



Evolution of NV40

Chris Seitz



Progression of Graphics



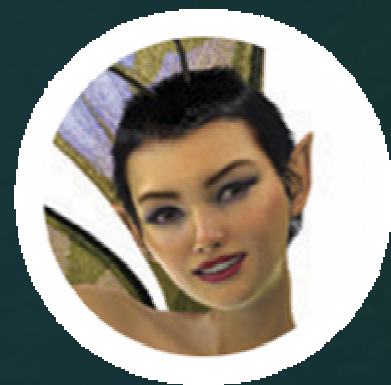
Virtual Fighter
NV1
1 Mtrans



Wanda
NV1x
22 Mtrans



Wolfman
NV2x
63Mtrans



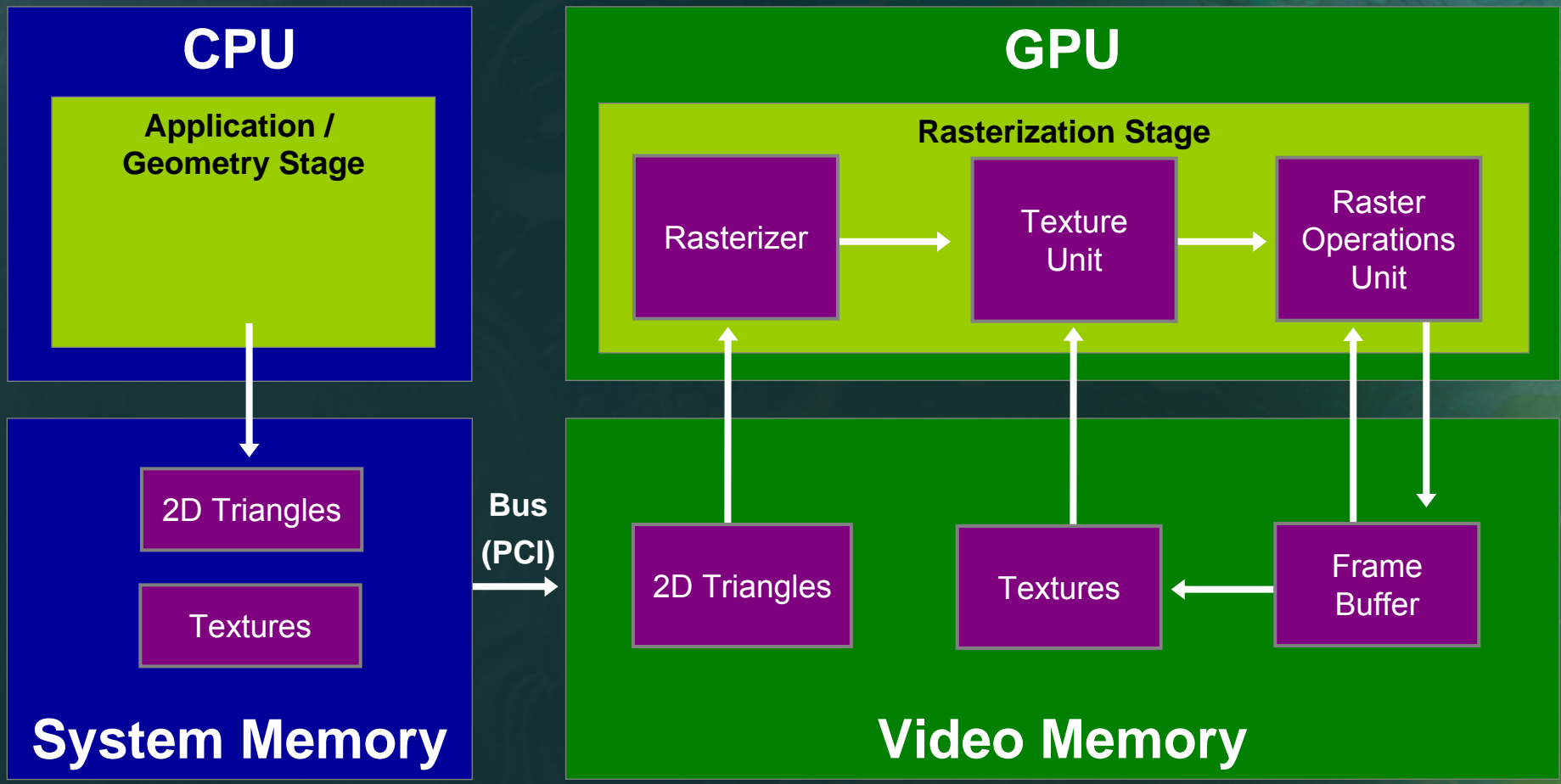
Dawn
NV3x
130Mtrans



Nalu
NV4x
222M



