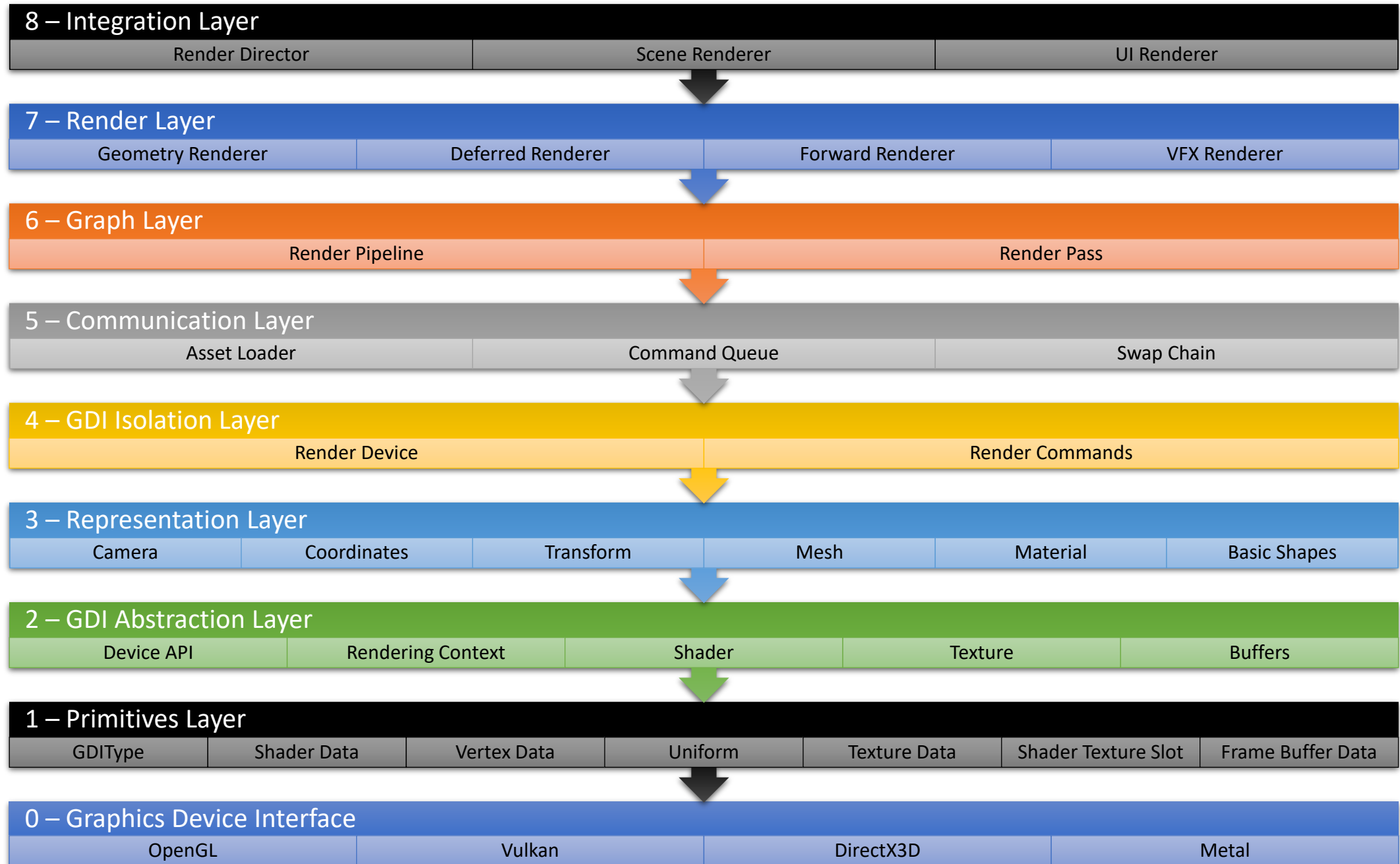
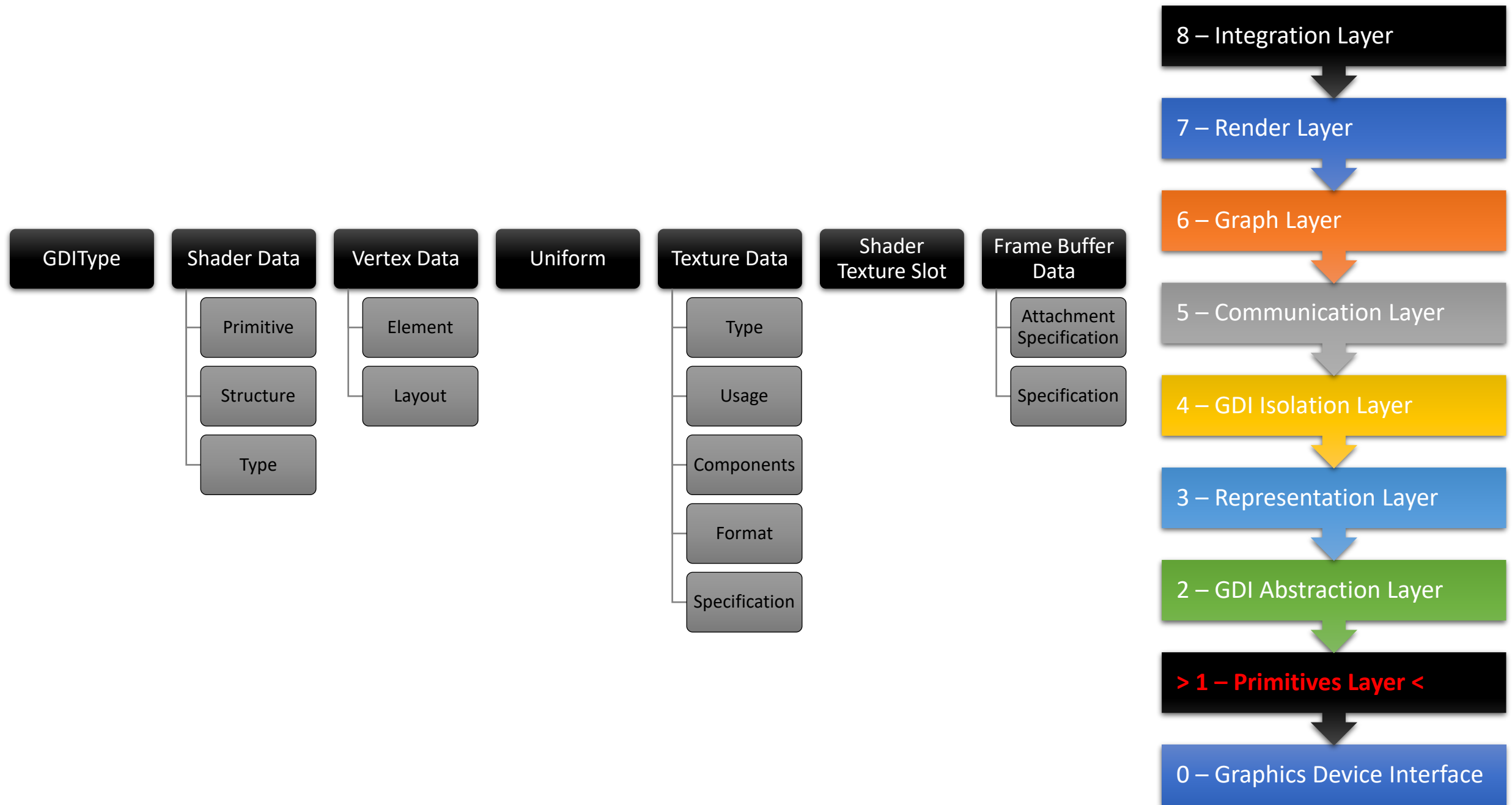
