
Game Developers Conference®
February 28 - March 4, 2011
Moscone Center, San Francisco
www.GDConf.com
GDC
UBM

Using Simulation Data In Real time With Video Textures

Bryan Moss
Senior Technical Artist
THQ Digital Studios Phoenix

Introduction

- Background in games being a Technical Artist
  - Prefer Graphics than Tools
  - Currently embedded with the graphics team
- THQ Digital Studios Phoenix
  - Small studio
  - Need to innovate quickly
  - Still need to maintain extremely high standards

Video Simulation Textures

- What is it? How did the idea come about?

- Siggraph 2009
  - The Light Kit: HDRI-Based Area Light System for “ The Curious Case of Benjamin Button”

Video Simulation Textures

- What problem was I looking to solve?
- Where would this be used?
- What ELSE can be achieved with this?
  - Didn’t want to create another closed system
  - Need to be flexible and able to be used elsewhere

Previous Solution

- Old cloth system was outdated and visually lacking
  - Used a system of “jiggle bones”
  - Not intuitive
  - Impossible to make the Art Director’s vision a reality
  - Fairly expensive
  - 11 bones * 12 riders = OUCH!!
  - Idea for video rendering seemed perfect
- Jiggle bones do have their uses

Comparison

Reflex (2009)

Alive (2011)

Planning

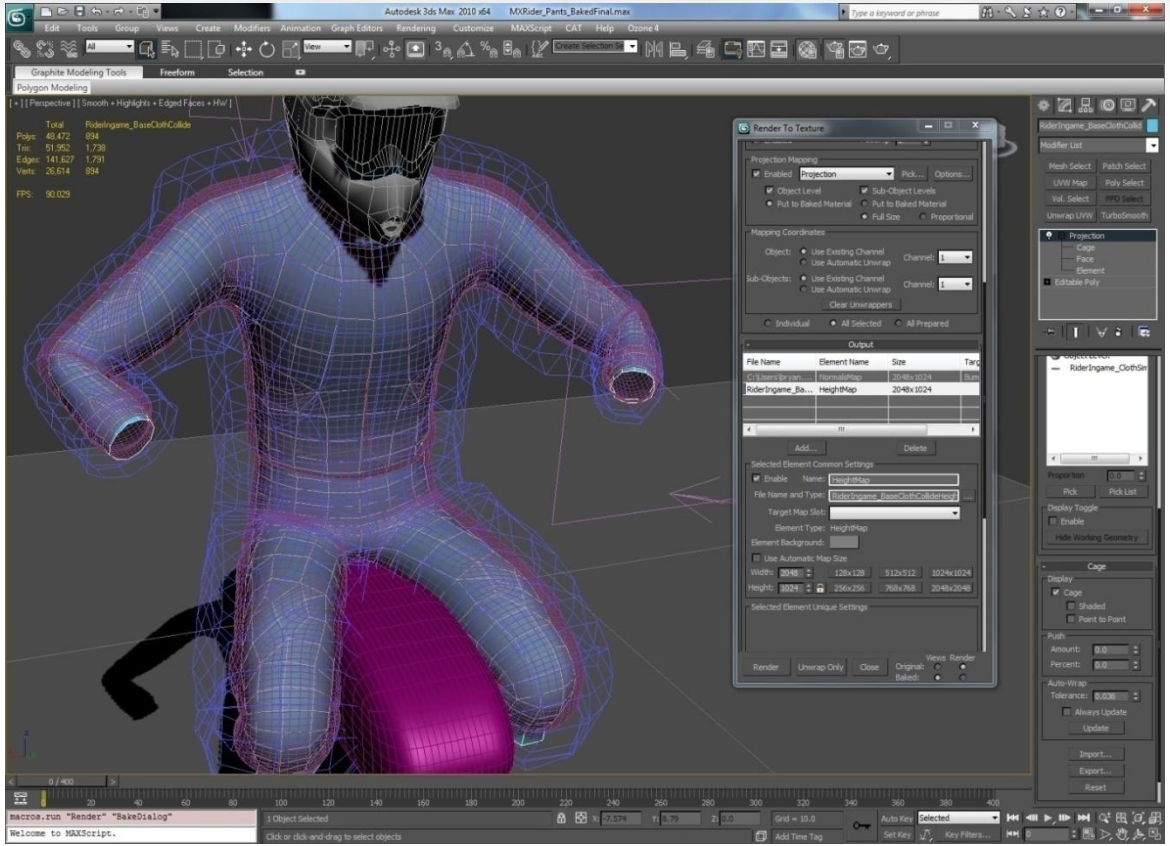
- What kind of new pipelines will we need?
- Cloth will be simulated using 3DStudio Max
- Use existing Video Replace entity node to embed video into shader
  - Entity defines any texture in a scene and replaces it with a defined video
  - The texture that is replaced is the one that the shader performs on all of the operations
- After initial R&D, discovered almost no new pipelines would be needed. GREAT!

