Using the New *Radiance* BSDF Material Primitive

Greg Ward, Anyhere Software

Talk Overview

- New *BSDF* primitive and relation to WINDOW 6
- Primitive arguments and behavior modes
- Examples
- Future Work

New BSDF Primitive

mod **BSDF** *id*
6+ *thick BSDFfile ux uy uz funcfile [transform..]*
0
0/3/6/9

ρ_{rf} ρ_{gf} ρ_{bf}

ρ_{rb} ρ_{gb} ρ_{bb}

τ_{r} τ_{g} τ_{b}

Basic Example:

```
void BSDF blinds-0
6 0 vb0.xml 0 0 1 .
0
0
```

Advantages:

- **Includes transmission & reflection from either side**
- **Supports multiple random ray samples with -ss option**
- **“Proxy” mode enabled with non-zero thickness**

Relation to WINDOW 6

- WINDOW 6 exports BSDF data for complex fenestration systems, supports “layers”
  - Only square matrix BSDFs are supported by WINDOW 6 (up to 145x145)
- Extensions to WINDOW 6 XML format for:
  - Non-square matrices & multiple bases
  - CFS geometry
  - Variable-resolution BSDFs

} Produced by **genBSDF**

Front/Back Confusion

- WINDOW 6 considers the exterior of a building to be the “front” and labels XML files using this convention
- *Radiance* considers window and other surface normals as pointing *into* room, reversing this notion
- BSDF library thus swaps conventions, applying “Reflection Back” data to front side of surface
- **genBSDF +backward** generates interior data for “backwards ray-tracing,” i.e., *Radiance*

Dissecting the *BSDF* Primitive

mod **BSDF** *id*

6+ *thick BSDFfile ux uy uz funcfile [transform..]*

0

0/3/6/9

ρ_{rf} ρ_{gf} ρ_{bf}

ρ_{rb} ρ_{gb} ρ_{bb}

τ_{r} τ_{g} τ_{b}

What does it all mean?

BSDF Modifier

mod **BSDF** *id*
6+ *thick BSDFfile ux uy uz funcfile [transform..]*
0
0/3/6/9

ρ_{rf} ρ_{gf} ρ_{bf}

ρ_{rb} ρ_{gb} ρ_{bb}

τ_r τ_g τ_b

- **Textures perturb surface normal in the usual way**
- **Patterns affect all components except non-diffuse reflection**

BSDF Up Vector

mod **BSDF** *id*
6+ *thick BSDFfile ux uy uz funcfile [transform..]*
0
0/3/6/9

ρ_{rf} ρ_{gf} ρ_{bf}
ρ_{rb} ρ_{gb} ρ_{bb}
τ_{r} τ_{g} τ_{b}

- **Usually a constant vector**
- **Need not be normalized**
- **Extra effort needed to attach *BSDF* to sphere**

BSDF Diffuse Components

mod **BSDF** *id*

6+ *thick BSDFfile ux uy uz funcfile [transform..]*

0

0

ρ_{rf} ρ_{gf} ρ_{bf}

ρ_{rb} ρ_{gb} ρ_{bb}

τ_r τ_g τ_b

Remember that BSDF data may have diffuse component(s) also

- **Zero float arguments means no extra diffuse components**
- **Three float arguments adds diffuse front reflection**
- **Six float arguments adds diffuse back reflection**
- **Nine float arguments adds diffuse transmission**

BSDF Thickness

mod **BSDF** *id*
6+ *thick BSDFfile ux uy uz funcfile [transform..]*
0
0/3/6/9

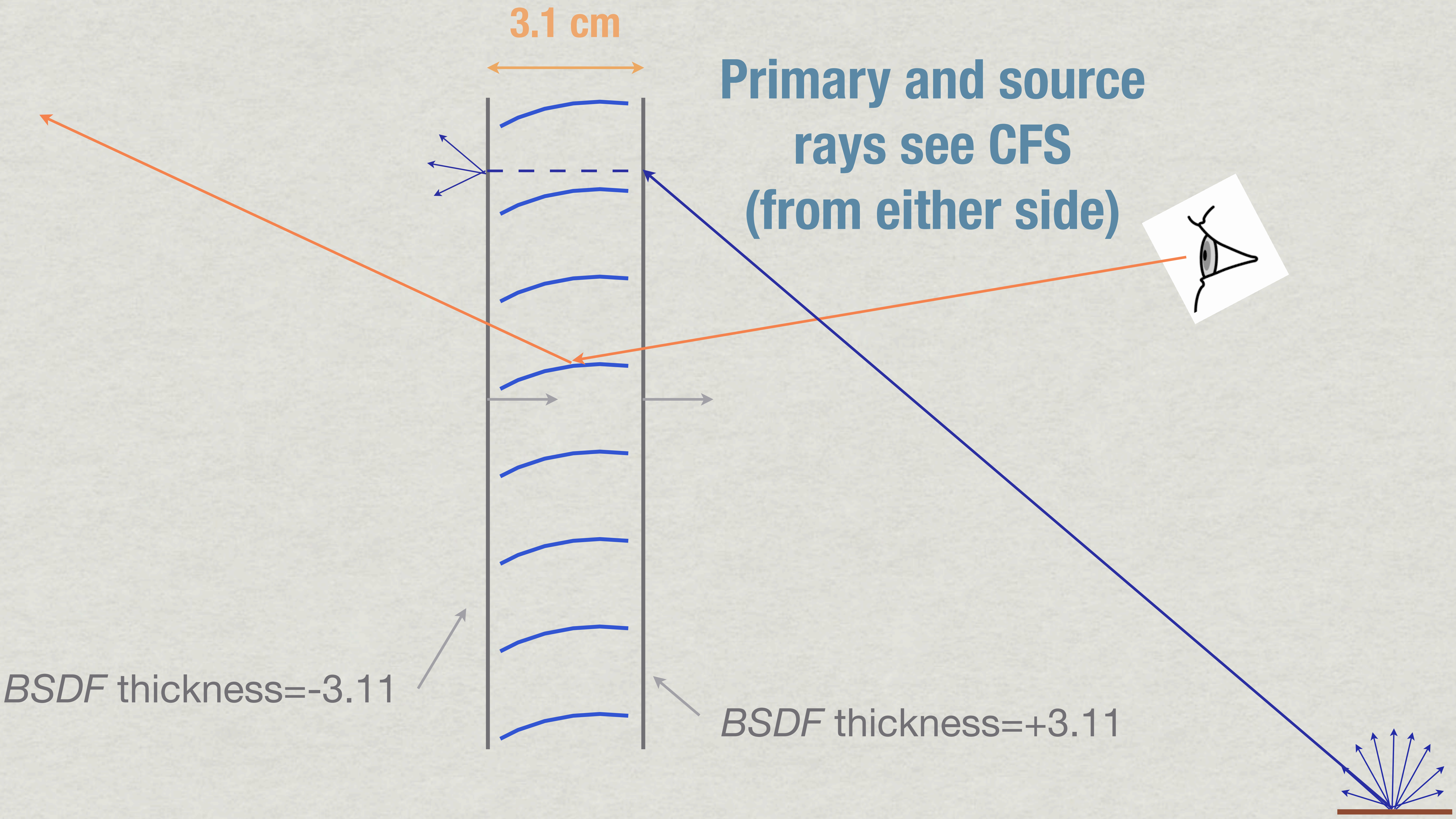
ρ_{rf} ρ_{gf} ρ_{bf}

ρ_{rb} ρ_{gb} ρ_{bb}

τ_r τ_g τ_b

- **If thickness=0,** *BSDF* **surface is all there is**
- **Positive thickness indicates proxied geometry "behind" surface**
- **Negative thickness places proxied geometry in front of surface**
- **Two** *BSDF* **surfaces may be used to "sandwich" CFS geometry**