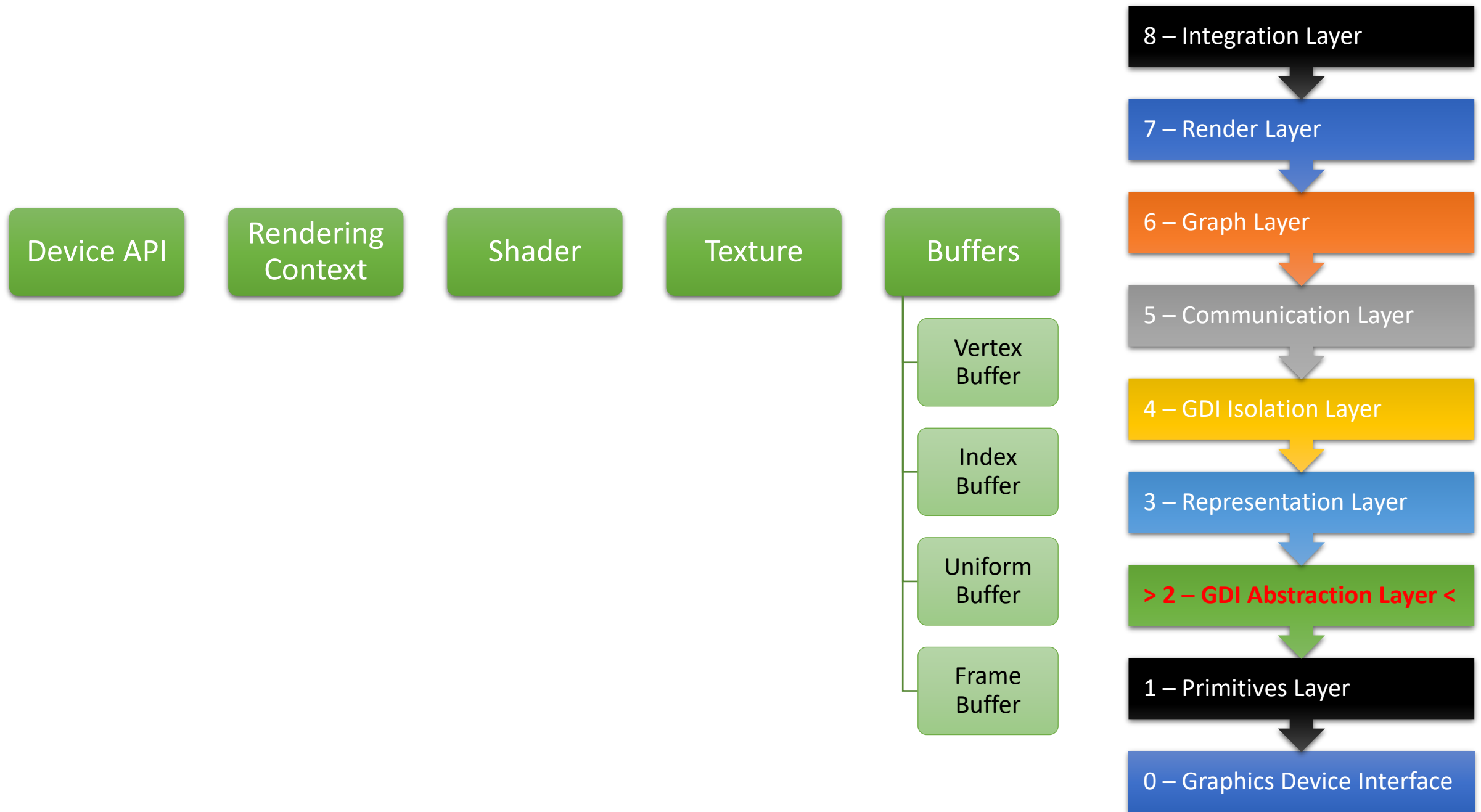