Start In 3DStudio Max

- Use in game mesh as collision
- Duplicate mesh and re-topologize
- Max Cloth plug-in

Render to Texture

- One pass for Normal Maps
- Another for Height Maps



Limitless possibilities?

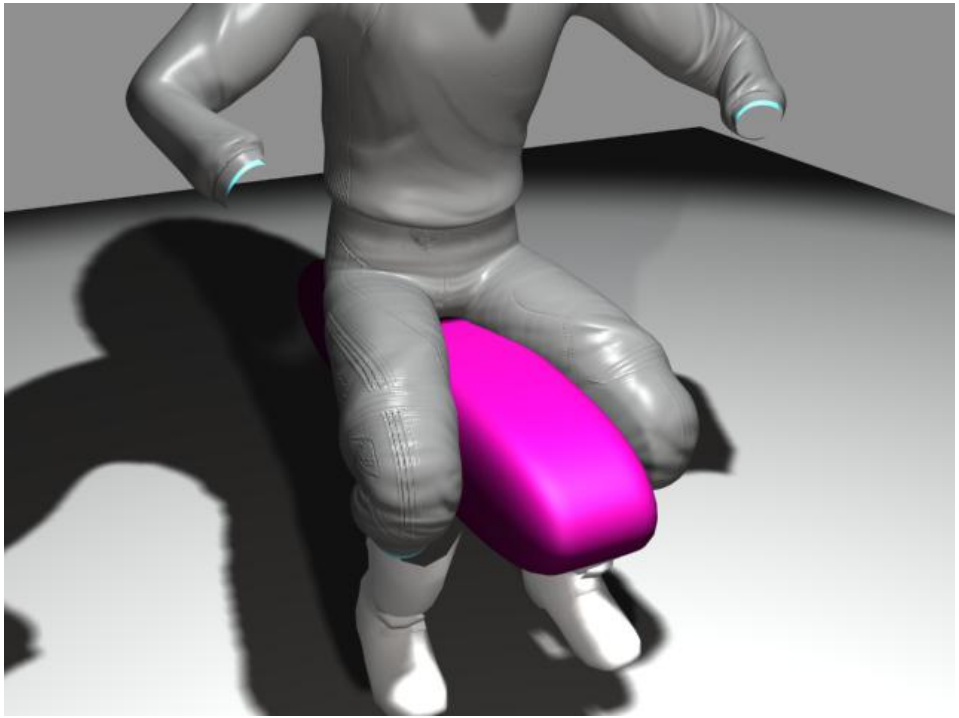
- Cloth sim was 400k triangles
- Collision pieces are added to simulate the vehicle blocking the wind
- 3 different wind sources

