
Image: Arrowhead Game Studios
Deferred Shading

Overview

- Motivation: why not conventional shading?
  - Shader permutation problem
  - Combinatorial explosion
- Deferred shading method
- G-Buffers
- Other post-processing effects
- Advantages/disadvantages

Conventional Forward Rendering

- After rasterization
- Shading calculations in fixed function pipeline or fragment shader
- Complexity = O(Light sources * Objects)

Overdraw

- Complex scenes, large virtual environments → overdraw
- Overdraw → redundant calculations
- Why shade a pixel, if it gets overdrawn in final image?
- Idea: perform shading at the end of rendering
  → we need only one shading operation per pixel

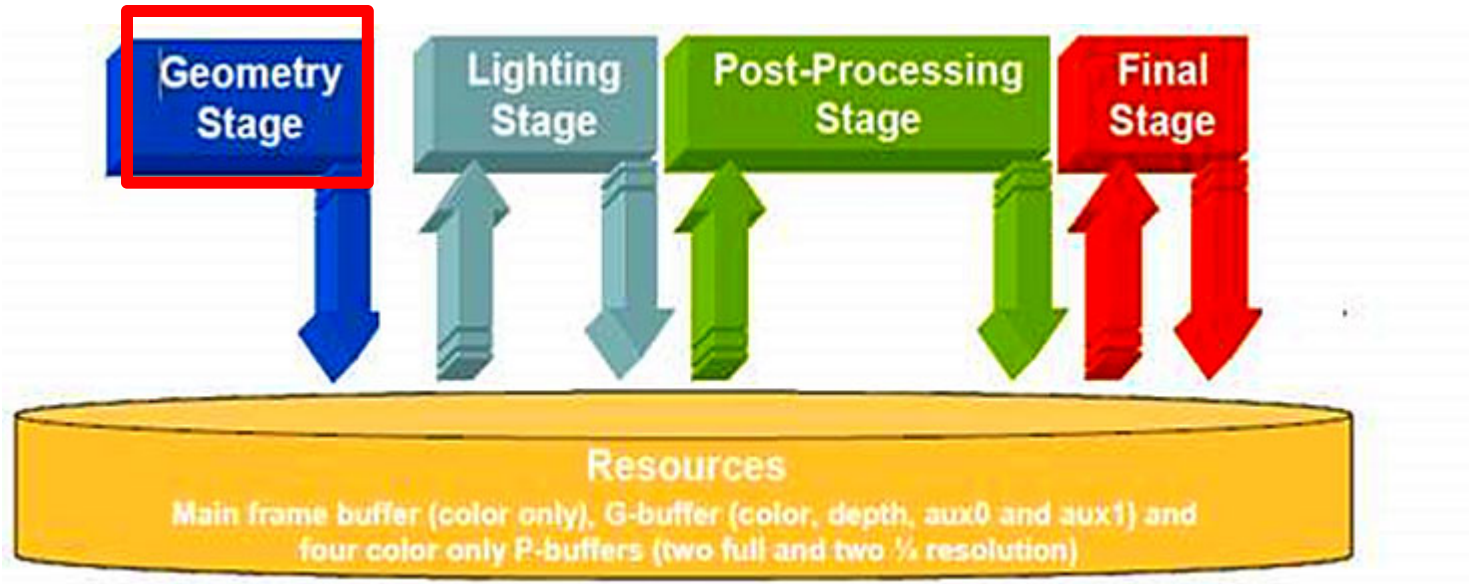
Intermediate:

Final:

