Cinematic Effects II:
The Revenge

Kevin Bjorke, NVIDIA

September 2004

Overview

- Films and Games: Differences and Similarities
  - Visual Qualities & “Look Development”
  - Scale of Production
- Ideas from Cinema, Realized
  - New Tools, Shaders, Ideas
  - Live Examples
  - Getting it into your game engine
  - Getting it into your art pipeline
- Source Code!
  - Source code from all examples is on http://developer.nvidia.com

“MRT” visualization of texture coordinates

“Revenge”???

- *The Story So Far…**
  - Cinematic Effects, via Programmable Shading, are the Most Powerful Artistic Tool Yet for Games
  - But it’s an Uphill Battle
    - Hard to implement and experiment
    - Hard to get into game engines
    - Even harder to debug
- Payback Time.

“Thad” from Animatrix – Character © Silver Pictures

* Part I available at http://developer.nvidia.com/

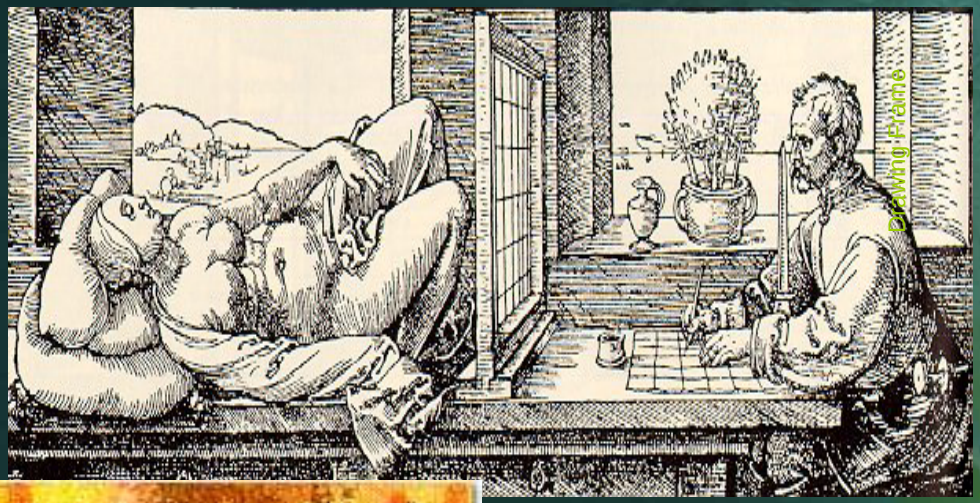
Visual Art and Gaming

- They have always been connected
- World's Oldest Art may be "High Score" tags!

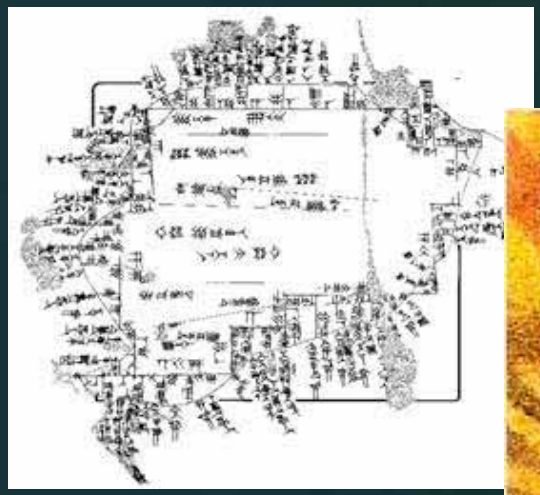
Chauvet Cave – maybe 25,000 B.C.

Geometry and Light

- Computer Graphics is the latest development
- Cinema & Photo
- Optics and Geometry
- “Measured Seeing”



Albrecht Durer



Mesopotamian Survey Map, ca. 2500 B.C.

Chauvet Cave – maybe 25,000 B.C.

Films and Realism

- Films aren't documentaries
- They are vivid stylized illustrations
  - Subjective, Not Objective
  - "Bigger Than Life"
- "Documentary Style" is just that – a *style*
  - "Reality TV" is scripted secretly
  - BBC's "The Office" is scripted obviously

Sylvia ©2003 Focus Features

CGI and “Photorealism”

- “Photorealism” is just another style
- Photos can be highly abstract!
- Nature is full of more stuff than we can handily write a single set of equations to recreate
- Film borrows many image-making ideas from earlier media

Vogue cover, 1950 - Penn

Abstraction Mixed with Precision

Mesopotamian Survey Map, ca. 2500 B.C.

- Maps, and then perspective (as well as literature) developed from… Accounting!
- Accurate depiction was important, but less than abstract issues such as ownership and taxation
- “Artful ommision” was important even back then

CGI, Films, and Painting

- Film borrows lighting and composition from media like painting
- Lighting leads attention
- Lighting sets emotional tone

