papers/Tutorials/ThreePhaseTutorial/lightpipe/bsdf/glass.xml
Save Image
Show Help
Show Numbers
Show Info
Equidistant
Orthographic
Log Scale
Linear Scale
Visible Transmission Back
Direct Hemispherical = 86.7%
Percent of Incident Flux Exiting Patch
100.0
39.81
15.84
6.309
2.511
1.000
0.398
0.158
0.063
0.025
0.010
BERKELEY LAB
Lawrence Berkeley
National Laboratory

Glass BSDF



Characterizing the pipe transmission

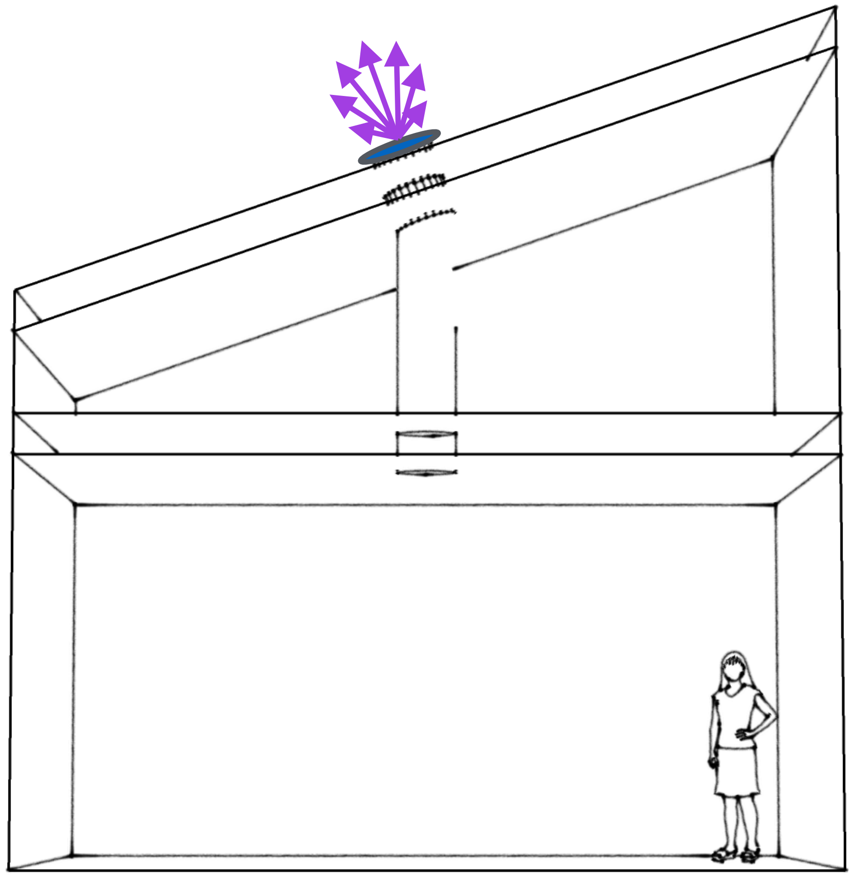
```
rfluxmtx -n 8 -ab 12 -ad 1000 -lw 1e-4 \
    objects/LP_bottom.rad objects/LP_top.rad \
    materials_glow.rad objects/LP_sides.rad \
    > results/LP_trans.mtx
```



Warning: The simulation parameters for a specular light pipe used here are untested. Convergence testing is recommended to determine appropriate settings.

The ‘Daylight’ Matrix

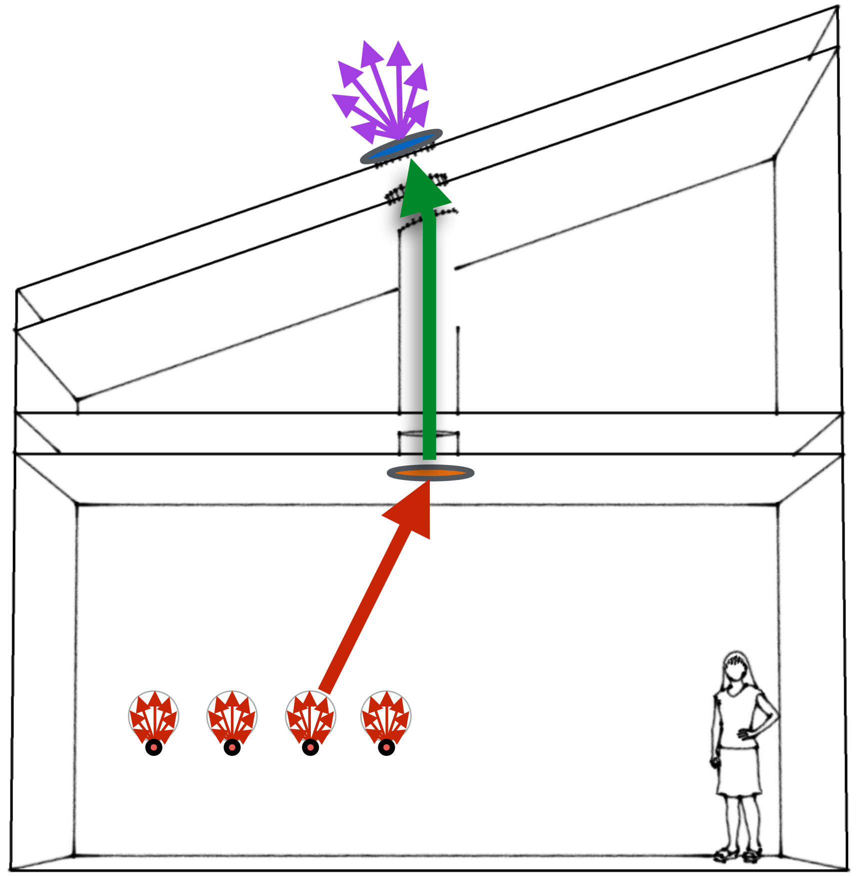
rfluxmtx -n 8 -ab 2 -ad 5000 -lw 1e-4 -c 10000 \
objects/LP_top.rad skies/sky.rad \
materials.rad model.rad > results/exterior.dmx



Putting it all together (sensor points)

```
rmtxop  results/points.vmx bsdf/lens.xml \
        results/LP_trans.mtx bsdf/glass.xml \
        results/exterior.dmx skies/12_21_15.skv
   | rmtxop -fa -c 47.4 119.9 11.6 - > illum_12_21_15.txt
```

(Change from Irradiance to Illuminance)

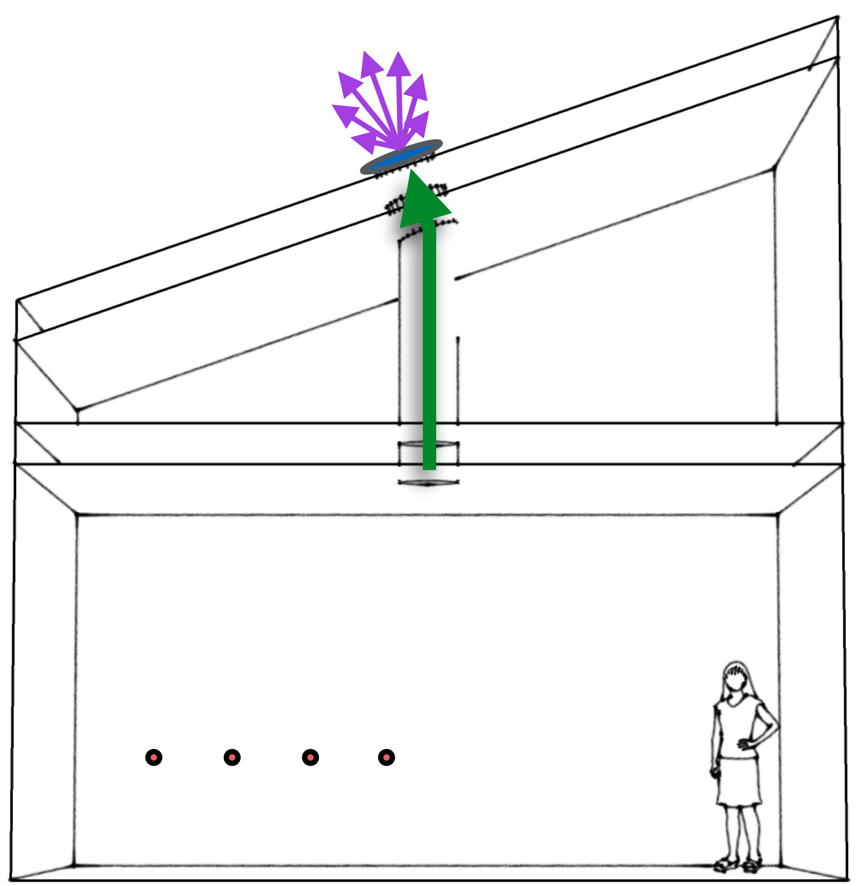


```
#?RADIANCE
rmtxop -fa -c 179 0 0 -
NROWS=5
NCOLS=1
NCOMP=1
FORMAT=ascii

2.767935412636226e+00
2.971198976613583e+00
3.337874915476797e+01
3.173601257922114e+00
3.558548860092117e+00
```

Putting it all together (renderings)

rmtxop can't handle image view matrices. We have to arrange the matrices so that we can use dctimestep.

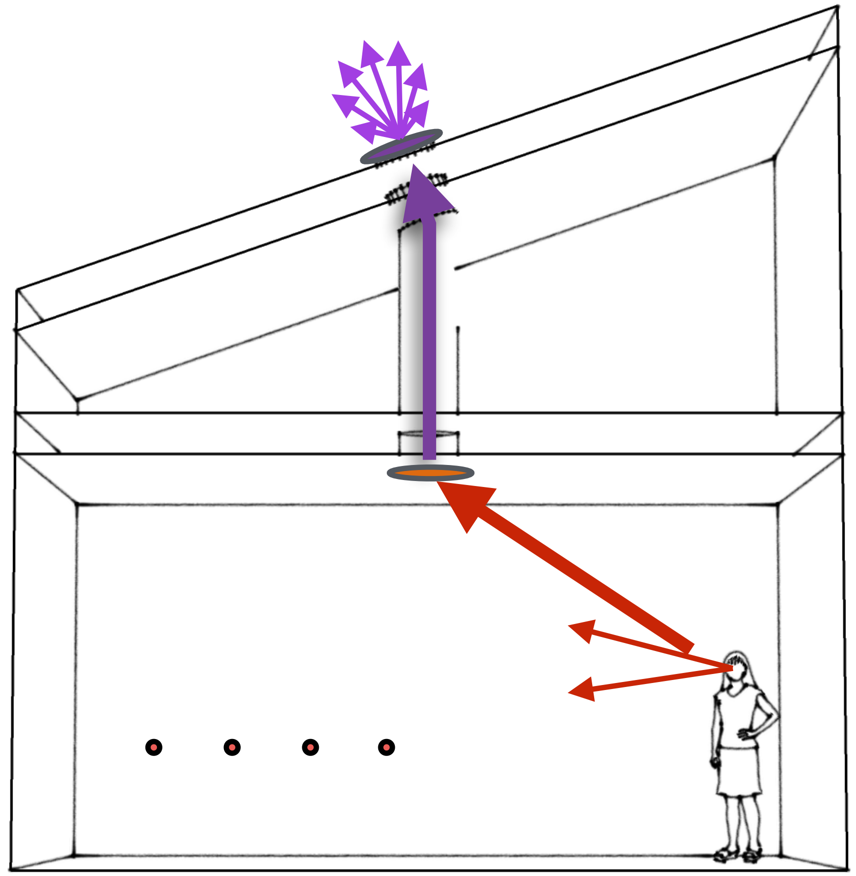
1. Combine pipe, skylight and daylight matrix into a pseudo-daylight matrix.



```
rmtxop   results/LP_trans.mtx \
         bsdf/glass.xml \
         results/exterior.dmx \
         > pipe2exterior.dmx
```

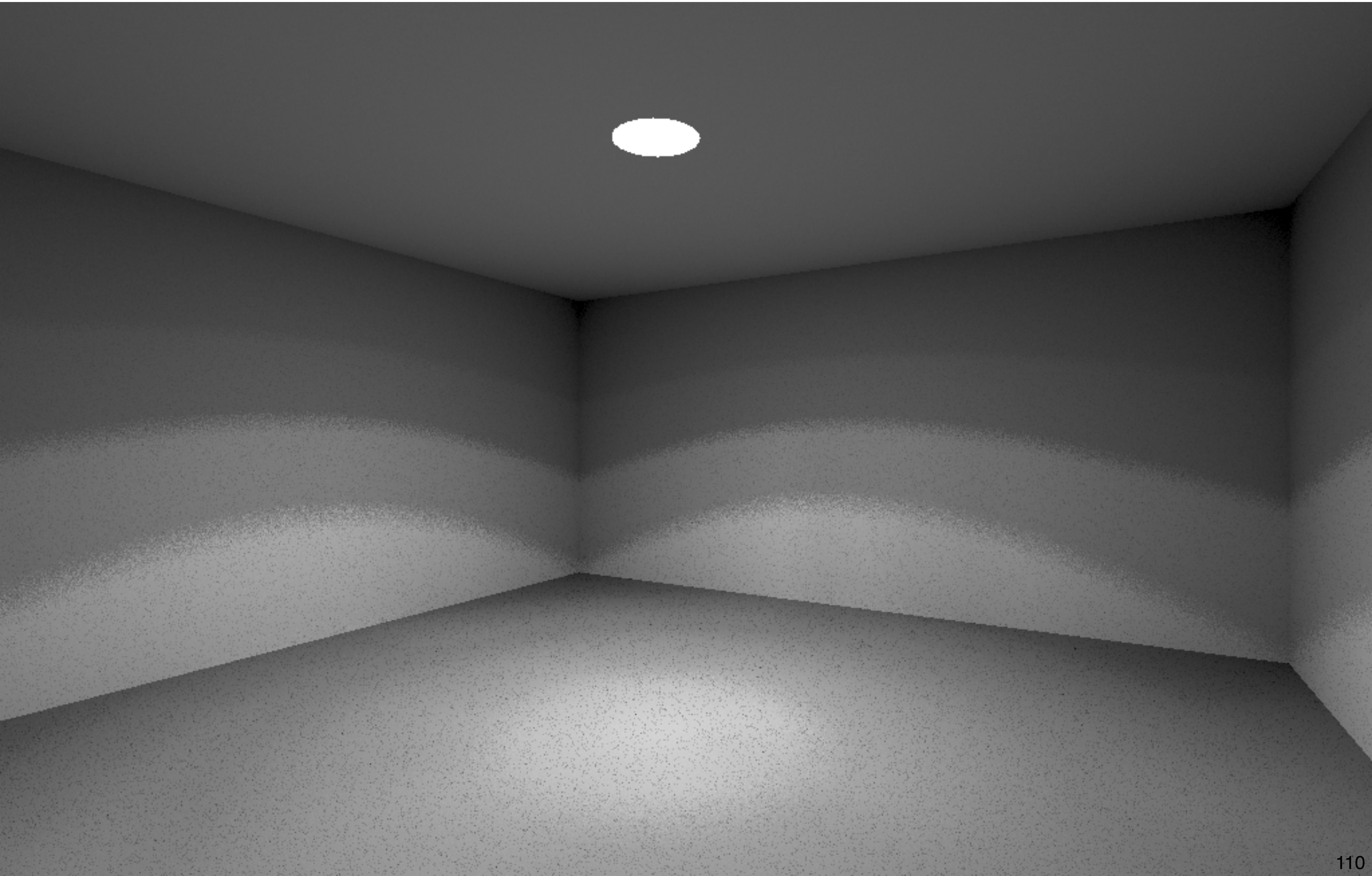
Putting it all together (renderings)

2. Use dctimestep to run the three-phase



```
dctimestep \
    results/v2_vmx/v2_%03d.hdr \
    bsdf/lens.xml \
    results/pipe2exterior.dmx \
    skies/12_21_15.skv
    > 12_21_15.hdr
```

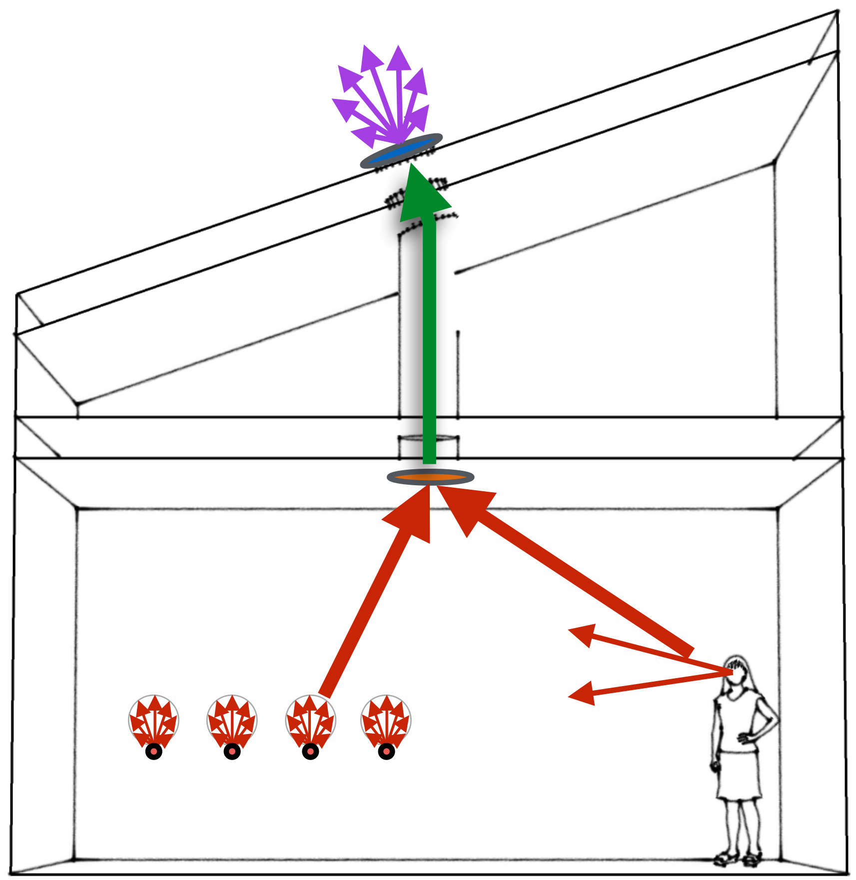
Rendered Result



Changing something

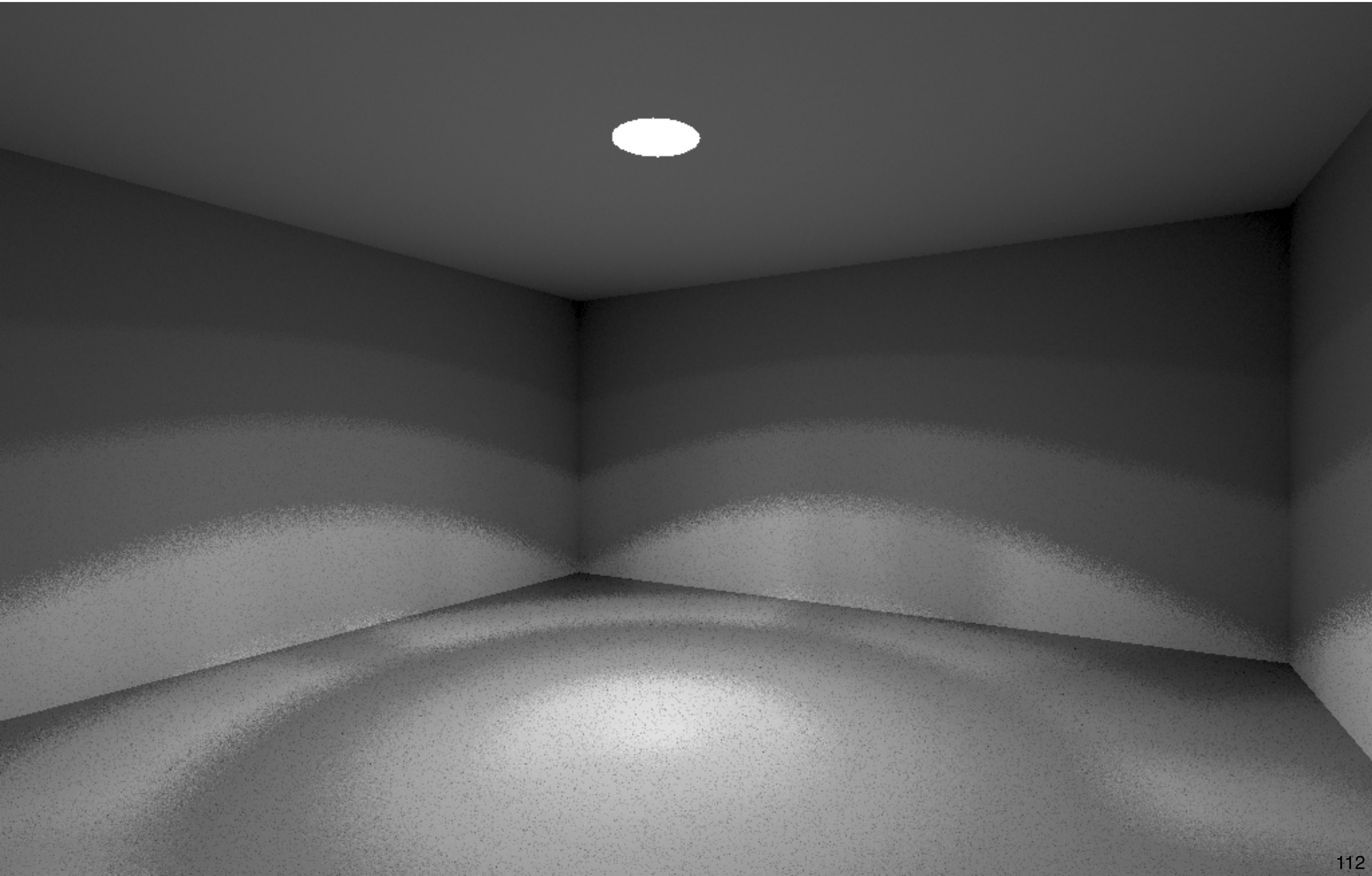
Let's say we wanted to try a different lens at the end of the lightpipe.

Just change the matrix!