Basic Deferred Rendering

Split rendering pipeline in two separate stages

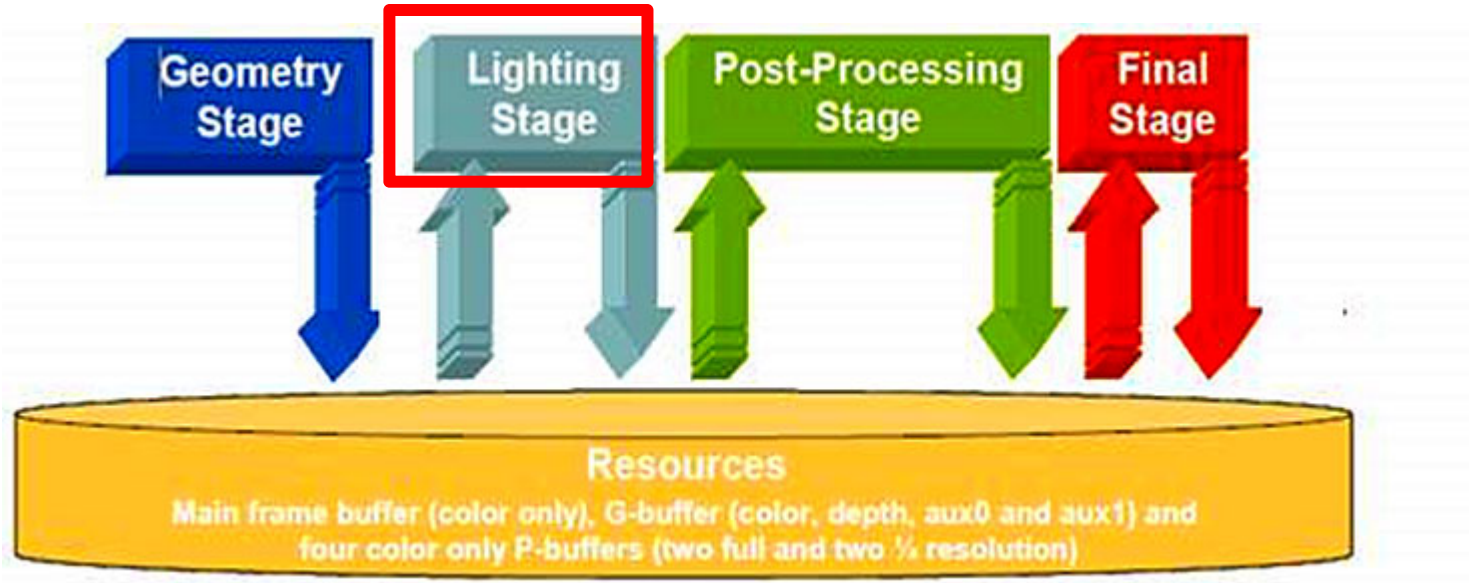
1. Geometry transformation and rasterization