Scott's *Blade Runner*

Rafael's *Transfiguration*

Look Development

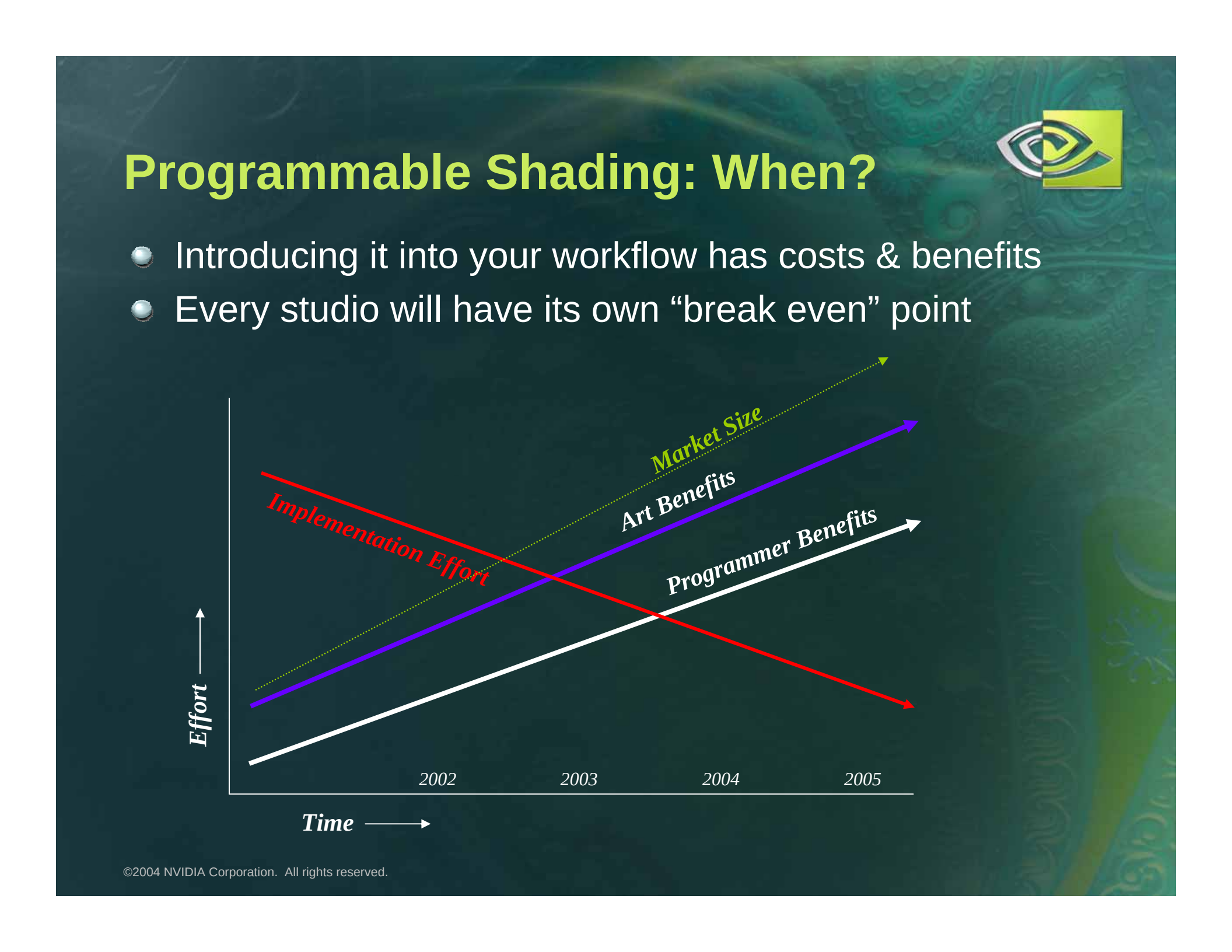
- “Look Development” is when we decide what’s important (and what’s not), and lay down the elements of style for any project (or part of a project)
- The earlier in development that the “look” is determined, the better it is (and the cheaper it is to use)

Look Development in Games

- In gaming, look is often a byproduct of engine design
  - Hard for Artists to Guess at anything other than “Lowest Common Denominator”
  - Design tends to be conservative and safe
  - Concerned with technical limitations
- In films, development is usually done initially without slavish attention to implementation “details” like budget
  - Artists completely free
  - Concerned with story

Test from *Animatrix* – © Silver Pictures

Programmable Shading: When?



- Introducing it into your workflow has costs & benefits
- Every studio will have its own “break even” point

Developing Shaders: Programmers

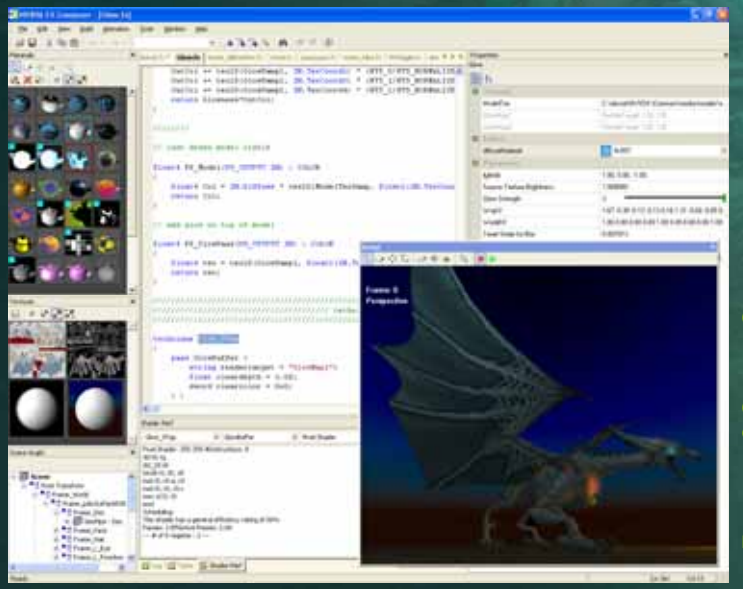
- Shading tools are important for both Programmers and Designers
- To be complete for modern game engines, tools have to support ideas like:
  - Render-To-Texture (RTT)
  - Multiple Render Targets (MRT)
  - Render States like stencil, alpha blend, etc.
  - Custom Texture Maps (e.g. Normalization cubes, noise)
  - Management details to make sure complex ordering matches any specific game engine's render loop
  - Scriptable
- How to get results in *and out* of your game engine, at every stage of production?