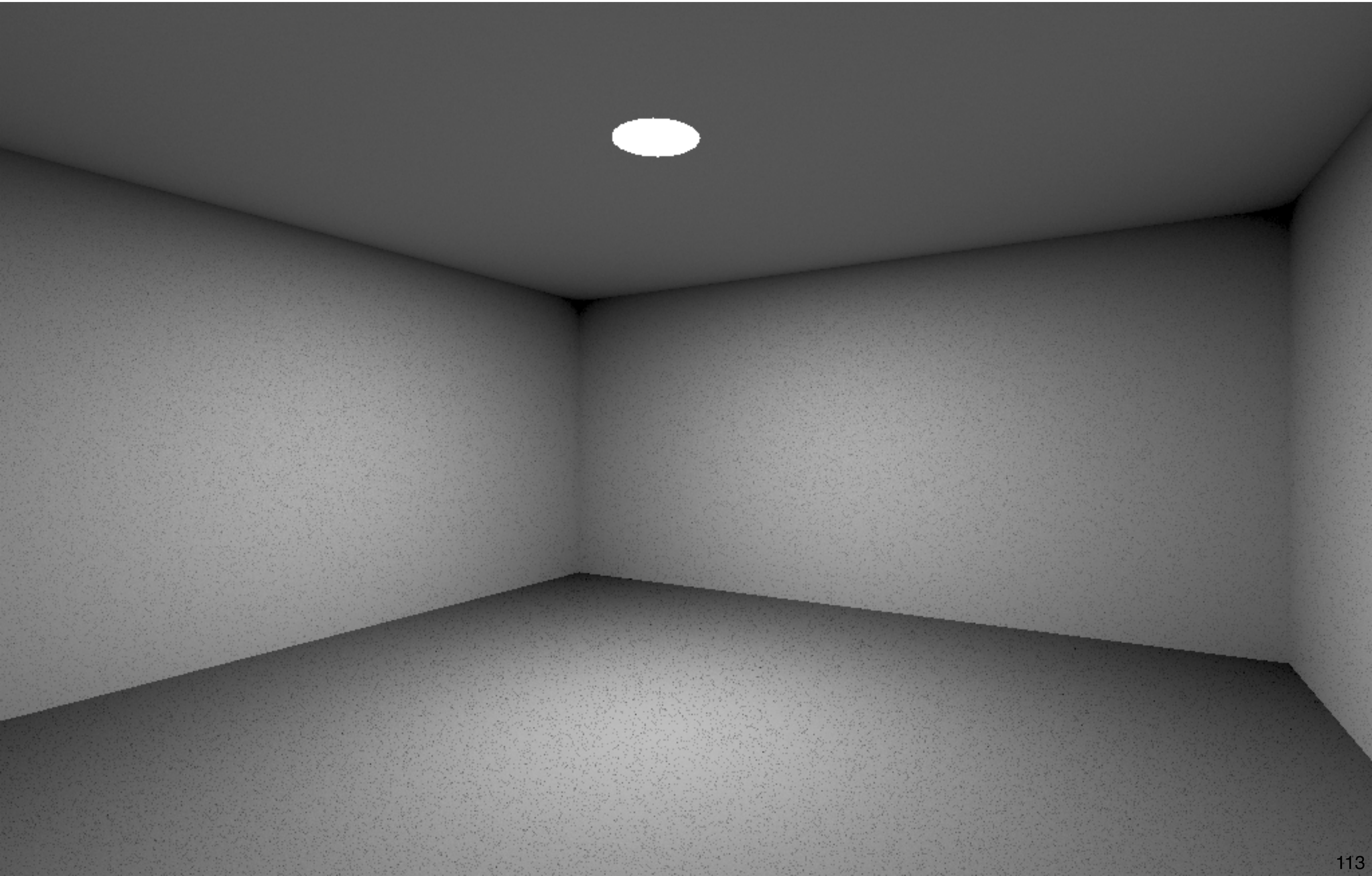


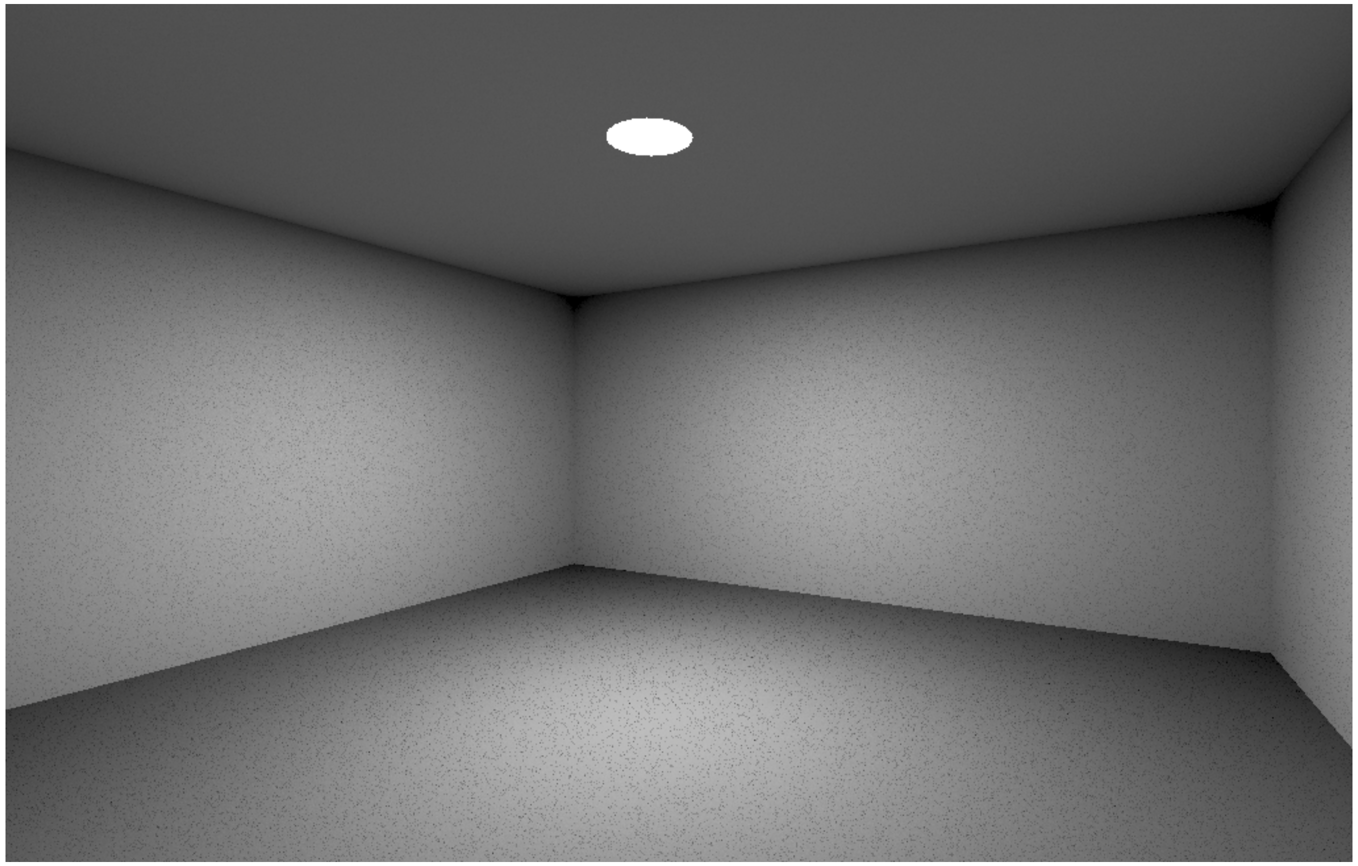
```
dctimestep \
    results/v2_vmx/v2_%03d.hdr \
    bsdf/glass.xml \
    results/pipe2exterior.dmx \
    skies/12_21_15.skv
    > 12_21_15.hdr
```

Rendered Result - clear lens

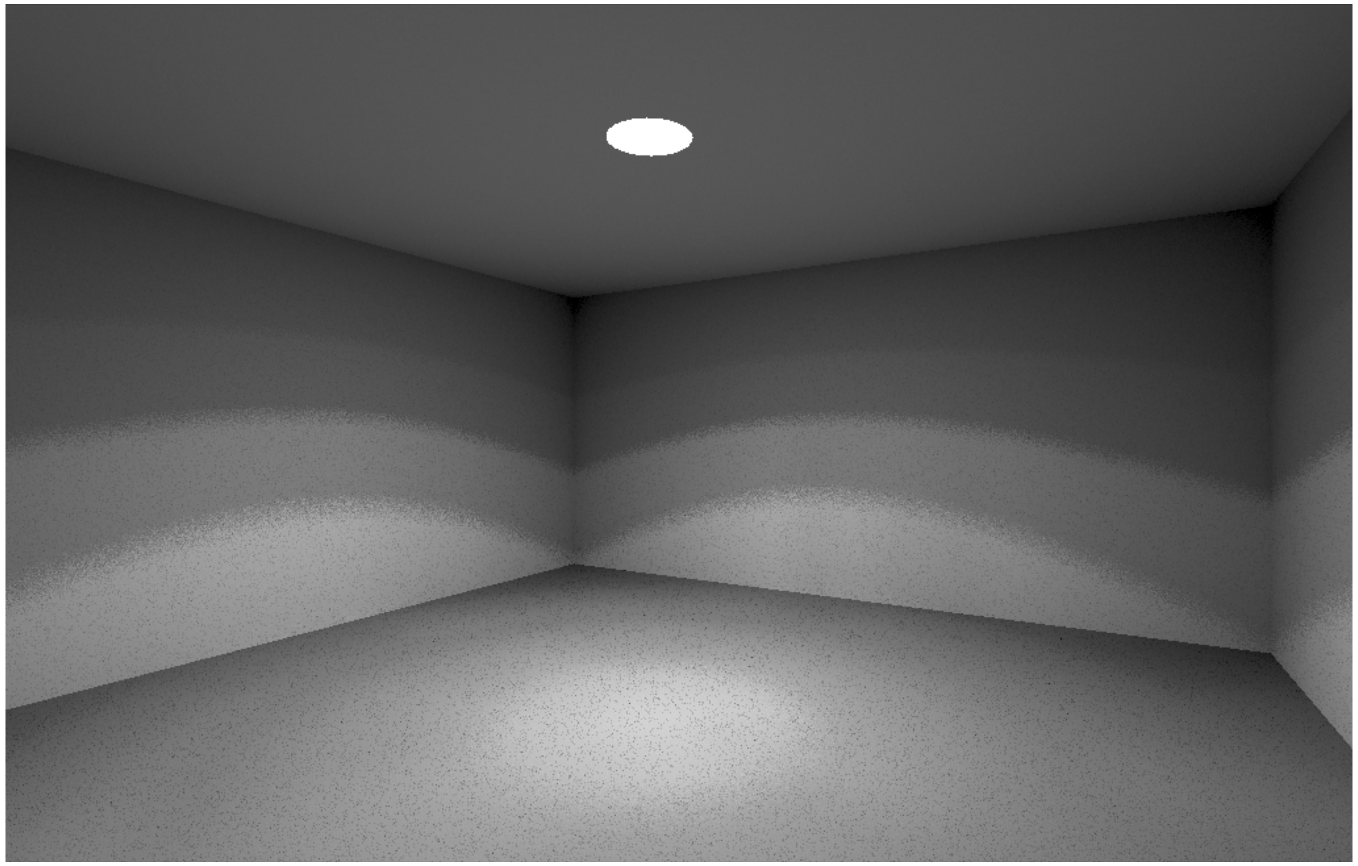


Rendered Result - lambertian lens

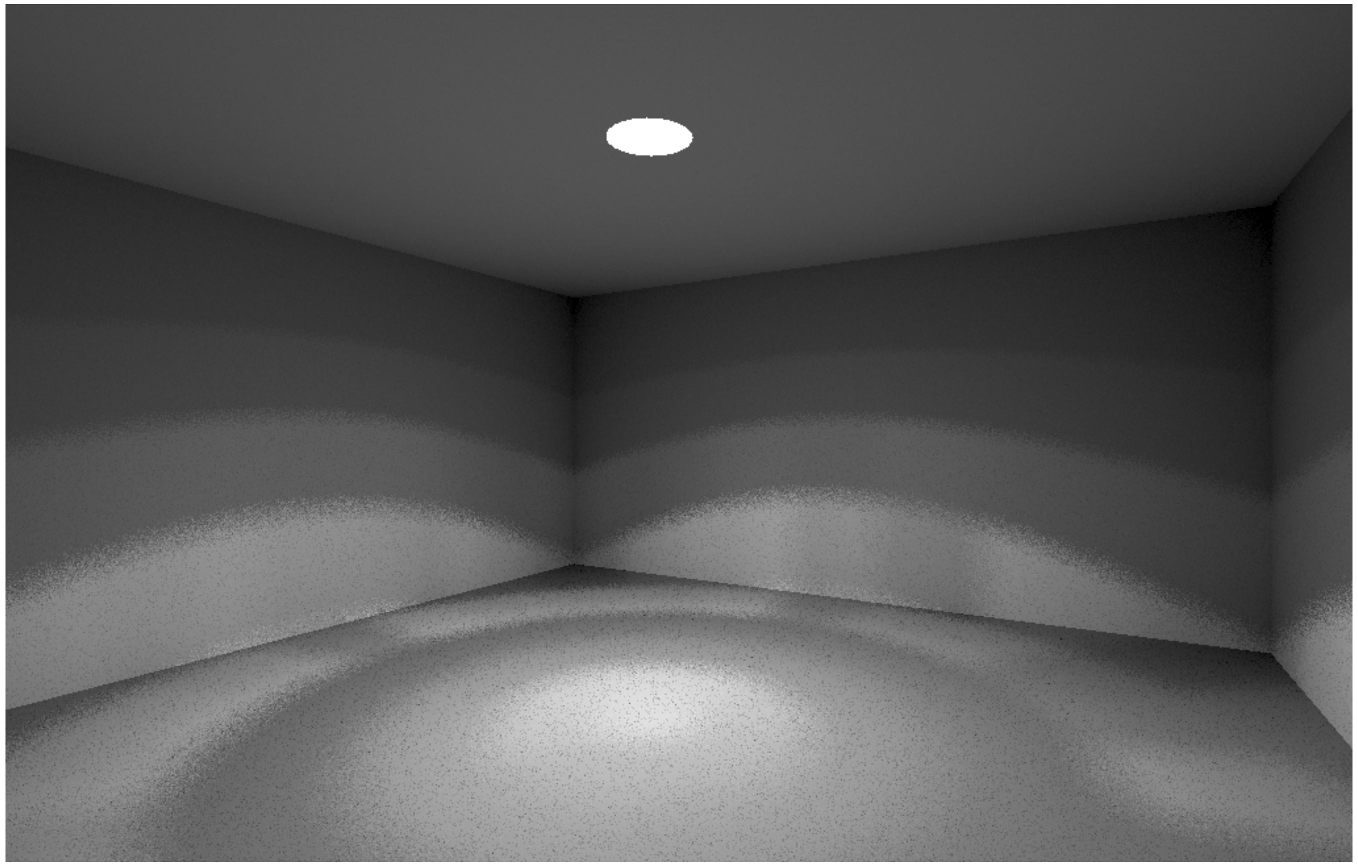




Lambertian



Semi Diffuse

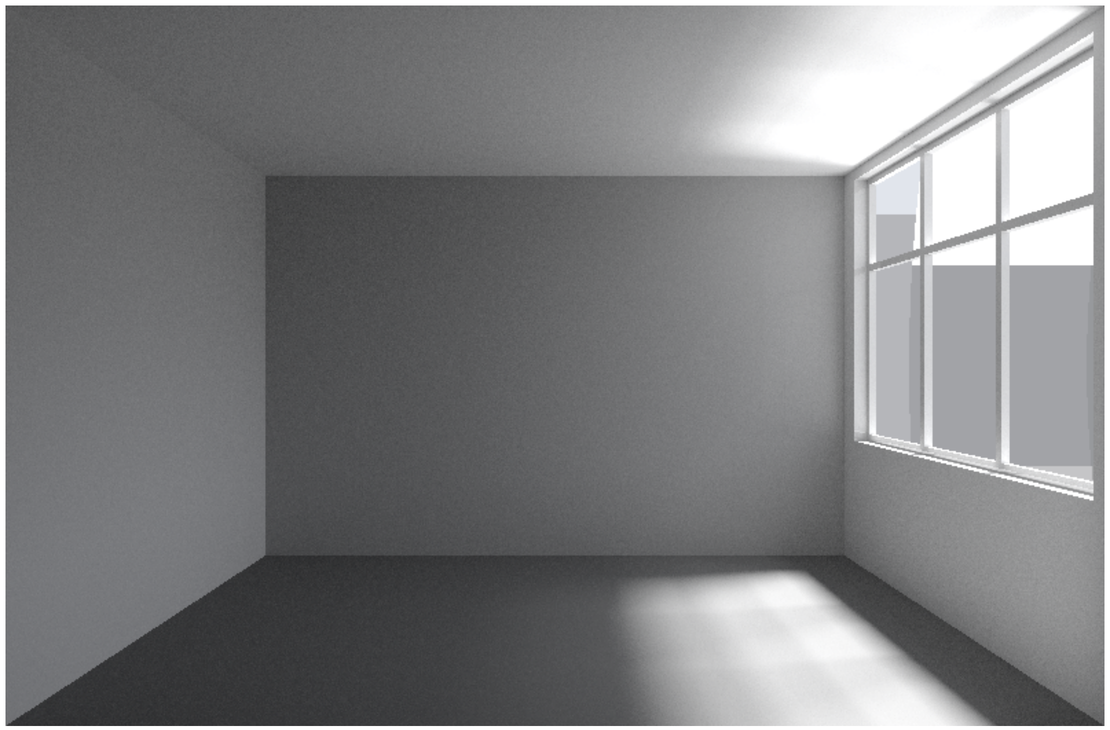


Clear

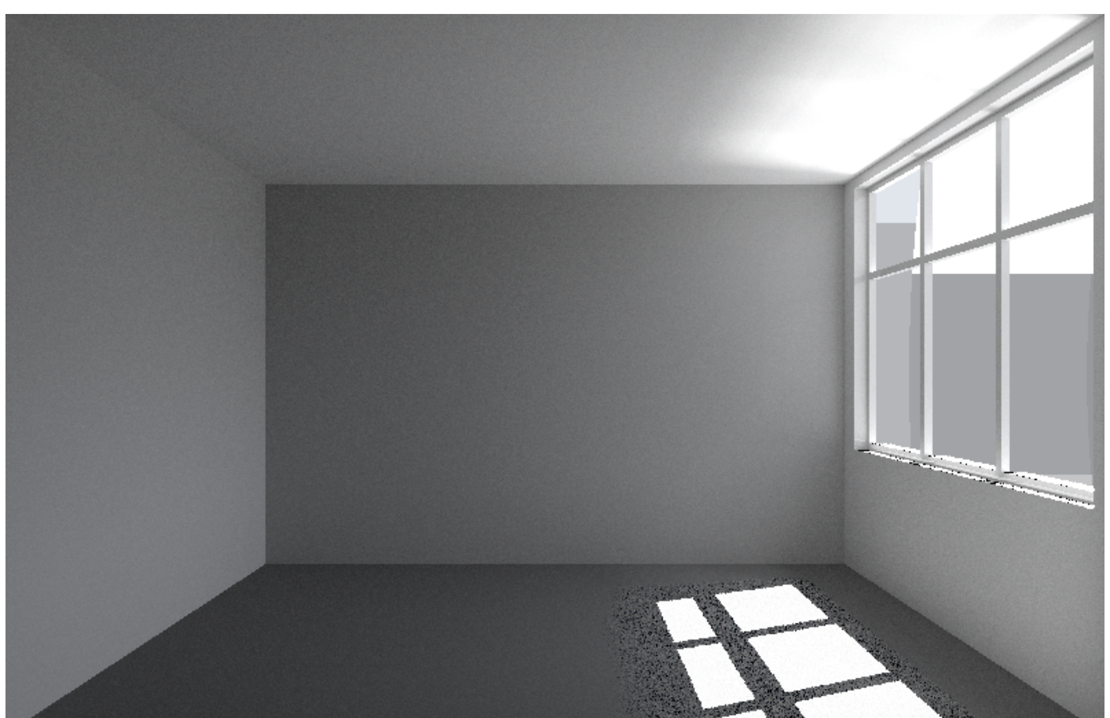
Now with 67% more phases!

The 5-phase method

Why do we need more %@#!$ phases?

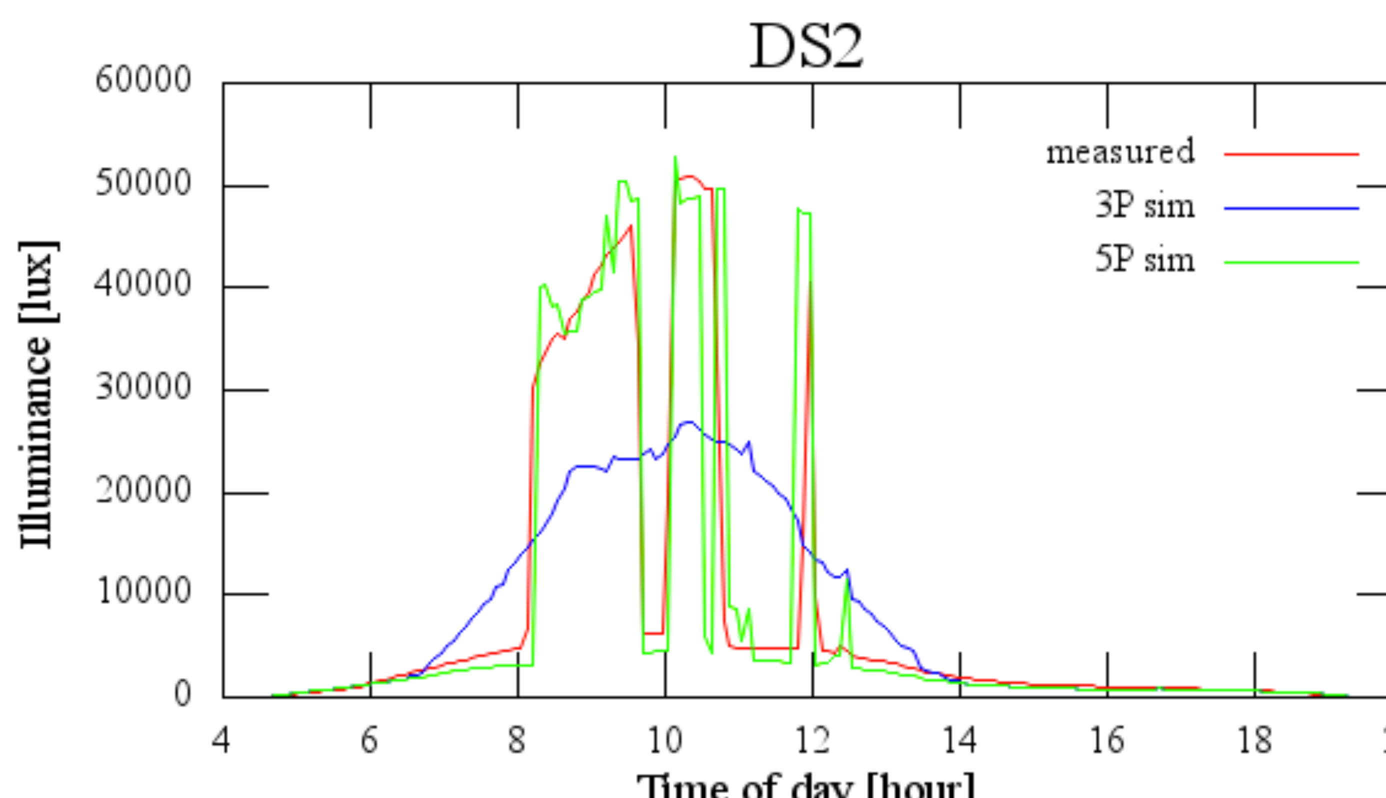


3-phase simulation result



5-phase simulation result

- The three-phase method disperses energy passing through the window.
- Particularly evident with direct solar component.
- Five-phase method uses high-resolution BSDF or actual BSDF geometry for direct solar component.