Super realistic cloth was not ideal

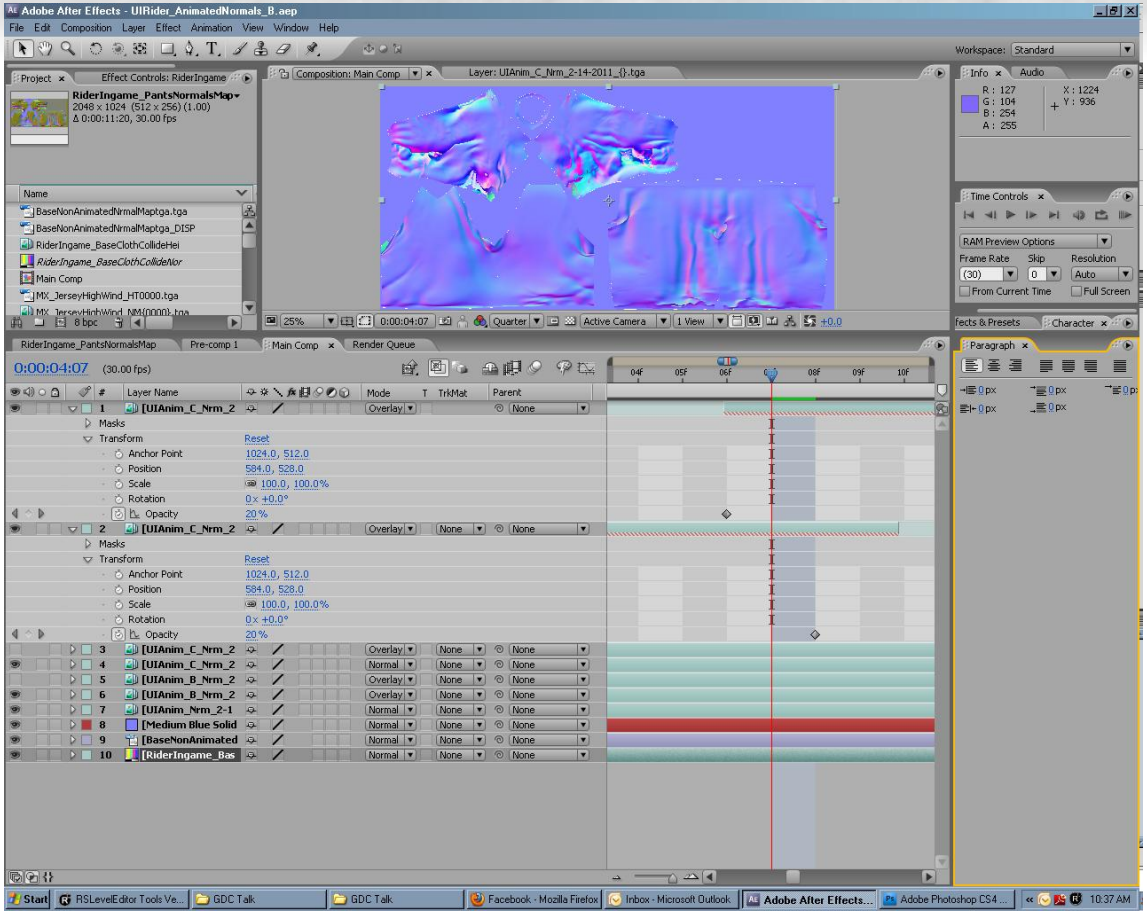
DON’T BE SUBTLE!

