NVIDIA SLI and stutter avoidance: a recipe for smooth gaming and perfect scaling with multiple GPUs

Lars Nordskog and Iain Cantlay
Stutter - A Vast Topic

- Two topics today only:
  - One measurement - percentiles
  - One cause - memory over-commitment
- See Cem Cebenoyan’s broader, extensive treatment from China GDC 2012
Stutter Matters

- Critical to game-play experience
- As important as average FPS?
- More important?
- Ultra-low tolerance in VR
“This is a confirmed fix”

“I was able to play smoothly without any problems”

“we see heavy perf drop/severe stutter”

“[the fix] didn't do anything for me”
No Supporting Data
Why Measure Stutter?

- Prove your fixes
- Regression test
- Quantify your claims
- Measure before you can fix
- Trust numbers (with time)
How to Grok Percentiles

99.87% of frames < 53ms
0.13% of frames > 53ms

99% of frames < 29ms
1% of frames > 29ms
How to Compute Percentiles

- Common in statistics
- Excel
- Fraps2percentile tool - source published: https://developer.nvidia.com/content/analysing-stutter-%E2%80%93-mining-more-percentiles-0
Example: Battlefield 4

- Fixed a stutter bug in game (not driver)
- Before/after clear picture (for >99.5%)
- Measured!
Example: GeForce Driver

- GeForce driver opts for shader compilation
- End-user improvement with no changes to game
- Measured!
- Stdev
- Count > threshold
- Count > median over window
- Fourier analysis
- Any spike, even one, == a Bad Thing
Limitations of Percentiles

- Definitely spikey
- Percentiles agree
Limitations of Percentiles

- Average frame time jumps up
- Masks the earlier spikes
- Analyse a small window instead?
Stutter is Complex and Varied

- Percentile graphs capture variety
Limitations of Percentiles

- Cannot graph many results
- Too complex for a single scalar representation

<table>
<thead>
<tr>
<th>99%-ile frame time analysis (ms)</th>
<th>Win 8.1</th>
<th>Win 7 SP1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1920x1200</td>
<td>2560x1600</td>
</tr>
<tr>
<td>Maximum Quality</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Extreme Quality</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maximum Quality</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Extreme Quality</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Delta (99%-ile frame time) – (1000/Avg FPS)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GTX 750</td>
<td>12.3</td>
<td>23.6</td>
</tr>
<tr>
<td>GTX 750 Ti</td>
<td>19.6</td>
<td>68.6</td>
</tr>
<tr>
<td>GTX 660</td>
<td>31.9</td>
<td>44.6</td>
</tr>
<tr>
<td>GTX 960</td>
<td>26.1</td>
<td>23.1</td>
</tr>
<tr>
<td>GTX 970</td>
<td>10.1</td>
<td>21.2</td>
</tr>
<tr>
<td>GTX 780 Ti</td>
<td>8.9</td>
<td>15.6</td>
</tr>
<tr>
<td>GTX 980</td>
<td>9.1</td>
<td>13.8</td>
</tr>
</tbody>
</table>
Stutter is Complex and Varied
Top Five Causes of Stutter

- See Cem’s China GDC 2012 presentation
  1. Shader compilation
  2. Video memory oversubscription
  3. Resource mis-management (mapping)
  4. Queued frames
  5. Improper queries
Top Five Causes of Stutter

- See Cem’s China GDC 2012 presentation
1. Shader compilation
2. Video memory oversubscription
3. Resource mis-management (mapping)
4. Queued frames
5. Improper queries
Fixing - A Simple Tip

- **GPUView**
  - (get Microsoft’s Windows Performance Toolkit)
- **Memory over-commitment**
- **P2 references > 0% correlate with stutter**
- **Plenty more fix tips in Cem’s presentation (see references)**
Fixing Stutter - GPUView
Fixing Stutter - GPUView
Fixing Stutter - Memory

Paging Packets (red)

64ms

P2 references are high
Fixing Stutter - Memory
Measure, measure, measure
Add measures to your game
Then fix
Questions / Further Reading


https://developer.nvidia.com/content/analysing-stutter-%E2%80%93-mining-more-percentiles-0

http://graphics.stanford.edu/~mdfisher/GPUView.html

http://developer.amd.com/gpu_assets/Using%20GPUView%20to%20Understand%20your%20DirectX%20Game.pps
GameWorks

- Get the latest information for developers from NVIDIA and continue the discussion
- gameworks.nvidia.com