The Equations

Daylight Coefficient Equation:

$$E_{DC} = C_{dc}S$$

Three-Phase Equation:

$$E_{3ph} = VTDS$$

Five-Phase Equation:

$$E_{5ph} = VTDS - V_{d}TD_{d}S_{d} + C_{ds}S_{sun}$$

Direct sun calculated using high-resolution BSDF or system geometry.

Direct sun component calculated using the 3-phase method.

3-phase simulation.

Graphically

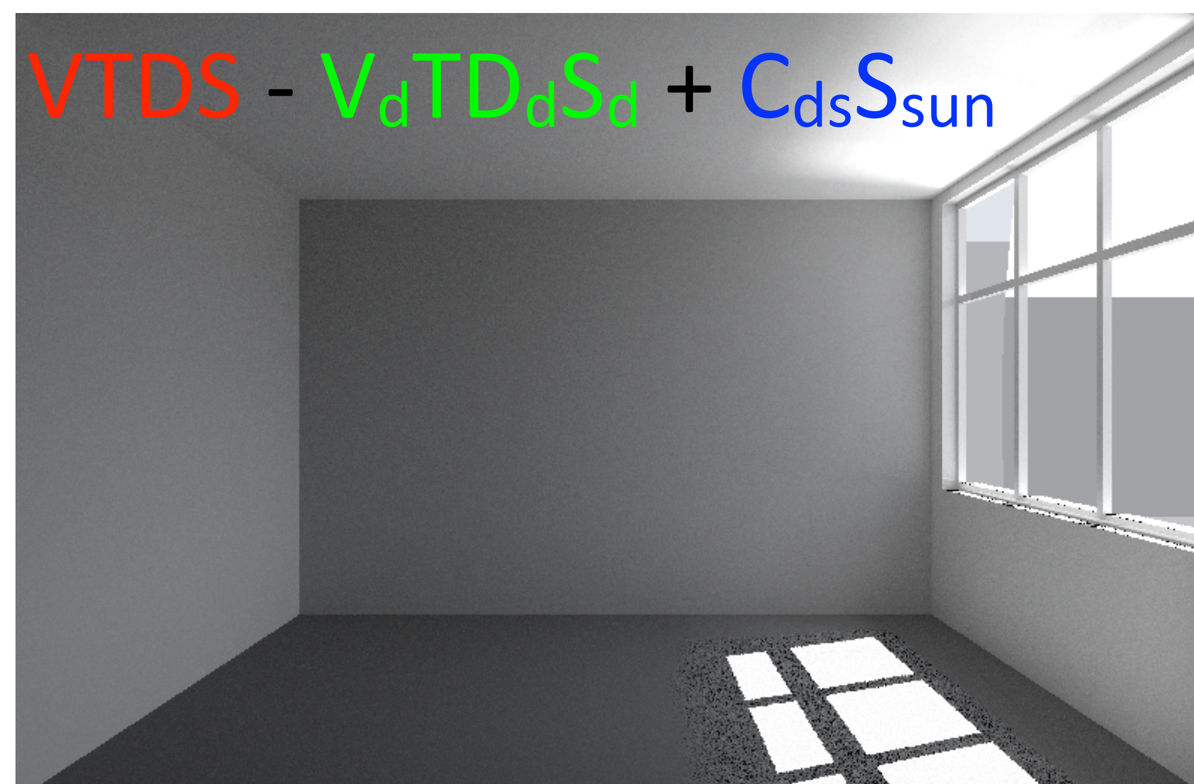
$I_{5ph} = VTDS - V_dTD_dS_d + C_{ds}S_{sun}$

-

$I_{5ph} = VTDS - V_d T D_d S_d + C_{ds} S_{sun}$

Graphically

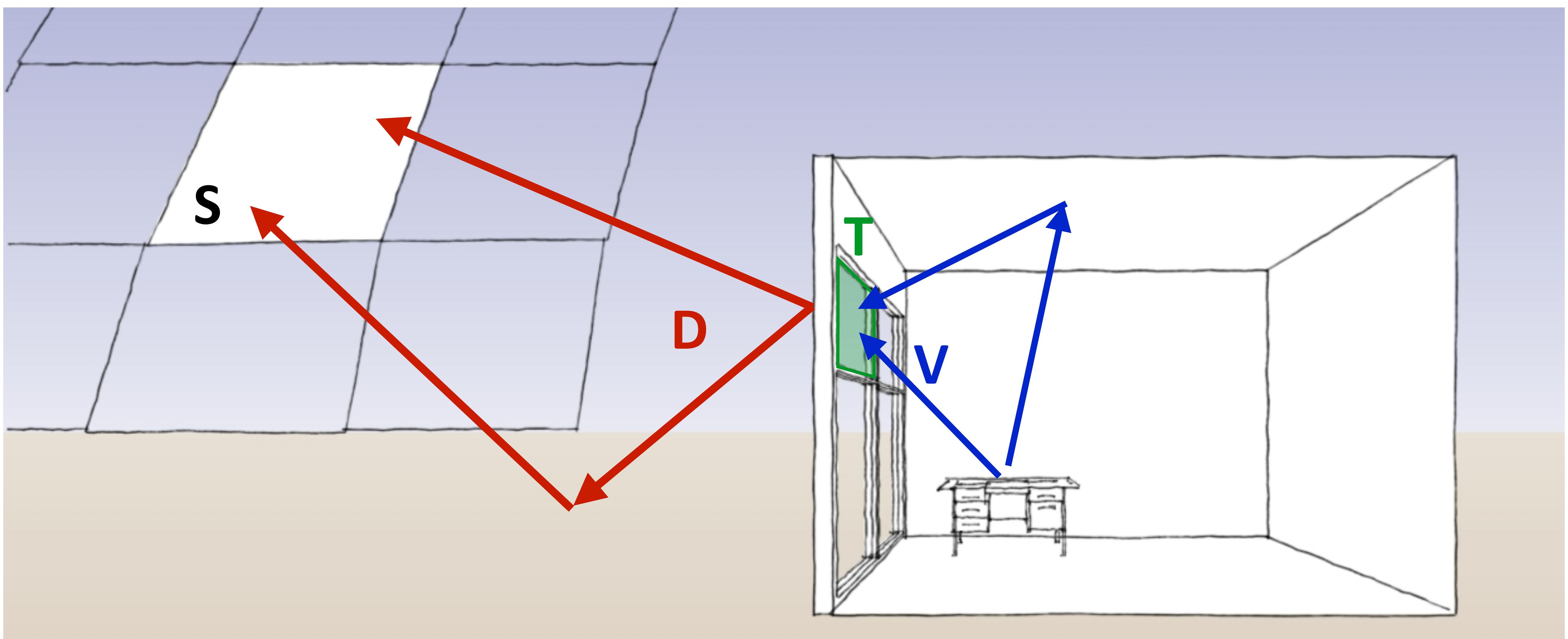
+



First Term: 3-phase

I_{5ph} = VTDS - $V_d T D_d S_d + C_{ds} S_{sun}$

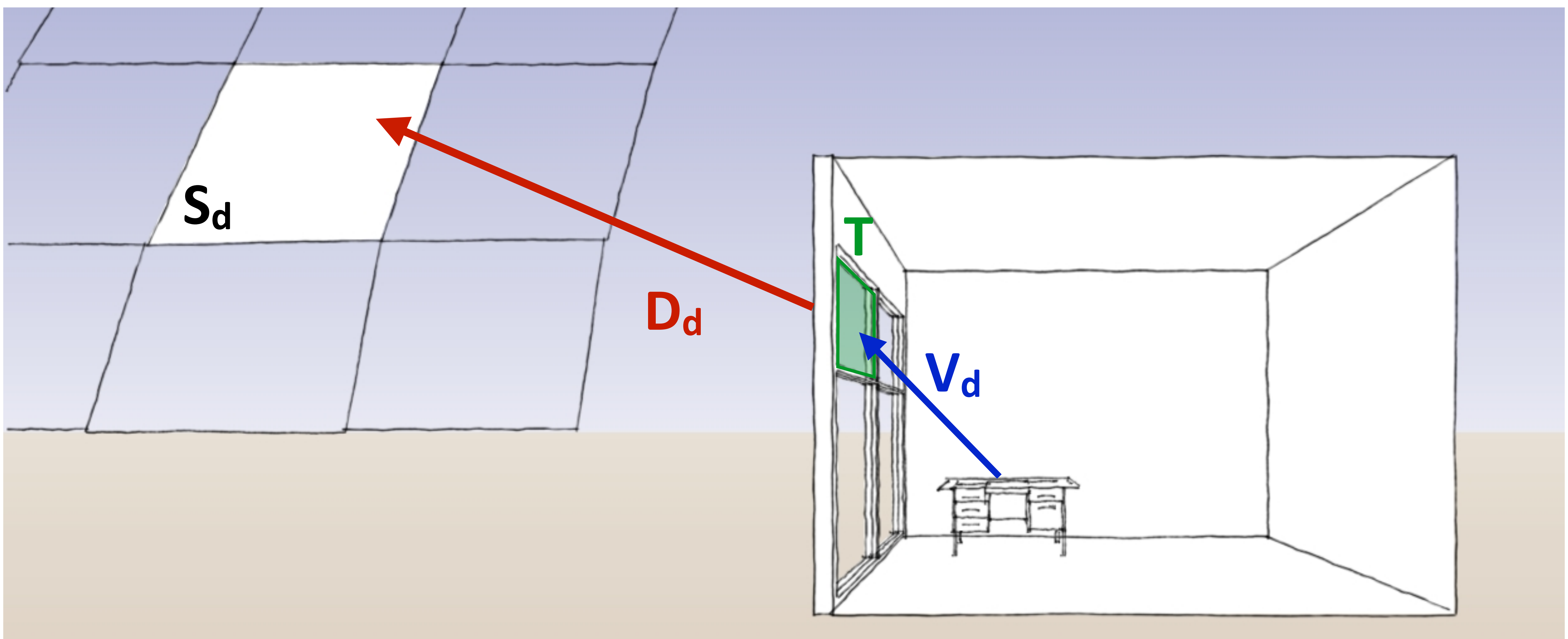
V = View Matrix (interior)
T = Transmission Matrix (BSDF)
D = Daylight Matrix (exterior)
S = Sky Matrix



$I_{5ph} = VTDS - V_dTD_dS_d + C_{ds}S_{sun}$

Second Term: Direct component of 3-phase

V_d = Direct View Matrix (interior)
T = Transmission Matrix (BSDF)
D_d = Direct Daylight Matrix (exterior)
S_d = Direct Sky Matrix

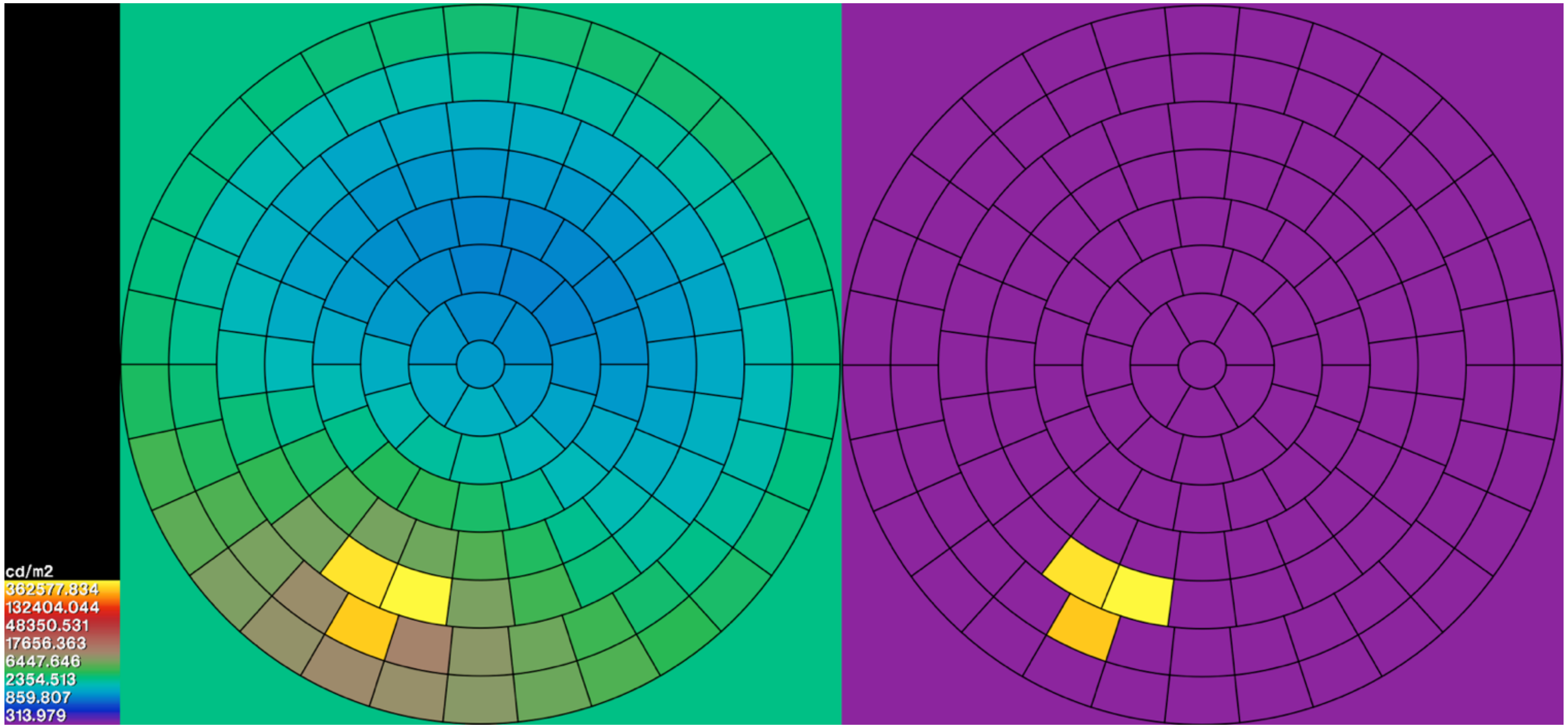


$I_{5ph} = VTDS - V_dTD_dS_d + C_{ds}S_{sun}$

Direct Sky Matrix (S_d)

The direct sky matrix contains only luminance from the sun.

gendaymtx has a -d option to generate a direct only sky matrix.



$$I_{5ph} = VTDS - V_dTD_dS_d + C_{ds}S_{sun}$$

Daylight matrix (Dd)

The daylight matrix contains coefficients relating energy leaving sky patches with energy incident on a window in a klems directional bin, without any external inter-reflection.

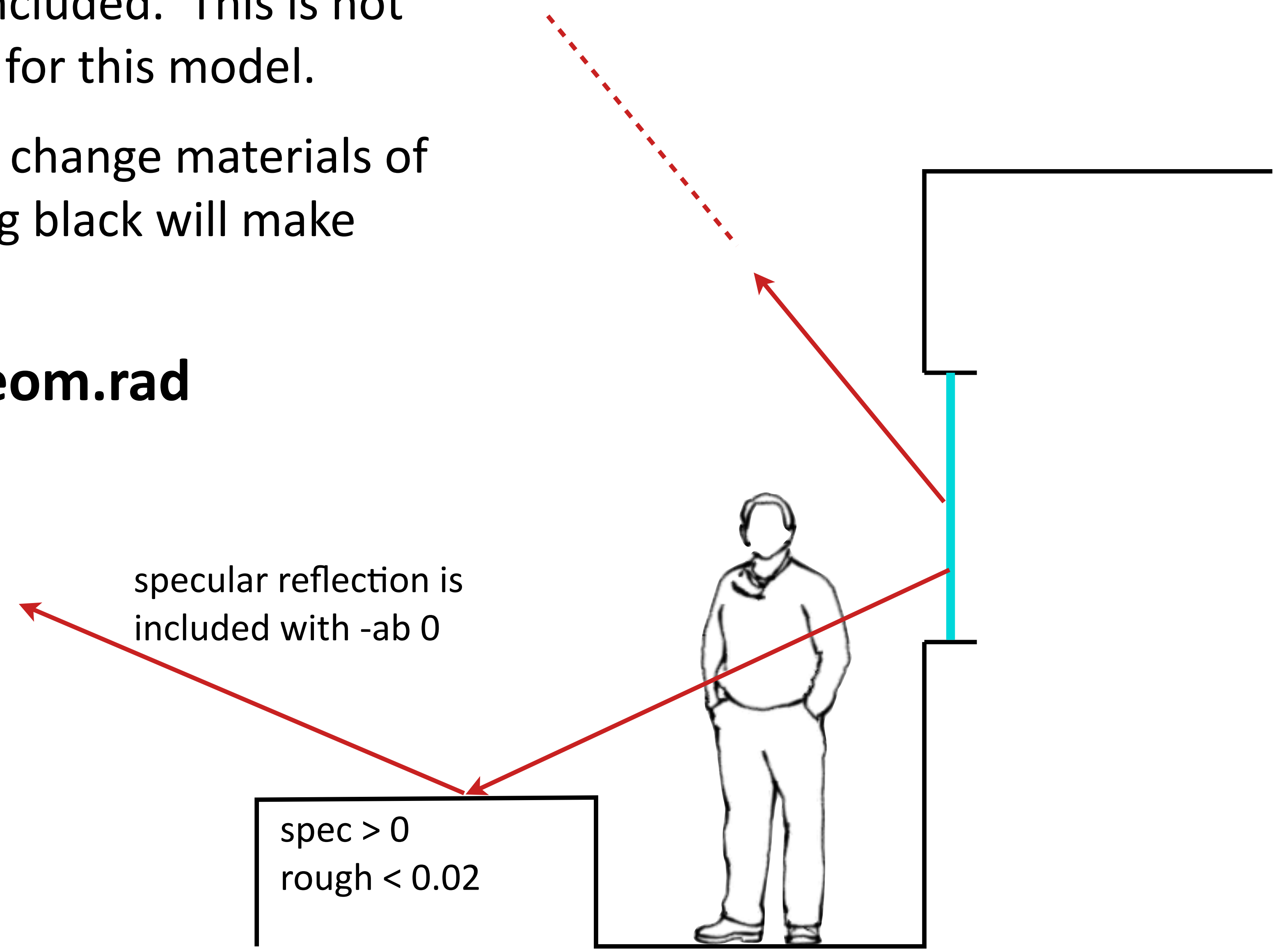
rcontrib is used with -ab 0

Even with -ab 0 specular reflections are included. This is not desired so materials need to be modified for this model.

So we create a material called ‘black’ and change materials of all geometry to black (the reason for using black will make sense later). to do this we’ll use:

xform -m black geom.rad > blackgeom.rad

rfluxmtx -ab 0



Direct View matrix (V_d)

The view matrix contains coefficients relating energy leaving a window in klems direction bins that is directly incident at a sensor point or image pixel.

rcontrib is used with -ab 1

As with the direct daylight matrix, specular reflections will be included (which is unwanted) so we'll have to change the materials in the model.

xform -m black geom.rad | oconv

rfluxmtx -ab 1

$I_{5ph} = VTDS - V_dTD_dS_d + C_{ds}S_{sun}$

Third Term: Direct sun component

C_{ds} = Sun Coefficient Matrix

S_{sun} = Sun Matrix

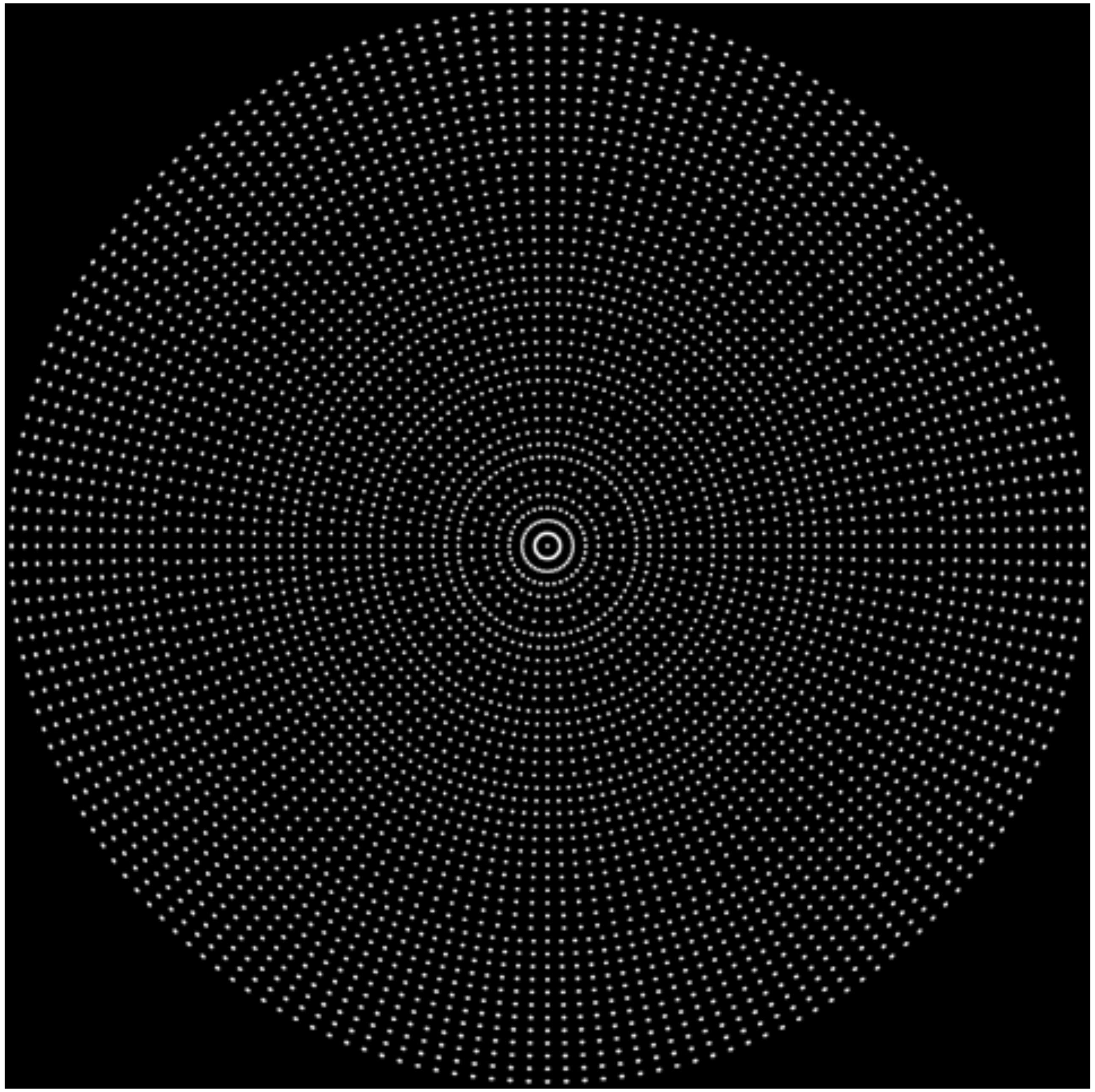
Generating suns

We need a radiance geometry file containing lots of suns centered in Reinhart sky patches.