'95-'98: Texture Mapping & Z-Buffer



Multitexturing



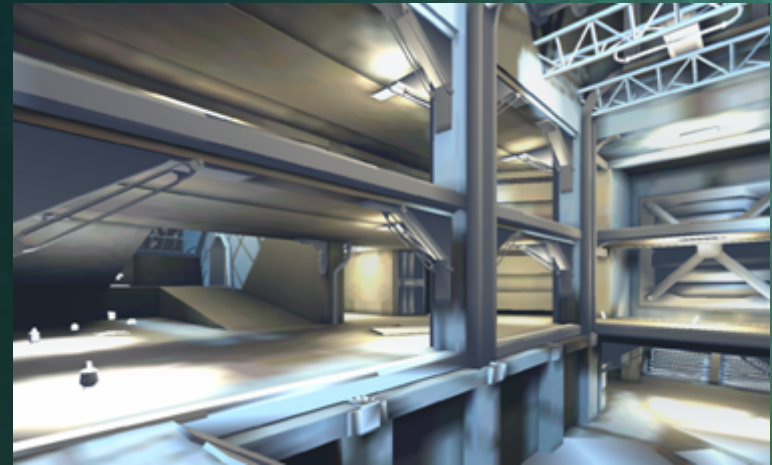
Base Texture

modulated by

Light Map



X



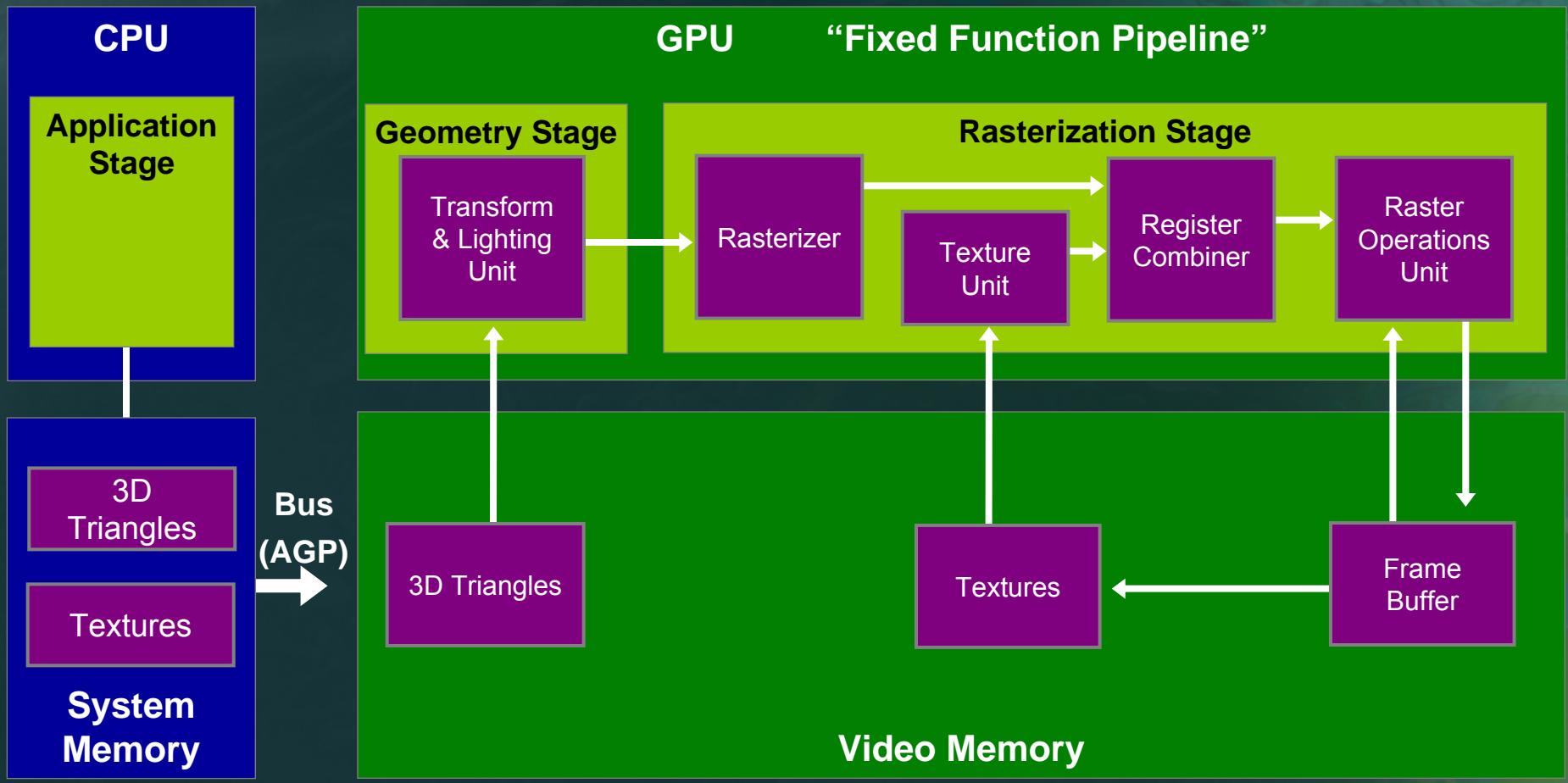
=



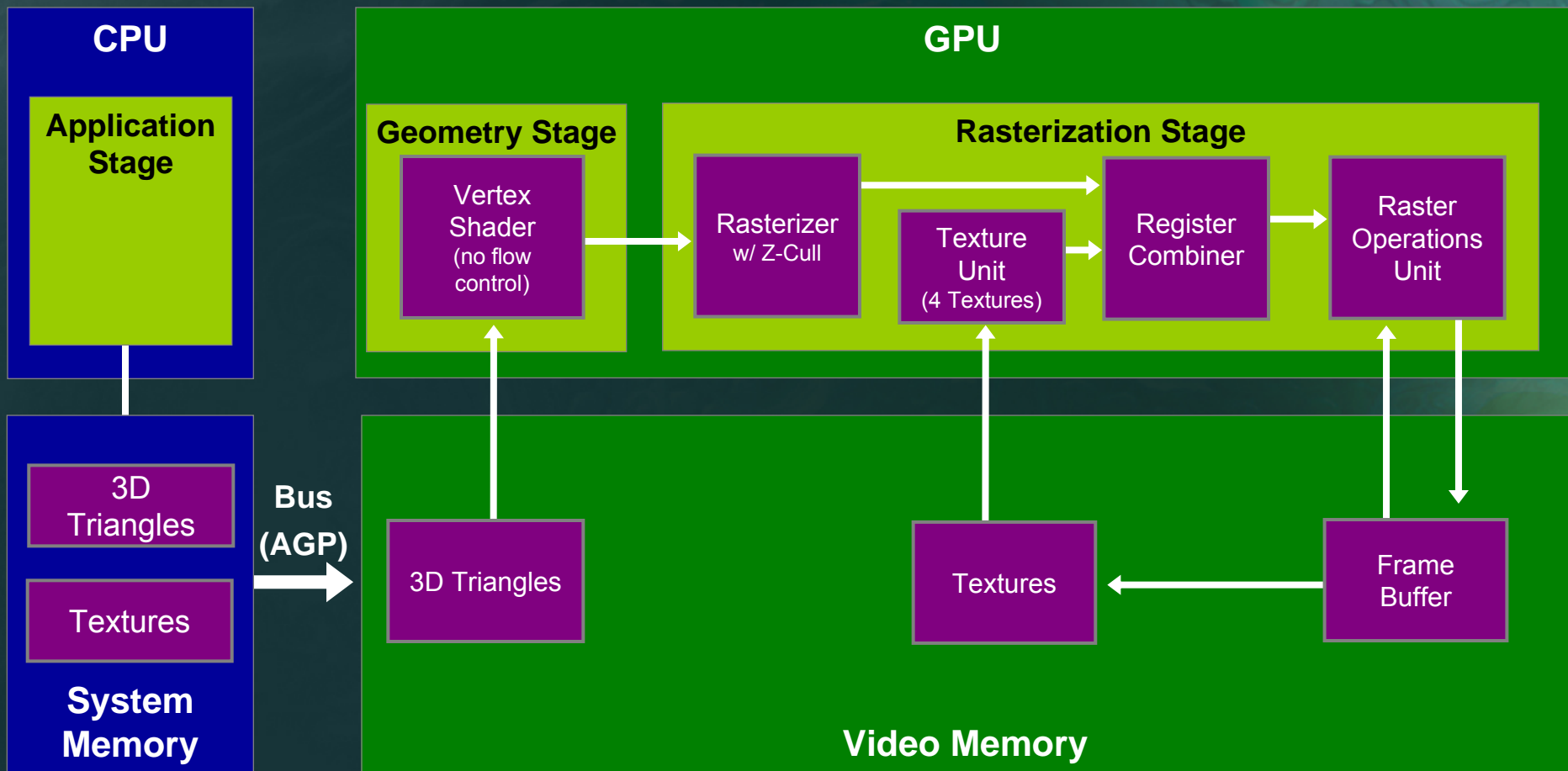
from UT2004 (c)
Epic Games Inc.
Used with permission