Developing Shaders: Artists

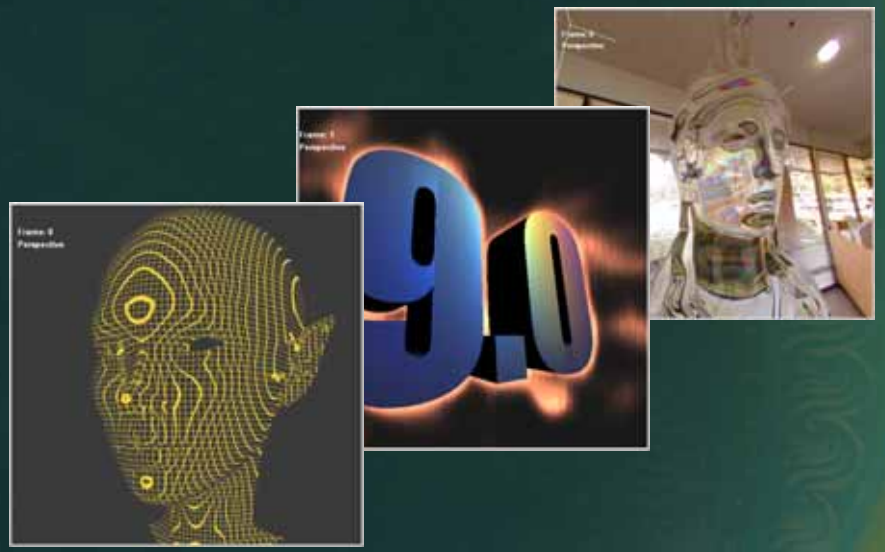
- Artists want to see what they design, not just guess at what it might look like later
- Not just using the correct models, but also the correct lighting environment, so that *the shaders and models developed can really be the ones used in-game.*
- Rendering implementations are typically different in each different DCC application (Maya versus Max versus XSI versus….) – none match real games
- We want to accommodate console-game designers, too – provide a way to see models for different versions of the same game (DX9, DX8, Xbox, PS…)

HLSL & FX Composer

- A Tool Built for the Task
- Combine shaders
- Customize shaders
- Move back and forth without rewrites or additional SDKs and runtime layers
- Performance tuning tools
- C# & VB scripting
- http://www.fxcomposer.com/

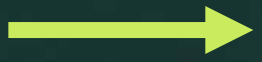


Everquest® Content Courtesy
Sony Online Entertainment Inc.



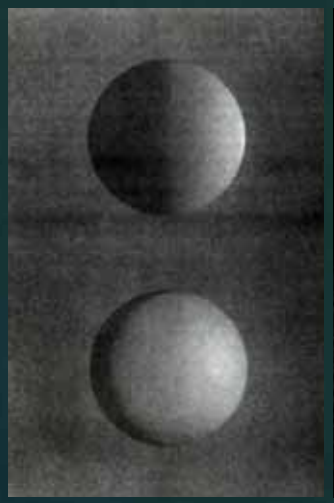
A Shading Sketchbook

- FX Composer gives artists and programmers an environment to play with complex ideas, without needing to write a whole C++ game engine to try them out!



Building a Library of Shaders

- FX Composer ships with lots of sample shaders
  - Any HLSL FX shader can be used, from other shader tools too
- Do experiments, save them and keep them around – you'll use them someday!
- Save, trade, and collect 'em



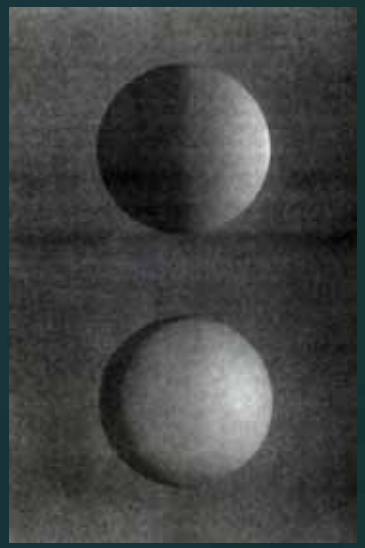
Ruskin's Shading Exercises, 1877

Bjorke's Dumb Mistake, 2003

Sketchbook Example: Turning Pencil Sketches into Shaders

- A shaded sphere is trivial to turn into a shader
- Useful as color reference
- Beware tiny details (like JPEG noise), they smear