# **Planned Rendering Architecture of Fools Engine**

Work in progress



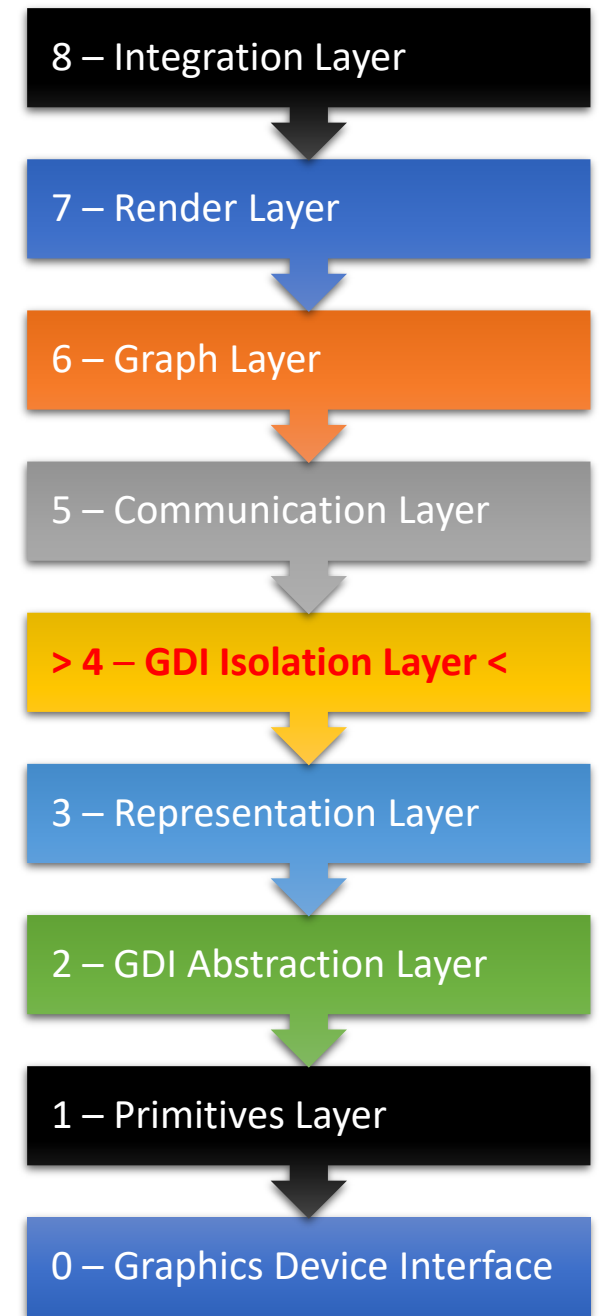






Render  
Device

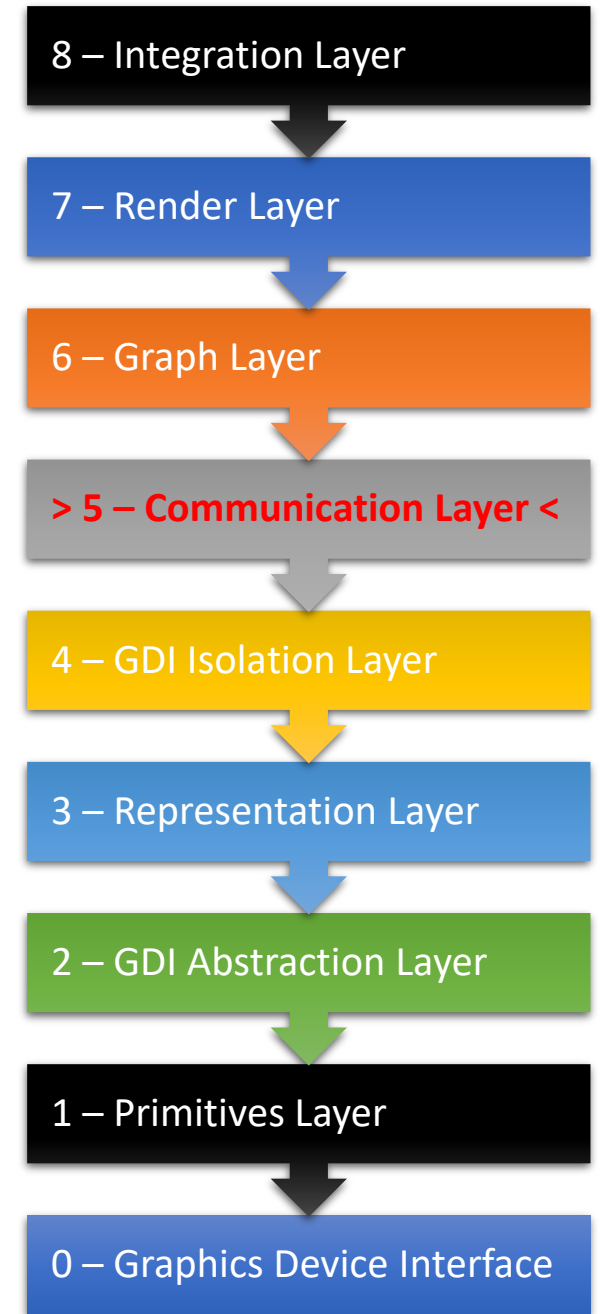
Render  
Commands

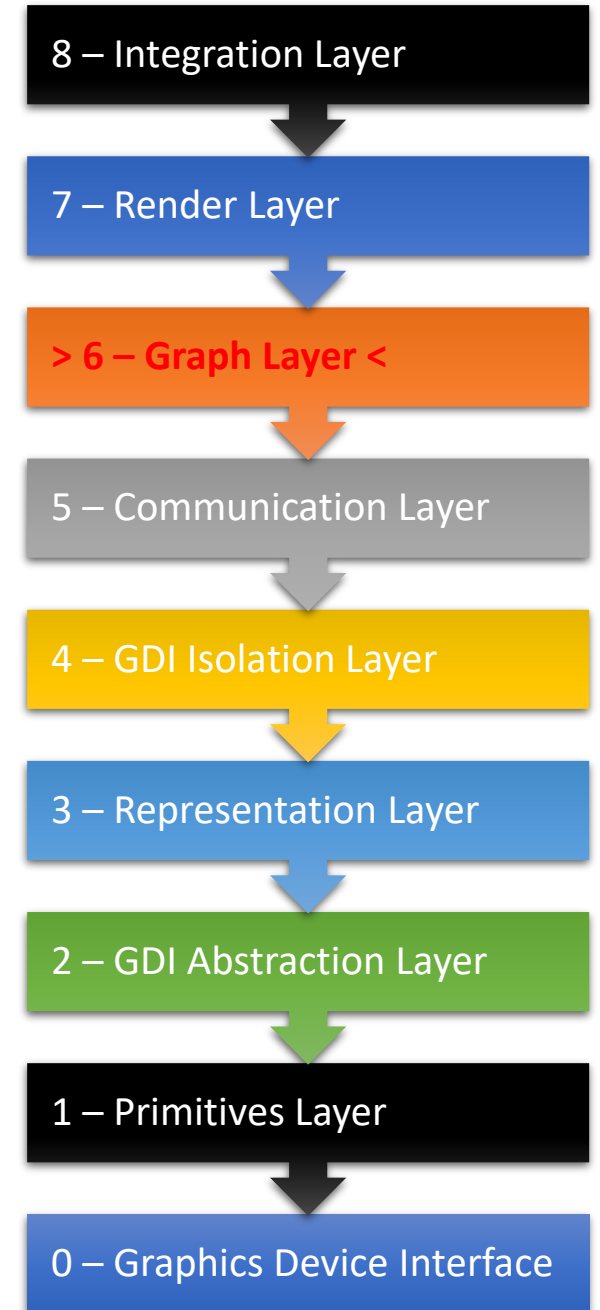


Asset  
Loader

Command  
Queue

Swap Chain







Geometry  
Renderer

Deferred  
Renderer

Forward  
Renderer

VFX  
Renderer

8 – Integration Layer

> 7 – Render Layer <

6 – Graph Layer

5 – Communication Layer

4 – GDI Isolation Layer

3 – Representation Layer

2 – GDI Abstraction Layer

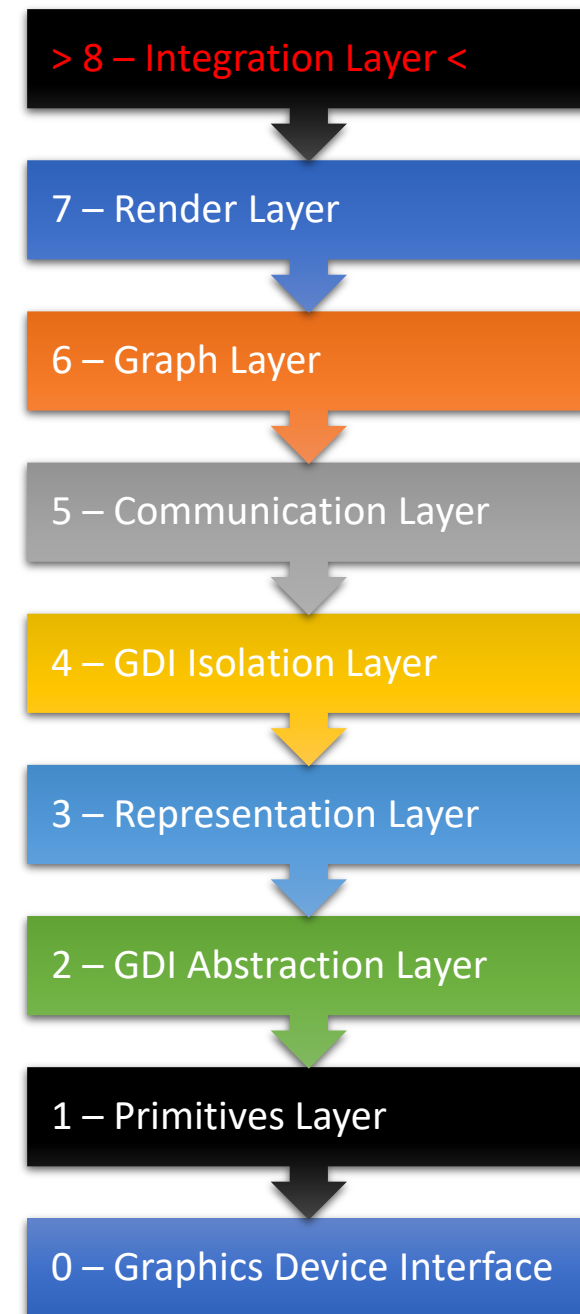
1 – Primitives Layer

0 – Graphics Device Interface