Thickness Example



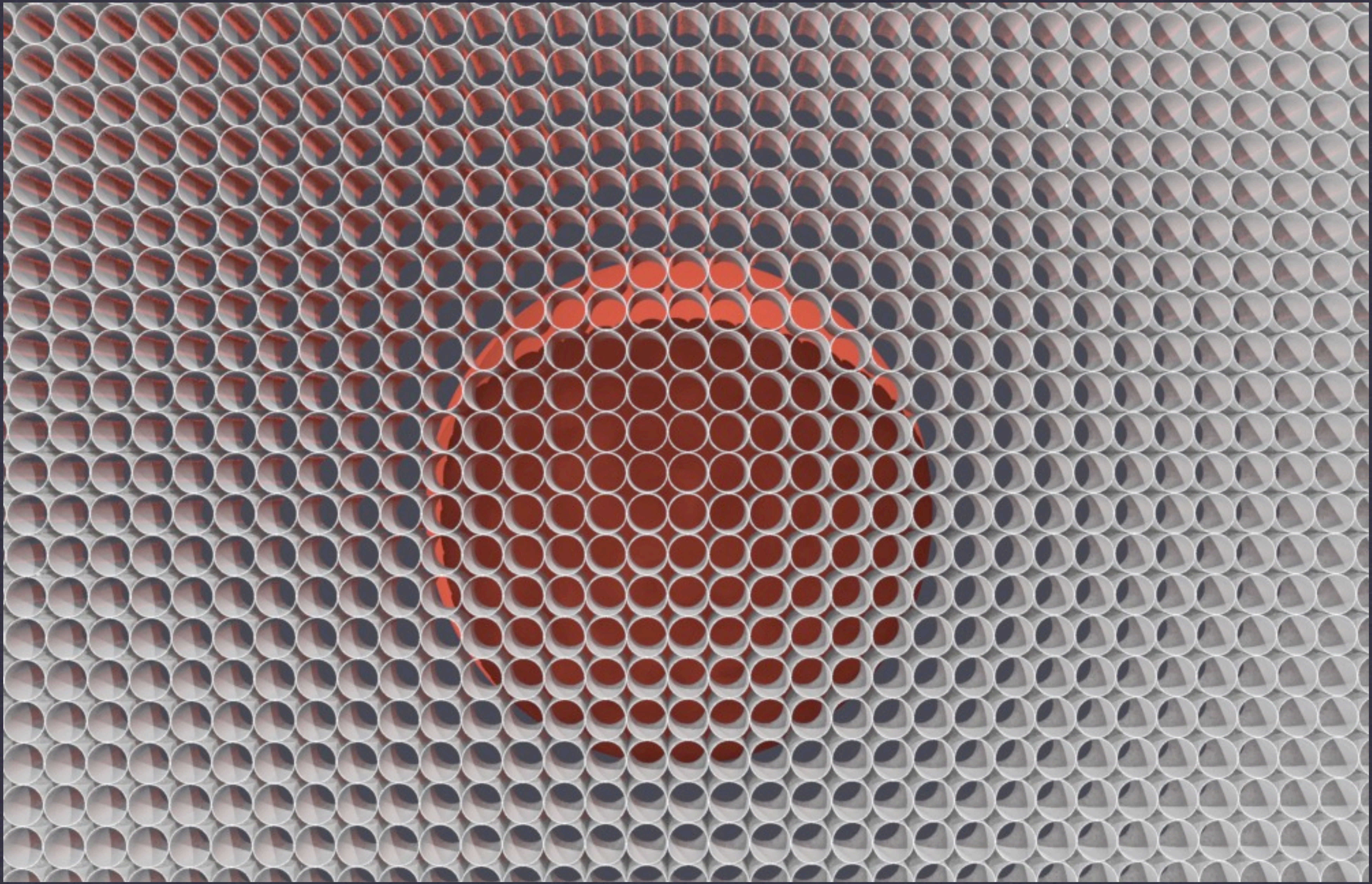
3.1 cm

Primary and source rays see CFS (from either side)