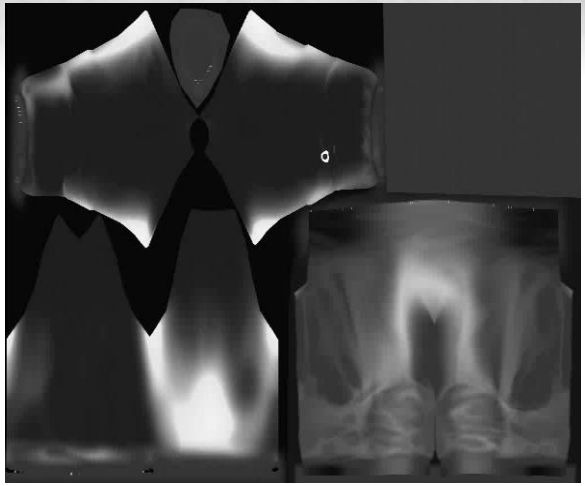
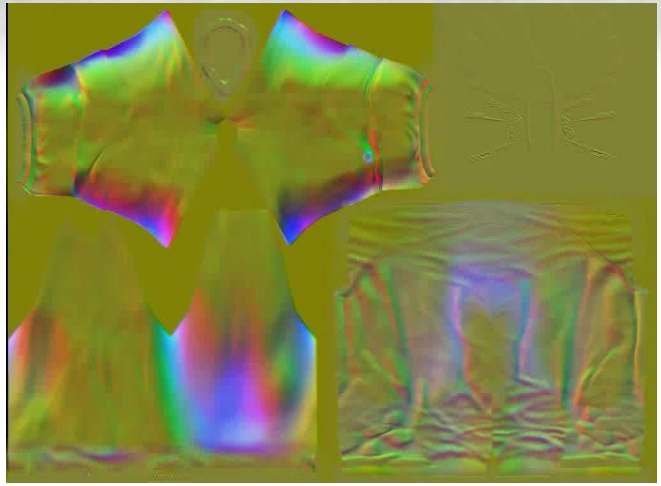
Important!

Use whatever DCC apps that you want



- Videos are like “Combo maps”
- Composite together
- “Fix it in Post” mentality
  - Extremely flexible part of process
  - Don’t like something? Comp it out!



Normal Map (R,G) + Height Map (B) = Final Composite (rgb)

Packing multiple videos together

- Blue channel in a normal map is a throw away
- Height map is already grey scale
- Pack the Height map into blue channel of normal map

Video Compression

- Bink for Video compression
- good documentation for their tools

- No new middleware to purchase
- this is great for small teams

- Find a compression solution
  to suit your needs

Cloth Shader

- Technical Artists are invaluable on this step.
  - Art Knowledge plus technical know how
  - Very little programmer oversight
  - Always run the numbers! (PIX, GPAD, etc)
- Can’t be too expensive on the GPU
  - Needs to run on 12 riders simultaneously

Features

- Extremely dynamic shader
  - 5 dynamic systems including cloth
  - Multiple specular types
  - Two sets of UV's
  - 5 textures plus one Video Texture
- Bulk of cloth is in the Normal and Parallax
  - Swizzle out Blue channel for parallax
  - Broken Parallax on UV 2 required programmer help
- Very efficient established shader pipeline tools helped tremendously

