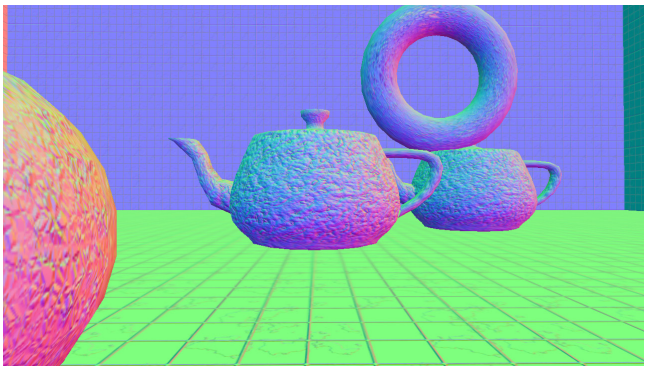
BurgerEngine Deferred Renderer

Rendering process:

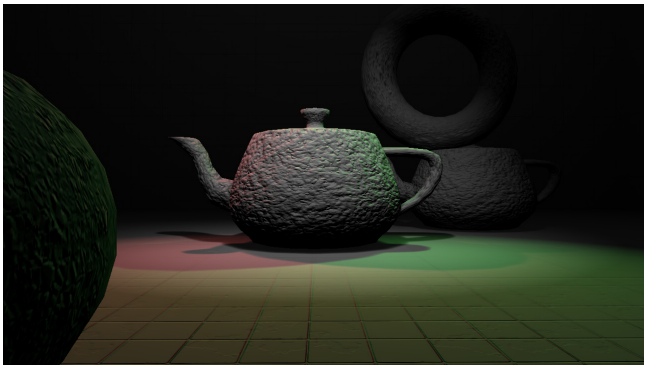
Step 01 : Culling
Frustum culling (meshes and lights)
Sort opaque object from front to back
Sort transparent object from back to front

Step 02 : Shadow maps rendering (Exponential Shadow Map)
For each spot shadow, render each each visible object (write depth into a 512x512 32bit buffer), then blur the render target.

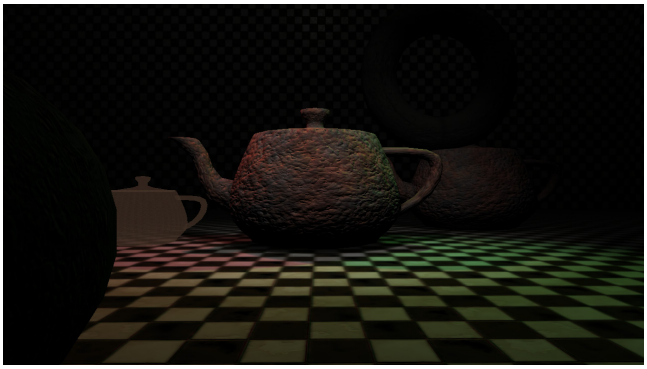
Step 03: GBuffer pass
For each object affected by light, write view space normal, gloss and depth.



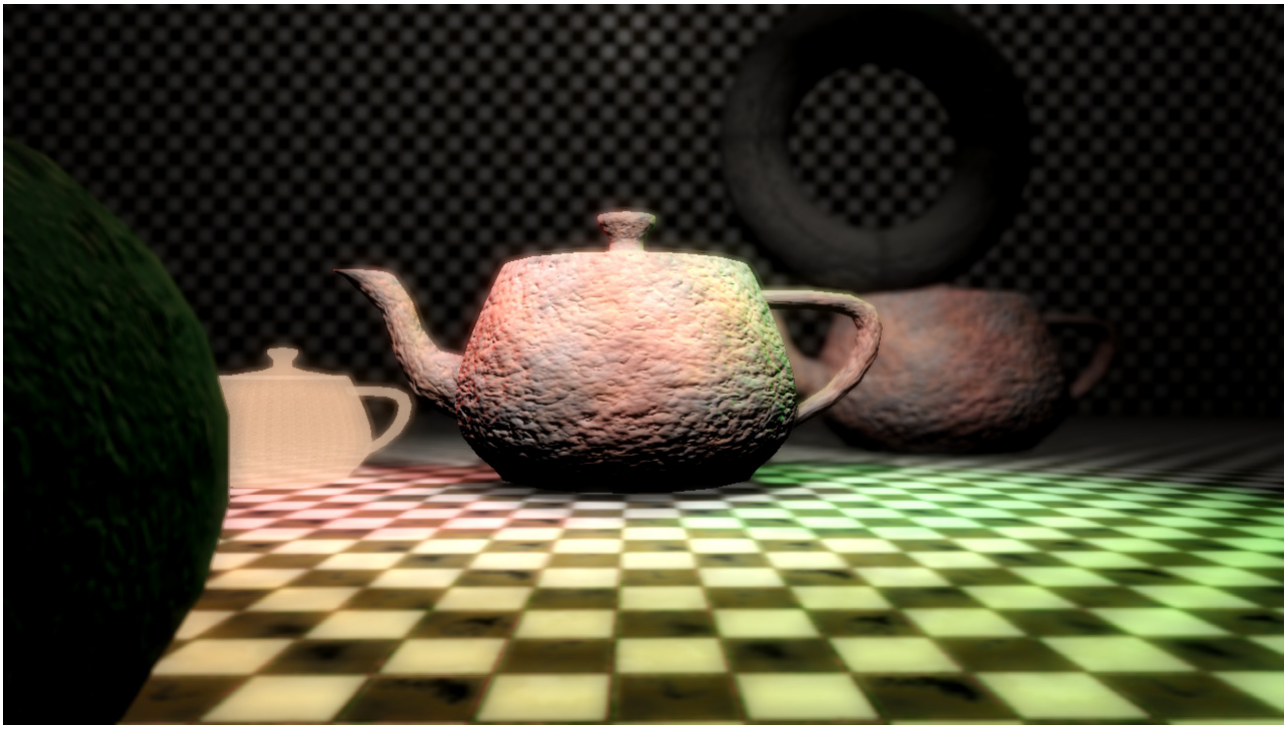
Step 04: Lighting pass
For each light source, use GBuffer to compute lighting and accumulate it to the LightBuffer.



Step 05: Material pass
For each opaque object, render with custom shader (use LightBuffer)
Render skybox
For each transparent object, render back faces with custom shader then render front faces with custom shader.
Every shader must write DOF data into a specific render target



Step 06: Post processing
Compute average luminance of the scene. Value is interpolated over time to simulate eye/camera adaptation.
Brightpass + blur of the colorbuffer.
Downsample and blur colorbuffer for DOF.
Final shader applies tone mapping, depth of field, color correction and glow.



Bonus Step:
Various debug render (bounding volumes, render targets etc.)