1999-2000: Transform and Lighting



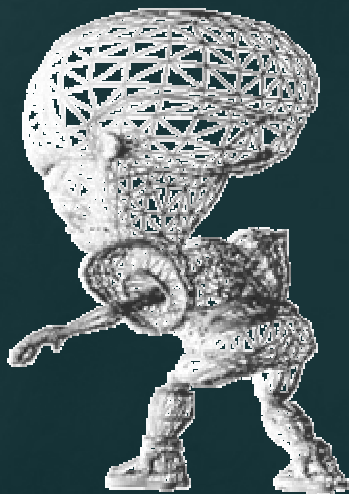
2001: Programmable Vertex Shader





Vertex Shader

- A programmable processor for any per-vertex computation

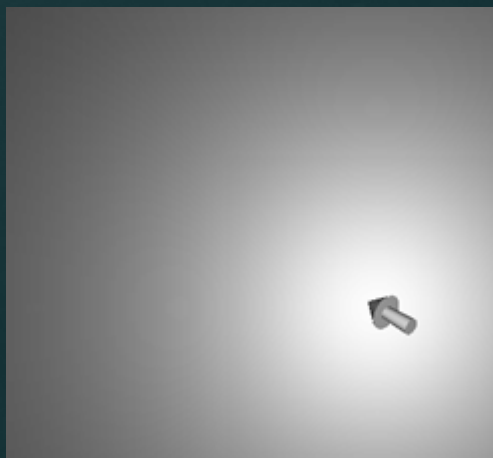


```
void VertexShader(  
    // Input per vertex  
    in float4 positionInModelSpace,  
    in float2 textureCoordinates,  
    in float3 normal,  
  
    // Input per batch of triangles  
    uniform float4x4 modelToProjection,  
    uniform float3 lightDirection,  
  
    // Output per vertex  
    out float4 positionInProjectionSpace,  
    out float2 textureCoordinatesOutput,  
    out float3 color  
)  
{  
    // Vertex transformation  
    positionInProjectionSpace = mul(modelToProjection, positionInModelSpace);  
  
    // Texture coordinates copy  
    textureCoordinatesOutput = textureCoordinates;  
  
    // Vertex color computation  
    color = dot(lightDirection, normal);  
}
```



Bump Mapping

- Bump mapping involves fetching the per-pixel normal from a **normal map** texture (instead of using the interpolated vertex normal) in order to compute lighting at a given pixel