Lighting and Shading Stage

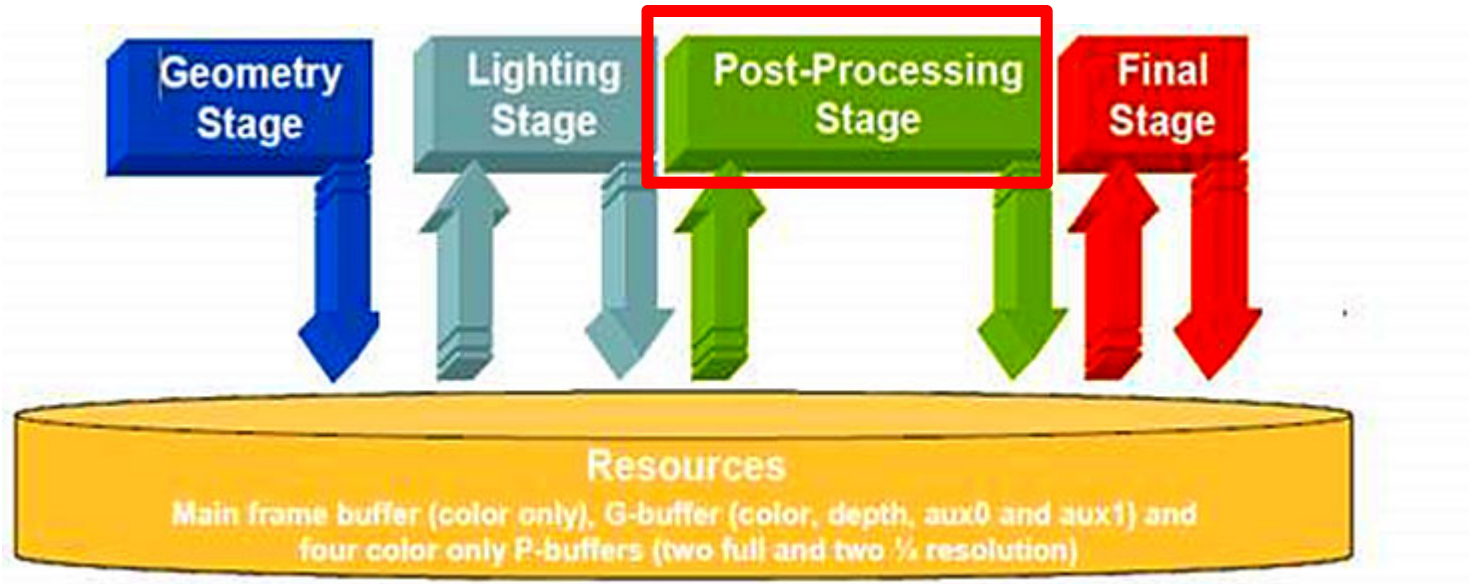
Split rendering pipeline in two separate stages