Render  
Director

Scene  
Renderer

UI  
Renderer



## Redeferred rendering – Experimental concept

Let's split diffused and specular contributions of the reflectance part of the rendering equation

$$\begin{aligned} L_o(\mathbf{p}, \omega_o) &= \\ &= \int_{\Omega} \left( k_d \frac{c}{\pi} + \frac{DFG}{4(\omega_o \cdot \mathbf{n})(\omega_i \cdot \mathbf{n})} \right) L_i(\mathbf{n} \cdot \omega_i) d\omega_i = \\ &= \int_{\Omega} \left( k_d \frac{c}{\pi} L_i(\mathbf{n} \cdot \omega_i) + \frac{DFG}{4(\omega_o \cdot \mathbf{n})(\omega_i \cdot \mathbf{n})} L_i(\mathbf{n} \cdot \omega_i) \right) d\omega_i = \\ &= \int_{\Omega} \left( k_d \frac{c}{\pi} L_i(\mathbf{n} \cdot \omega_i) + \frac{DFG}{4(\omega_o \cdot \mathbf{n})} L_i \right) d\omega_i = \\ &= \int_{\Omega} k_d \frac{c}{\pi} L_i(\mathbf{n} \cdot \omega_i) d\omega_i + \int_{\Omega} \frac{DFG}{4(\omega_o \cdot \mathbf{n})} L_i d\omega_i \end{aligned}$$

$$L_d(\mathbf{p}, \omega_o) = \int_{\Omega} k_d \frac{c}{\pi} L_i(\mathbf{n} \cdot \omega_i) d\omega_i$$

$$L_s(\mathbf{p}, \omega_o) = \int_{\Omega} \frac{DFG}{4(\omega_o \cdot \mathbf{n})} L_i d\omega_i$$

$$L_o(\mathbf{p}, \omega_o) = L_d(\mathbf{p}, \omega_o) + L_s(\mathbf{p}, \omega_o)$$