Diffuse light

+



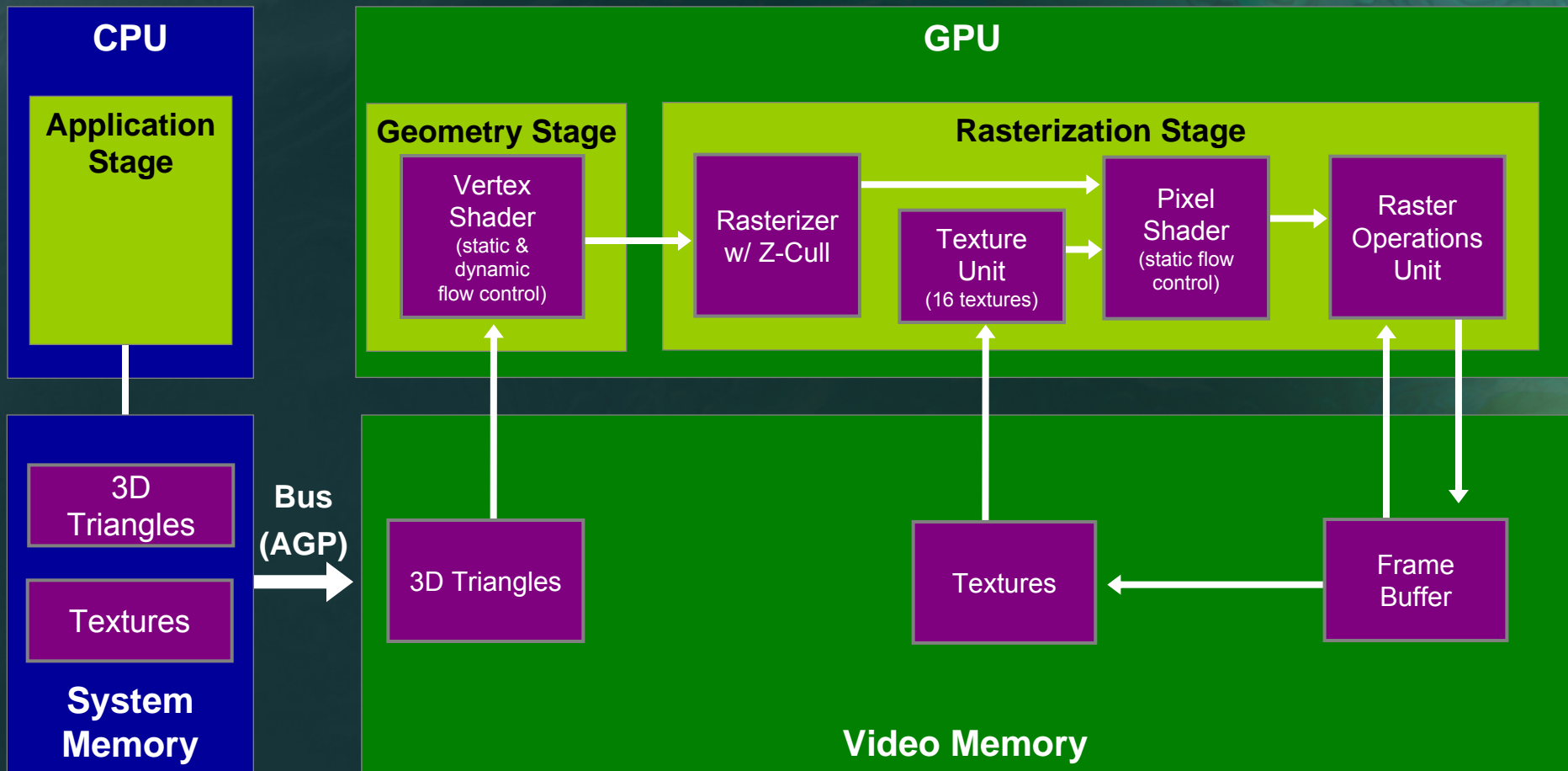
Normal Map

=



Diffuse light with bumps

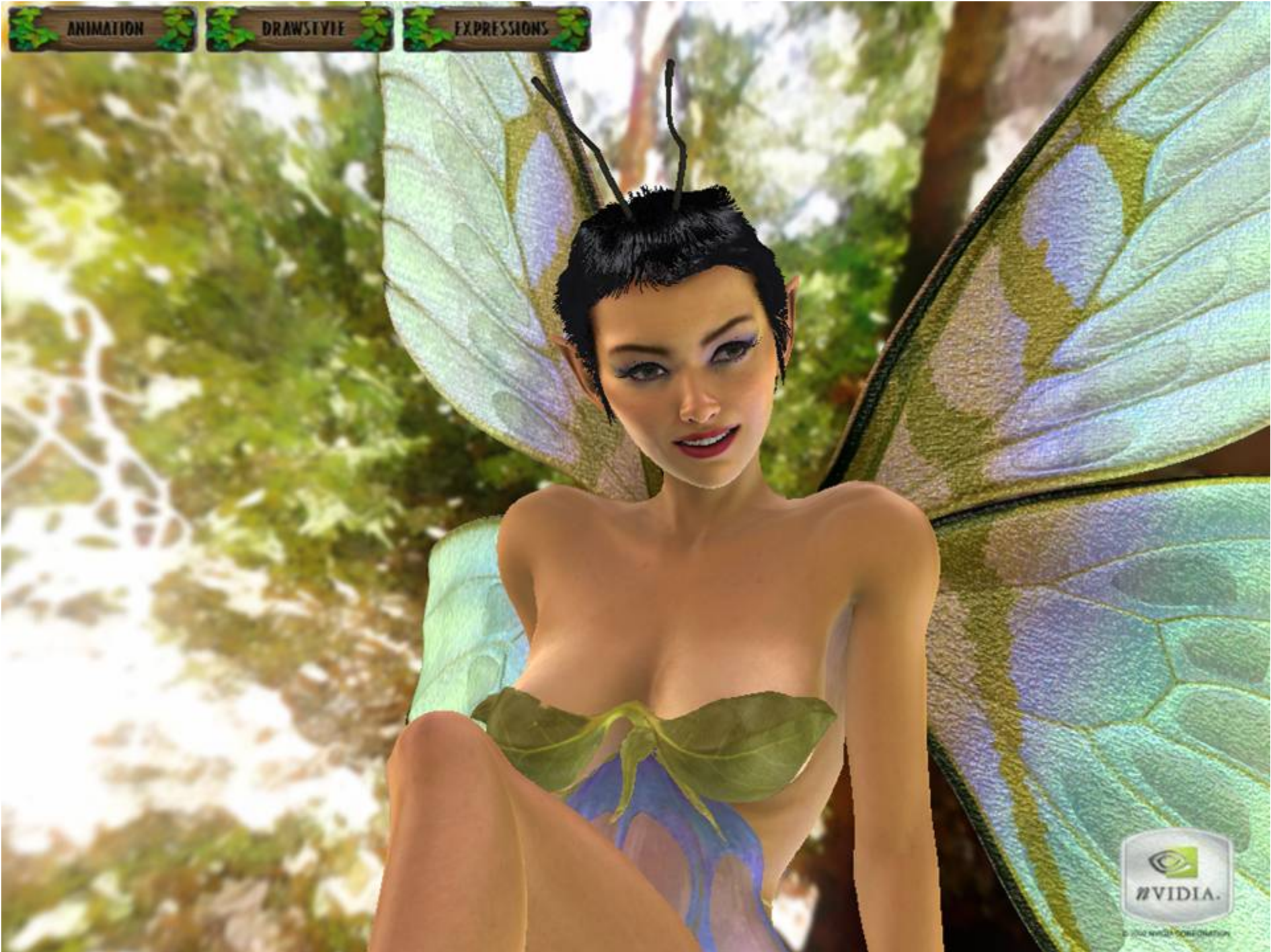
'02- '03: Programmable Pixel Shader



ANIMATION

DRAWSTYLE

EXPRESSIONS

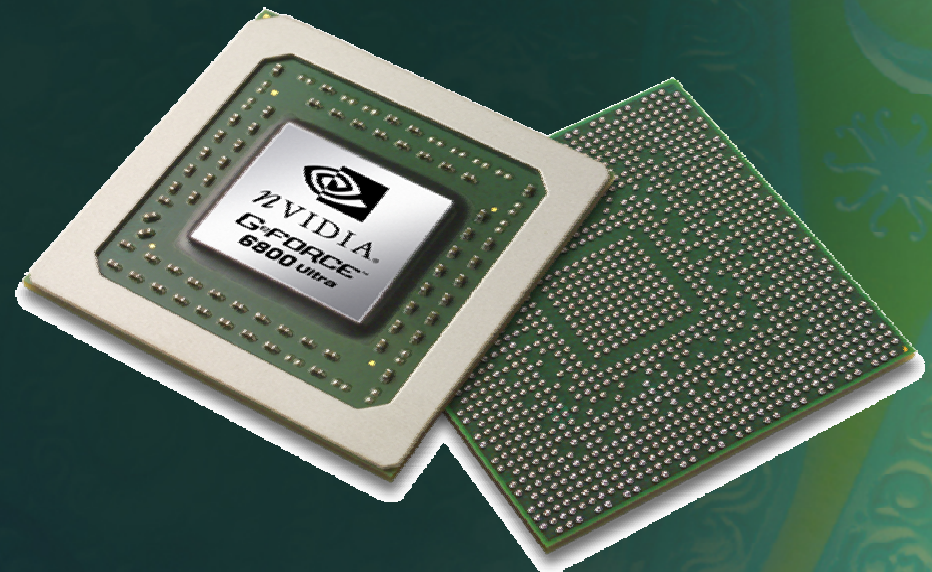


© 2012 NVIDIA CORPORATION

GeForce 6800

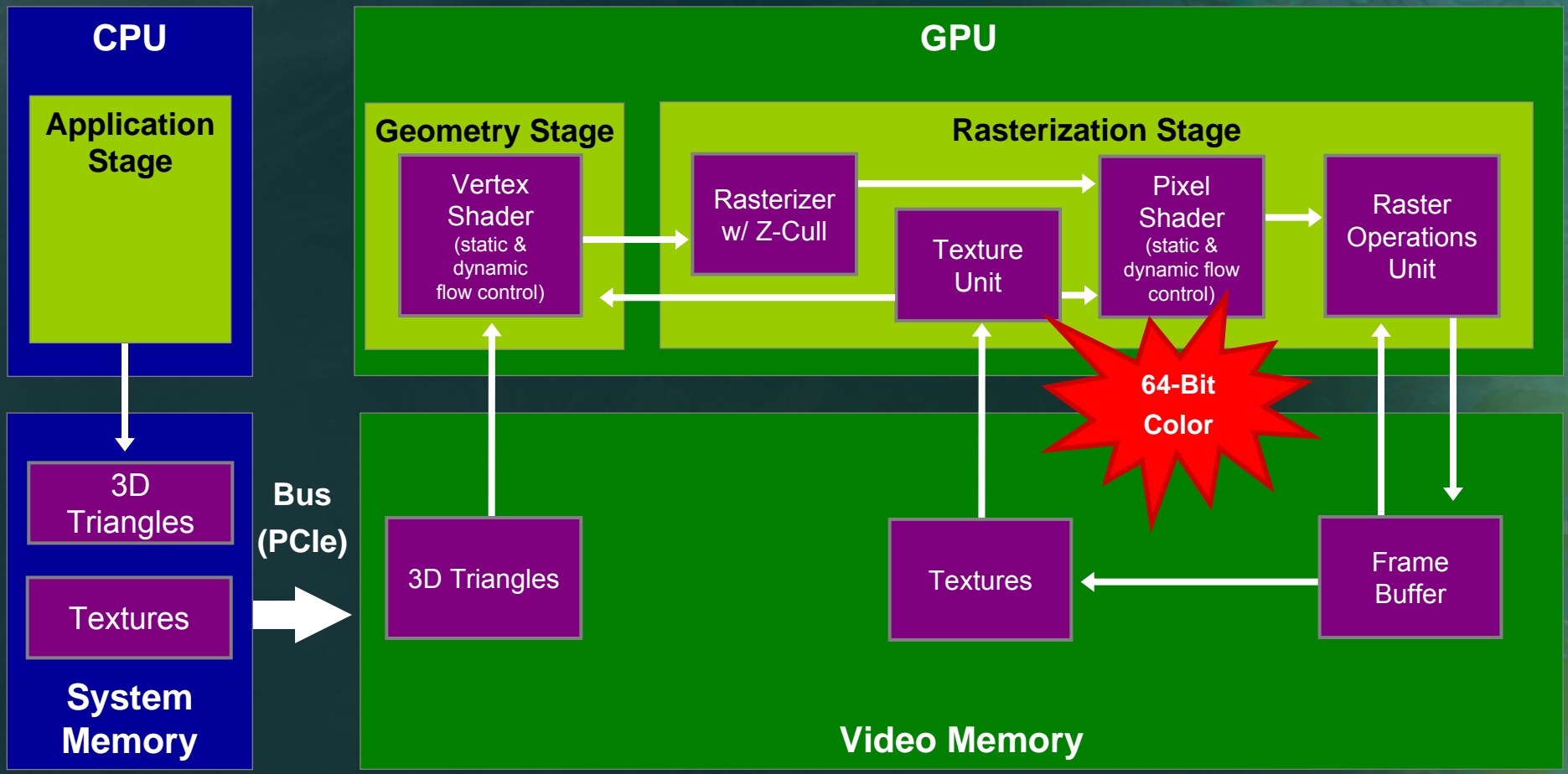
Revolutionary performance & complete Shader Model 3.0

- Complete Native Shader Model 3.0 Support
 - Full support for shader model 3.0
 - Vertex Texture Fetch / Long programs / Pixel Shader flow control
 - Full speed fp32 shading
- OpenEXR High Dynamic Range Rendering
 - Floating point frame buffer blending
 - Floating point texture filtering
- Unparalleled Performance
 - 222M xtors / 0.13um @ IBM
 - 6 vertex units / 16 pixel pipelines
- Next Generation Video
 - VMR / High quality compositing
 - Hardware MPEG encode / decode
 - HDTV Output
- PCI Express





2004: Shader Model 3.0 & 64-Bit Color Support



Shader Model 3.0

- Longer shaders → More complex shading
- Pixel shader:
 - Dynamic flow control → Better performance
 - Derivative instructions → Shader antialiasing
 - Support for 32-bit floating-point precision → Fewer artifacts
 - Face register → Faster two-sided lighting
- Vertex shader:
 - Texture access → Simulation on GPU, displacement mapping
- Geometry Instancing → Better performance



