From Terrain To Godrays: Better Use of DX11

Iain Cantlay
Andrei Tatarinov
Developer Technology Group, NVIDIA
Better Use of Tessellation

- One of the most recent additions to DirectX
- And one of the least explored too
Getting More Adoption

- Originally tessellation was available only on DX11-capable PCs
Getting More Adoption

- Originally tessellation was available only on DX11-capable PCs
- Wasn’t available on consoles
- Tessellation brings special requirements to the content
Getting More Adoption

- Next-generation consoles support it too!
Getting More Adoption

- Some AAA-titles already use it

Screenshots were made by Andrew Iain Burnes and published at GeForce.com
Outline

● Common use cases
  ● Terrain
  ● Super-static objects

● Novel approaches
  ● Tessellated particles
  ● Godrays

● Tessellation Tips and Tricks
Terrain
A classic task for tessellation

H.A.W.X. 2 courtesy of Ubisoft
Terrain

- Requires
  - Detail at wide range of scales
  - Highly mobile view, e.g., flying
  - Frequent, seamless LOD changes
- H.A.W.X 2
- Frostbite 2: BF3, NFS The Run
- Frostbite 3: BF4, NFS Rivals
Terrain

- Requires
  - Detail at wide range of scales
  - Highly mobile view, e.g., flying
  - Frequent, seamless LOD changes

- **H.A.W.X 2**
- Frostbite 2: **BF3**, NFS The Run
- Frostbite 3: BF4, **NFS Rivals**
Tessellation Patches

DX9

HAWX2

Replace

BF3

x0.5

DX11 Patches

Tessellate – HS

Details not accurate!!
**Extra Detail**

HAWX2

- DX9 offline tessellation
- DX9 tessellation on CPU

BF3

- Add fBm detail noise in DS

DX11 tessellation Sample height map in DS
DEMO
Battlefield 3

Battlefield 3 courtesy of EA DICE
Adaptive Tessellation

BF3

NFS Rivals + “Density” Map
Need For Speed Rivals courtesy of Ghost Games and EA
Need For Speed Rivals courtesy of Ghost Games and EA
<table>
<thead>
<tr>
<th>Scene</th>
<th>Brute force</th>
<th>Density map</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scene 1</td>
<td>667060</td>
<td>249222</td>
</tr>
<tr>
<td>Scene 2</td>
<td>511723</td>
<td>209485</td>
</tr>
</tbody>
</table>
Performance and Conclusions

- Add detail easily !/$
- Natural fit to terrain
- High perf on many platforms

<table>
<thead>
<tr>
<th>Battlefield 3 DX11</th>
<th>1920x1200</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>terrain</td>
<td>ultra</td>
</tr>
<tr>
<td>GTX 750 Ti (2GB)</td>
<td>46.1</td>
<td>43.7</td>
</tr>
<tr>
<td>GTX 760 (2GB)</td>
<td>78.1</td>
<td>73</td>
</tr>
<tr>
<td>GTX 770 (2GB)</td>
<td>91.8</td>
<td>86.7</td>
</tr>
<tr>
<td>R7 260X (2GB)</td>
<td>43.9</td>
<td>40.5</td>
</tr>
<tr>
<td>R9 270X (2GB)</td>
<td>65.4</td>
<td>60.3</td>
</tr>
<tr>
<td>R9 280X (3GB)</td>
<td>92.5</td>
<td>83.2</td>
</tr>
</tbody>
</table>
Need For Speed Rivals courtesy of Ghost Games and EA
Tessellating super-static objects

- Super-static objects are good candidates for tessellation.
We did this in Metro: Last Light

- A joint project of 4A Games and NVIDIA
  - Use super-static geometry that has bump maps
  - Implement hull and domain shaders
  - Add displacement to the geometry
Metro: Last Light

Tessellation OFF
Metro: Last Light

Tessellation ON
Metro: Last Light

Tessellation OFF
Metro: Last Light

Tessellation ON
Under-tessellation is bad

- Super-static objects are often modelled with large triangles
- Level of detail required to represent displacement can exceed DirectX tessellation factor limit
Under-tessellation is bad
Under-tessellation is bad
Under-tessellation is bad
Virtual dicing