- reinsrc.cal
- rcalc

echo void light solar 0 0 3 1e6 1e6 1e6 > suns.rad

```
cnt 5185 | rcalc -e MF:6 -f reinsrc.cal \
    -e Rbin=recno -o 'solar source sun 0 0 4 \
   ${ Dx } ${ Dy } ${ Dz } 0.533' >> suns.rad
```



Fisheye rendering looking up at a sky full of suns. These suns use the Reinhart MF:6 sky patches.

$I_{5ph} = VTDS - V_dTD_dS_d + C_{ds}S_{sun}$

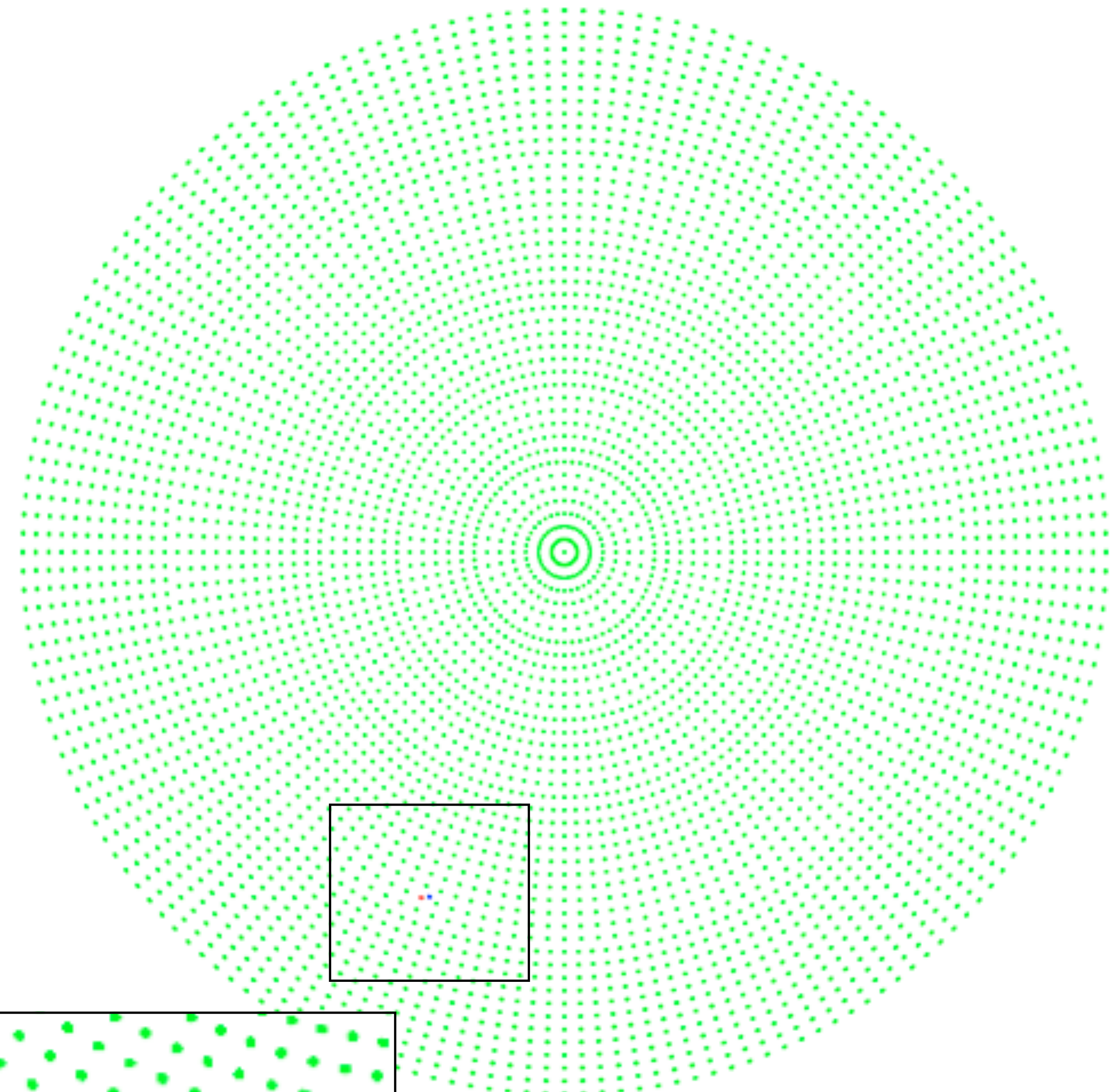
Sun Matrix

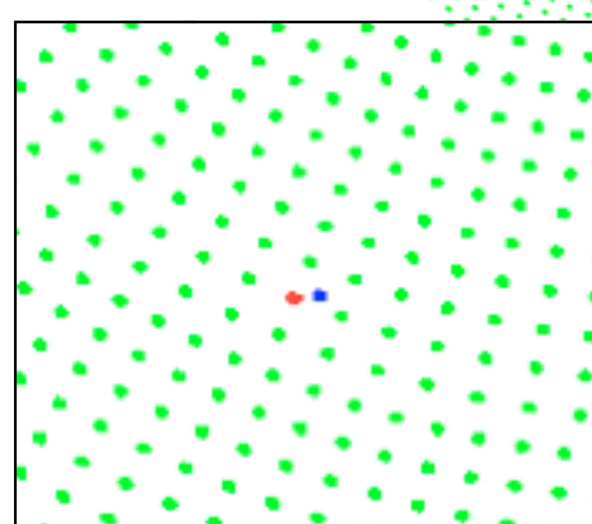
The sun matrix used in the third term of the 5-phase equation.

gendaymtx has a secret option for creating the sunmatrix:

```
gendaymtx -m 6 -5 -of city.wea \
   > city_direct_m6.smx
```

- Closest sun position is used per timestep (one position instead of three patches).
- A factor is included to compensate for the solid angle of reinhart patch vs. angular source (-5 option)





blue - actual timestep sun position
red - sun position used in calculation
difference = 1.15°

$I_{5ph} = VTDS - V_dTD_dS_d + C_{ds}S_{sun}$

Direct Sun Coefficient Matrix - sensor points

Fenestration Model:

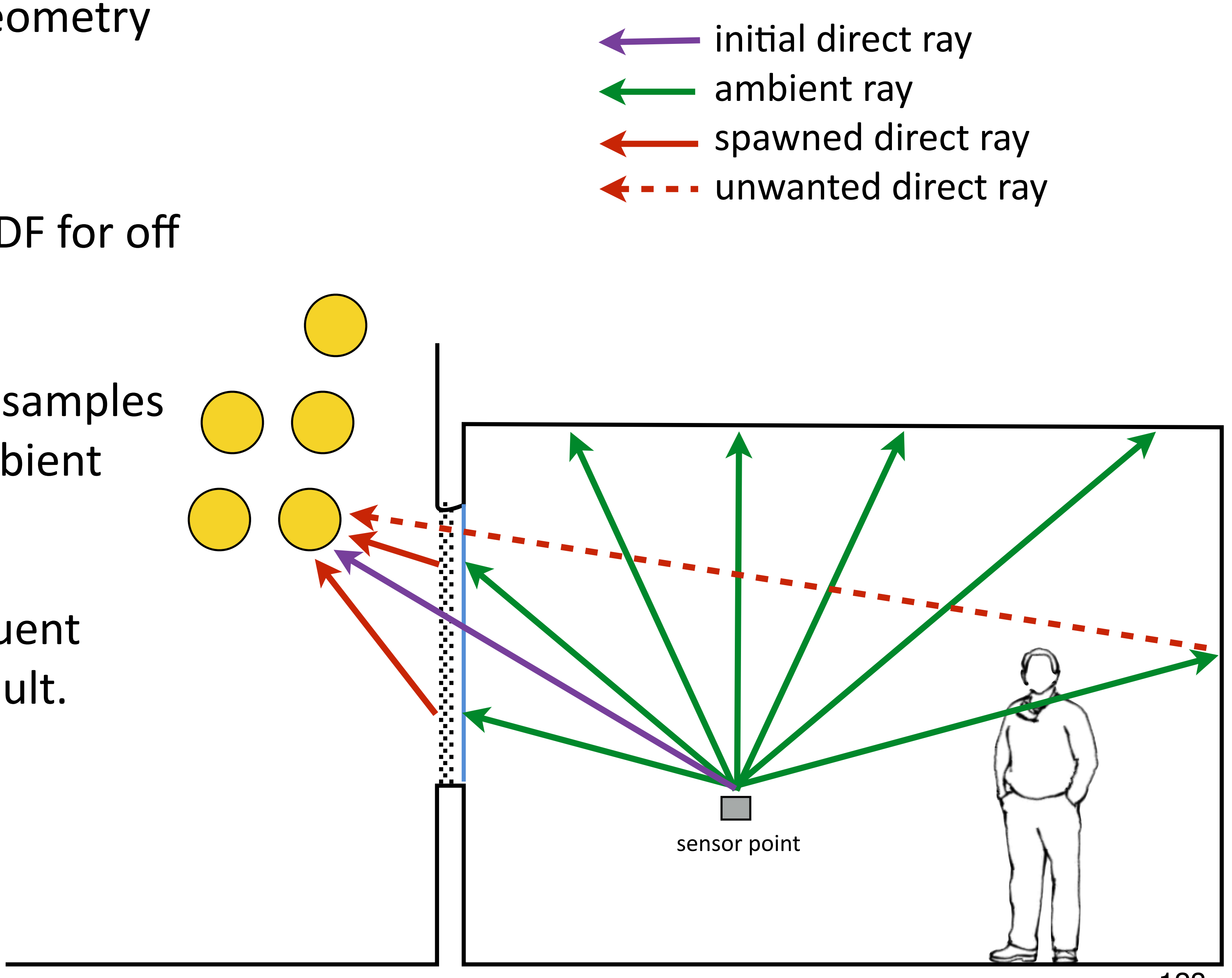
- Klems BSDF + Proxied geometry
- Tensor Tree BSDF + Proxied geometry
- Tensor Tree BSDF w/o proxied geometry

All Black Materials:

We want ambient rays to sample the BSDF for off angle transmission: **-ab 1**

When an ambient sample is sent, direct samples are sent from the termination of the ambient rays: **we don't want this behavior**

All-black geometry prevents the subsequent direct sample rays from affecting the result.



$L_{5ph} = VTDS - V_dTD_dS_d + C_{ds}S_{sun}$

Direct Sun Coefficient Matrix - renderings

Preventing unwanted rays gets more complex with renderings - a black model produces a black luminance rendering.

Step 1 - generate a illuminance coefficient rendering with black model:

vwrays | rcontrib -i

Step 2 - generate a material reflectance map - each pixel is equal to the material reflectance divided by pi:

rpict -av 0.31831 0.31831 0.31831

Step 3 - multiply step 1 times step 2.

This workaround assumes all reflectances are lambertian

This is terribly awkward. I'm still looking for a better way...

$I_{5ph} = VTDS - V_d T D_d S_d + C_{ds} S_{sun}$

Putting it all together

dctimestep view.vmx T.xml daylight.dmx city.smx > 1term.dat
dctimestep view_direct.vmx T.xml daylight_direct.dmx city_direct.smx > 2term.dat
dctimestep suncoefficient.mtx city_ds.smx > 3term.dat

rmtxop 1term.dat + -s -1 2term.dat + 3term.dat

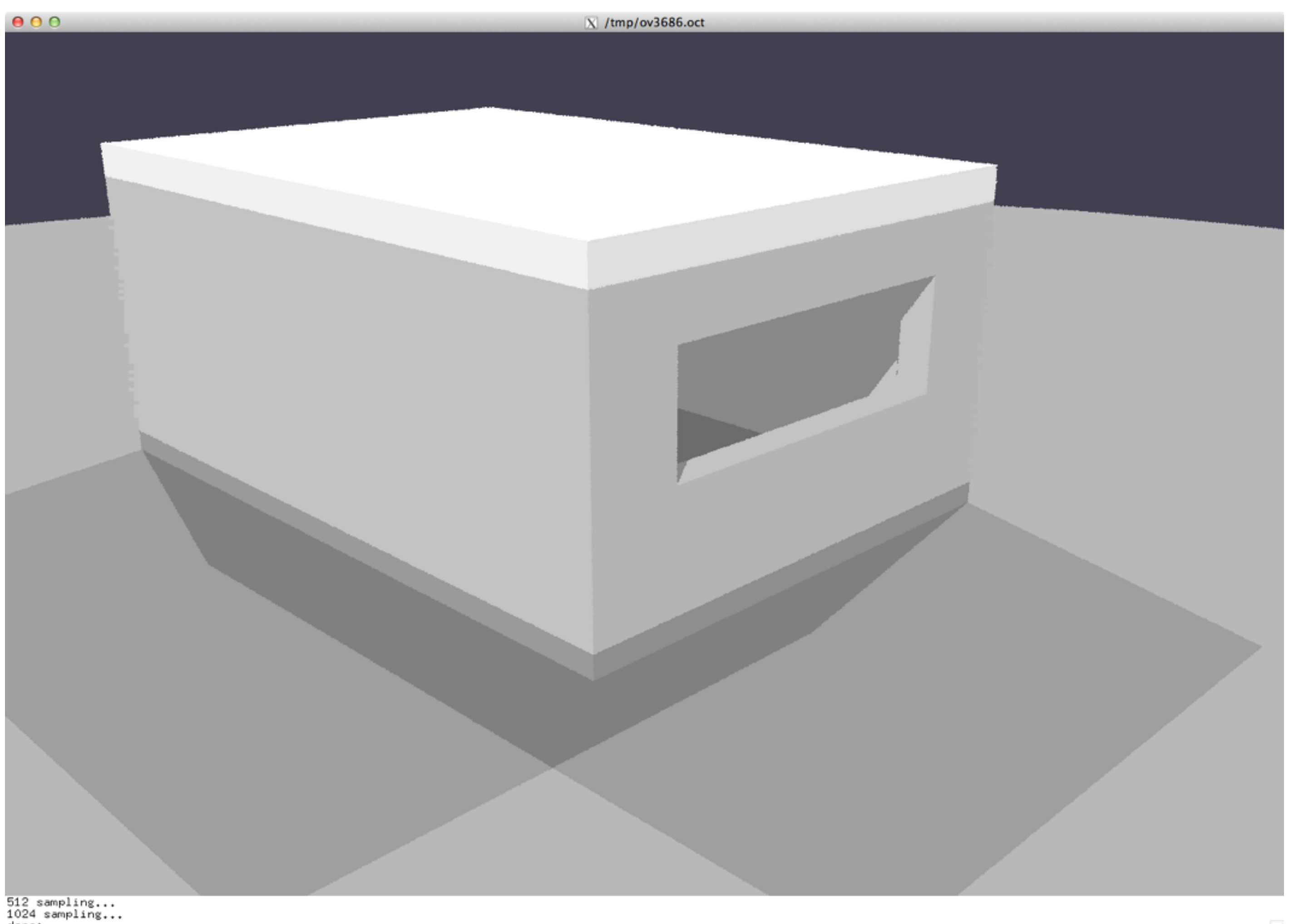
<u>before rmtxop:</u>

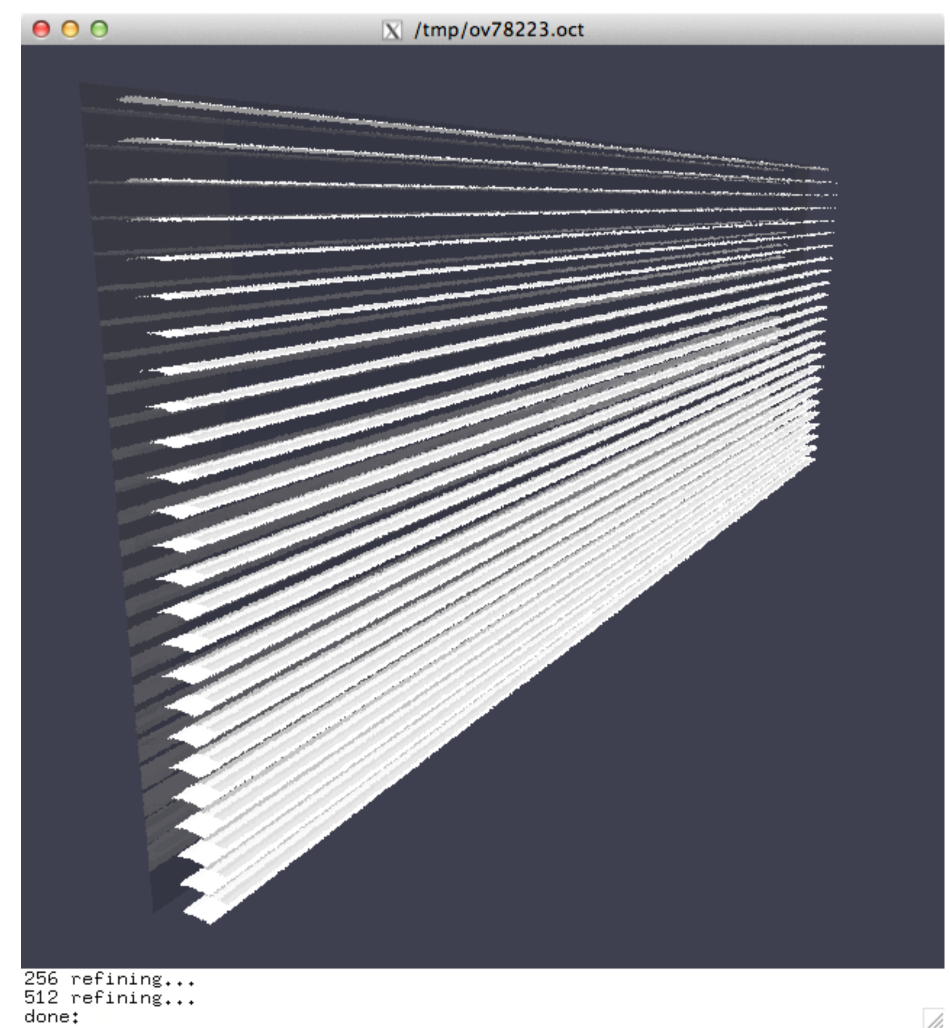
```
rlam -if3 1term.dat 2term.dat 3term.dat | \
    rcalc -if9 -e 'r=$1-$4+$7;g=$2-$5+$8;b=$3-$6+$9' \
    -e '$1=179*(.265*r+.670*g+.065*b)' | \
    awk '{printf("%f\t",$1);if(NR%8760==0) printf("\n")}' > illum.txt
```

An Example

Using the model from Axel’s tutorial

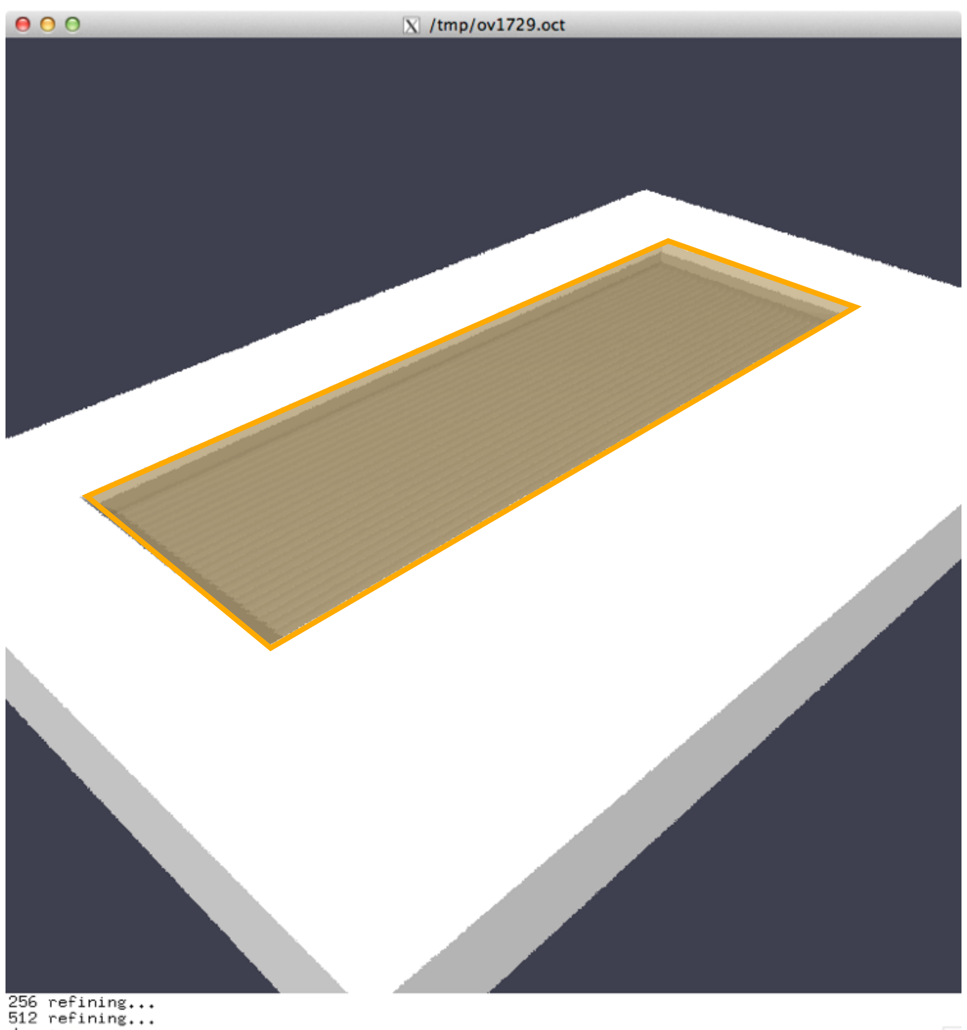
Fenestration: clear glazing with venetian blinds.