Lord of the Rings™
The Battle for Middle-earth™



Far Cry



SpeedTree

Complete Native DirectX 9 Support



	DX9	DX9.0c
Vertex Shader Model	2.0	3.0
Vtx Shader Instructions	256	2^{16} (65,535)
Displacement Mapping	-	✓
Vertex Texture Fetch	-	✓
Geometry Instancing	-	✓
Dynamic Flow Control	-	✓
Pixel Shader Model	2.0	3.0
Required Shader Precision	fp24	fp32
Pixel Shader Instructions	96	2^{16} (65,535)
Subroutines	-	✓
Loops & Branches	-	✓
Dynamic Flow Control	-	✓

GeForce 6800 Graphics

Unparalleled Performance



- Up to 8x pixel shading performance - *combination of 4x the pipes and 2x the math per pipe*
- 2x vertex shading performance - *MIMD architecture, dual-issue, very efficient branching*
- Next generation UltraShadow - *4x the performance of NV35 32ppc for z/stencil rendering*
- 256-bit DDR3 – *1.1 GHz DDR data rate.*

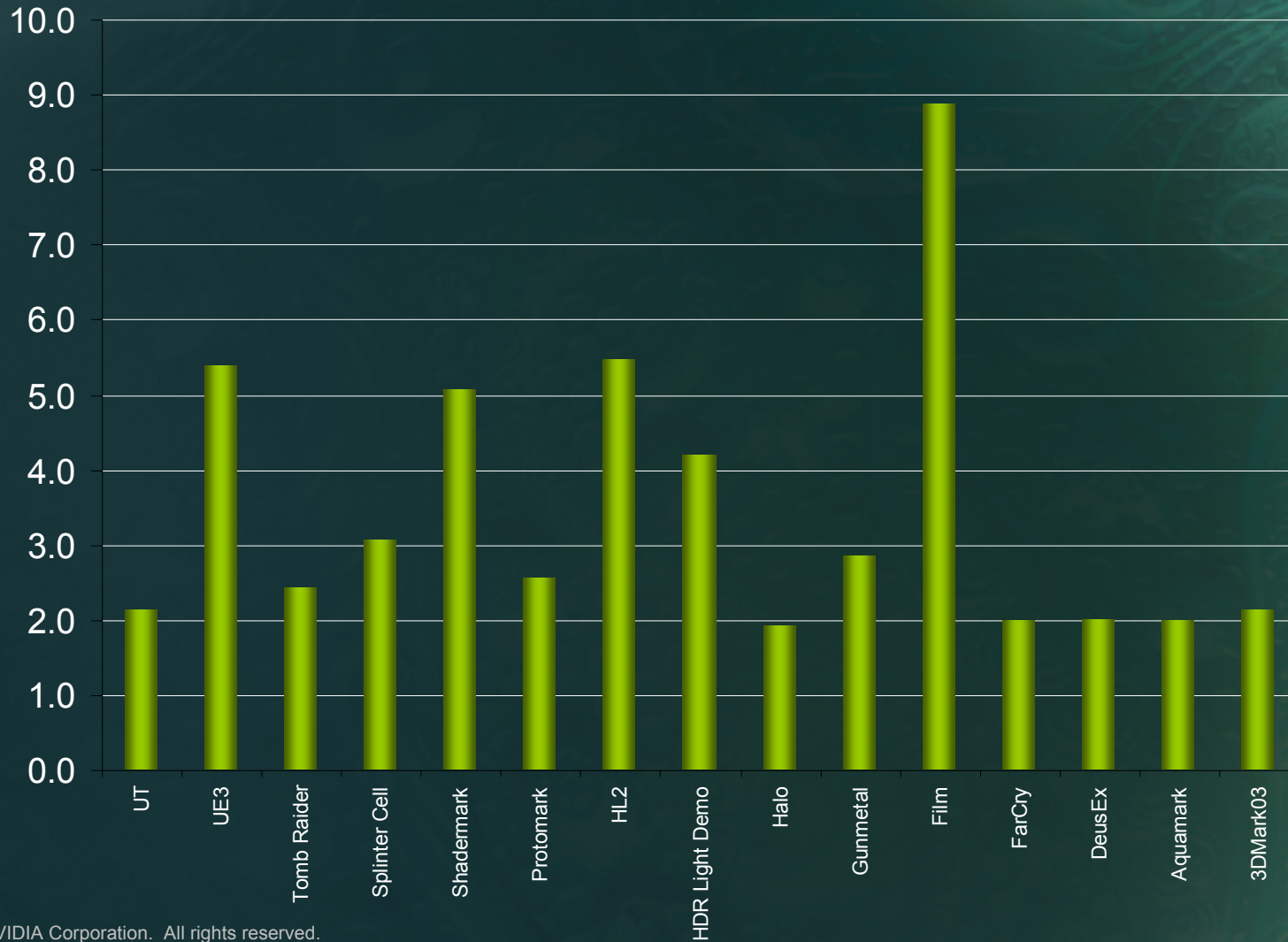


GeForce 6800 series 3D Pipeline



NV40 Shader Performance

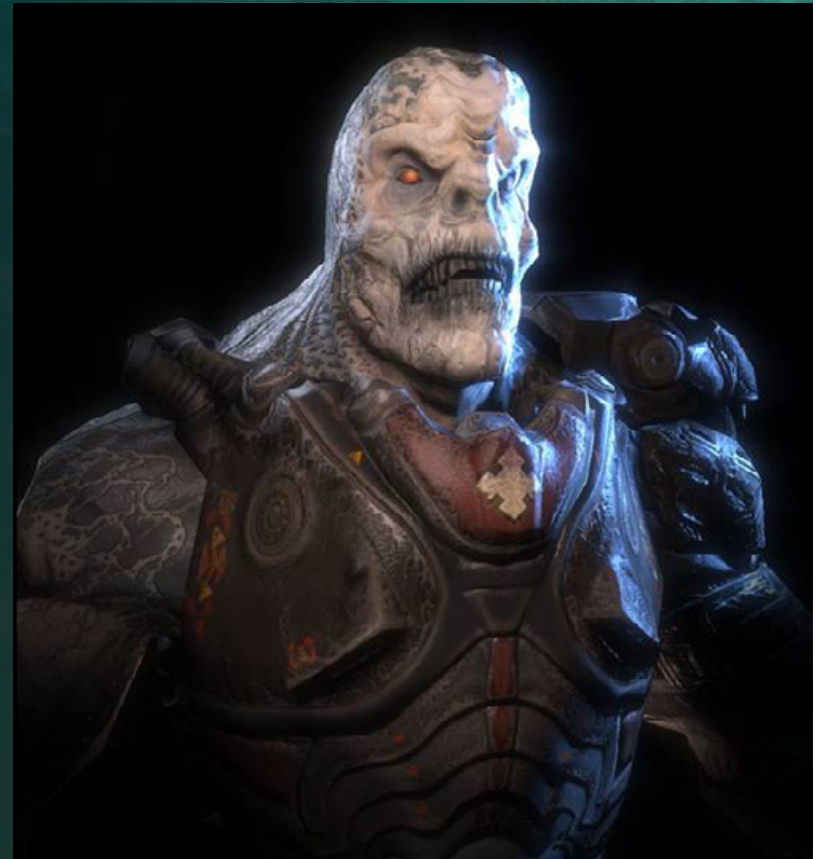
Multiples of GeForce FX 5950 performance on shaders