- Virtual dicing subdivides big triangles into smaller ones on-the-fly

- This can also be done offline
Virtual dicing in Metro: Last Light
Virtual dicing in Metro: Last Light

Original mesh
Virtual dicing in Metro: Last Light
Over-tessellating is wasteful

- Some areas on displacement maps don’t require high tessellation factors
Over-tessellating is wasteful

- Some areas on displacement maps don’t require high tessellation factors
Over-tessellating is wasteful

- Some areas on displacement maps don’t require high tessellation factors
Over-tessellating is wasteful

- Some areas on displacement maps don’t require high tessellation factors
Use adaptive tessellation

- Analyze the displacement map covered by the triangle
- Adjust the inside tessellation factor accordingly
Adaptive tessellation in detail

- Use a simple quad mesh as an example
Adaptive tessellation in detail

• Use a simple quad mesh as an example
Adaptive tessellation in detail

- Add displacement map
Adaptive tessellation in detail

- Smooth shapes require high expansion
Adaptive tessellation in detail

- Flat areas can use lower expansion
Adaptive tessellation in detail

- Take the average from coarse mip-level
Adaptive tessellation in detail

- Use finer mip-level to calculate variance
Adaptive tessellation in detail

- Calculate the metric based on variance
Adaptive tessellation in detail

- Use threshold to control tessellation factors
DEMO

- Metro: Last Light
Performance and conclusions

- Metro: Last Light, Undercity level, 1920x1200, Very High, SSAA OFF

<table>
<thead>
<tr>
<th></th>
<th>FPS</th>
<th>Adaptive OFF</th>
<th>Adaptive ON</th>
<th>Gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>GTX 750Ti</td>
<td>17.2</td>
<td>17.2</td>
<td>31.1</td>
<td>2X</td>
</tr>
<tr>
<td>GTX 760</td>
<td>27.8</td>
<td>27.8</td>
<td>47.2</td>
<td>2X</td>
</tr>
<tr>
<td>GTX 770</td>
<td>35.1</td>
<td>35.1</td>
<td>60.7</td>
<td>2X</td>
</tr>
<tr>
<td>R7 260X</td>
<td>16.2</td>
<td>16.2</td>
<td>29.4</td>
<td>2X</td>
</tr>
<tr>
<td>R9 270X</td>
<td>13.9</td>
<td>13.9</td>
<td>41.6</td>
<td>3X</td>
</tr>
<tr>
<td>R9 280X</td>
<td>14.9</td>
<td>14.9</td>
<td>57.7</td>
<td>4X</td>
</tr>
</tbody>
</table>
Performance and conclusions

- Adaptive tessellation doubles performance on tessellation-heavy levels in Metro: Last Light
- Don’t be afraid to tessellate densely where needed
- But use your triangles efficiently!
Another meaning of tessellation

- Before DX11, we could only perform calculations at vertex, geometry or pixel rates.
Another meaning of tessellation

Same Blinn-Phong shading done at different rates:

- Gouraud
- Flat
- Phong
Another meaning of tessellation

Same Blinn-Phong shading done at different rates:

- Gouraud
- Flat
- Phong

Only Phong looks nice
Another meaning of tessellation

- Before DX11, we had to choose between three “fixed” rates
- Vertex or geometry rate is too low
- For some effects, pixel rate is too high
Another meaning of tessellation

- Before DX11, we had to choose between three “fixed” rates
- Vertex or geometry rate is too low
- For some effects, pixel rate is too high
Another meaning of tessellation

- Tessellation is a stage with adjustable shading rate
Adjustable shading rate

- Hull shader is a “slider” that allows to adjust shading rate
- Domain shader does the actual shading
Adjustable shading rate

● Similar to Reyes pipeline
  ● Shading is done in object space
  ● Sampling (rasterization) is only used to interpolate results
Which effects can benefit from it?