## Diffused

$$\begin{aligned}
L_d(\mathbf{p}, \omega_o) &= \int_{\Omega} k_d \frac{c}{\pi} L_i(\mathbf{n} \cdot \omega_i) d\omega_i = \frac{c}{\pi} \int_{\Omega} k_d L_i(\mathbf{n} \cdot \omega_i) d\omega_i = \\
&= \frac{c}{\pi} \int_{\Omega} (1 - k_s)(1 - m) L_i(\mathbf{n} \cdot \omega_i) d\omega_i = \\
&= (1 - m) \frac{c}{\pi} \int_{\Omega} (1 - k_s) L_i(\mathbf{n} \cdot \omega_i) d\omega_i \approx \\
&\approx (1 - m) \frac{c}{\pi} \int_{\Omega} (1 - (F_0 + (1 - F_0)P_i)) L_i(\mathbf{n} \cdot \omega_i) d\omega_i = \\
&= (1 - m) \frac{c}{\pi} \int_{\Omega} (1 - F_0 - (1 - F_0)P_i) L_i(\mathbf{n} \cdot \omega_i) d\omega_i = \\
&= (1 - m) \frac{c}{\pi} \int_{\Omega} ((1 - F_0) - (1 - F_0)P_i) L_i(\mathbf{n} \cdot \omega_i) d\omega_i = \\
&= (1 - m) \frac{c}{\pi} \int_{\Omega} (1 - F_0)(1 - P_i) L_i(\mathbf{n} \cdot \omega_i) d\omega_i = \\
&= (1 - m) \frac{c}{\pi} (1 - F_0) \int_{\Omega} (1 - P_i) L_i(\mathbf{n} \cdot \omega_i) d\omega_i \\
L_{d_g} &= \int_{\Omega} (1 - P_i) L_i(\mathbf{n} \cdot \omega_i) d\omega_i \\
L_d(\mathbf{p}, \omega_o) &= (1 - m) \frac{c}{\pi} (1 - F_0) L_{d_g}
\end{aligned}$$

Fresnel-Schlick approximation:

$$k_s = F = F_0 + (1 - F_0)(1 - (h_i \cdot \omega_o))^5$$

$$P_i = (1 - (h_i \cdot \omega_o))^5$$

$$k_s = F = F_0 + (1 - F_0)P_i$$

## Specular

$$\begin{aligned}
L_s(\mathbf{p}, \omega_o) &= \int_{\Omega} \frac{DFG}{4(\omega_o \cdot \mathbf{n})} L_i d\omega_i = \frac{1}{4(\omega_o \cdot \mathbf{n})} \int_{\Omega} DFGL_i d\omega_i = \\
&= \frac{1}{4(\omega_o \cdot \mathbf{n})} \int_{\Omega} (F_0 + (1 - F_0)P_i) DGL_i d\omega_i = \\
&= \frac{1}{4(\omega_o \cdot \mathbf{n})} \int_{\Omega} F_0 DGL_i + (1 - F_0)P_i DGL_i d\omega_i = \\
&= \frac{1}{4(\omega_o \cdot \mathbf{n})} \left( \int_{\Omega} F_0 DGL_i d\omega_i + \int_{\Omega} (1 - F_0)P_i DGL_i d\omega_i \right) = \\
&= \frac{1}{4(\omega_o \cdot \mathbf{n})} \left( F_0 \int_{\Omega} DGL_i d\omega_i + (1 - F_0) \int_{\Omega} P_i DGL_i d\omega_i \right) = \\
&= \frac{1}{4(\omega_o \cdot \mathbf{n})} \left( F_0 \int_{\Omega} D \frac{(\omega_o \cdot \mathbf{n})}{(\omega_o \cdot \mathbf{n})(1 - k) + k} \times \frac{(\omega_i \cdot \mathbf{n})}{(\omega_i \cdot \mathbf{n})(1 - k) + k} L_i d\omega_i + (1 - F_0) \int_{\Omega} P_i D \frac{(\omega_o \cdot \mathbf{n})}{(\omega_o \cdot \mathbf{n})(1 - k) + k} \times \frac{(\omega_i \cdot \mathbf{n})}{(\omega_i \cdot \mathbf{n})(1 - k) + k} L_i d\omega_i \right) = \\
&= \frac{1}{4(\omega_o \cdot \mathbf{n})} \times \frac{(\omega_o \cdot \mathbf{n})}{(\omega_o \cdot \mathbf{n})(1 - k) + k} \left( F_0 \int_{\Omega} D \frac{(\omega_i \cdot \mathbf{n})}{(\omega_i \cdot \mathbf{n})(1 - k) + k} L_i d\omega_i + (1 - F_0) \int_{\Omega} P_i D \frac{(\omega_i \cdot \mathbf{n})}{(\omega_i \cdot \mathbf{n})(1 - k) + k} L_i d\omega_i \right) = \\
&= \frac{1}{4((\omega_o \cdot \mathbf{n})(1 - k) + k)} \left( F_0 \int_{\Omega} D \frac{(\omega_i \cdot \mathbf{n})}{(\omega_i \cdot \mathbf{n})(1 - k) + k} L_i d\omega_i + (1 - F_0) \int_{\Omega} P_i D \frac{(\omega_i \cdot \mathbf{n})}{(\omega_i \cdot \mathbf{n})(1 - k) + k} L_i d\omega_i \right) = \\
&= \frac{1}{4((\omega_o \cdot \mathbf{n})(1 - k) + k)} \left( F_0 \int_{\Omega} \frac{\alpha^2}{\pi((\mathbf{n} \cdot \mathbf{h}_i)^2(\alpha^2 - 1) + 1)^2} \times \frac{(\omega_i \cdot \mathbf{n})}{(\omega_i \cdot \mathbf{n})(1 - k) + k} L_i d\omega_i + (1 - F_0) \int_{\Omega} P_i \frac{\alpha^2}{\pi((\mathbf{n} \cdot \mathbf{h}_i)^2(\alpha^2 - 1) + 1)^2} \times \frac{(\omega_i \cdot \mathbf{n})}{(\omega_i \cdot \mathbf{n})(1 - k) + k} L_i d\omega_i \right) = \\
&= \frac{\alpha^2}{4\pi((\omega_o \cdot \mathbf{n})(1 - k) + k)} \left( F_0 \int_{\Omega} \frac{(\omega_i \cdot \mathbf{n})}{((\mathbf{n} \cdot \mathbf{h}_i)^2(\alpha^2 - 1) + 1)^2((\omega_i \cdot \mathbf{n})(1 - k) + k)} L_i d\omega_i + (1 - F_0) \int_{\Omega} P_i \frac{(\omega_i \cdot \mathbf{n})}{((\mathbf{n} \cdot \mathbf{h}_i)^2(\alpha^2 - 1) + 1)^2((\omega_i \cdot \mathbf{n})(1 - k) + k)} L_i d\omega_i \right) =
\end{aligned}$$