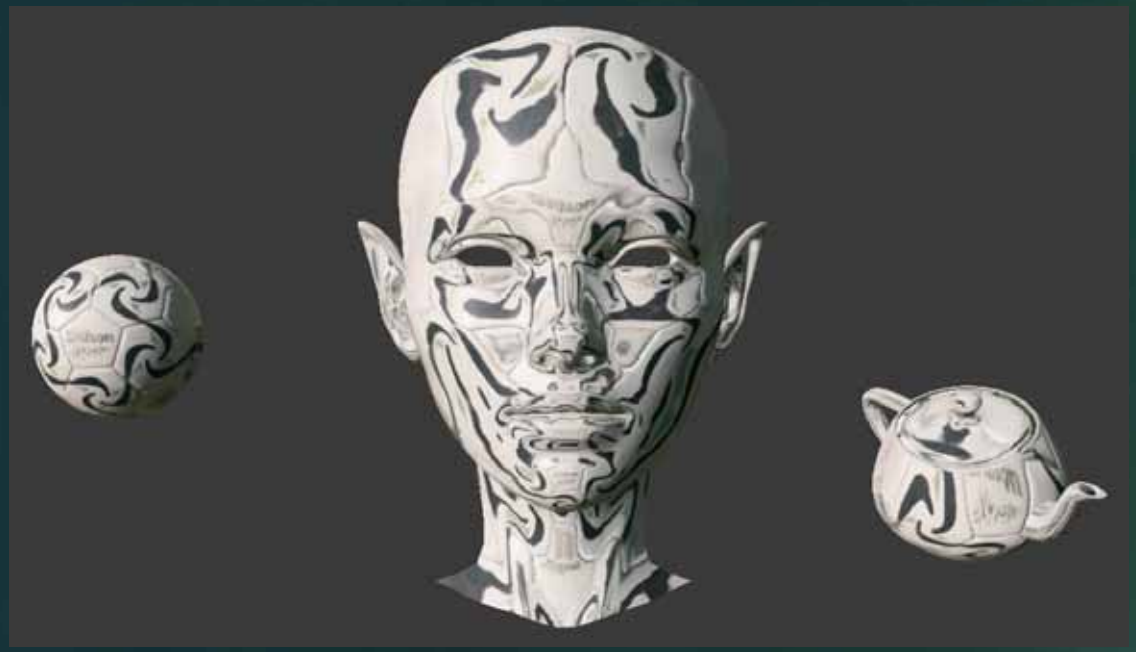
1877



2004

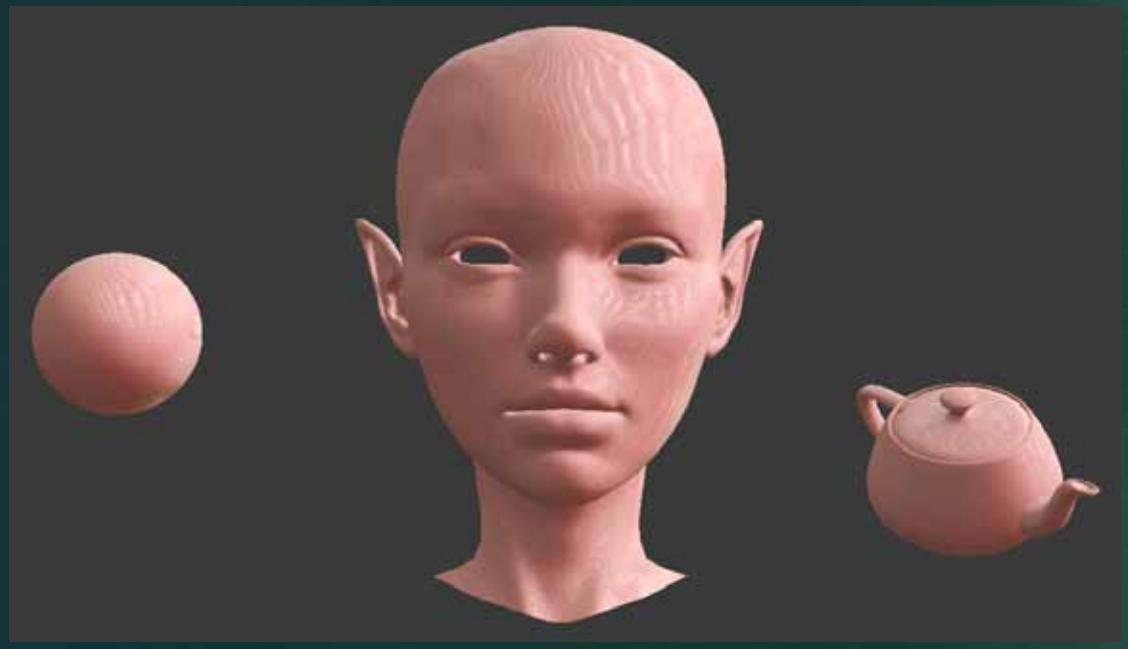
Fanciful examples

- Photos will distort
- This is probably rarely useful, but *cheap* – only ONE cycle!
- Can we do something generically useful with this?



Does the shape have to be a sphere?

- Not if we're willing to do some work with Photoshop
  - I like "Liquify" and the Smudge/Stamp Tools