1. Geometry transformation and rasterization
2. Shading



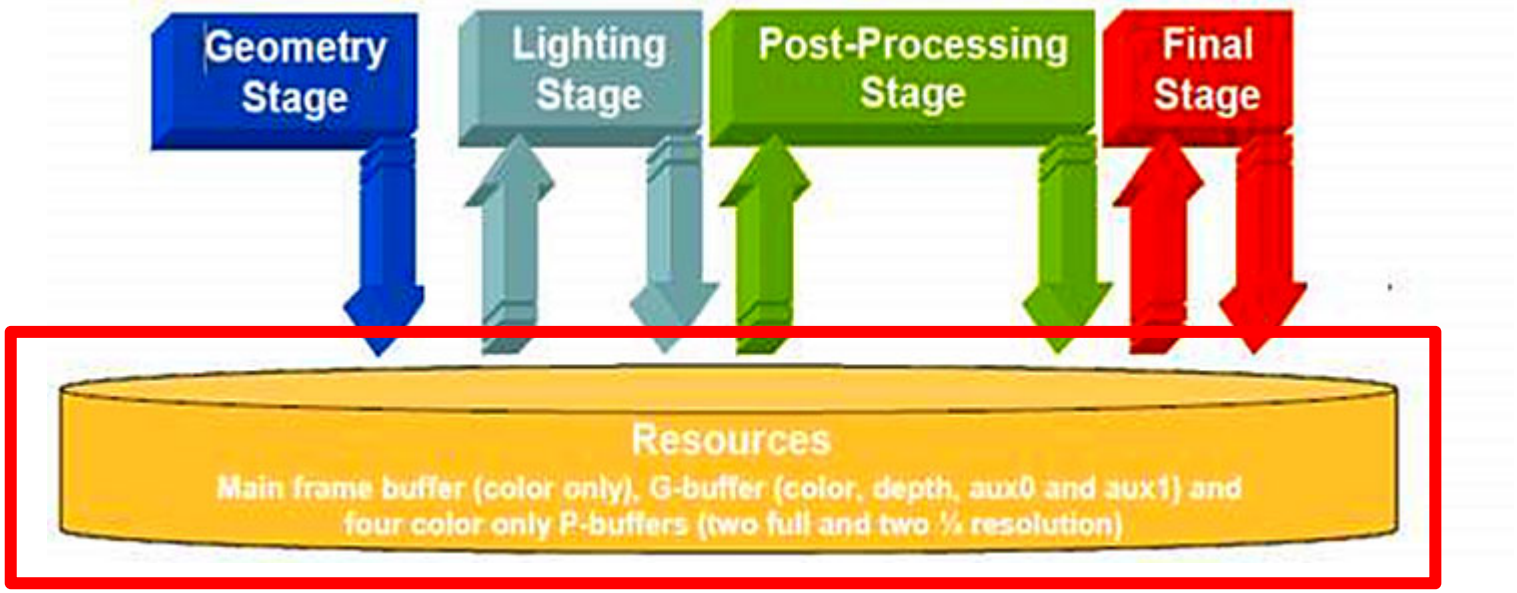
Post-Processing

- In an optional final stage, apply image post-processing



G-Buffers

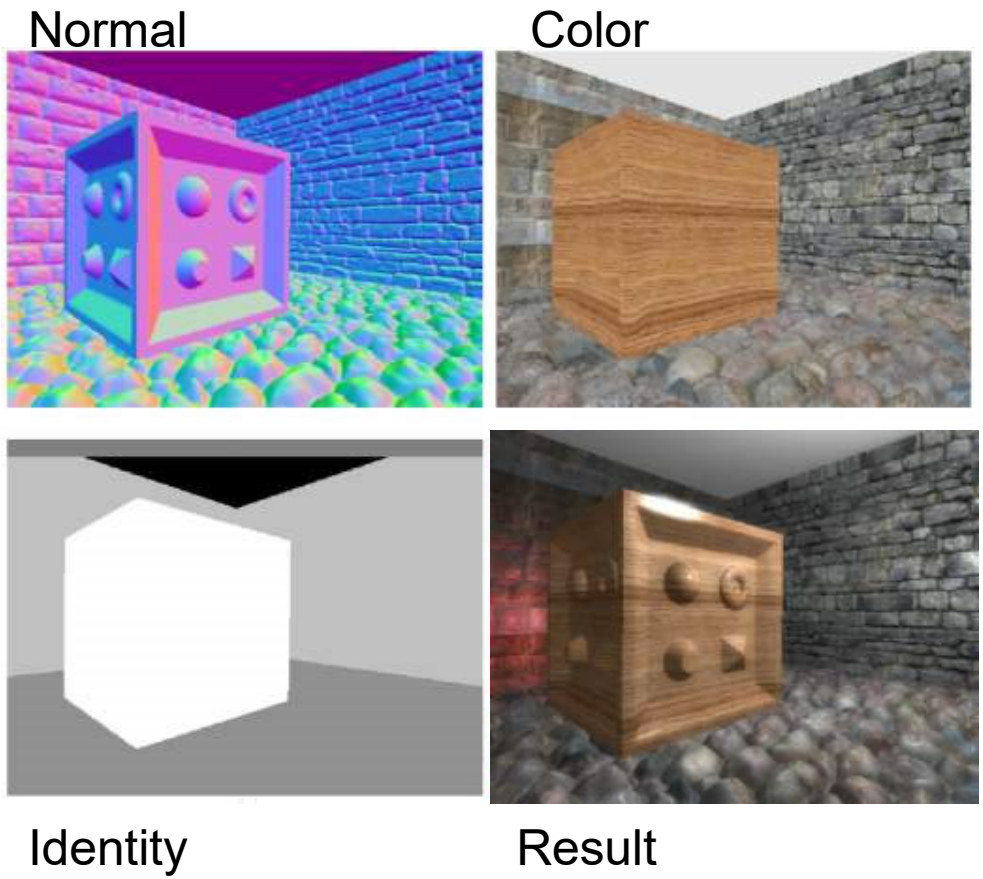
- G-Buffers pass data from one stage to next



Types of G-Buffers

G-Buffers store per pixel:

- Color
- Depth
- Normal vector
- Position
- Object identity
- etc.