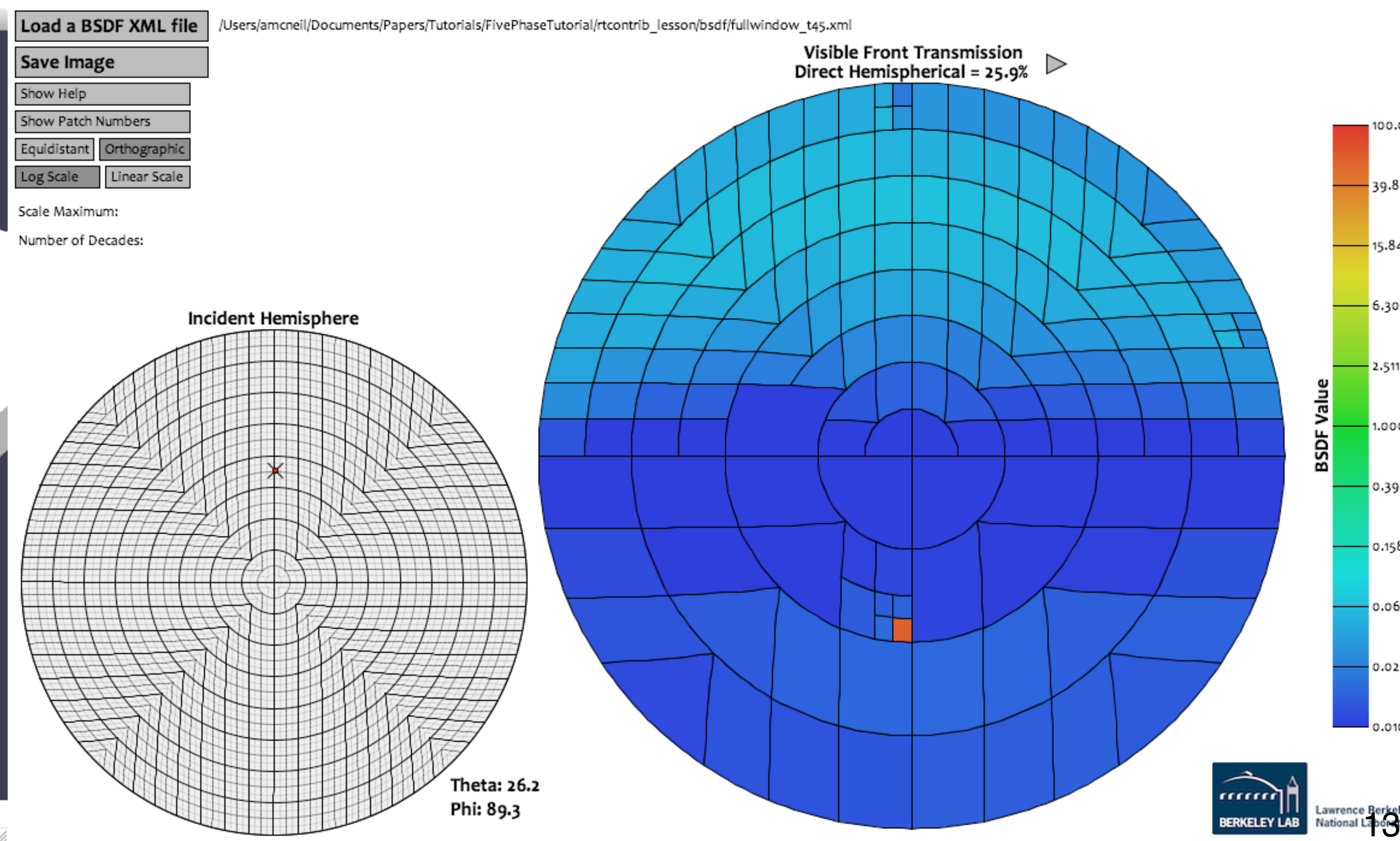




Creating BSDFs

genBSDF +f +b -geom meter **-dim 0.5 3.5 1 2 -.3 0** -t4 5 \
    bsdf/fullwindow.rad > bsdf/fullwindow_t45.xml



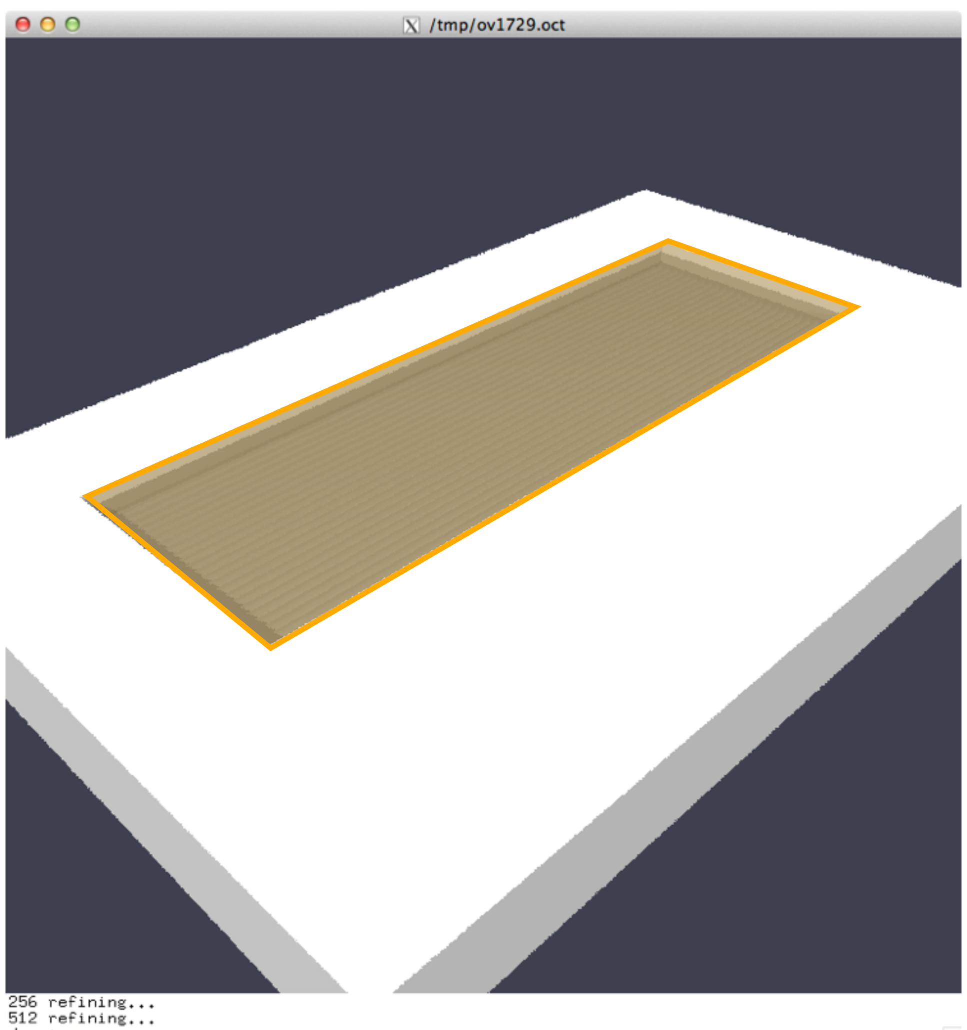


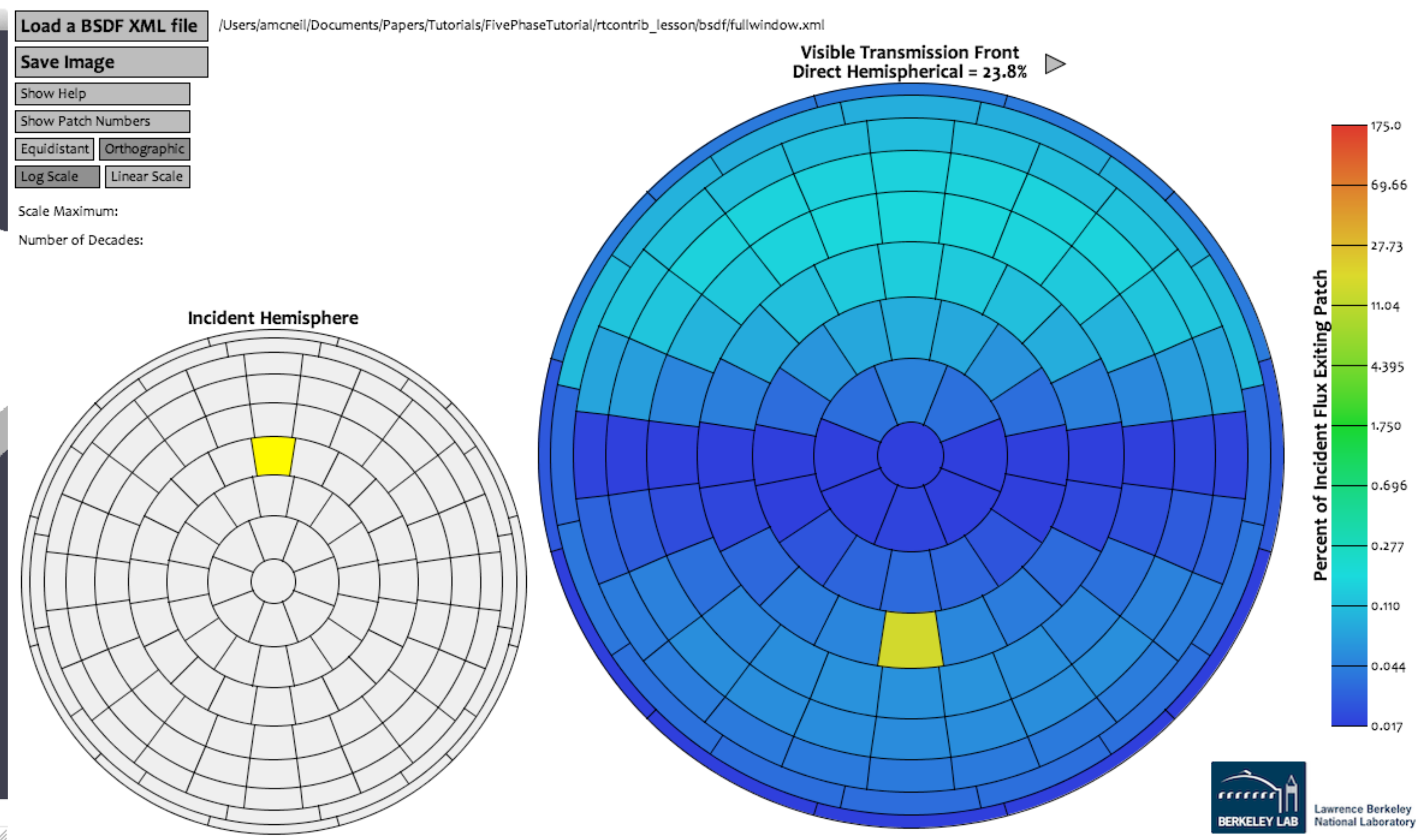
Creating BSDFs

bsdf2klems bsdf/fullwindow_t45.xml > bsdf/fullwindow_klems.xml

or

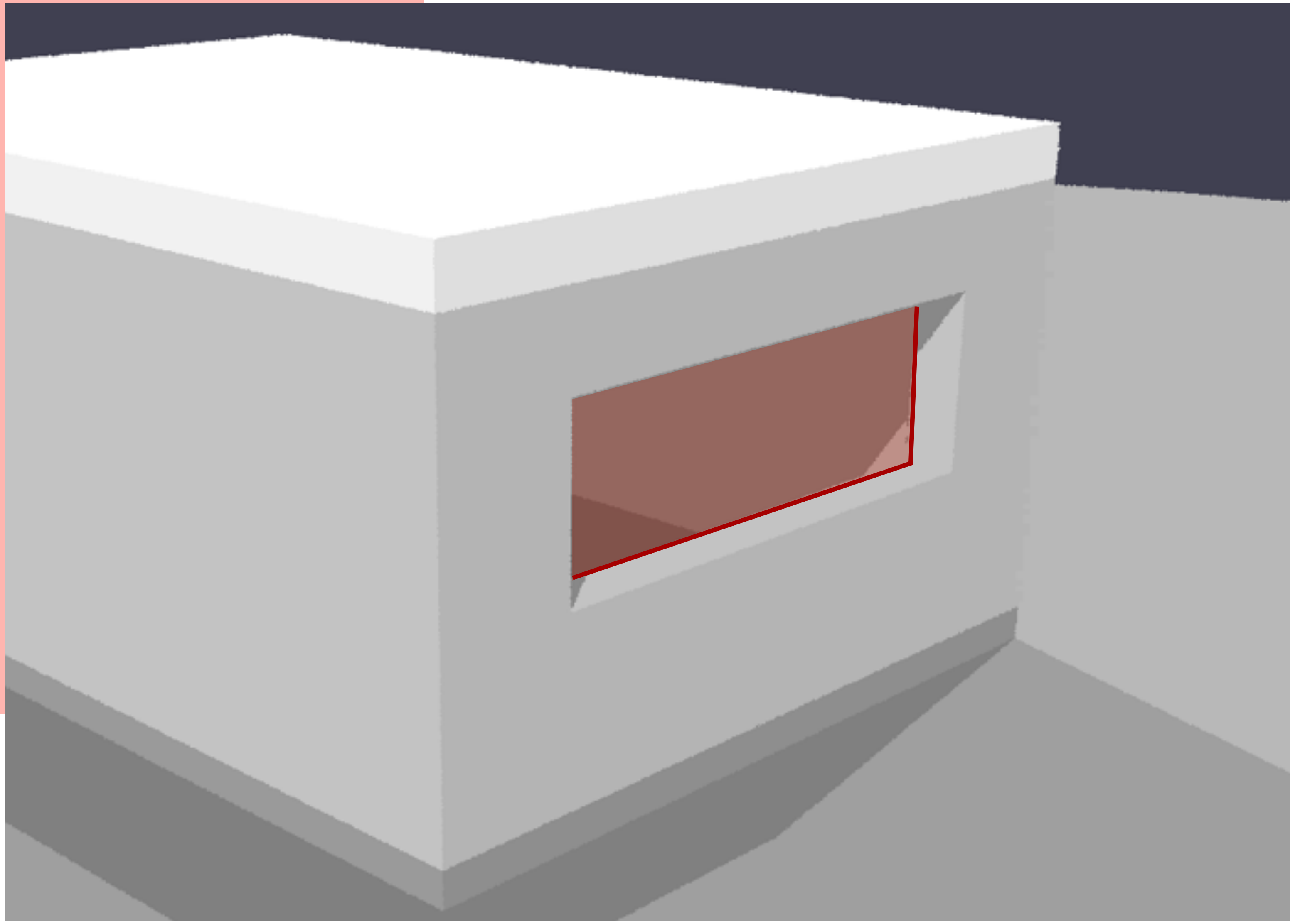
genBSDF -n 4 +f +b -geom meter **-dim 0.5 3.5 1 2 -.3 0** \
    bsdf/fullwindow.rad > bsdf/fullwindow_klems.xml





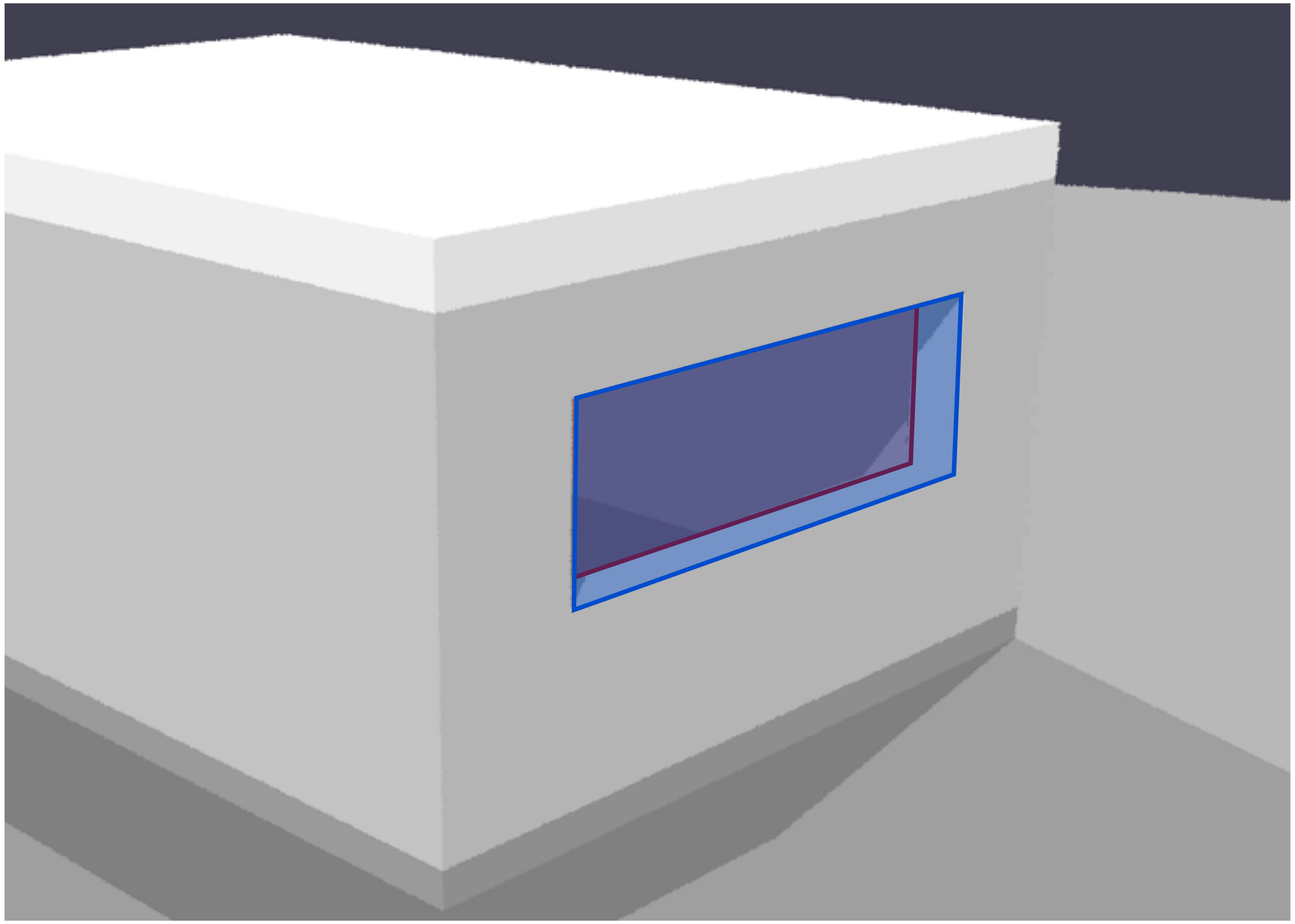
Create Sending / Receiving Surfaces

```
#@rfluxmtx h=kf u=Z o=vmx/w_%03d.hdr

void glow viewsurf
0
0
4 1 1 1 0

viewsurf polygon inside
0
0
12    0.5   0   1
      0.5   0   2
      3.5   0   2
      3.5   0   1
```



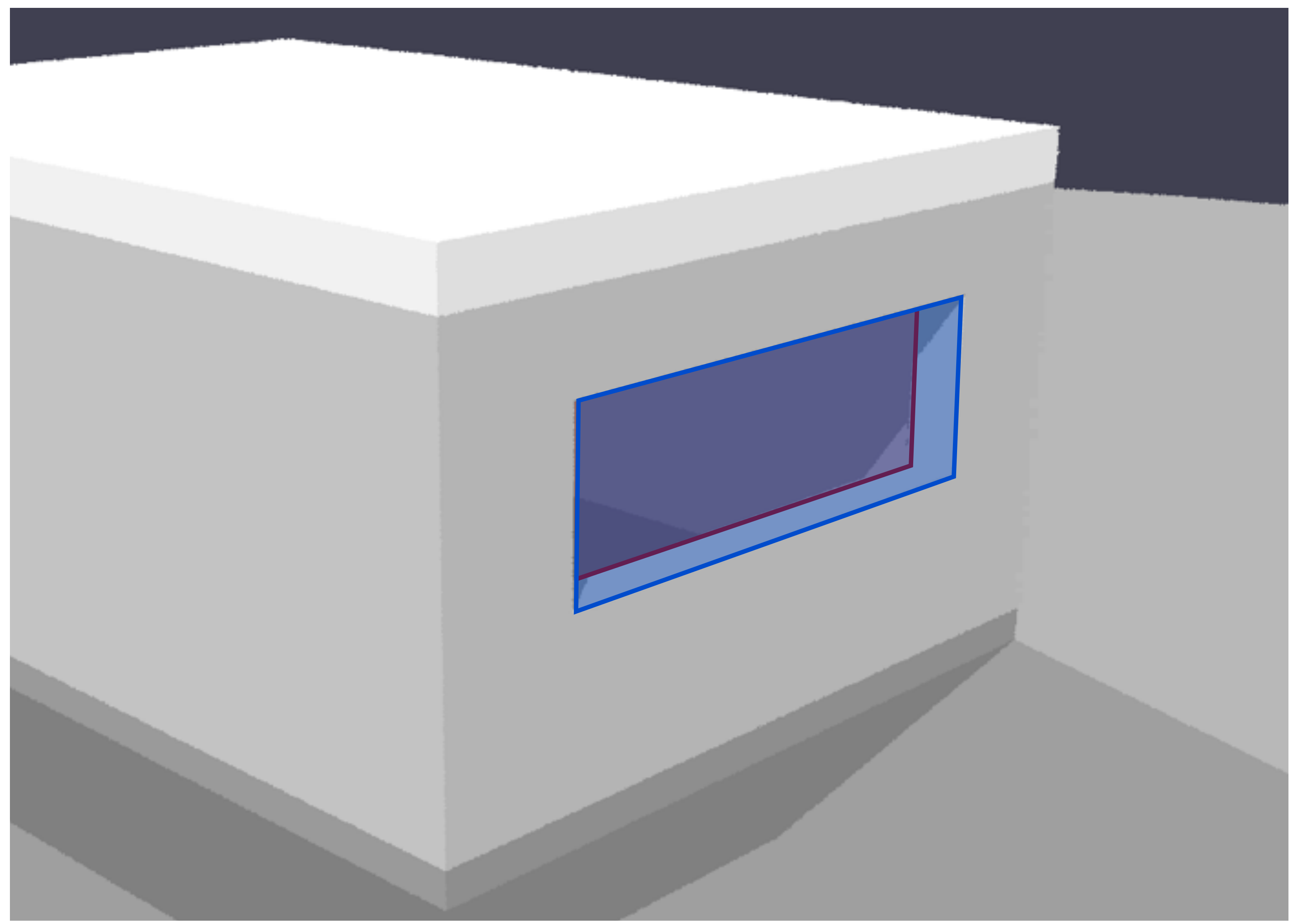
Create Sending / Receiving Surfaces

```
#@rfluxmtx h=kf u=Z

void glow daymtxsurf
0
0
4 1 1 1 0

daymtxsurf polygon outside
0
0
12    0.5   0   1
      0.5   0   2
      3.5   0   2
      3.5   0   1
```



Create Sending / Receiving Surfaces

```
#@rfluxmtx h=u u=Y
void glow ground_glow
0
0
4       1 1 1 0

ground_glow source ground
0
0
4       0 0 -1  180

#@rfluxmtx h=r1 u=Y
void glow sky_glow
0
0
4       1 1 1 0

sky_glow source sky
0
0
4       0 0 1  180
```



Create Fenestration model with BSDF

```
void BSDF BSDFproxy
6 0.24 bsdf/fullwindow_t45.xml 0 0 1 .
0
0

BSDFproxy polygon inside
0
0
12    0.5   0   1
      0.5   0   2
      3.5   0   2
      3.5   0   1
```

Generating View Matrices (Sensor points)

View Matrix:

```
rfluxmtx -I+ -ab 10 -ad 65536 -lw 1.52e-5 -y 6 \
    < data/photocells.pts - objects/viewmtxsurf.rad \
    materials/testroom.mat objects/testroom_Swall.rad \
    objects/testroom.rad objects/ground.rad\
    > matrices/viewmatrix.vmx
```

Direct View Matrix:

```
xform -m black objects/testroom_Swall.rad objects/testroom.rad objects/
ground.rad > testroom_black.rad

rfluxmtx -I+ -ab 1 -ad 65536 -lw 1.52e-5 -y 6 \
    < data/photocells.pts - objects/viewmtxsurf.rad \
    materials/testroom.mat testroom_black.rad \
    > matrices/viewmatrix_direct.vmx
```

Generating Daylight Matrices

Daylight Matrix:

```
rfluxmtx -c 1000 -ab 2 -ad 1024  objects/daymtxsurf.rad skies/sky.rad \
    materials/testroom.mat objects/testroom_Swall.rad \
    objects/testroom.rad objects/ground.rad > matrices/daylightmatrix.dmx
```