Fresnel-Schlick approximation:

$$k_s = F = F_0 + (1 - F_0)(1 - (h_i \cdot \omega_o))^5$$

$$P_i = (1 - (h_i \cdot \omega_o))^5$$

$$k_s = F = F_0 + (1 - F_0)P_i$$

Smith's method with Schlick-GGX:

$$G = \frac{(\omega_o \cdot \mathbf{n})}{(\omega_o \cdot \mathbf{n})(1 - k) + k} \times \frac{(\omega_i \cdot \mathbf{n})}{(\omega_i \cdot \mathbf{n})(1 - k) + k}$$

$$k = \frac{(\alpha+1)^2}{8} \text{ or } k = \frac{\alpha^2}{2}$$

Trowbridge-Reitz GGX:

$$D = \frac{\alpha^2}{\pi((\mathbf{n} \cdot \mathbf{h}_i)^2(\alpha^2 - 1) + 1)^2}$$

Specular

$$L_s(\mathbf{p}, \omega_o) = \frac{\alpha^2}{4\pi((\omega_o \cdot n)(1 - k) + k)} \left( F_0 \int_{\Omega} \frac{(\omega_i \cdot n)}{((n \cdot h_i)^2(\alpha^2 - 1) + 1)^2((\omega_i \cdot n)(1 - k) + k)} L_i \, d\omega_i + (1 - F_0) \int_{\Omega} P_i \frac{(\omega_i \cdot n)}{((n \cdot h_i)^2(\alpha^2 - 1) + 1)^2((\omega_i \cdot n)(1 - k) + k)} L_i \, d\omega_i \right)$$

$$R_i = ((n \cdot h_i)^2(\alpha^2 - 1) + 1)^2((\omega_i \cdot n)(1 - k) + k)$$

$$L_s(\mathbf{p}, \omega_o) = \frac{\alpha^2}{4\pi((\omega_o \cdot n)(1 - k) + k)} \left( F_0 \int_{\Omega} \frac{(\omega_i \cdot n)}{R_i} L_i \, d\omega_i + (1 - F_0) \int_{\Omega} P_i \frac{(\omega_i \cdot n)}{R_i} L_i \, d\omega_i \right)$$

$$L_{sg_1} = \int_{\Omega} \frac{L_i(\omega_i \cdot n)}{R_i} \, d\omega_i$$

$$L_{sg_2} = \int_{\Omega} \frac{P_i L_i(\omega_i \cdot n)}{R_i} \, d\omega_i$$

$$L_s(\mathbf{p}, \omega_o) = \frac{\alpha^2}{4\pi((\omega_o \cdot n)(1 - k) + k)} \left( F_0 L_{sg_1} + (1 - F_0) L_{sg_2} \right)$$

## Recombination

$$k = \frac{(\alpha + 1)^2}{8} \text{ or } k = \frac{\alpha^2}{2}$$

$$R_i = ((n \cdot h_i)^2 (\alpha^2 - 1) + 1)^2 ((\omega_i \cdot n)(1 - k) + k)$$

$$P_i = (1 - (h_i \cdot \omega_o))^5$$

$$L_{d_g} = \int_{\Omega} (1 - P_i) L_i(n \cdot \omega_i) d\omega_i$$

$$L_{s_{g1}} = \int_{\Omega} \frac{L_i(\omega_i \cdot n)}{R_i} d\omega_i$$

$$L_{s_{g2}} = \int_{\Omega} \frac{P_i L_i(\omega_i \cdot n)}{R_i} d\omega_i$$

$$L_d(p, \omega_o) = (1 - m) \frac{c}{\pi} (1 - F_0) L_{d_g}$$

$$L_s(p, \omega_o) = \frac{\alpha^2}{4\pi((\omega_o \cdot n)(1 - k) + k)} (F_0 L_{s_{g1}} + (1 - F_0) L_{s_{g2}})$$

$$L_o(p, \omega_o) = L_d(p, \omega_o) + L_s(p, \omega_o)$$

We are going to calculate  $L_{d_g}$ ,  $L_{s_{g1}}$  and  $L_{s_{g2}}$  first, looping through all illumination sources (lights / rays).