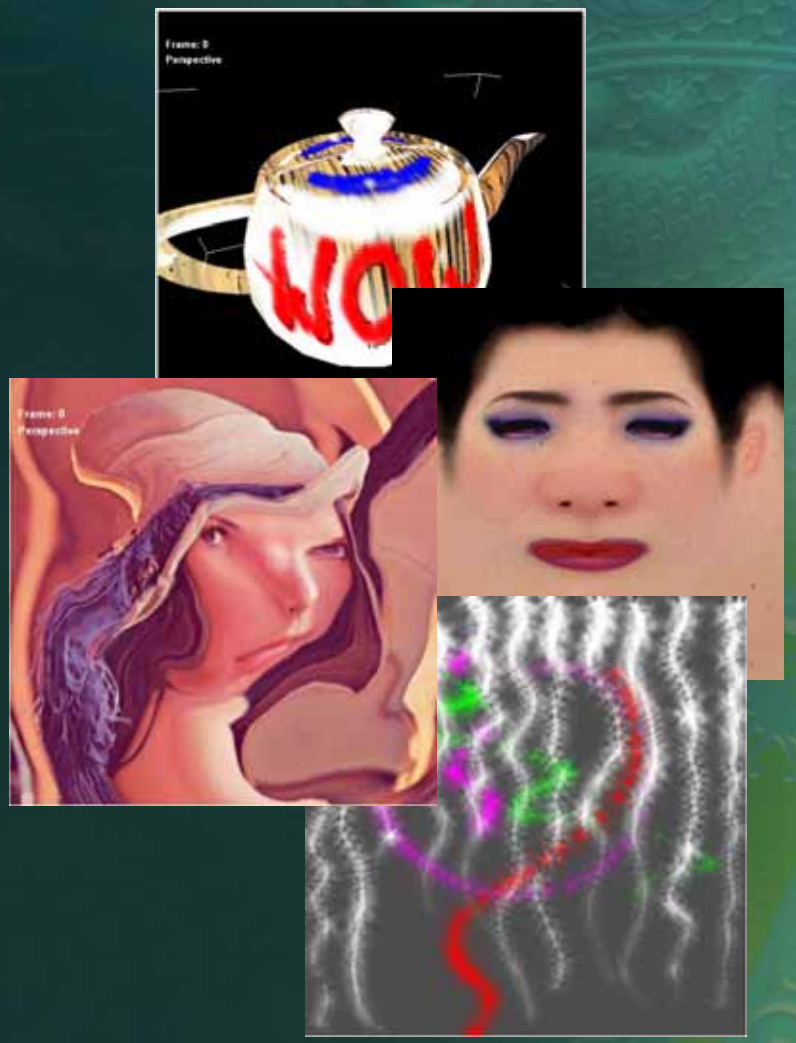


Refining the Color

- Gaussian Blur in the texture to isolate the color
- Great to mix with other shading models

Sketching In FX Composer

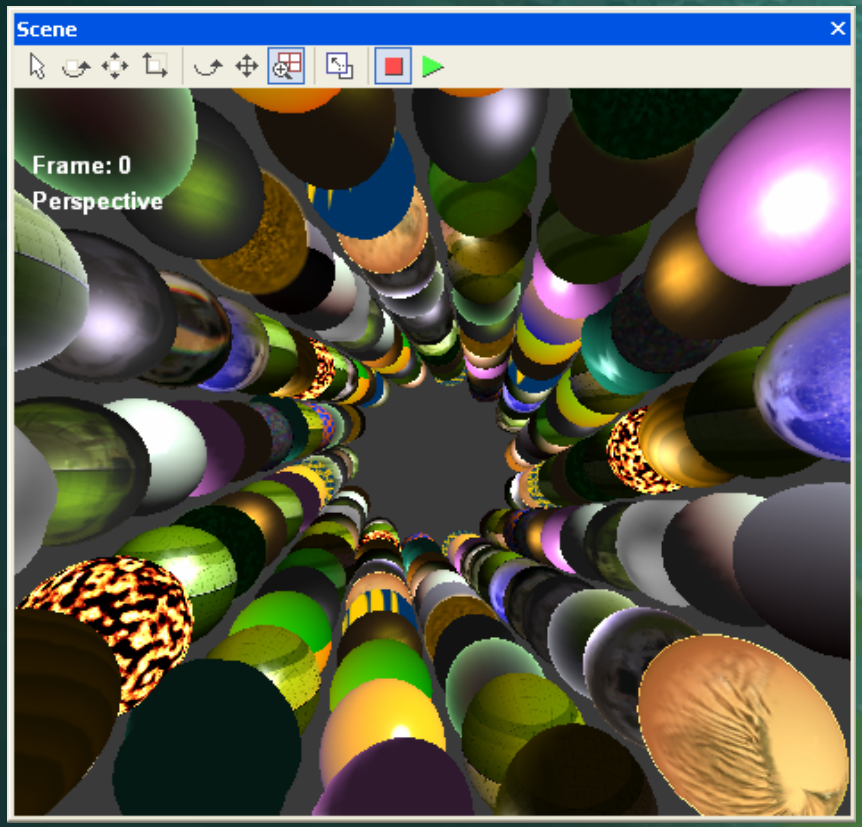
- While we're on the subject of sketching:
- FX Composer lets us intercept mouse events
- We can use this to build mini-apps entirely from FX shaders



Movies: Managing Scale

- Films have Massive Scale
- Lots of Geometry
- *Lots of Shaders*
  - *Toy Story:* 1300 shaders
  - *Bugs:* Double
  - *Monsters Inc:* “thousands”
- Lots of Compositing Layers (sometimes hundreds)
- Scriptable Tools (Perl, C#, VB, Python, Mel… you name it!)
- *Long Schedules*
  - Instant Rendering shaves off *some* schedule…

FX Composer: Managing Scale



"See all the shaders in a directory"
-- Scene Generated by C# Script

- The Scale of Games is Increasing Too
- Managing Lots of Shaders and Models Can Be a Chore
- FXComposer uses .NET assemblies so that .NET can control FXComposer to build scenes, export images, assign shaders, export data, etc, rapidly
- Use C# or Visual Basic