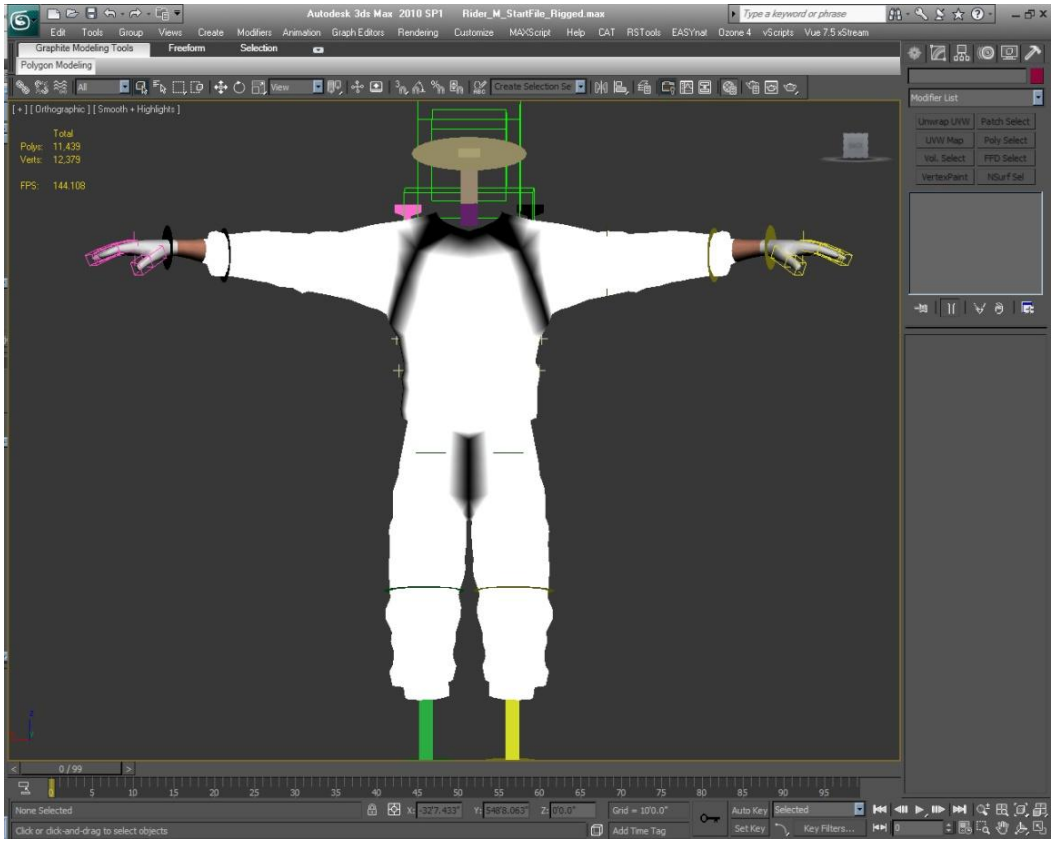
First Pass

It Works!!

This was the prototype used to get approval for use in-game.

Iterate and Polish

- Lots of shader bugs
  - Still too expensive
  - Shader is very complex
  - Stack based interface can be confusing
- Wasn't happy visually
  - Blending Static and Animated Maps together
  - Animated map now drives the static Normal Map
  - Parallax "smears" across uv boundaries
  - Lack of silhouette movement irritating
- Was not dynamic
- Video texture played when the rider was stopped
  - Treat the video texture as a dynamic entity



Breakthroughs

- Float switches and Controllable parameters
  - These are key for dynamic control
  - Created a float “switch” to blend between states
  - Switch is tied to the velocity of the vehicle
    - 0 = static map
    - 1 = full animated video + static map
- Total cost on GPU
  - .36ms, +100 instructions

Optimization

- GPU
  - Use PIX and GPAD
  - Adding parallax is biggest expense to the shader
  - Video size is fixed (256x256)
    - Video didn't make a difference, every frame is constant cost
    - Packing video was a huge win
  - Final cost .24ms, 96 instructions
- Uh oh……. Forgot about the CPU!
  - It costs HOW MUCH?
  - Video decompression is expensive!!

- 4ms per frame initial CPU cost
  - Not exactly a true cost
- Video decompression is not a constant expense
  - Each frame varied as much as .25 ms
- Solution
  - Negotiate a fair cost
  - Bink Tools are robust and fairly well documented
  - Take a 5 frame average cost and use that as a constant benchmark
  - Smaller size and better compression
  - Final average is .831 ms on CPU
  - Final video texture is 62 frames at 256x256
  - 150k

Downsides

- Process can be cumbersome
  - Art setup is high
  - Artists need to know multiple packages
  - Can be tech heavy when designing , coding, and maintaining shaders
  - Technical artists are a must

- CPU cost is still a big factor

Conclusion

- Huge win artistically
- Art Director is able to fully call the shots
- Able to visualize complex effects without massive simulation systems
- Extremely flexible
  - Video texture is able to be handled in a shader
  - DCC app agnostic
  - What else can we do?
    - Water Flow simulations
    - Fracture
    - Morphing
    - Very untapped system

Final Look