© 2004 NVIDIA Corporation. All rights reserved.

Shader Model 3.0 / 64-bit Floating Point Processing Unreal Engine 3.0 Running On GeForce 6 Series



***2 Million Triangle Detail Mesh Models
High Dynamic Range Rendering
Fully Customizable Shaders***

UnrealEngine3
Copyright (C) 2004, Epic Games

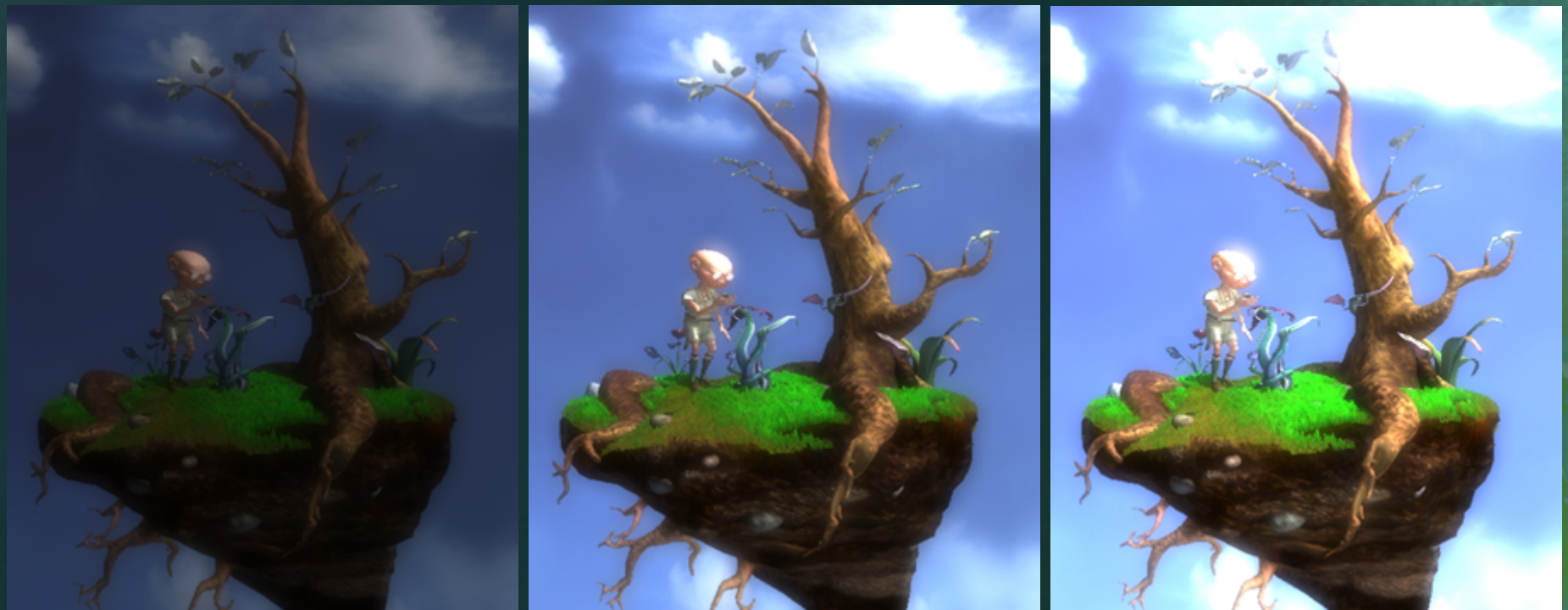
Shader Model 3.0 / 64-bit Floating Point Processing Unreal Engine 3.0 Running On GeForce 6 Series