BSDF thickness=-3.11

BSDF thickness=+3.11

Indirect rays use *BSDF* sampling methods



Frivolous CFS Example

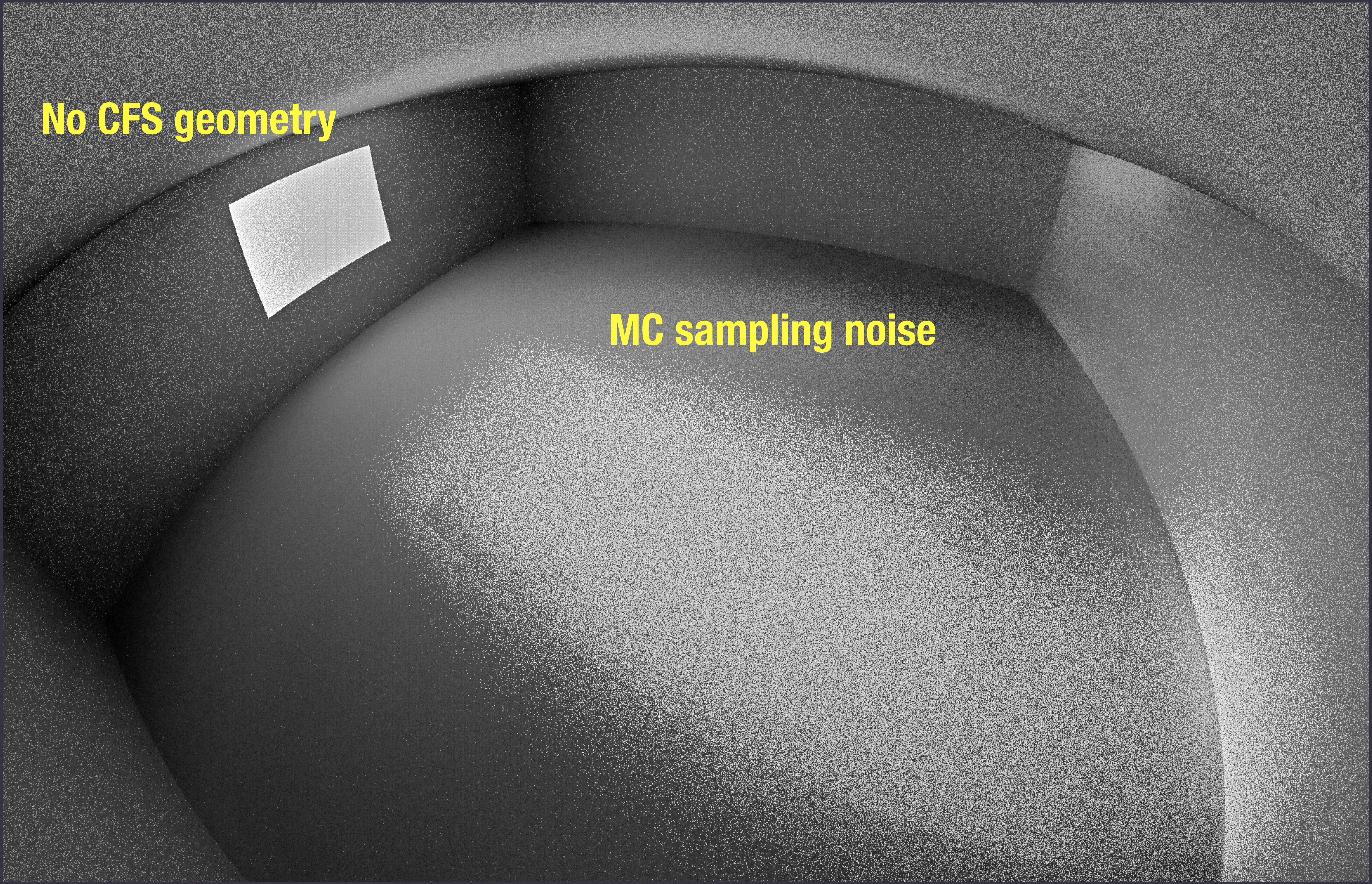
Assymetrically reflecting cylinders

genBSDF Command

```
genBSDF +geom meter +backward tblinds.rad > tblindsBack.xml
```

Straight *BSDF* **surface (zero thickness):**

```
void BSDF winBSDF
6 0 tblindsBack.xml 0 0 1 .
0
0

winBSDF polygon window
0
0
12
                         0                        4                        1
                         0                     6.25                        1
                         0                     6.25                      2.5
                         0                        4                      2.5
```



Straight *BSDF* Surface

No proxied CFS geometry

Using *BSDF* as Proxy

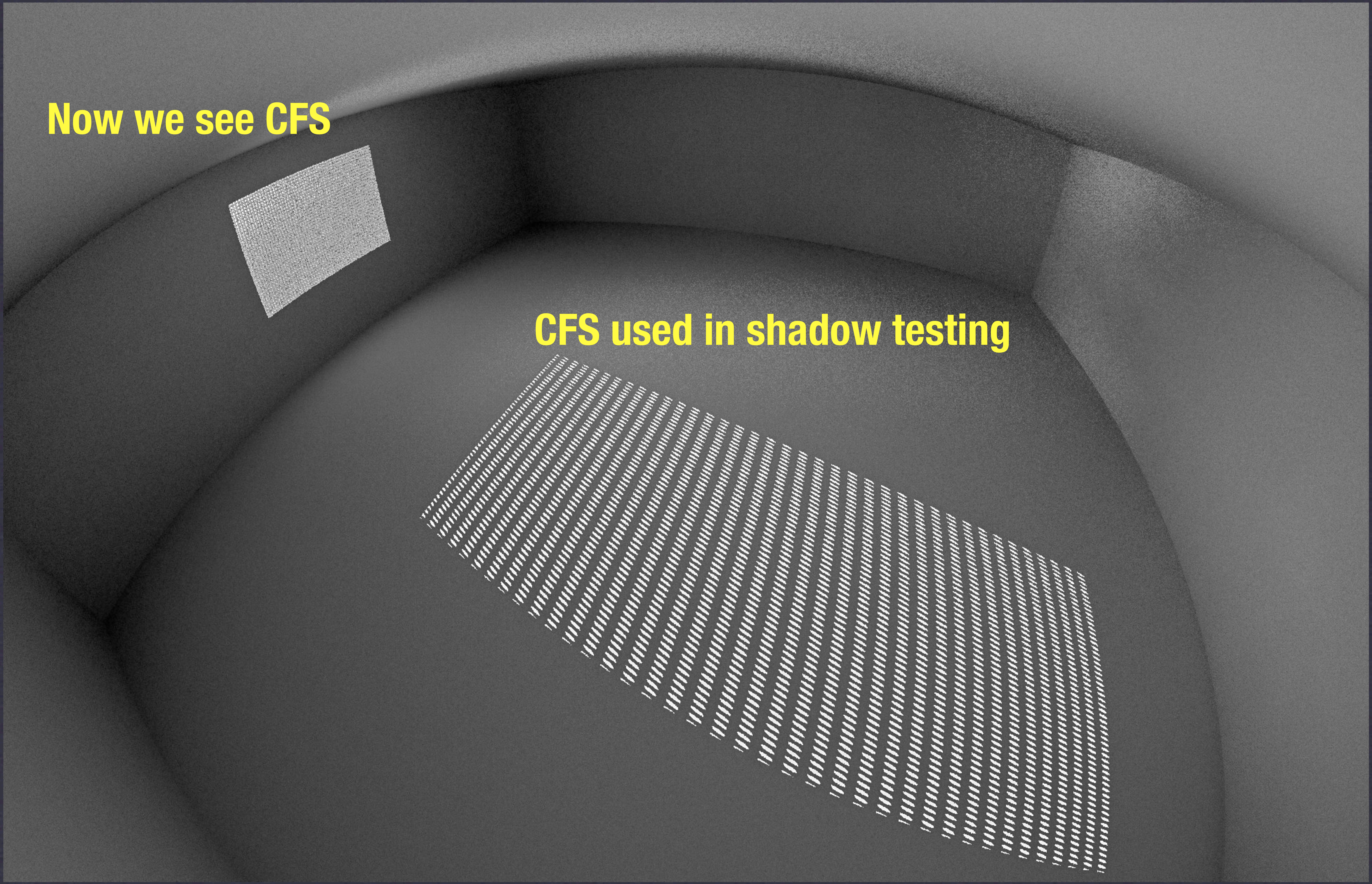
```
void BSDF winBSDF
6 0.055 tblindsBack.xml 0 0 1 .
0
0
```

thickness of geometry behind

```
winBSDF polygon window
0
0
12
                          0                      4                       1
                          0                     6.25                     1
                          0                     6.25                     2.5
                          0                      4                       2.5

!xform -rx 90 -rz 90 -t -.001 3.625 .75 tblinds.rad
```