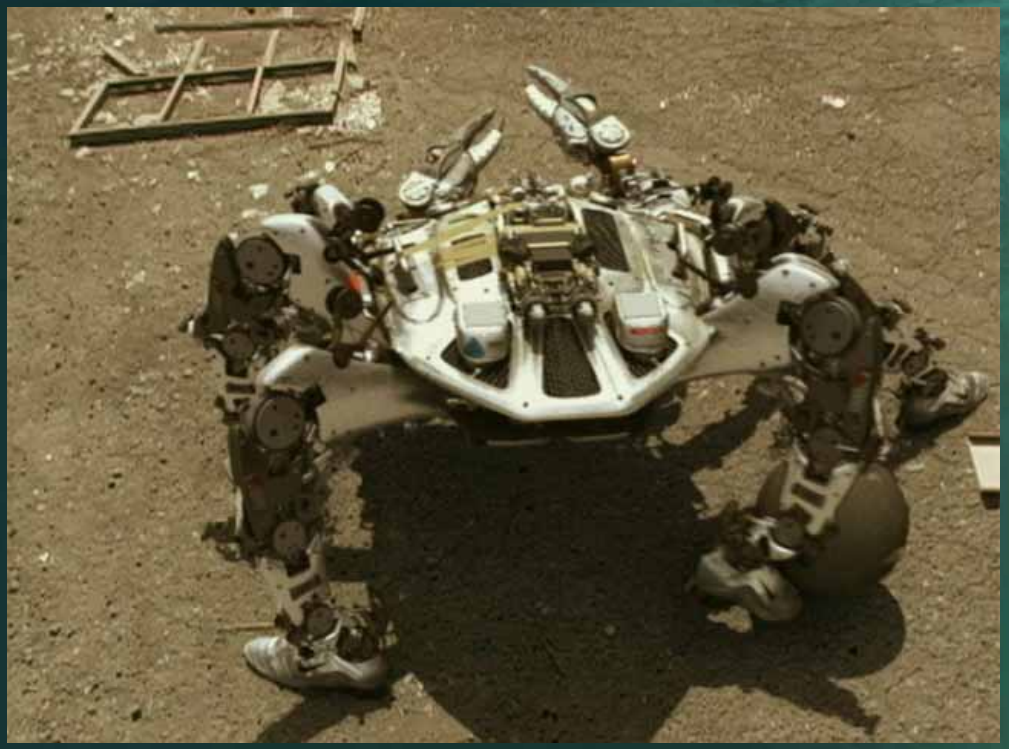
Long Schedules

- Movies have lots of money and time, so they have the potential to develop cool technologies
- BUT: Those technologies need to be locked down early enough so that shots on the last day of production look like they belong with shots from the first day of production
- This sometimes limits innovation
- Fastest turnaround of innovation: *TV Commercials*

Nike.Com campaign, Weiden + Kennedy, Dir Neill Blomkamp http://www.theembassyvfx.com/

NikeLab.COM

- Produced by The Embassy Visual Effects
- In Four Weeks!
- Using Lightwave, Shake, and NVIDIA Quadro GPUs



Nike campaign, Weiden + Kennedy, Dir Neill Blomkamp
http://www.theembassyvfx.com/

Shadows

- Shadows are often more important than illumination
- Once shadows appear, it’s hard to go back!

1998

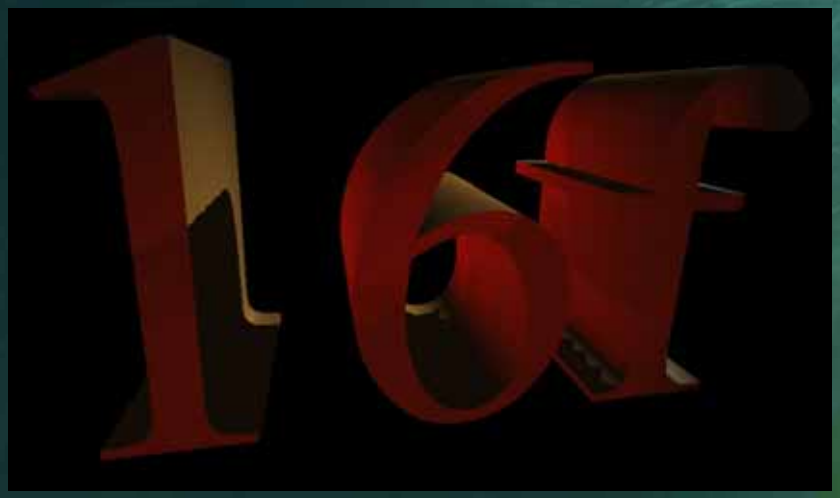
2004

Art Lesson: Shadows

The Art Lesson

- Simple shadows: Stencil Volumes or Render to Texture
- Where's the light?
- Sharing Lights
- "Advanced Shading" talk will address some of these issues too

Fancy Shadows -- Translucence



- Shadow Z values can be used behind objects, too
- “Advanced Shading” will speak in depth about this technique (and more)