Direct Daylight Matrix:

```
rfluxmtx -c 1000 -ab 0 -ad 1024 \
    objects/daymtxsurf.rad skies/sky.rad \
    materials/testroom.mat testroom_black.rad \
     > matrices/daylightmatrix_direct.dmx
```

Generating Sun Coefficient Matrix
(Sensor points)

First the suns:

```
echo void light solar 0 0 3 1e6 1e6 1e6 > skies/suns.rad

cnt 5185  |  rcalc -e MF:6 -f reinsrc.cal -e Rbin=recno \
      -o 'solar source sun 0 0 4 ${ Dx } ${ Dy } ${ Dz } 0.533' >> skies/suns.rad
```

Then the octree:

```
xform -m black objects/testroom_Swall.rad objects/testroom.rad objects/ground.rad \
      | oconv materials/testroom.mat - objects/glazing.rad objects/venetianblind.rad \
          objects/glazing_bsdf.rad skies/suns.rad > octs/model_suns.oct
```

Finally the Coefficient Matrix:

```
rtcontrib < data/photocells.pts -I -ab 1 -ad 65536 -lw 1.52e-5  -dc 1 -dt 0 -dj 1 -st 1 -ss 0 -faf  \
      -e MF:6 -f reinhart.cal -b rbin -bn Nrbins -m solar \
      octs/model_suns.oct > matrices/directsun.dsmx
```

Generating Sky Matrices

First convert an epw file to wea:

```
epw2wea skies/USA_CA_Oakland.Intl.AP.724930_TMY3.epw skies/OakLand.wea
```

Then the three sky matrices:

Normal sky matrix:

```
gendaymtx -of skies/OakLand.wea > matrices/OakLand.smx
```

Direct only sky matrix:

```
gendaymtx -of -d skies/OakLand.wea > matrices/OakLand_direct.smx
```

Direct sun sky matrix

```
gendaymtx -5 -d -m 6 -of skies/OakLand.wea > matrices/OakLand_direct_m6.smx
```

Putting it together (Sensor Points)

First term:

```
rmtxop matrices/viewmatrix.vmx bsdf/fullwindow.xml \
    matrices/daylightmatrix.dmx matrices/OakLand.smx > i_3ph.txt
```

Second term:

```
rmtxop matrices/viewmatrix_direct.vmx bsdf/fullwindow.xml \
    matrices/daylightmatrix_direct.dmx matrices/OakLand_direct.smx > i_ds3ph.txt
```

Third term:

```
rmtxop matrices/directsun.dsmx matrices/OakLand_direct_m6.smx \
   > i_ds5ph.txt
```

Adding the terms:

```
rmtxop i_3ph.txt + -s -1 i_ds3ph.txt + i_ds5ph.txt \
    | rmtxop -fa -c 47.4 119.9 11.6 -t - > E5ph.txt
```

Putting it together (Sensor Points)

Creating unformatted binary data files:

```
tr -s '\t\r\n ' '\n' < i_3ph.txt | rcalc -of -e '$1=$1' > i_3ph.dat
tr -s '\t\r\n ' '\n' < i_ds3ph.txt | rcalc -of -e '$1=$1' > i_ds3ph.dat
tr -s '\t\r\n ' '\n' < i_ds5ph.txt | rcalc -of -e '$1=$1' > i_ds5ph.dat
```

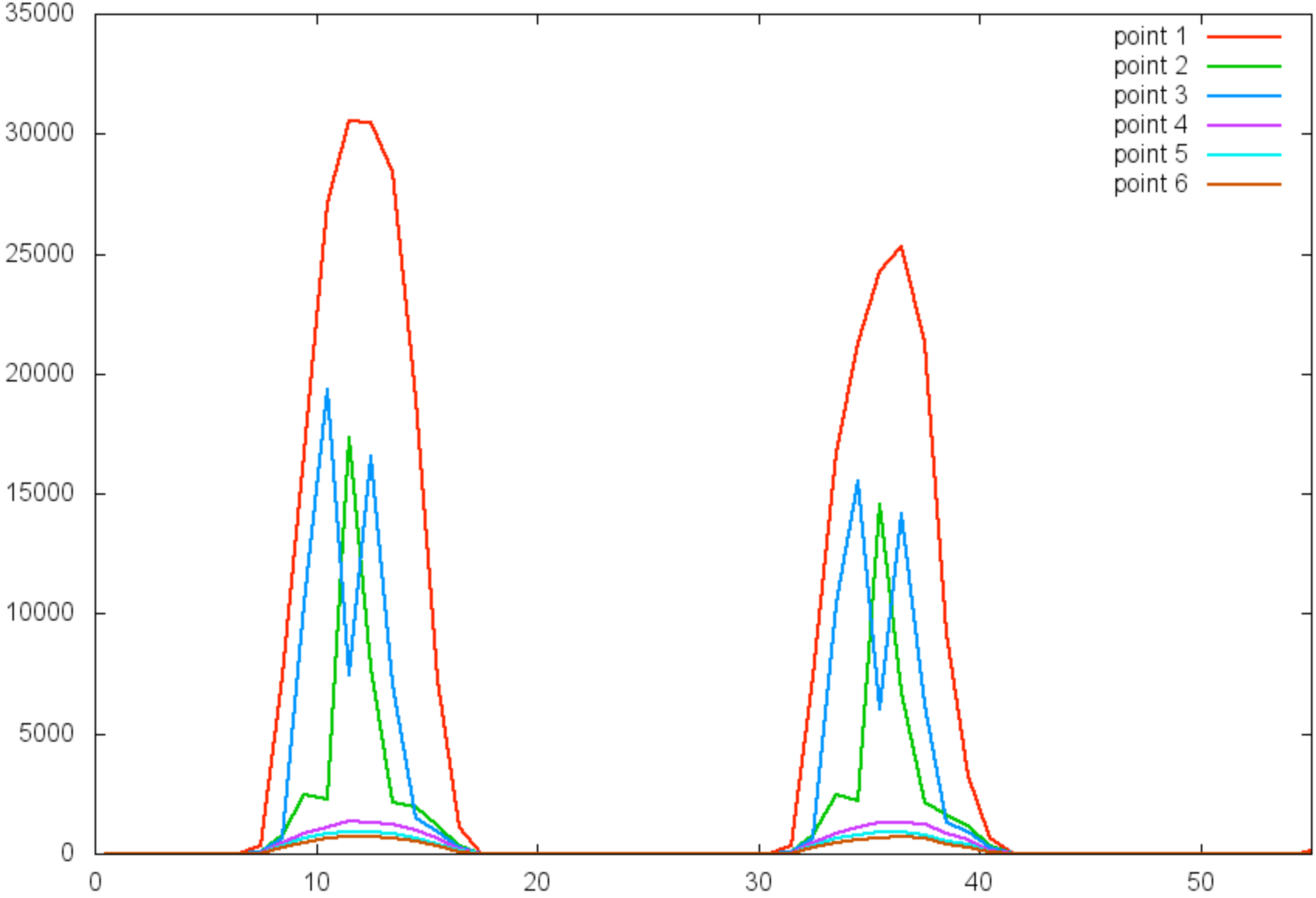
Combining the files and doing the calculation:

```
rlam -if3 i_3ph.dat i_ds3ph.dat i_ds5ph.dat | \
    rcalc -if9 -e 'r=$1-$4+$7;g=$2-$5+$8;b=$3-$6+$9' \
    -e '$1=179*(.265*r+.670*g+.065*b)' | \
    awk '{printf("%f\t",$1);if(NR%8760==0) printf("\n")}'  > illum.txt
```

Adding the terms:

```
rmtxop i_3ph.txt + -s -1 i_ds3ph.txt + i_ds5ph.txt \
    | rmtxop -fa -c 47.4 119.9 11.6 -t - > E5ph.txt
```

Finally, an illuminance result



Generating View Matrices - Renderings

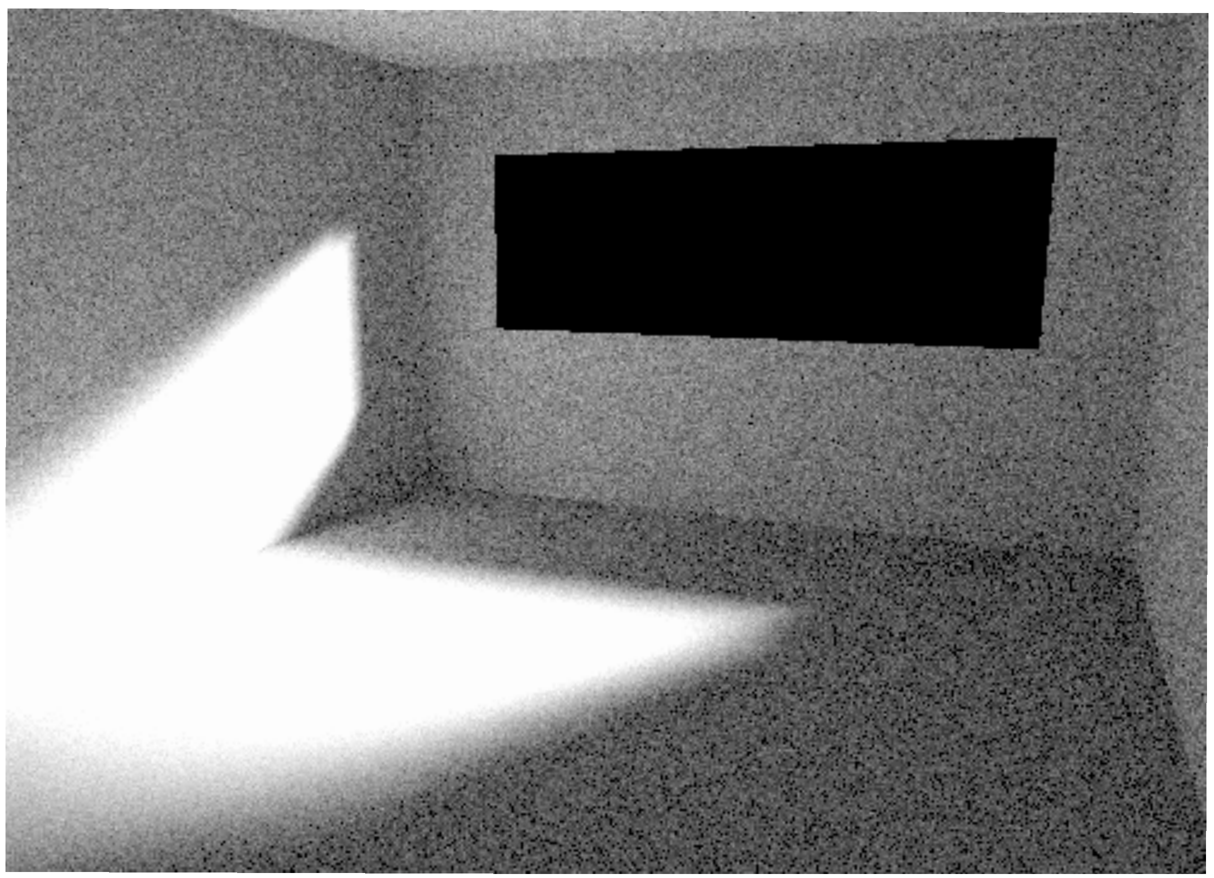
View Matrix:

```
vwrays -vf views/back.vf -ff -x 600 -y 600 \
    | rfluxmtx `vwrays -vf views/back.vf  -x 600 -y 600 -d` \
        -ffc -ab 10 -ad 65536 -lw 1.52e-5 - objects/viewmtxsurf.rad \
       materials/testroom.mat objects/testroom_Swall.rad \
       objects/testroom.rad objects/ground.rad
```

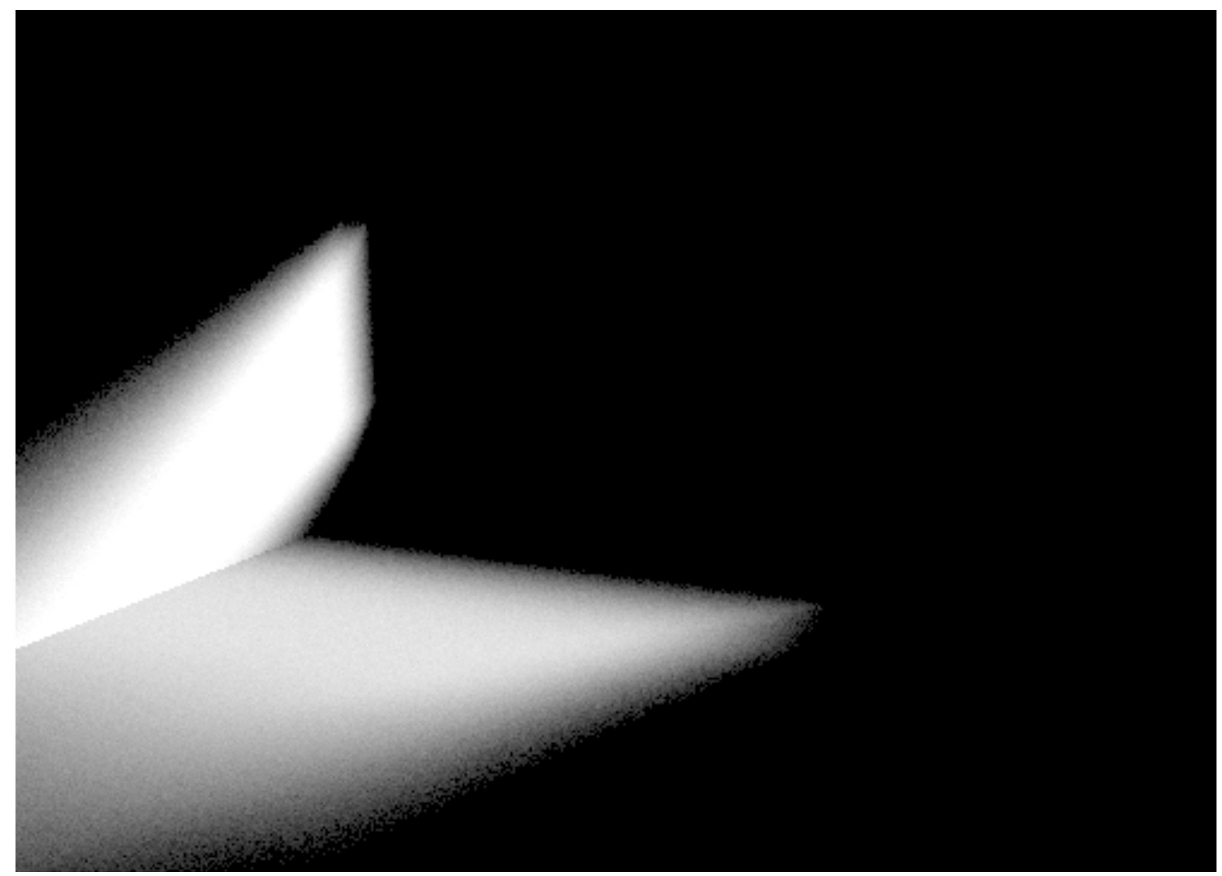
Direct View Matrix:

```
vwrays -vf views/back.vf -ff -x 600 -y 600 \
    | rfluxmtx `vwrays -vf views/back.vf  -x 600 -y 600 -d` \
        -ffc -ab 1 -ad 65536 -lw 1.52e-5 - objects/viewmtxsurf.rad \
        materials/testroom.mat objects/testroom_Swall.rad \
        objects/testroom.rad objects/ground.rad
```

Rendered View Matrix - example



View Matrix



Direct View Matrix

Generating Sun Coefficient Matrix - Renderings

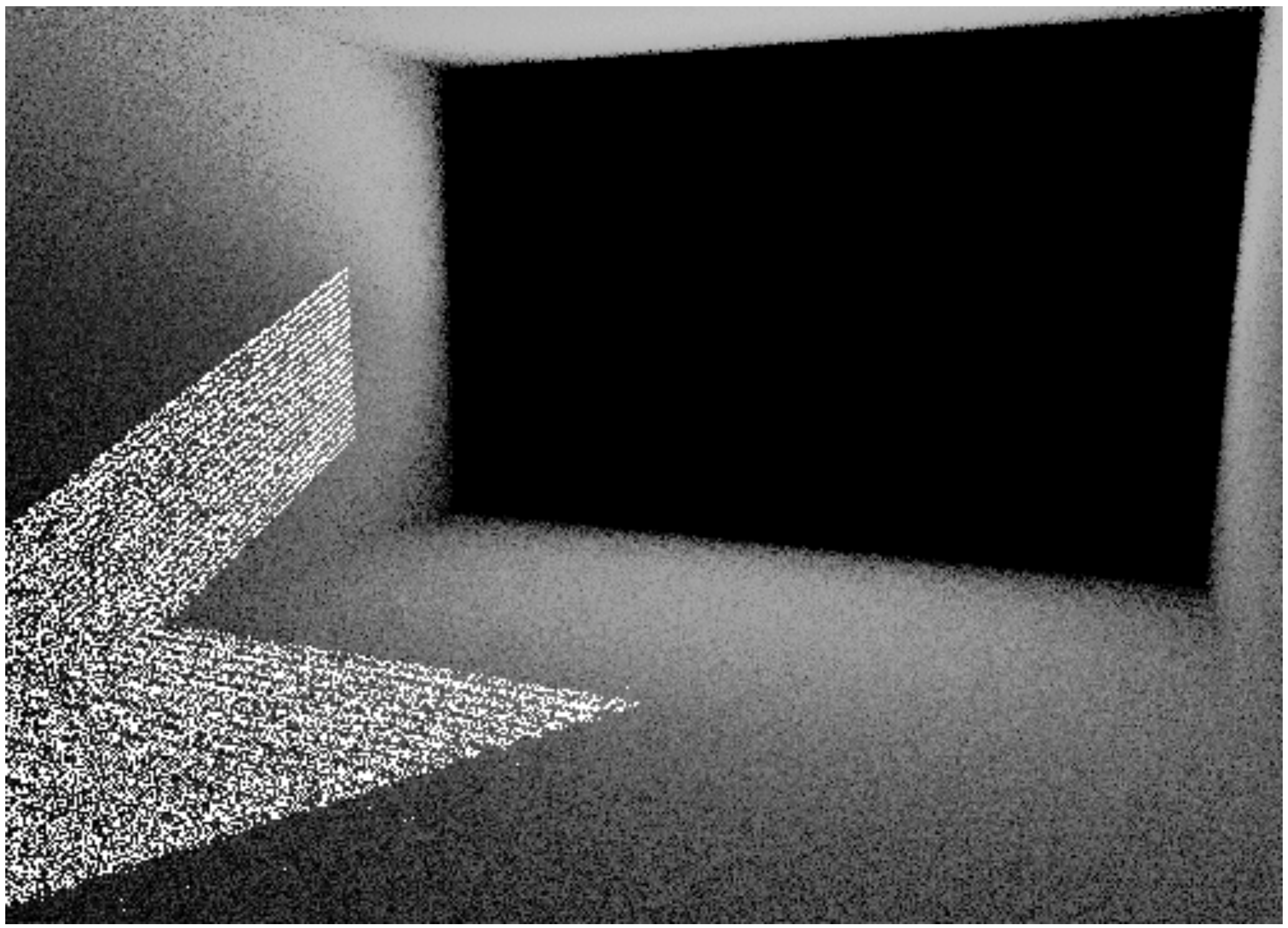
First the coefficient renderings:

```
ulimit -n 9999
vwrays -ff -vf views/back.vf -x 600 -y 600 \
    | rcontrib `vwrays -vf views/back.vf -x 600 -y 600 -d` -ffc -fo -o viewpics_ds/back_%04d.hdr \
        -e MF:6 -f reinhart.cal -b rbin -bn Nrbins -m solar -i -ab 1 -ad 1000 -lw 1e-3 \
        octs/model_suns.oct
```

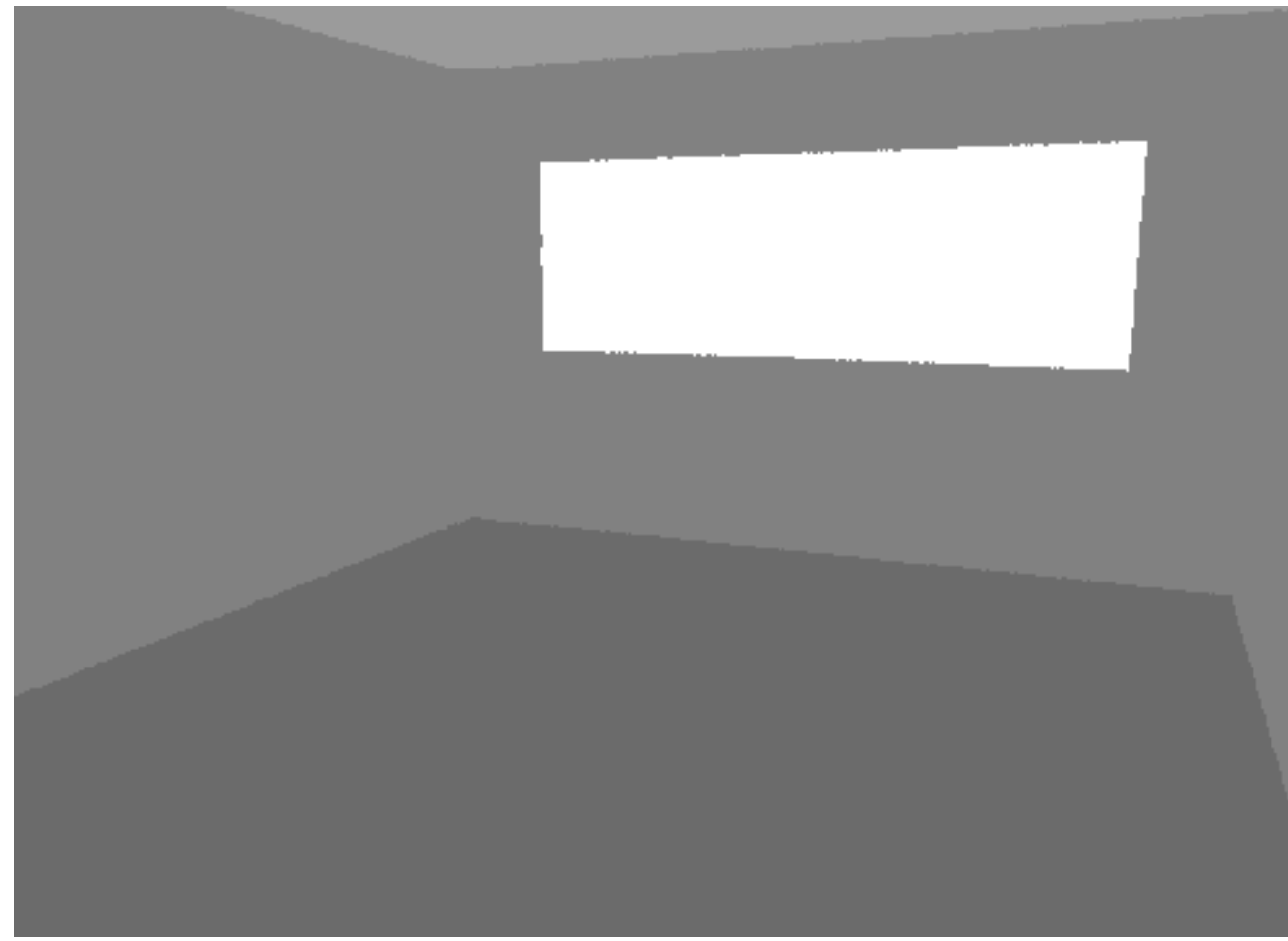
Then the material map rendering:

```
rpict -x 600 -y 600 -vf views/back.vf -av 0.31831 0.31831 0.31831 -aa 0 octs/model_3ph.oct \
    > materialmap.hdr
```