Alternatively, the new **pkgBSDF** program could be applied:

```
!pkgBSDF -s tblindsBack.xml | xform -rx 90 -rz 90 -t -.001 3.625 .75
```

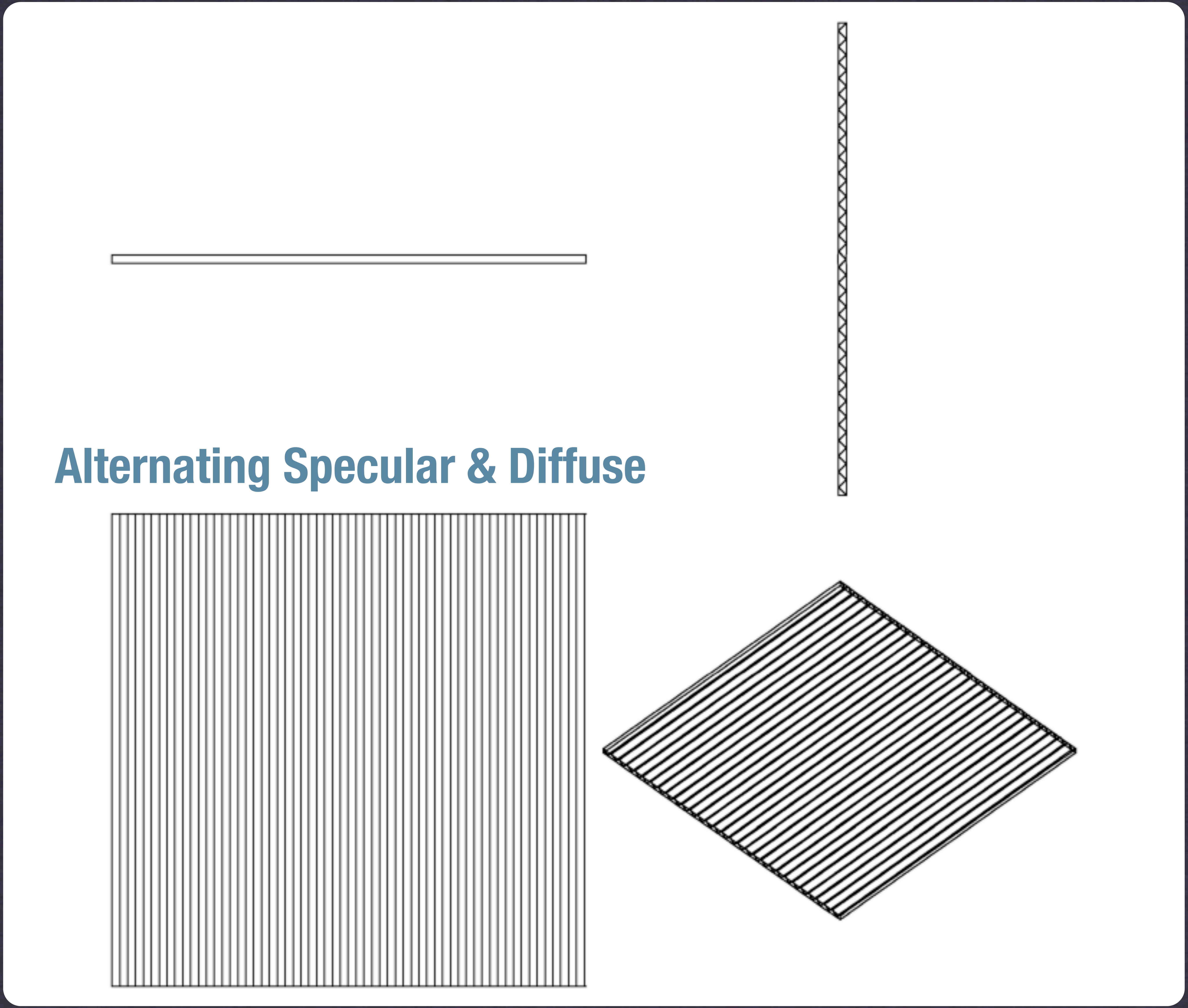


BSDF with Thickness

Proxied CFS geometry

Reflective Example

Specular and diffuse sawtooth



Alternating Specular & Diffuse

MGF Description

```
# Sawtooth surface with diffuse and
# specular materials alternating
#
m mirror =
    c
    rs .7 0
m diffuse =
    c
    rd .5
o sawtooth
```

Repeating part

```
xf -a 30 -t .02 0 0
v v1 =
    p 0 0 0
v v2 =
    p .01 0 -.01
v v3 =
    p .01 .6 -.01
v v4 =
    p 0 .6 0
v v5 =
    p .02 .6 0
v v6 =
    p .02 0 0
m diffuse
f v1 v2 v3 v4
m mirror
f v2 v3 v5 v6
xf
```

Frame part

```
v v1 =
    p 0 0 0
v v2 =
    p 0 .6 0
v v3 =
    p .6 .6 0
v v4 =
    p .6 0 0
v v5 =
    p 0 0 -.01
v v6 =
    p 0 .6 -.01
v v7 =
    p .6 .6 -.01
v v8 =
    p .6 0 -.01
m diffuse
f v1 v5 v6 v2
f v2 v6 v7 v3
f v3 v7 v8 v4
f v4 v8 v5 v1
o
```