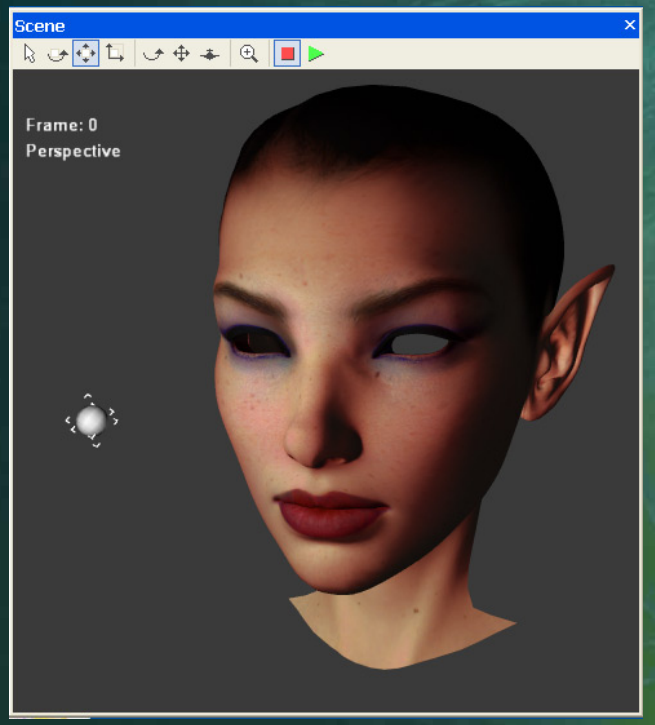
DXSAS – Scriptable FX/HLSL

- DXSAS = “DirectX Standard Annotations and Semantics” & is a Microsoft Standard * part of XNA
- Includes a “Script” semantic for each pass and technique
- Scripts define Render Targets, can loop, and can call on each other
- HLSL “Virtual Machine” (VM) does numerics like matrix math

The Art Lesson

Skin and Shading

- Diffuse Subsurface Scattering on the Cheap:
  - By remapping “(N·L)” in our diffuse-shading calculations to “((N·L)+w)/(1+w)” we can “wrap” light around the contours of an object
  - (Don’t worry about the math details – an example awaits!)
  - Since this is all in the diffuse lighting, it’s sometimes okay to do the job in the vertex shader



Skin and Direct Reflectance

- The younger you are, the less dead skin
- Live skin cells reflect like little cat's eye reflectors
- Therefore, a flat skin tone = youthful appearance
- Oren-Nayar Shading (expensive) and "grisaille" shading (cheap!)
- Combining ideas

One Modern Variation

Traditional Grisaille Relief

Lighting

- Shade what's lit – not what's not lit
- Use PS_3_0 early outs
  - Bonus: Using "if" can also benefit batch sizes
  - Write one shader, compile for ps_3 or ps_2
- For defered shading, only shade lit pixels
- "Gloominance" is perfectly safe in all cases for floating-point pixels

Spotlight

Smart Light Placement

- Magy Seif El-Nasr's "ELE": The Expressive Lighting Engine
- http://ist.psu.edu/SeifElNasr/
- Uses robotics load-balancing equations to maximize visibility and "mood" for a limited set of lights

Mirage, El-Nasr et al, CIRA

Reflections

- Can replace all specular in some circumstances
- Can use VM to generate CUBE maps
- Can have finite radius (see talk later)
- Can have distance with quadratic falloff (see talk later)

Environment-mapped background, reflected card-shaped light source, 16-bit blending with overbright bloom

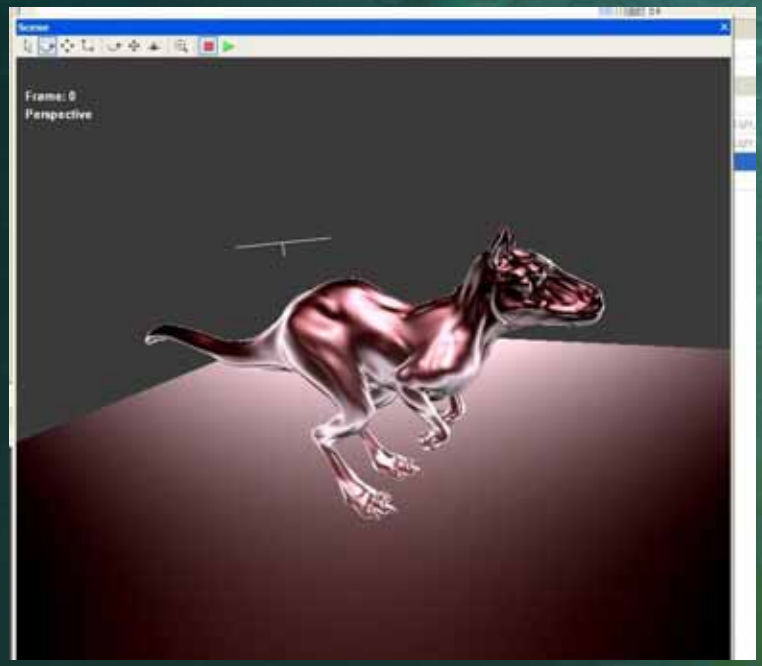
New Territory: Camera Effects

- "Accumulation Buffer" technique gives us:
  - Motion Blur
  - Depth of Field
  - Soft Shadows
  - More…
- No special shading requirements, but shaders must go *fast*

Motion blur

Making the Most of the Direct X VM

- Texture Generation
- “Texture Shaders” on the CPU can generate images, or create textures containing predictable functions
- Matrix Manipulation using HLSL intrinsics make complex shadowing a functional reality