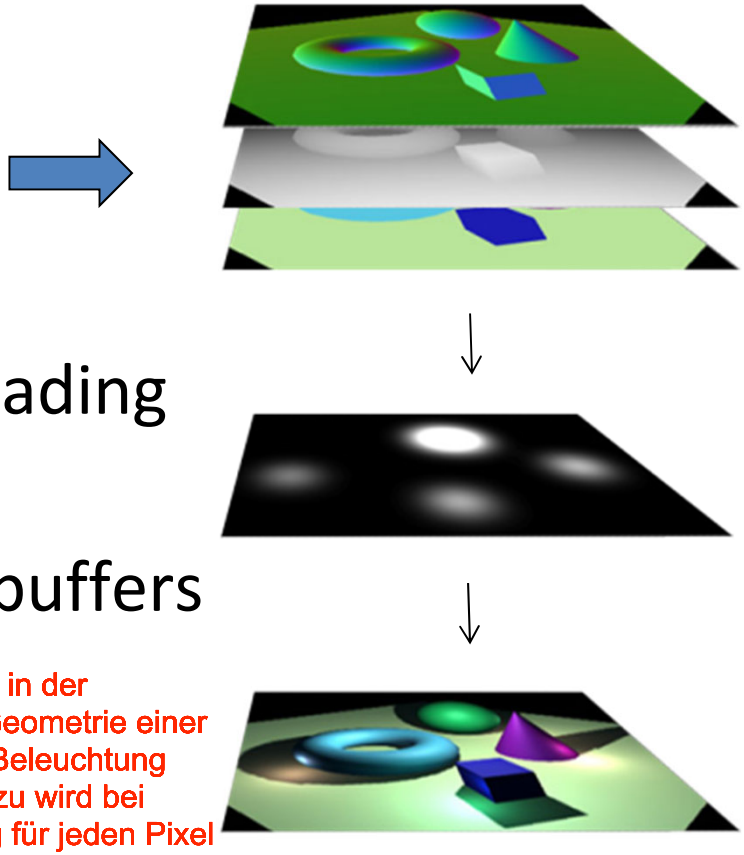


Deferred Rendering Overview

- DR is a framework supporting various effects
  - Screen-space ambient occlusion
  - Non photo-realistic rendering
  - High-dynamic range rendering
  - Deferred shading
  - …
- Implemented as a post-processing effect
- Most important sub-type: deferred shading

Deferred Shading

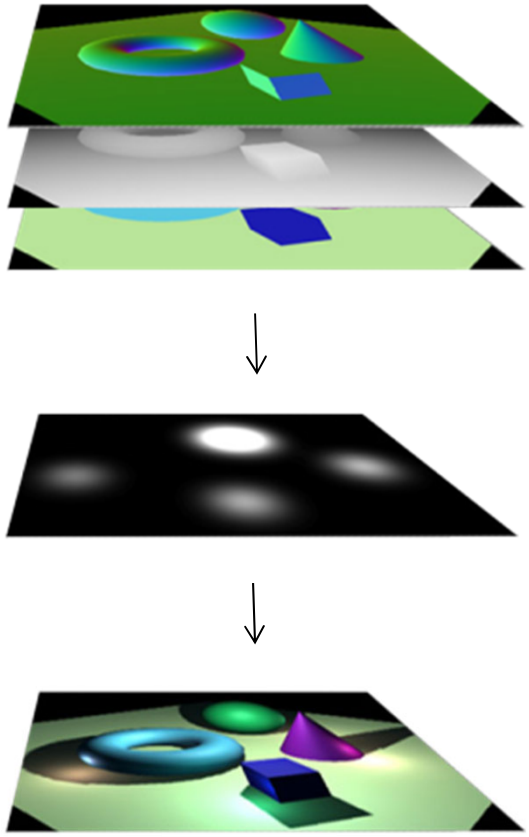
- Geometry stage as usual
  1. Geometry transformations
  2. Rasterization
  3. Material, texturing, but no shading
- Instead of shading,
  store intermediate results in G-buffers
  - Diffuse color
  - Depth
  - Position
  - Normal vectors