*100 Million Triangle Source Content Scene
High Dynamic Range Rendering*



Demo: Real-Time Tone Mapping

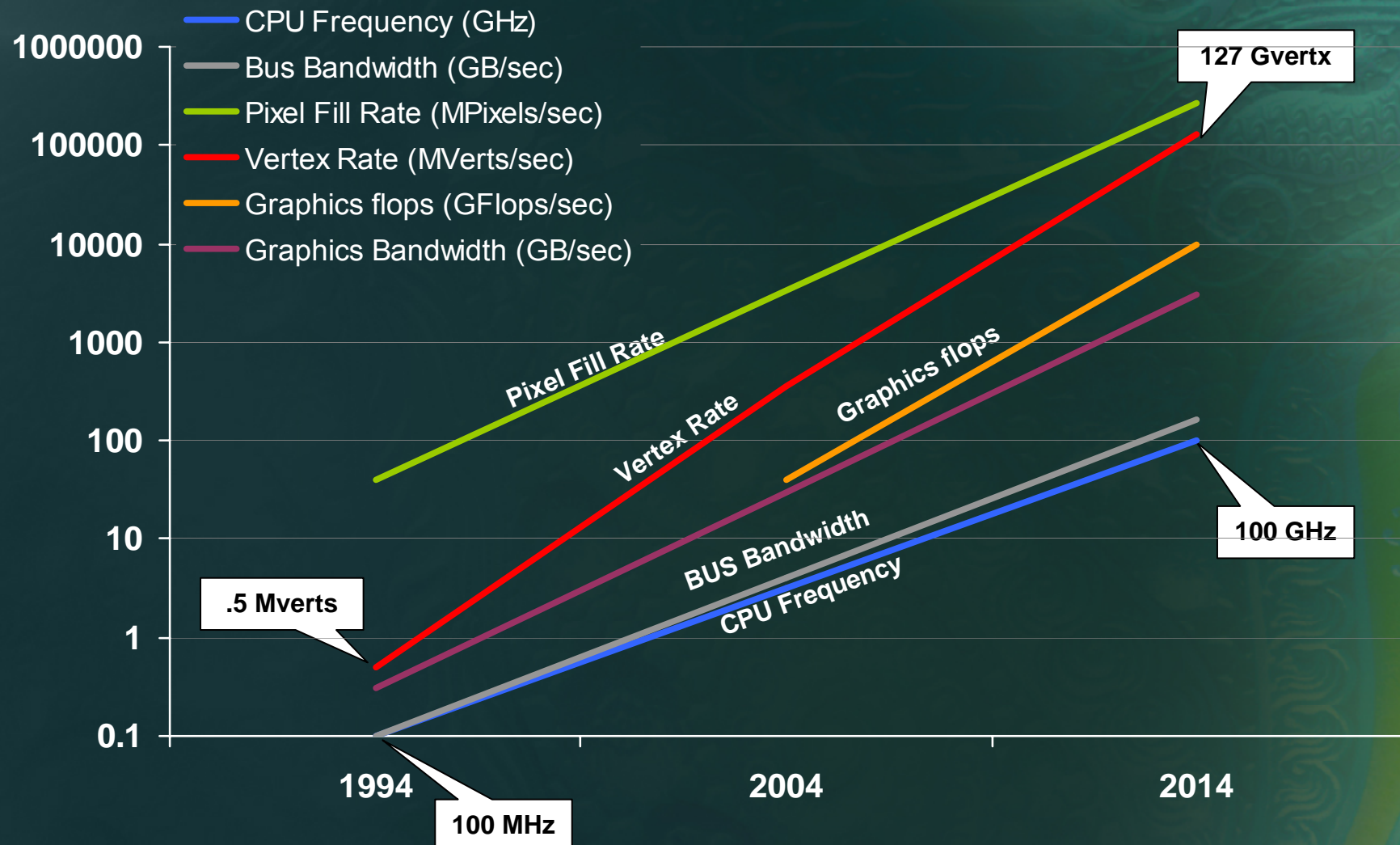
The image is entirely computed in 64-bit color and tone-mapped for display



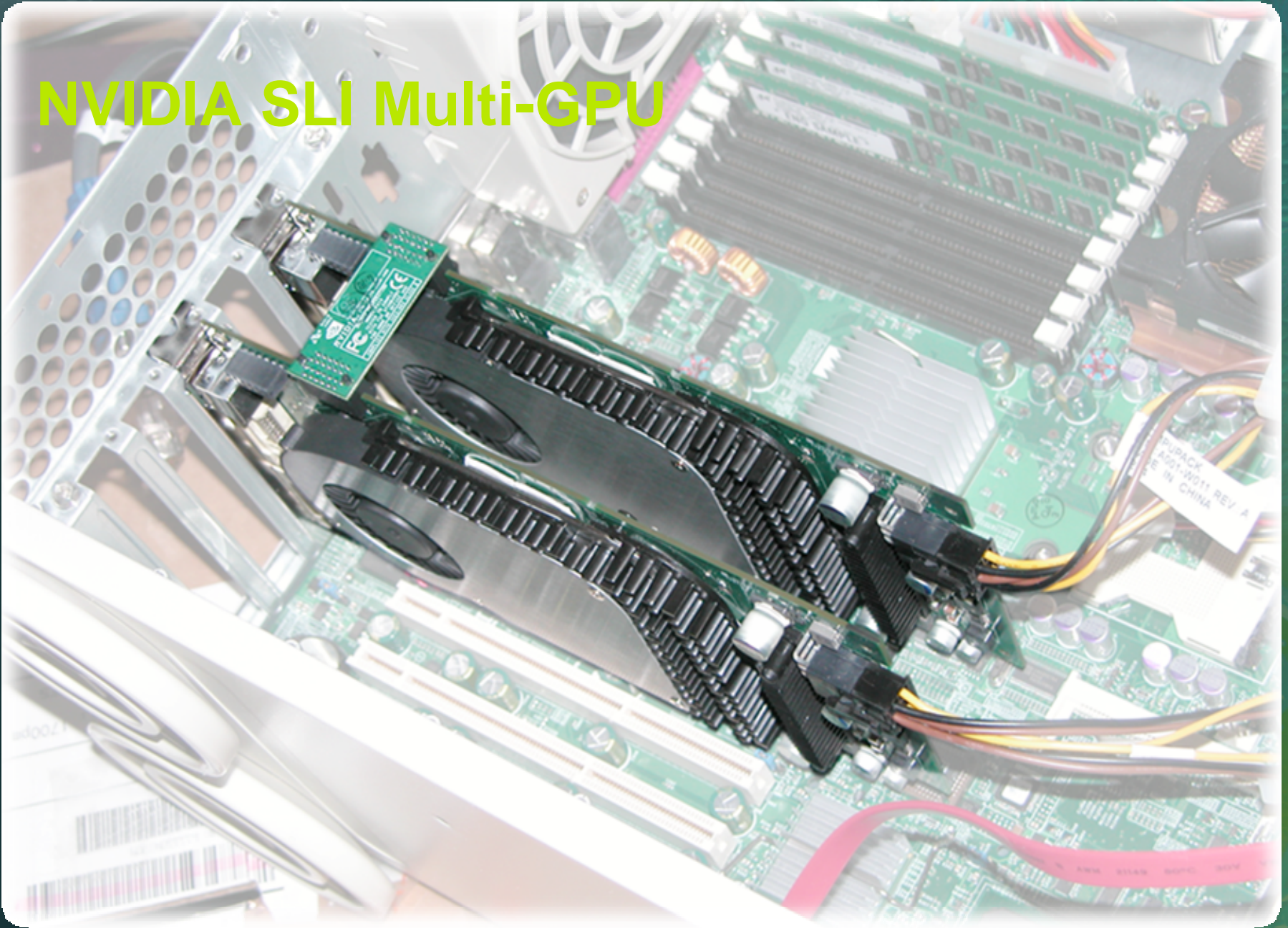
Renderings of the same scene, from low to high exposure



Looking Ahead: Now + 10 years



NVIDIA SLI Multi-GPU



Thank You