Sun Coefficient Rendering Example



Sun Coefficient Rendering



Material Map Rendering

Putting it all together (Renderings)

First term:

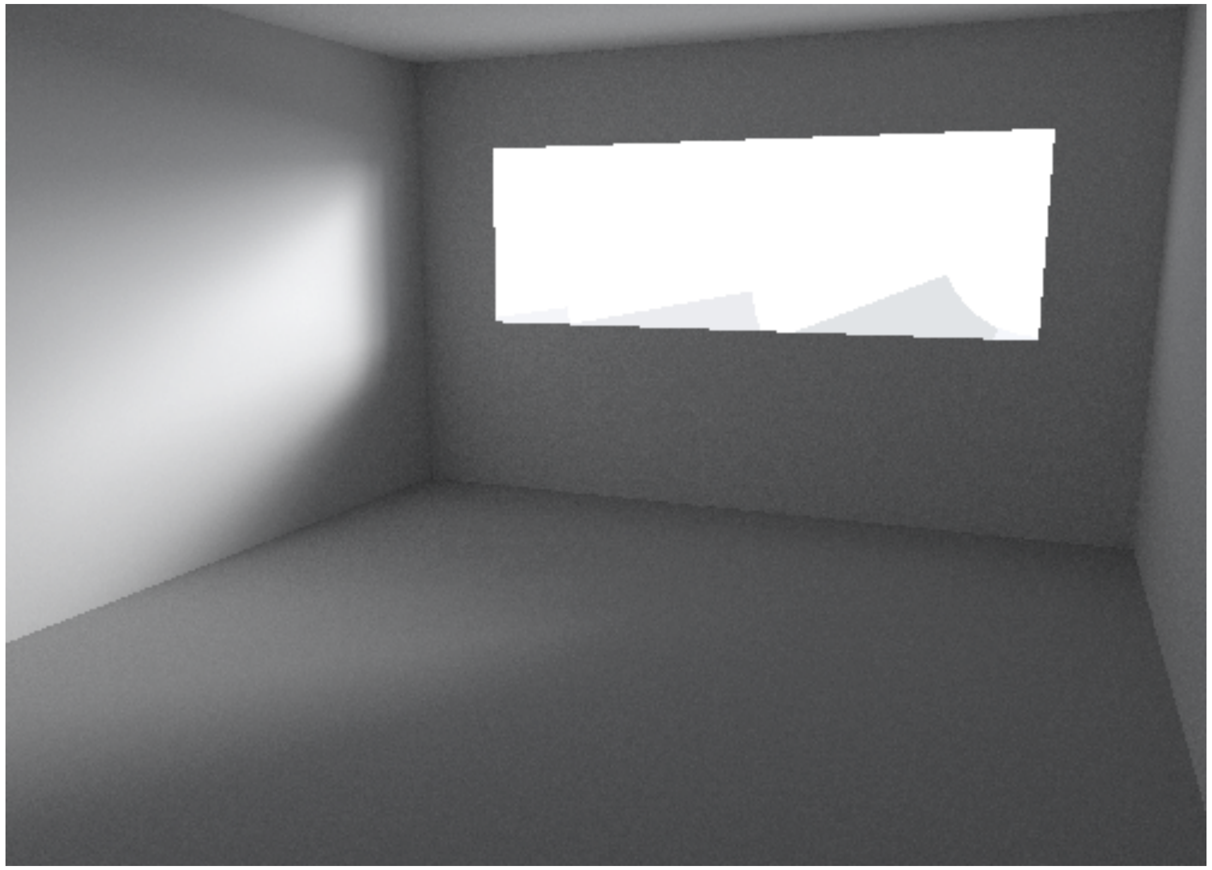
```
dctimestep -n 8760 -if -o hourlypics/back_%04d.hdr viewpics/back_%03d.hdr \
    bsdf/fullwindow.xml matrices/daylightmatrix.dmx matrices/OakLand.smx
```

Second term:

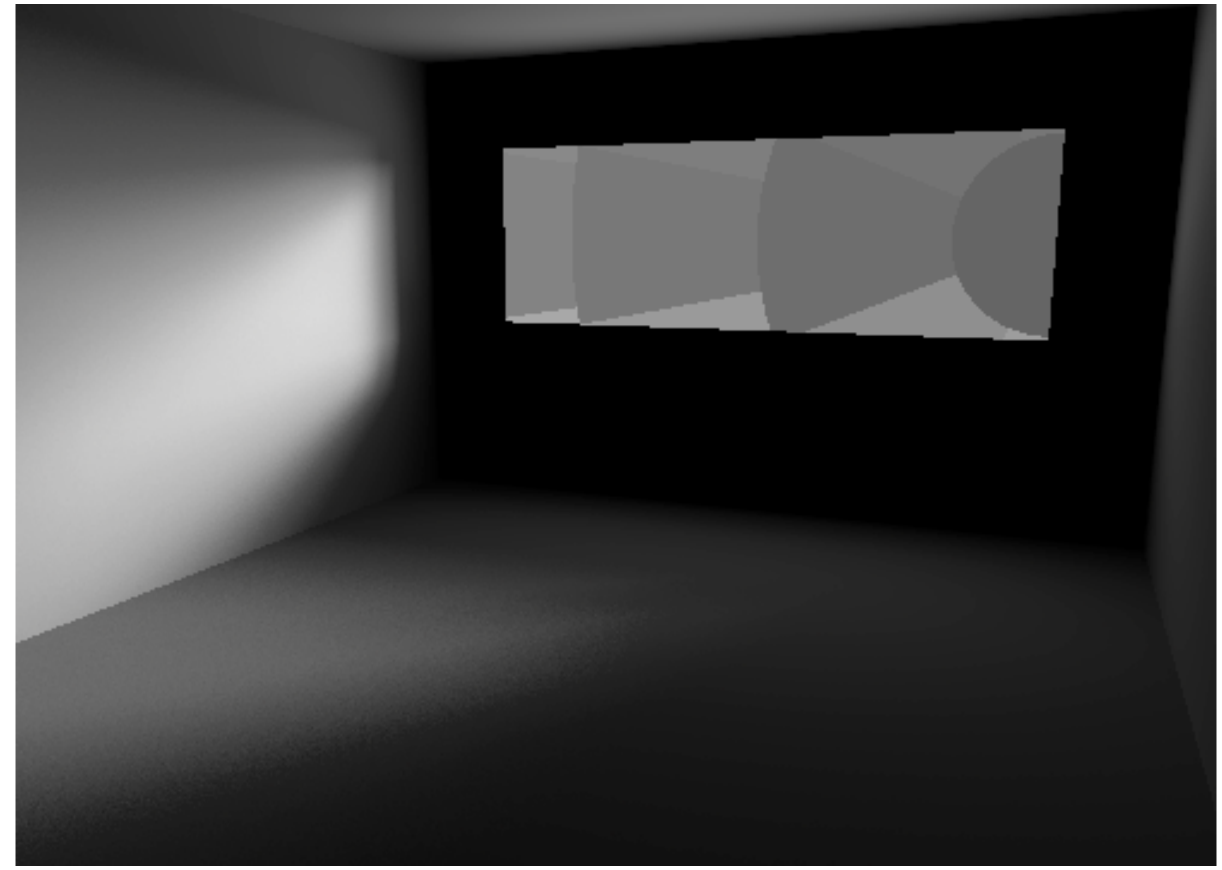
```
dctimestep -n 8760 -if -o hourlypics_dir/back_%04d.hdr viewpics_dir/back_%03d.hdr \
    bsdf/fullwindow.xml matrices/daylightmatrix_direct.dmx matrices/OakLand_direct.smx
```

Third term:

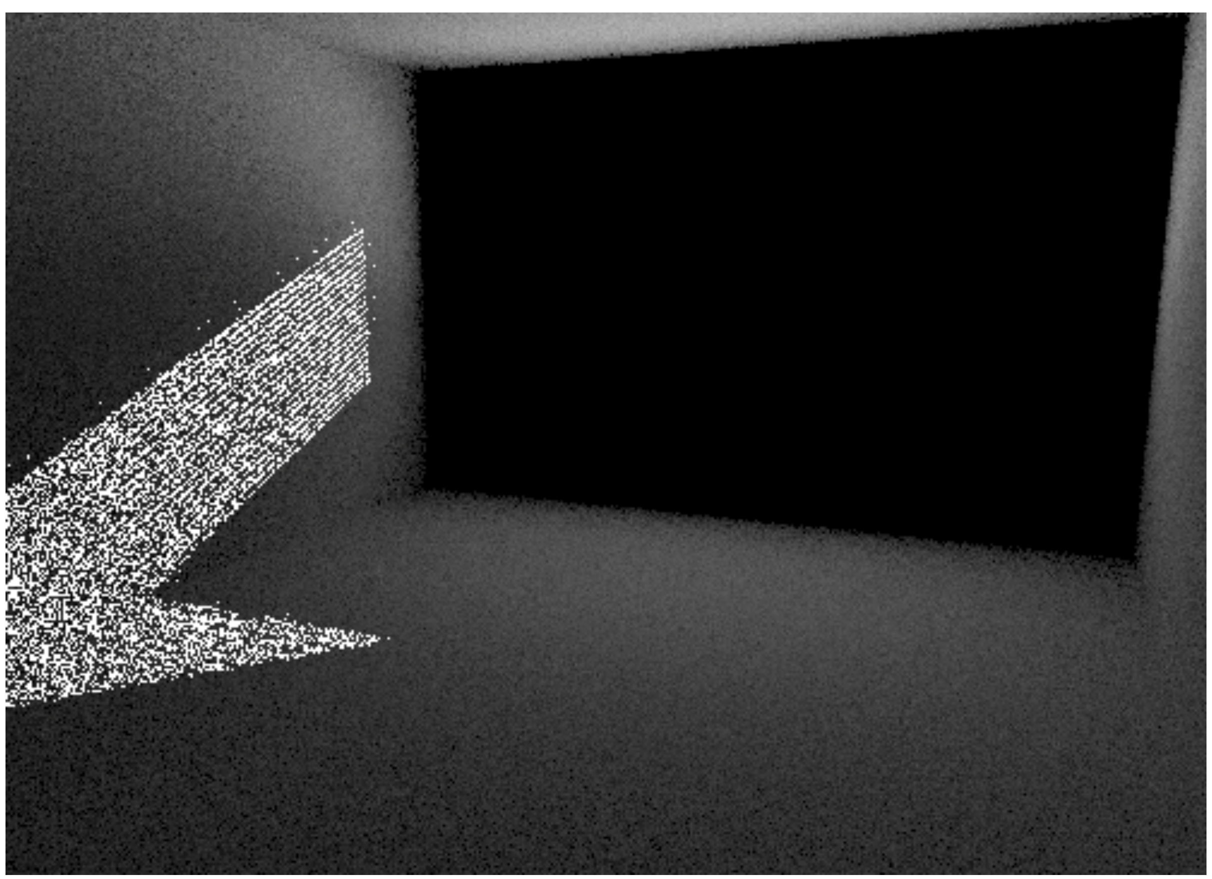
```
dctimestep -n 8760 -if -o hourlypics_ds/back_%04d.hdr viewpics_ds/back_%04d.hdr \
    matrices/OakLand_direct_m6.smx
```



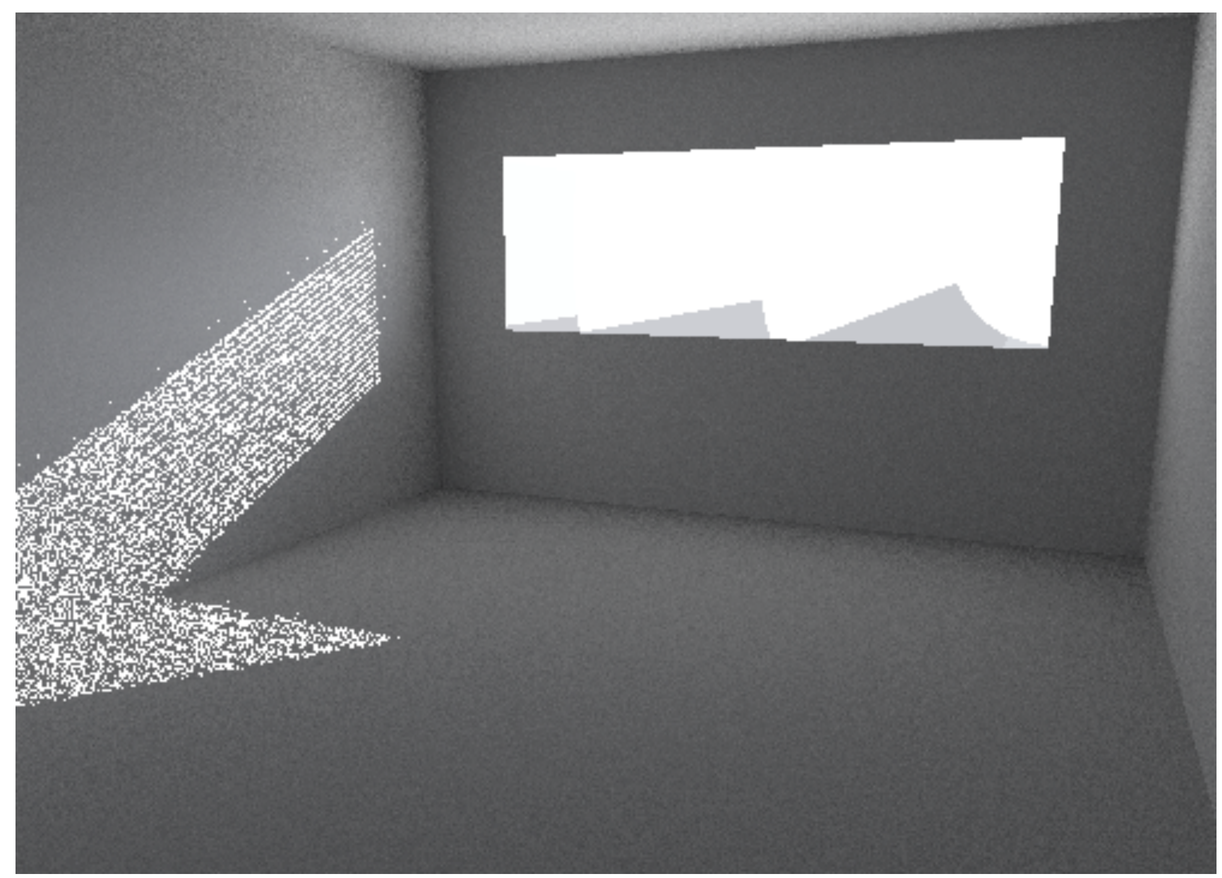
First Term (three-phase result)