Then we are going to calculate  $L_d$ ,  $L_s$  and finally  $L_o$  outside the loop.

A portion of calculations is now pushed out of the loop and performed only once, instead of on a per illumination source basis.

## Redeffered rendering

Lets split shading into two stages.

First stage (Illumination pass) will produce  $L_{dg}$ ,  $L_{sg1}$  and  $L_{sg2}$  values and store them in a dedicated buffer, let's call it I-buffer.

Second stage (Material pass) will use those values to calculate  $L_o$ .

I-buffer retains fundamental characteristic of light to be cumulative. This allows first stage to be split into multiple passes using specialized shaders for different types of lights, shadows. Redeferred rendering can be used analogously with most of indirect illumination techniques too and their results can be accumulated to the same I-buffer with a common second stage to be applied.

The first stage depends only on lights and geometry (from the PBR point of view roughness is a simplified representation of microscopic geometry). The variability of the I-buffer is expected to be low (imagine a white world with colorful lights). That makes this stage a good candidate for shading rate reducing techniques. We can safely apply a reduced shading rate everywhere unless there is an illumination discontinuity (hard shadow coming from an illumination source considered by a particular shader) or geometry edge – this will require tuning based on visual quality.

The second stage does not have to iterate over illumination sources, so it will be used to refine results by integrating them with materials at full resolution. We still consider geometry during the refining second stage, providing us with differentiation of pixels shaded together in the first pass. This fully covers artifacts of the reduced shading rate from the first stage, making them completely imperceptible.

$$k = \frac{(\alpha + 1)^2}{8} \text{ or } k = \frac{\alpha^2}{2}$$

$$R_i = ((n \cdot h_i)^2(\alpha^2 - 1) + 1)^2((\omega_i \cdot n)(1 - k) + k)$$

$$P_i = (1 - (h_i \cdot \omega_o))^5$$

$$L_{dg} = \int_{\Omega} (1 - P_i) L_i(n \cdot \omega_i) d\omega_i$$

$$L_{sg1} = \int_{\Omega} \frac{L_i(\omega_i \cdot n)}{R_i} d\omega_i$$

$$L_{sg2} = \int_{\Omega} \frac{P_i L_i(\omega_i \cdot n)}{R_i} d\omega_i$$

$$L_d(p, \omega_o) = (1 - m) \frac{c}{\pi} (1 - F_0) L_{dg}$$

$$L_s(p, \omega_o) = \frac{\alpha^2}{4\pi((\omega_o \cdot n)(1 - k) + k)} (F_0 L_{sg1} + (1 - F_0) L_{sg2})$$

$$L_o(p, \omega_o) = L_d(p, \omega_o) + L_s(p, \omega_o)$$



## Redeferred rendering - considerations

Bottleneck of memory throughput:

- A. Precalculating  $(\omega_o \cdot n)$  in geometry pass – no need for fetching normal and depth in the second stage
- B. Smart packing of G-Buffer to separate values used only in the second stage from those used in both stages

Light bleeding over geometry edges, aliased shadow edges and aliasing in strongly illuminated pixels due to greater aliasing in I-Buffer than in G-Buffer:

- A. VRCS with dedicated shading rate attachments per illumination source (direct, shadows, reflections, indirect diffuse, etc.) and shading into an intermediary render target (to have a base for producing attachment texture)
- B. Separate, specialized upscaling for diffuse and specular
- C. Using G-Buffer as additional information for upscaling (edge preservation)
- D. Temporal upscaling

Separate diffuse and specular aspects of illumination when used with temporal techniques separately improve quality (e.g. different types of reprojection: hit point and motion vector - [Tomasz Stachowiak: Stochastic all the things: Raytracing in hybrid real-time rendering](#))

Final frame still needs antialiasing

Other shading models may not be compatible with this technique!

Complete incompatibility with translucency.

$$k = \frac{(\alpha + 1)^2}{8} \text{ or } k = \frac{\alpha^2}{2}$$

$$R_i = ((n \cdot h_i)^2(\alpha^2 - 1) + 1)^2((\omega_i \cdot n)(1 - k) + k)$$

$$P_i = (1 - (h_i \cdot \omega_o))^5$$

$$L_{dg} = \int_{\Omega} (1 - P_i) L_i(n \cdot \omega_i) d\omega_i$$

$$L_{sg1} = \int_{\Omega} \frac{L_i(\omega_i \cdot n)}{R_i} d\omega_i$$

$$L_{sg2} = \int_{\Omega} \frac{P_i L_i(\omega_i \cdot n)}{R_i} d\omega_i$$

$$L_d(p, \omega_o) = (1 - m) \frac{c}{\pi} (1 - F_0) L_{dg}$$

$$L_s(p, \omega_o) = \frac{\alpha^2}{4\pi((\omega_o \cdot n)(1 - k) + k)} (F_0 L_{sg1} + (1 - F_0) L_{sg2})$$

$$L_o(p, \omega_o) = L_d(p, \omega_o) + L_s(p, \omega_o)$$

# GPU Memory Buffers Lifetimes