genBSDF command

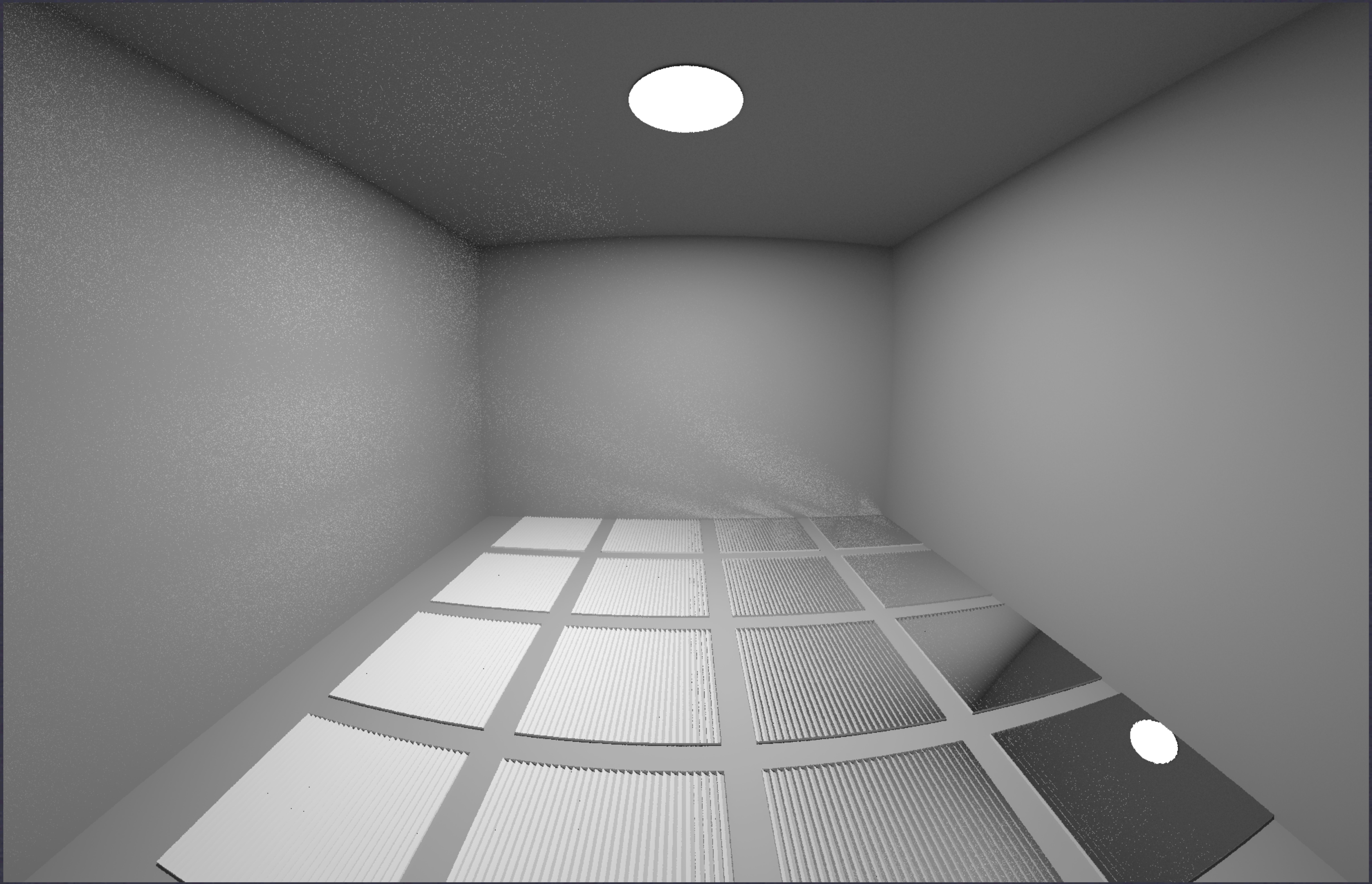
```
genBSDF +geom meter +mgf sawtooth.mgf > sawtooth.xml
```

Test room with proxied BSDF:

```
!pkgBSDF -s sawtooth.xml | xform -t .6 .6 .01 -a 4 -t .7 0 0 -a 4 -t 0 .7 0
```

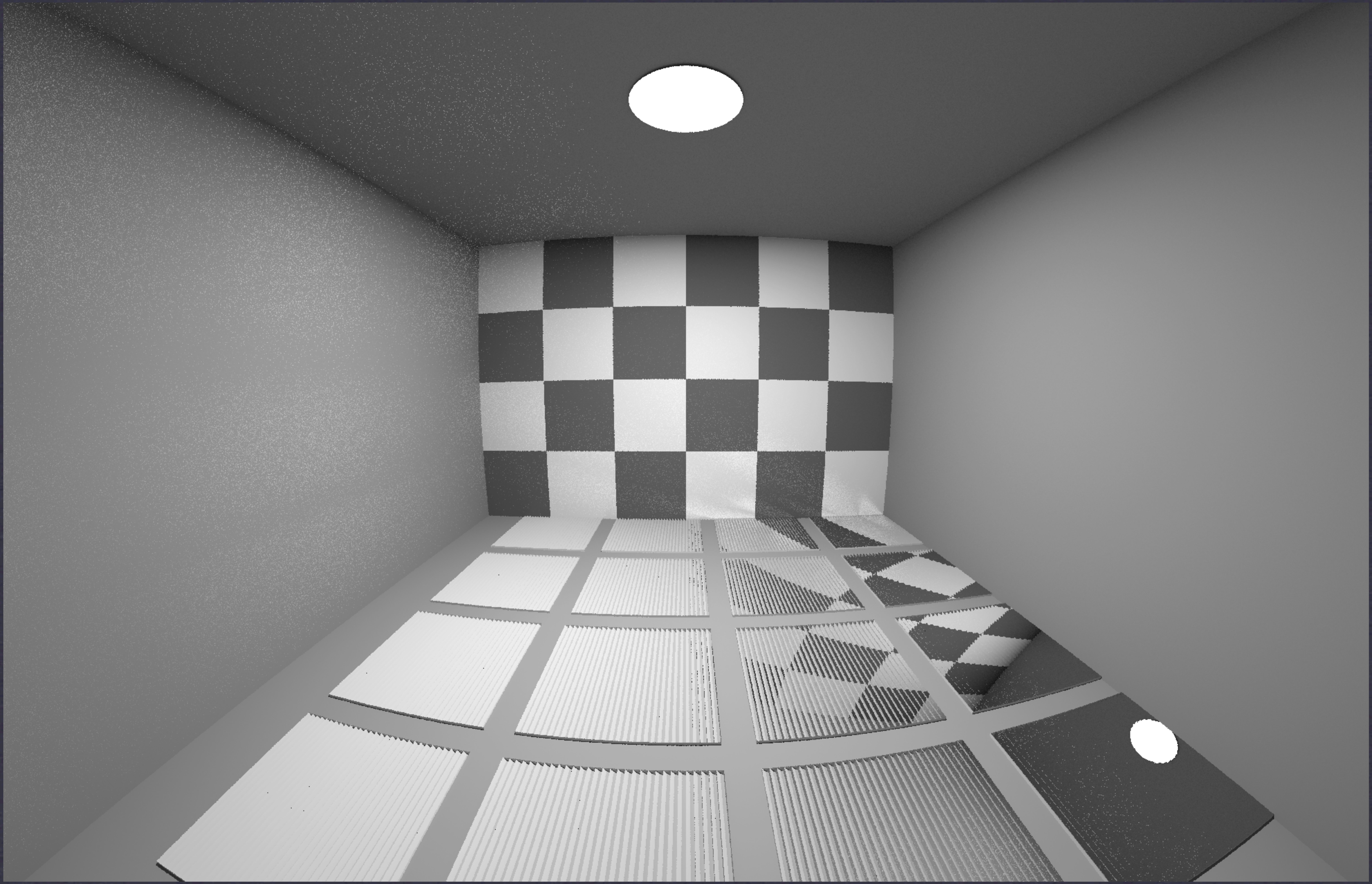
Leaves 10 cm gaps between tiles

```
void plastic dark
0
0
5 .2 .2 .2 0 0

!genbox dark room 3 3 2 -i

void light bright
0
0
3 100 100 100

bright ring src1
0
0
8
        1.5     1.5     1.99
        0       0       -1
        0       .2
```