Deferred Shading ist eine Technik in der Computergrafik, bei der erst die Geometrie einer Szene berechnet wird, bevor die Beleuchtung berechnet wird. Im Gegensatz dazu wird bei Forward Shading die Beleuchtung für jeden Pixel einzeln berechnet, während die Geometrie berechnet wird. Deferred Shading ermöglicht es, dass man mehrere Lichtquellen und deren Wechselwirkungen mit Oberflächen gleichzeitig berechnen kann, was zu realistischeren Ergebnissen führt. Es ist jedoch auch rechenintensiver als Forward Shading.

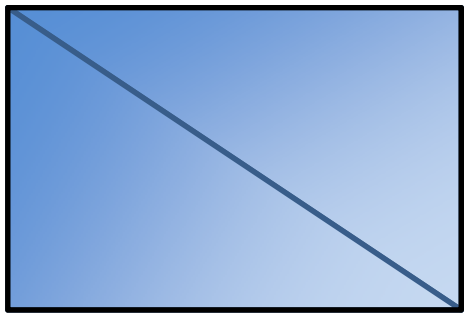
Actual Shading Stage

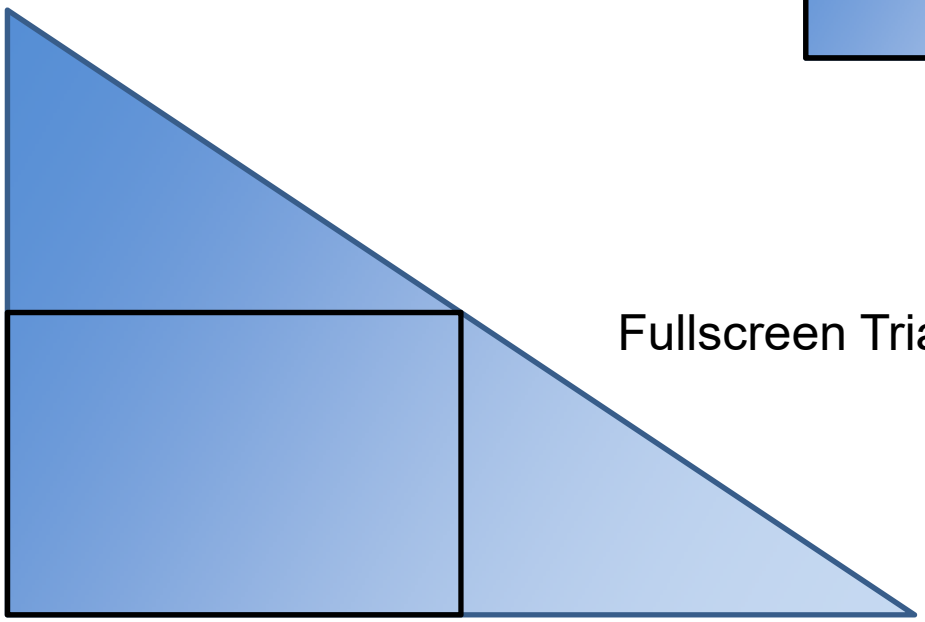
- Render screen-sized quad
- Fragment shader reads G-Buffers
- Perform shading and postprocessing
- Store result in framebuffer



Screen-filling Primitives

Fullscreen Quad





Fullscreen Triangle

Deferred Rendering Advantages

- Render geometry only once
- Perform complex shading and post-processing per pixel
- Complexity O(Light sources + Objects) instead of O(Lights*Objects)
- Independent of geometry and depth complexity
- Time for shading can be predicted well
  → good for games

Deferred Rendering Disadvantages

- Requires more memory and frequent read/write operations
- Advanced effects (transparency, ghostings) require per-pixel sorting
- Cannot use hardware anti-aliasing

- Forward shading may be faster, if
  - Low number of light sources
  - Low depth complexity
  - No need for post-processing effects