The Ultimate Developers Toolkit

Jonathan Zarge
Dan Ginsburg

February 20, 2008
Agenda

GPU PerfStudio
GPU ShaderAnalyzer
RenderMonkey
Additional Tools
  • Tootle
  • GPU MeshMapper
  • CubeMapGen
  • The Compressonator
  • OpenGL ES 2.0 Emulator

All tools available today at: http://ati.amd.com/developer/
GPU PerfStudio

http://ati.amd.com/developer/tools.html
GPU PerfStudio – Why use it?

• Is the graphics performance of your game not meeting expectations?
• Are you wondering what the heck the graphics hardware is doing?

The solution is GPU PerfStudio:
• Interactive tool for enabling graphics performance optimization
• View into API, driver and hardware
• Current being used by:
  – Leading game developers
  – AMD graphics ISV Engineering team
  – AMD demo and driver teams
GPU PerfStudio

- Performance analysis and optimization tool
- Real-time visualization:
  - API statistics
  - Hardware/driver data
- Override rendering states
- Launch remote applications
- Flexible data visualization
- Bottleneck analysis
GPU PerfStudio – New Features

- D3D10 and OpenGL support
- ATI Radeon HD2000 and HD3000 support
- Automated bottleneck analysis:
GPU ShaderAnalyzer

http://ati.amd.com/developer/tools.html
GPU ShaderAnalyzer

Shader performance analysis tool:
• View hardware disassembly
• Instant feedback as you tune
• Estimated cycle counts for all AMD GPUs

Support for all shading languages:
• HLSL, DX assembly
• DX9, DX10, DX10.1
• GLSL, ARB assembly
• Brook+, IL
• Vertex, fragment, and geometry shaders

http://ati.amd.com/developer/tools.html
GPU Shader Compilation

HLSL DX9 → DX9 ASM
HLSL DX10 → DX10 Blob
GLSL → IL
GL Assembly → IL
Brook+ → IL

IL → Shader Compiler

Shader Compiler → HW Instructions
RenderMonkey

Shader Development Environment
- Rapid Prototyping of Shader Effects

Multiple Shading Languages
- DirectX HLSL
- DirectX Assembler
- OpenGL Shading Language
- OpenGL ES Shading Language
RenderMonkey – Why use it?

Full IDE for shader effect development
  – Programmer and artist view for rapid iteration

Easy integration into game pipeline
  – Plug-in SDK for custom import/export
    • Effects, models, textures, variables, etc.
  – Support for many standard formats
    • DDS, BMP, TGA, X, OBJ, 3DS, DAE, FX, COLLADA FX

Encompasses all effect resources
  • Render state, texture state, variables, render targets, textures, models, etc...
RenderMonkey – What’s new?

V1.7:
• Support for OpenGL ES 2.0
• ES Shading Language v1.00
• ES syntax highlighting
• ES render/sampler states
• Large suite of ES examples

V1.8:
• OpenGL COLLADA Effects Exporter
• Import COLLADA Geometry
RenderMonkey - Demo
Additional Tools

http://ati.amd.com/developer/tools.html
Tootle

• A Triangle Order Optimization Tool

• Provided as a library for integration into your tool-chain

• Improves vertex cache hit rate
  – Shade fewer vertices

• Reduces overdraw
  – Shade fewer pixels
  – View independent
Tootle: Overdraw Reduction

• Example Scene
  – 70k polygons
  – 10 materials

• Reduced overdraw by factor of two

• 3-7% performance increase compared to D3DXOptimizeMesh.
GPU MeshMapper

Use low and high detail models to generate normal maps, displacement maps, and ambient occlusion maps

http://ati.amd.com/developer/tools.html
CubeMapGen

- Standard mip-maps
  - Filters individual faces
  - Results in artifacts

- CubMapGen mip-maps
  - Pre-filters across faces
  - Removes visible edge

- Smaller, better cube maps at no rendering cost!

http://ati.amd.com/developer/tools.html
The Compressonator

Compresses textures to all AMD-supported formats

Visual diff to assess compression quality

http://ati.amd.com/developer/tools.html
OpenGL ES 2.0 Emulator

- Provides an OpenGL ES 2.0 development environment on the PC
  - Develop your OpenGL ES 2.0 applications today
  - Minimizes porting effort once hardware is available

- Full implementation of OpenGL ES 2.0 and EGL 1.3
Summary

GPU PerfStudio

GPU ShaderAnalyzer

RenderMonkey

Additional Tools
  • Tootle
  • GPU MeshMapper
  • CubeMapGen
  • The Compressorator
  • OpenGL ES 2.0 Emulator

http://ati.amd.com/developer/tools.html
<table>
<thead>
<tr>
<th>Time</th>
<th>Session</th>
<th>Speaker(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>9:00 – 10:00 AM</td>
<td><strong>The Ultimate Developers Toolkit</strong>: Jonathan Zarge, Dan Ginsburg</td>
<td></td>
</tr>
<tr>
<td>10:30 – 11:30 AM</td>
<td><strong>Harnessing the Power of Multiple GPUs</strong>: Jon Story and Holger Gruen</td>
<td></td>
</tr>
<tr>
<td>12:00 – 1:00 PM</td>
<td><strong>Ultimate Graphics Performance for DirectX10 Hardware</strong>: Nicolas Thibieroz</td>
<td></td>
</tr>
<tr>
<td>2:30 – 3:30 PM</td>
<td><strong>Optimization Techniques for Attacking CPU Data Bottlenecks in PC Games</strong>: Michael Wall</td>
<td></td>
</tr>
<tr>
<td>4:00 – 5:00 PM</td>
<td><strong>Tessellation in a Low Poly World</strong>: Bill Bilodeau and Peter Lohrmann</td>
<td></td>
</tr>
</tbody>
</table>
Questions?

Jonathan.Zarge@amd.com
Dan.Ginsburg@amd.com
GPUTools.Support@amd.com
http://ati.amd.com/developer/

http://ati.amd.com/developer/tools.html