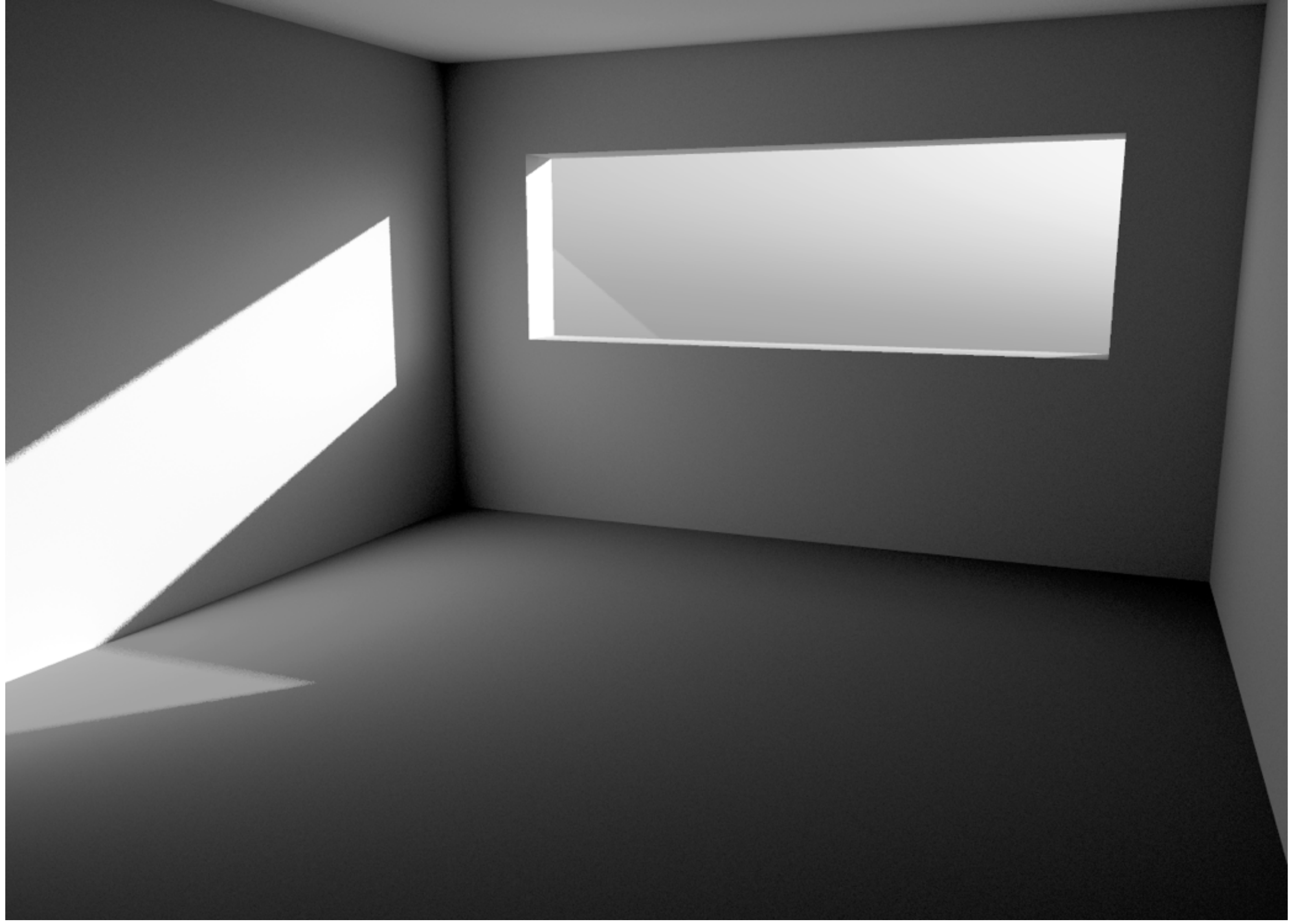
Second Term

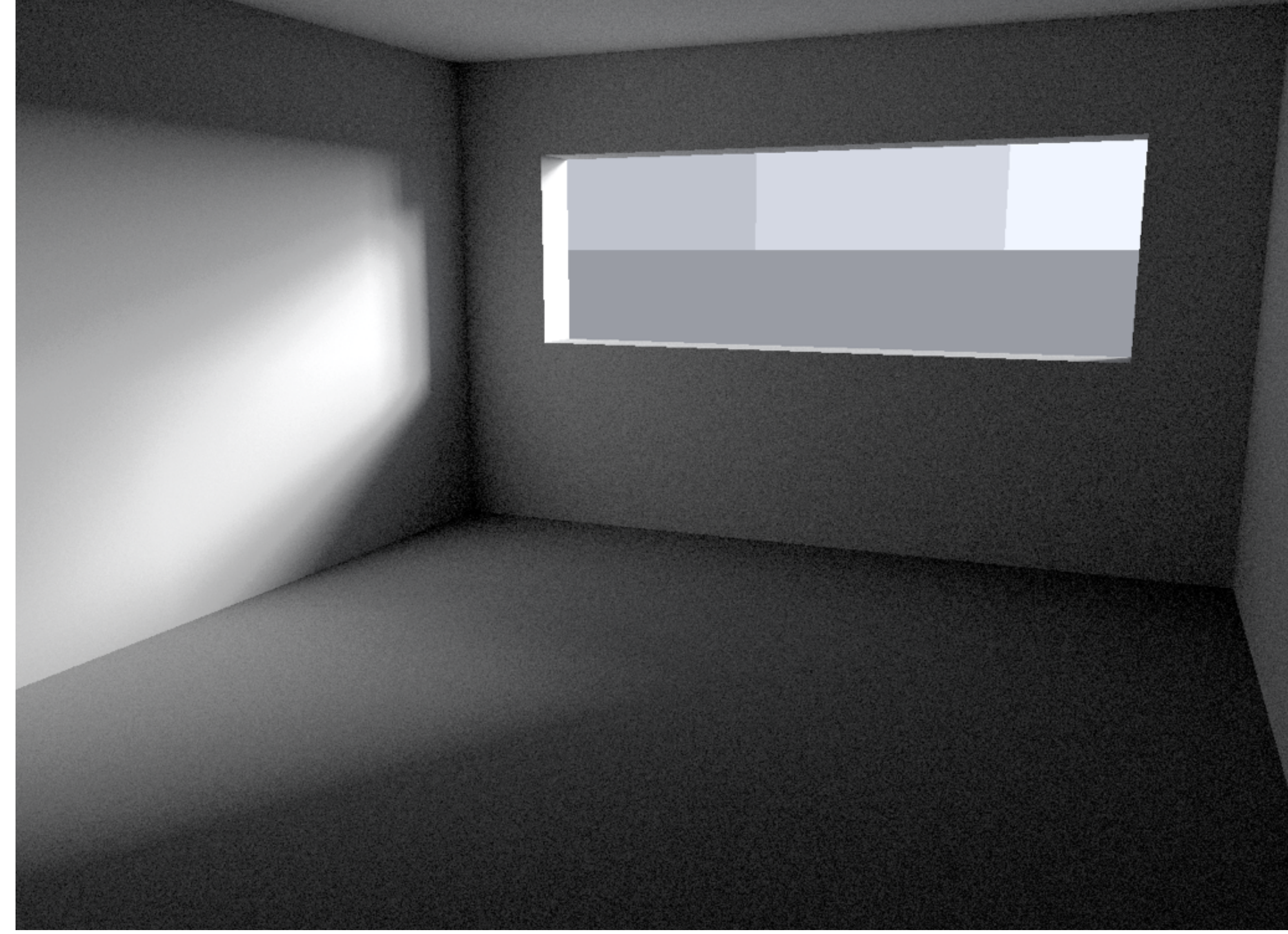


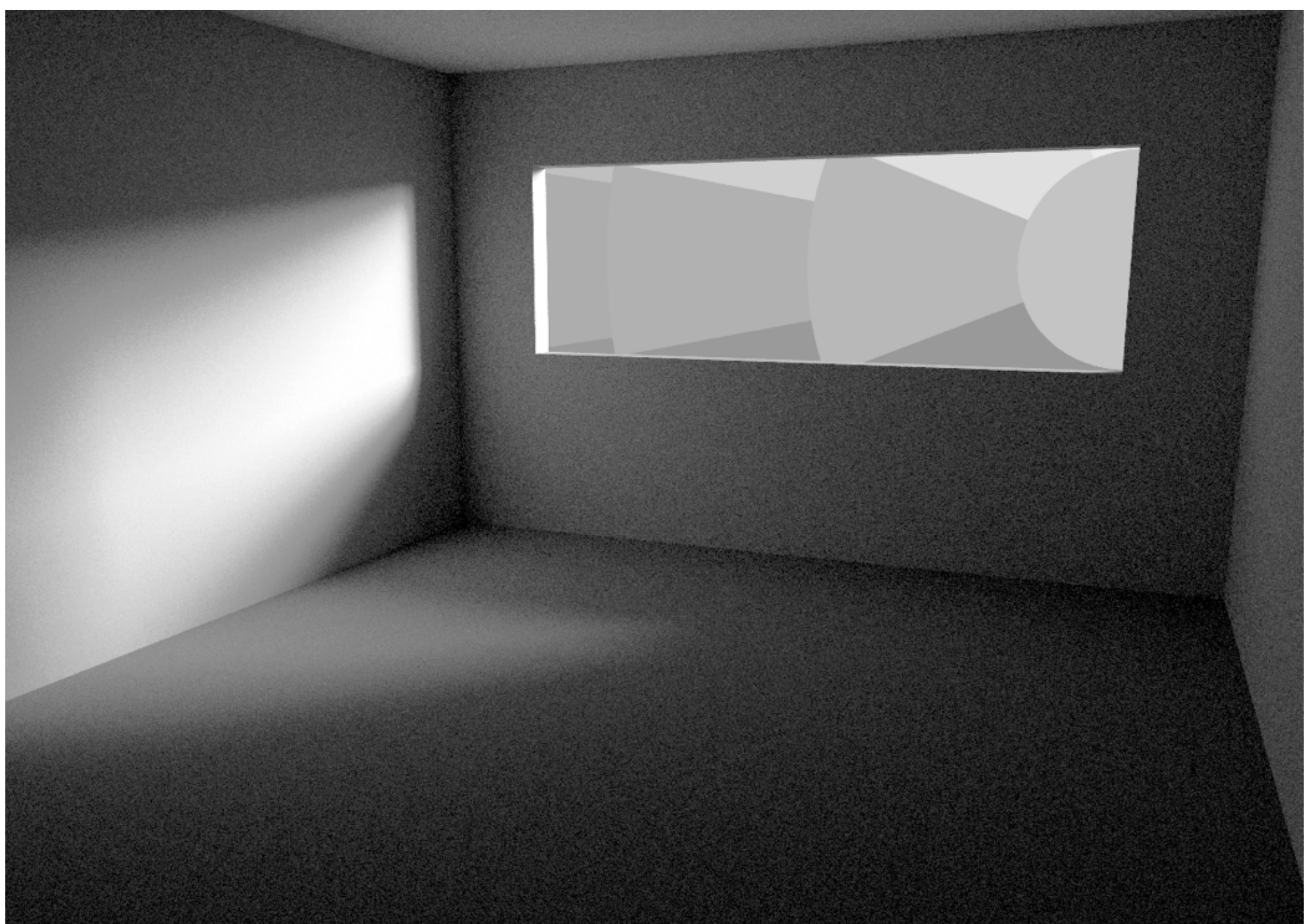
Third Term

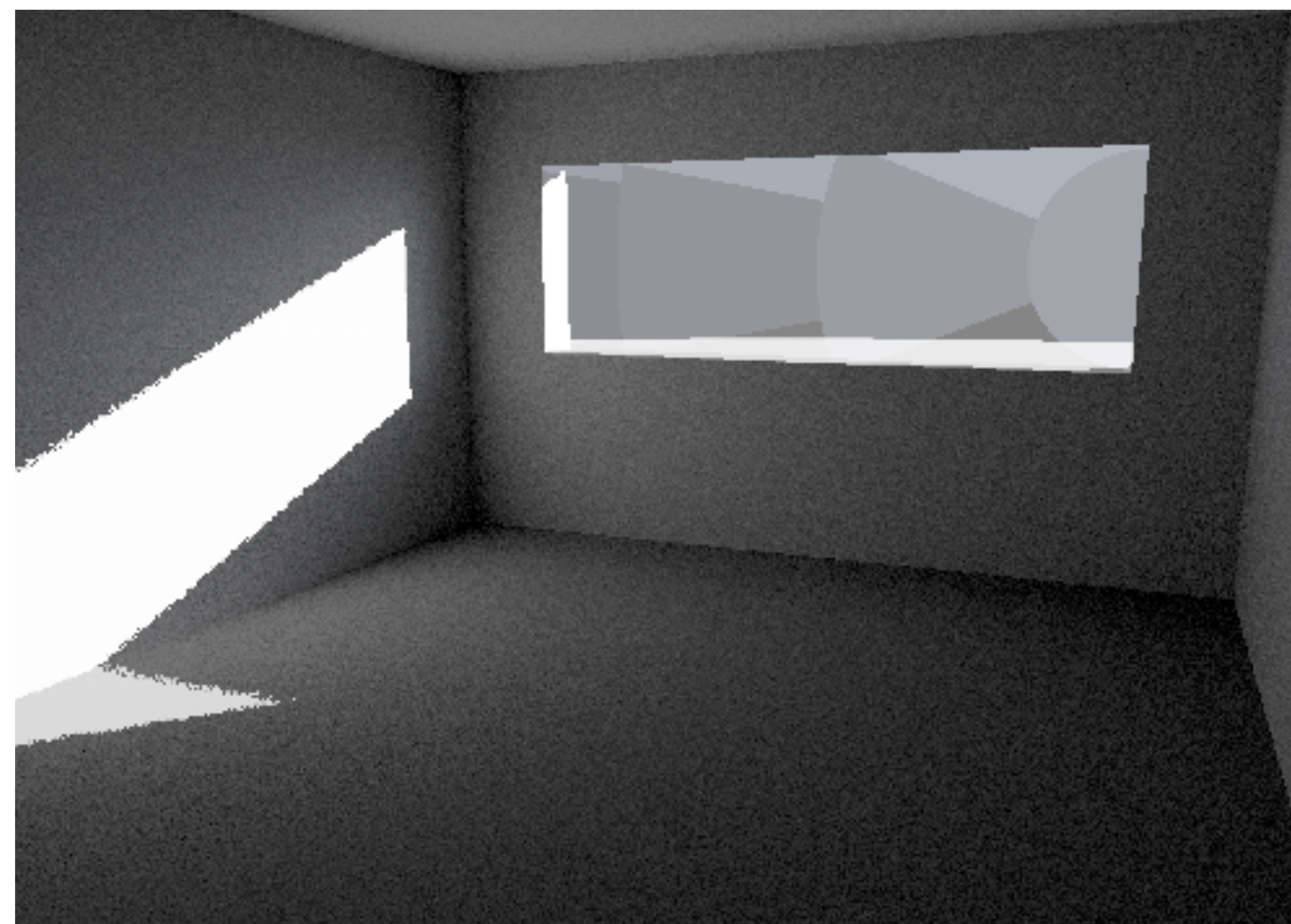


Five-phase Result


Clear Glazing, ground truth simulation


Clear Glazing, daylight coefficient simulation


Clear Glazing, three-phase simulation


Clear Glazing, five-phase simulation

What does the future hold?

- Five-phase simulation
  - Integration in DAYSIM - providing support for CFS
  - Openstudio/COMFEN other tools - via DAYSIM
  - Scripting / command line? - only for super hardcore (I'd rather not...)
- BSDF Data
  - Unhappy with the lack of independent testing for BSDF?
  - LBNL will start to offer BSDF measurement services for a fee to help kick start the industry.

What’s new at LBNL?

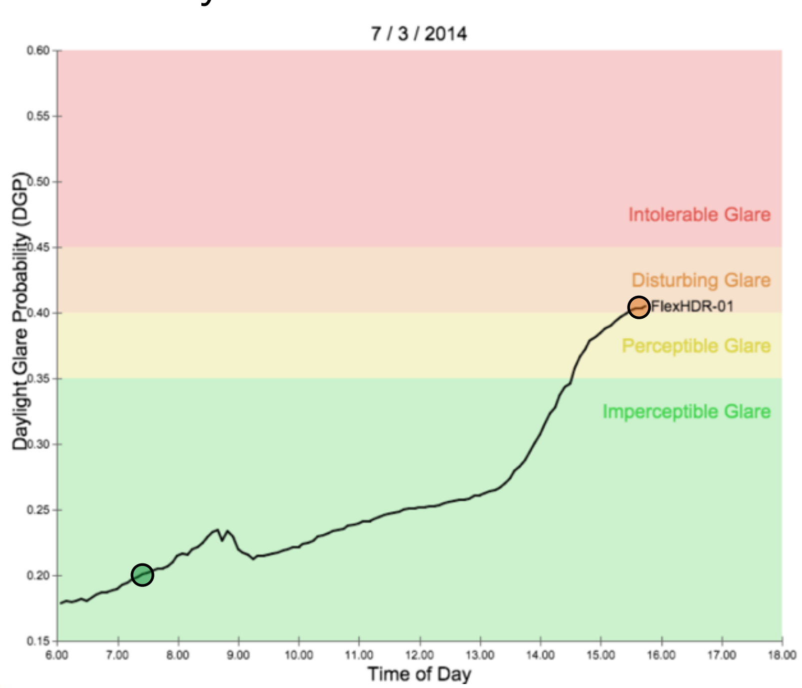
I’m glad you asked!

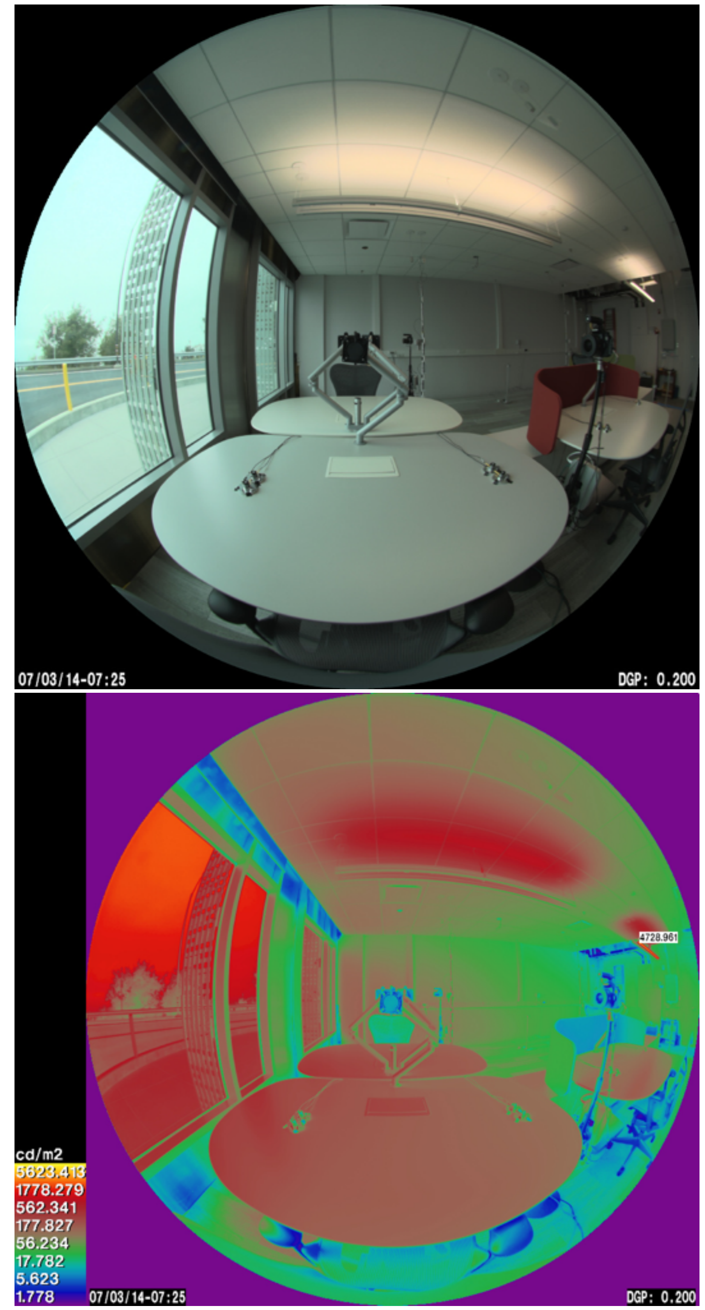
Flexlab

Image Based Glare Analysis

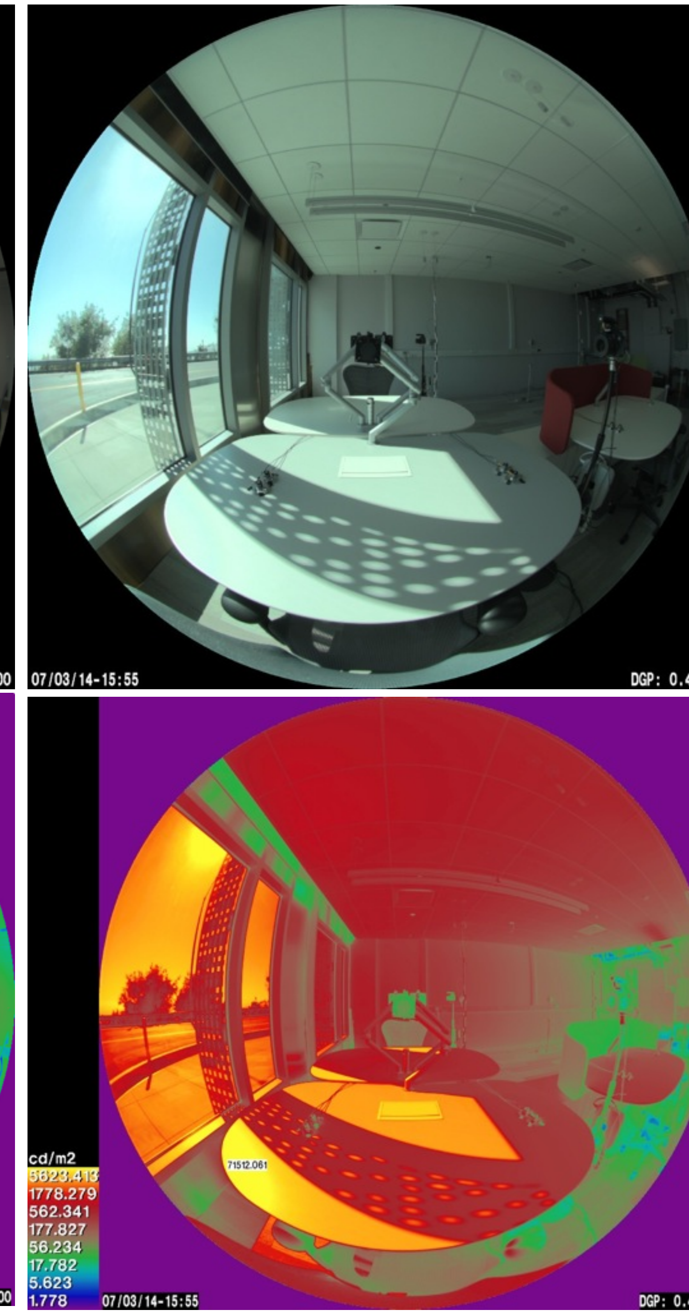
A dedicated camera takes bracketed exposure photos in FLEXLAB. An accompanying computer processes these photos into a single high dynamic range (HDR) image.

HDR images are evaluated for glare. The images to the left represent two conditions on July 3, 2014. The graph below shows glare rating throughout the day.





7:25 am: DGP=0.20, Imperceptible Glare (left images). An overcast morning with electric lighting on to supplement daylight.



3:55 pm: DGP=0.40, Disturbing Glare (right images). A sunny afternoon with sun shining on the desk. The electric light is dimmed to save energy. Glare could be mitigated by lowering the window shade.

Workplane Illuminance & Lighting Power Measurements

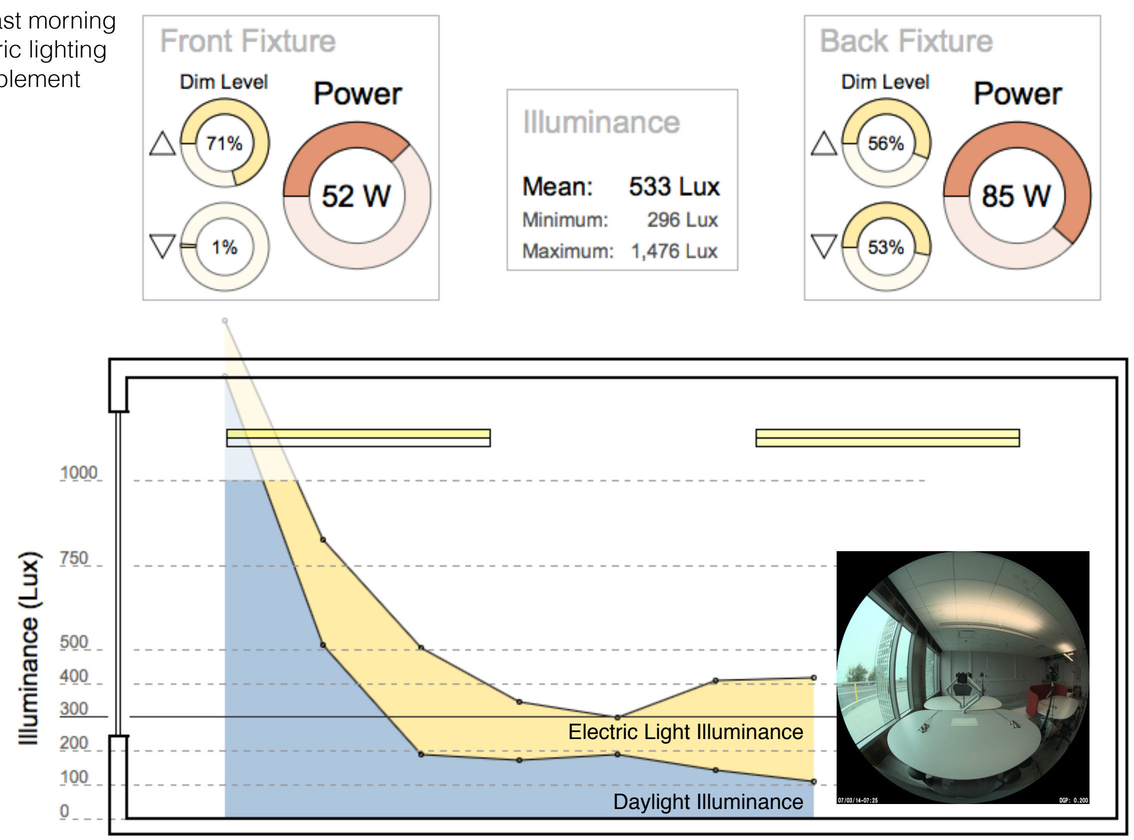
LED lighting fixtures have photocell control that dim the fixtures based on available daylight. Photometers distributed in the FLEXLAB mockup measure illuminance on the work plane. The target illuminance for this mockup is 300 lux.

The graphs below show the illuminance in the space measured by the photometers overlaid on a cross-section of the mockup. The window is shown on the left of the cross-section. Illuminance provided by daylight is shown in blue and supplemental illuminance provided by electric lighting is shown in yellow. Dial charts show the dimming levels for top and bottom light produced by each fixture along with the measured power consumed by each fixture.

Image of a photometer used in the FLEXLAB mockup

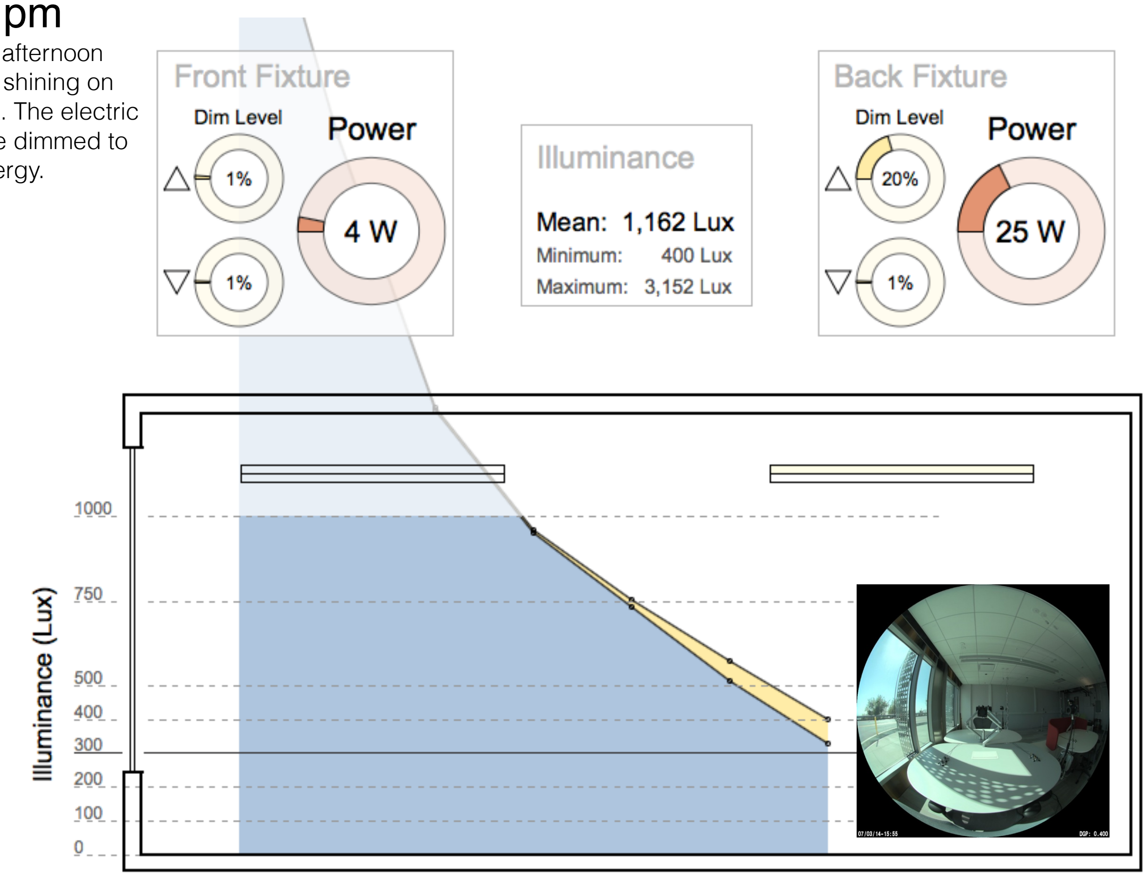
7:25 am

An overcast morning with electric lighting on to supplement daylight.



3:55 pm

A sunny afternoon with sun shining on the desk. The electric lights are dimmed to save energy.



SkyCam @ Flexlab

http://flexskycam.lbl.gov/


terrestrial LIGHT