- Computation-heavy effects with low frequency
  - Particle shadows
  - Volumetric effects
  - Global illumination
  - ...

Which effects can benefit from it?

- Computation-heavy effects with low frequency
  - Particle shadows
  - Volumetric effects
  - Global illumination
  - ...

AC4BF screenshot was made by Andrew Iain Burnes and published at GeForce.com
Particle Shadow Mapping

- Calculate shadow from a particle system
- Calculate particle system self-shadowing
Particle Shadow Mapping

- Particle systems contain thousands of particles
- Shadowing has to be calculated for every pixel of every particle
- Or maybe not?
Particle Shadow Mapping

- Let’s calculate it in DS!
- Tessellate the particle sprites
- Use HS to determine shading rate
Fourier Opacity Mapping

● A sample by Jon Jansen and Louis Bavoil

Fourier Opacity Mapping

- A sample by Jon Jansen and Louis Bavoil

DEMO

- Fourier Opacity Mapping sample
## Performance

- Fourier Opacity Mapping sample, 1920x1200

<table>
<thead>
<tr>
<th></th>
<th>FPS</th>
<th>Per-Pixel</th>
<th>Tessellated</th>
<th>Gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>GTX 750Ti</td>
<td>17.2</td>
<td></td>
<td>68.7</td>
<td>4X</td>
</tr>
<tr>
<td>GTX 760</td>
<td>34.2</td>
<td></td>
<td>118.7</td>
<td>3.5X</td>
</tr>
<tr>
<td>GTX 770</td>
<td>48.2</td>
<td></td>
<td>155.2</td>
<td>2X</td>
</tr>
<tr>
<td>R7 260X</td>
<td>15.1</td>
<td></td>
<td>65.3</td>
<td>4X</td>
</tr>
<tr>
<td>R9 270X</td>
<td>21.9</td>
<td></td>
<td>85.3</td>
<td>4X</td>
</tr>
<tr>
<td>R9 280X</td>
<td>32.8</td>
<td></td>
<td>100.5</td>
<td>3X</td>
</tr>
</tbody>
</table>
Particle Irradiance in 3DMark

- This approach was successfully used in 3DMark for Windows 8
Which effects can benefit from it?

- Computation-heavy effects with low frequency
  - Particle shadows
  - Volumetric effects
  - Global illumination
  - ...

GAME DEVELOPERS CONFERENCE® 2014

MARCH 17-21, 2014

GDCONF.COM
Which effects can benefit from it?

- Computation-heavy effects with low frequency
  - Particle shadows
  - **Volumetric effects**
  - Global illumination
  - ...

Godrays screenshot was taken from a sample by Alexey Panteleev
Volumetric effects

- Typically use ray-marching to integrate over the medium inside the volume
Godrays

- We have a medium and objects that occlude it from the light
- Occluders are typically represented as rendered into shadowmap
Godrays

At each ray-marching step

- Medium transmittance is calculated
- Shadowmap is fetched
Godrays

- We don’t need to do that many ray-marching steps if the medium is uniform
- We need to know the contents of the volume
Godrays

- Instead of rendering the volume, let’s render the actual geometry of godrays!
Tessellated godrays

- Scene
Tessellated godrays

- Shadowmap
Tessellated godrays

- Render grid
Tessellated godrays

- Tessellate it
Tessellated godrays

- Fetch from shadowmap and offset vertices
Tessellated godrays

- Don’t forget the cap
Tessellated godrays

- Integrate with positive sign for backfaces
- Integrate with negative sign for frontfaces
Tessellated godrays

- Integrate with positive sign for backfaces
- Integrate with negative sign for frontfaces
Tessellated godrays

- Integrate with positive sign for backfaces
- Integrate with negative sign for frontfaces
Tessellated godrays

- Integrate with positive sign for backfaces
- Integrate with negative sign for frontfaces
Tessellated godrays

- Result
Why use tessellation?

- Tessellation allows making grid resolution adaptive
Why use tessellation?