Dinosaur with Physically-based car paint BRDF

Compositing & 2D Effects

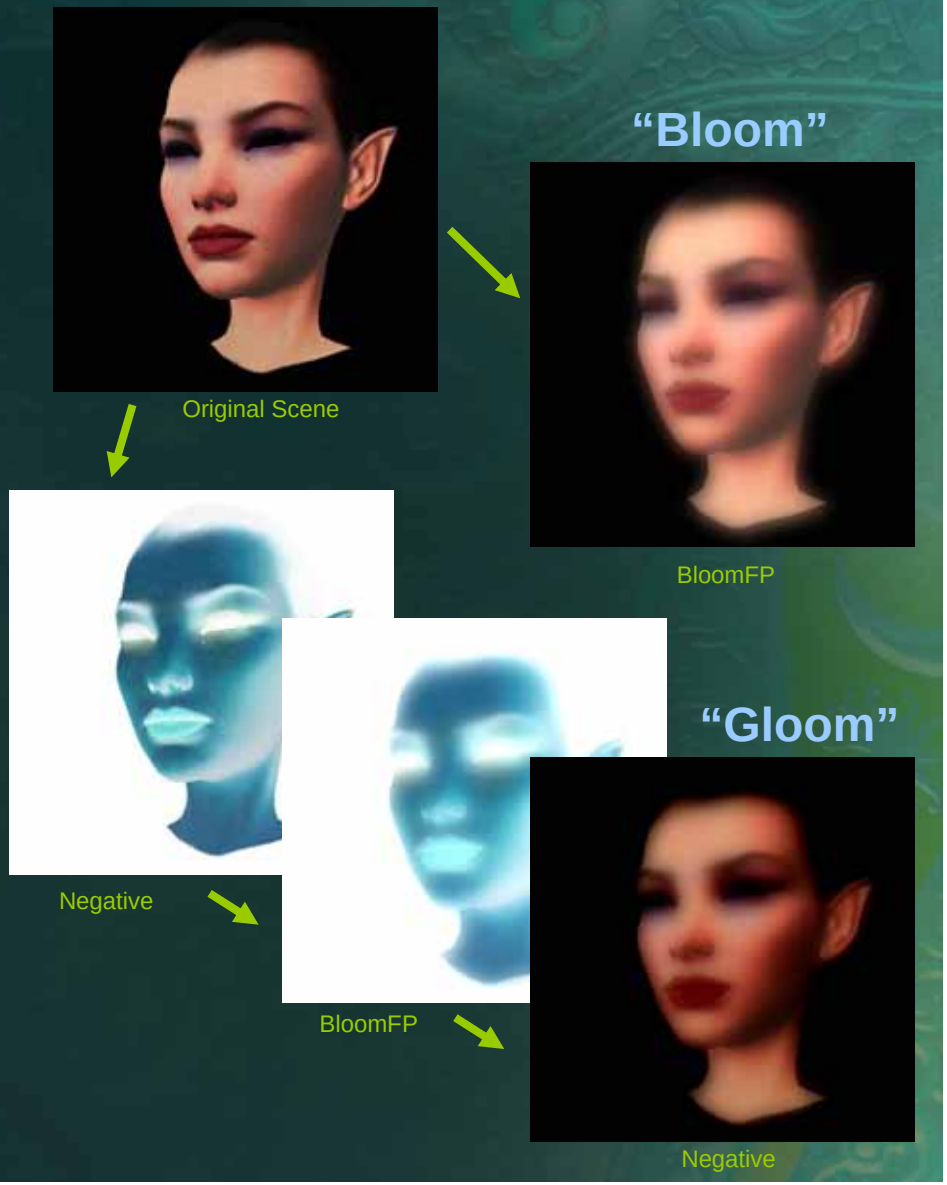
- FP buffers make things more powerful than ever
- Lots of fun…
- Color controls
- Final “sweetening”
- Blend modes
- Mix 2D/3D sprites
- Floating-point pixels

Halftoning Patterns

Image Trails

Post-Process: Bloom & Gloom

- Glow and “over-bright” bloom give a great illusion of complexity and scale
  - A bit like “echo” in the recording studio, it’s hard to resist using it!
- We can stack image effects in FX Composer for new, more-complex effects



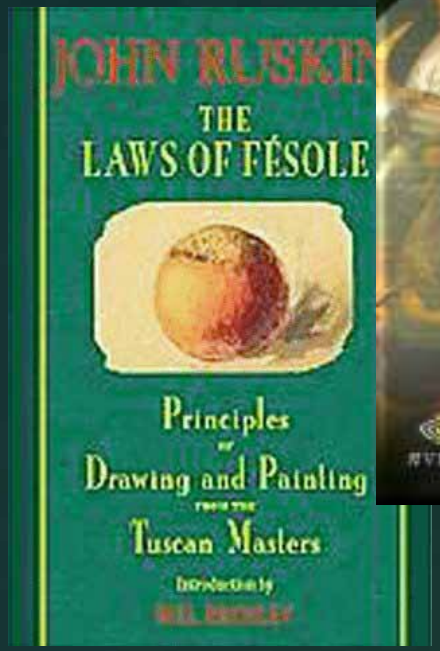
That’s a Wrap!

The End

- Games now have the capacity to match film shading, in character if not pixel-to-pixel
  - Get used to *lots* of shaders
  - Get tools that let you play
  - http://www.fxcomposer.com/
  - Play with shaders, try everything, keep a “sketchbook” of useful ideas

Some Recommended Books

- Randima Fernando: *GPU Gems*
- John Alton: *Painting with Light*
- Jon Ruskin: *The Laws of Fésole, Priciples of Drawing and Painting from the Tuscan Masters*



http://developer.nvidia.com/object/GPU_Gems_home.html

More On These Topics

- http://developer.nvidia.com/
- http://www.fxcomposer.com/
- http://developer.nvidia.com/object/sdk_effects.html
- kbjorke@nvidia.com