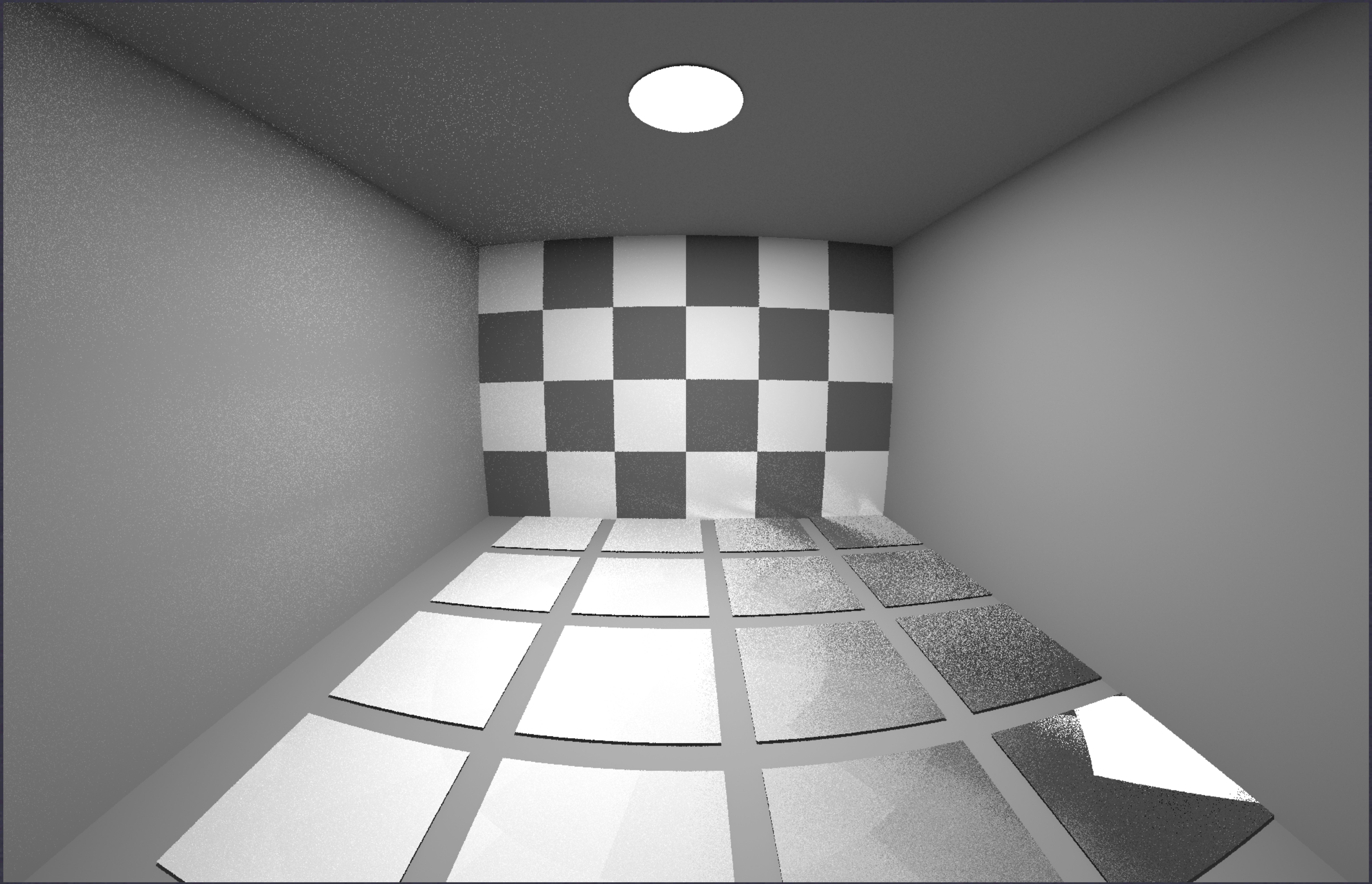


Rendered Result

Using BSDF proxy mode

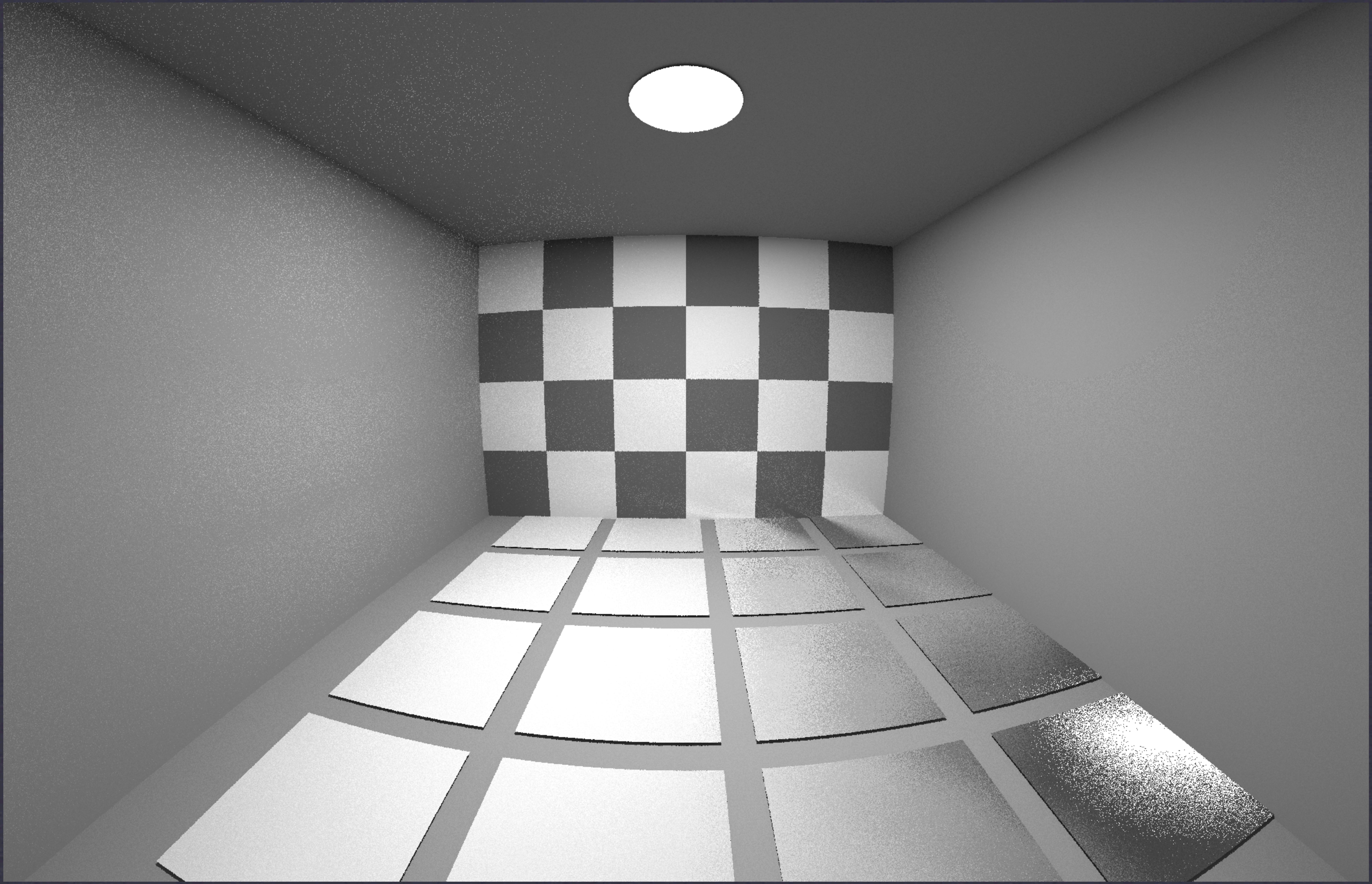


Adding Checkerboard



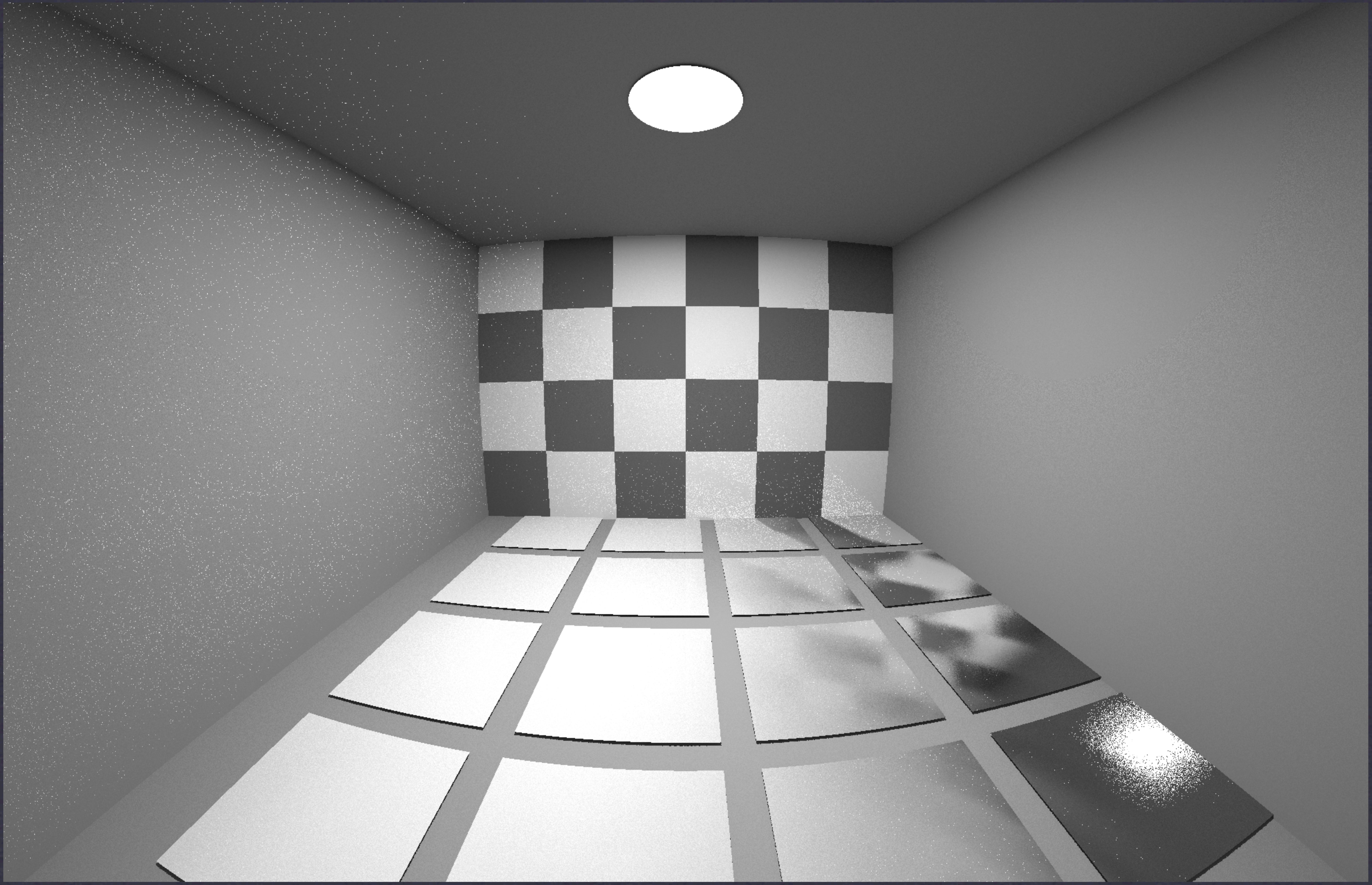
Removing Proxied Geometry

BRDF matrix becomes visible



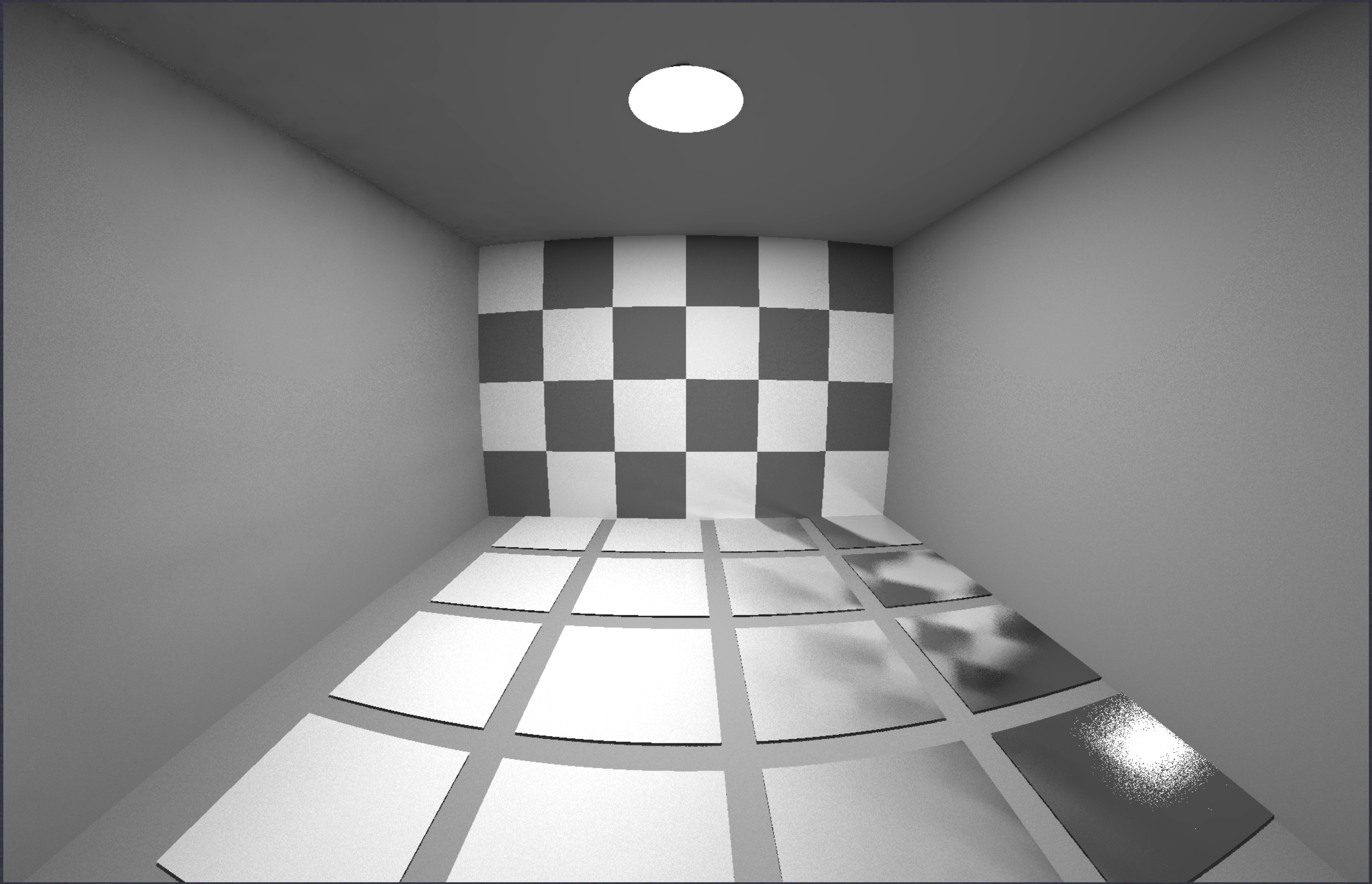
Additional Specular Sampling

-ss 16 -dj 1 (10% increase in run time)



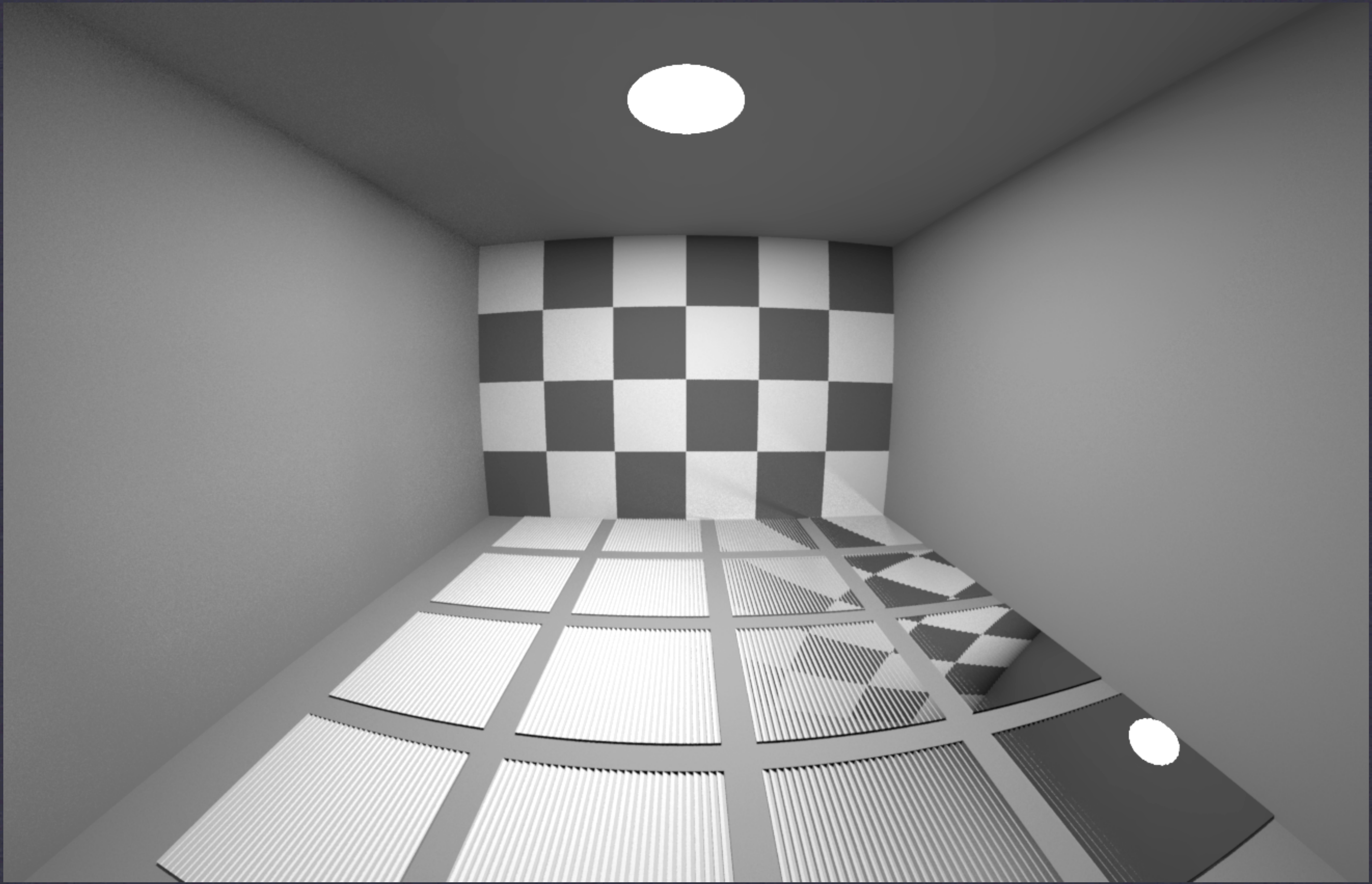
Using Variable-resolution BRDF

`genBSDF +t4 6` (took 1 CPU month to compute distribution at 4Kx4K resolution)



Variable-resolution BRDF with Ambient Cache

Indirect sampling noise is reduced, but reflections still not resolved very well



Ground Truth Rendering

Using *mirror* material to generate virtual light sources

Future Work

- Much validation still to be done
  - Compare to alternative simulations
  - Compare to physical measurements
- Color/spectral distributions

BSDF Library Availability

- BSDF C library designed to be cross-platform and separable from *Radiance* source tree
  - Reads matrix and variable-resolution BSDF data
  - Supports queries for BSDF value, resolution, hemispherical reflectance and transmittance
  - Generates ray samples with stratified Monte Carlo
  - Converts to and from local BSDF coordinates