- Tessellation allows making grid resolution adaptive
Adaptive tessellation

- Tessellated grid
Adaptive tessellation

- Optimized grid
Adaptive tessellation

- Geometry of godrays
Advantages

- Up to 4X performance improvement
- No banding, no aliasing
- Ability to represent small details
Assassin’s Creed IV Black Flag

- We integrated tessellation-based godrays into Assassin’s Creed IV Black Flag
- A joint project of Ubisoft Kiev and NVIDIA
Assassin’s Creed IV Black Flag

Godrays OFF
Assassin’s Creed IV Black Flag

Godrays ON
Assassin’s Creed IV Black Flag

Godrays OFF
Assassin’s Creed IV Black Flag

Godrays ON
Assassin’s Creed IV Black Flag

Godrays OFF
Assassin’s Creed IV Black Flag

Godrays ON
Assassin’s Creed IV Black Flag

Godrays OFF
Assassin’s Creed IV Black Flag

Godrays ON
Which effects can benefit from it?

- Computation-heavy effects with low frequency
  - Particle shadows
  - Volumetric effects
  - Global illumination
  - ...

GAME DEVELOPERS CONFERENCE® 2014

MARCH 17-21, 2014  GDCONF.COM
Which effects can benefit from it?

- Computation-heavy effects with low frequency
  - Particle shadows
  - Volumetric effects
  - Global illumination
  - ...

AC4BF screenshot was made by Andrew Iain Burnes and published at GeForce.com
Tips and Tricks

- Adding tessellation to your game is not that straightforward
- These corner-cases require attention:
  - Tessellation vs. Depth Pre-Pass
  - Tessellation vs. Shadowmapping
  - Tessellation vs. Decals
Tips and Tricks

- Tessellation vs. Depth Pre-Pass
- Tessellation vs. Shadowmapping
- Tessellation vs. Decals
Tips and Tricks

- **Tessellation vs. Depth Pre-Pass**
- **Tessellation vs. Shadowmapping**
- **Tessellation vs. Decals**
Tessellation vs. Depth pre-pass

- Tessellating during depth pre-pass can kill the performance benefits of depth pre-pass
Tessellation vs. Depth pre-pass

- Turn depth pre-pass off
- Don’t use tessellation in depth pre-pass
  - Use always positive tessellation
  - Configure depth test properly
Tessellation vs. Depth pre-pass

- Not tessellating in depth pre-pass
Tessellation vs. Depth pre-pass

- Not tessellating in depth pre-pass
Tessellation vs. Depth pre-pass

- Not tessellating in depth pre-pass
Tessellation vs. Depth pre-pass

- Use always-positive displacement
Tips and Tricks

- Tessellation vs. Depth Pre-Pass
- Tessellation vs. Shadowmapping
- Tessellation vs. Decals
Tips and Tricks

- Tessellation vs. Depth Pre-Pass
- **Tessellation vs. Shadowmapping**
- Tessellation vs. Decals
Tessellation vs. Shadowmapping

- Tessellating while rendering to shadowmap can kill performance
Tessellation vs. Shadowmapping

- Turning tessellation off in shadowmaps can introduce artifacts
Tessellation vs. Shadowmapping

- We decided to turn tessellation off in shadowmaps in Metro: Last Light
- This introduced artifacts that artists had to fix by tuning the content
Metro: Last Light

Tessellation OFF
Metro: Last Light

Tessellation ON
Metro: Last Light

Tessellation OFF
Metro: Last Light

Tessellation ON
Tessellation vs. Shadowmapping

- Use always-positive displacement
Tessellation vs. Shadowmapping

- Use always-positive displacement
Tessellation vs. Shadowmapping

- Use always-positive displacement
Tessellation vs. Shadowmapping

- If performance is not a problem, what tessellation factor to choose for shadowmap?
  - The same as was used for main screen rendering or
  - Calculated relative to shadowmap camera
Tessellation vs. Shadowmapping

- Problem of camera and light opposing each other
Tessellation vs. Shadowmapping

- Problem of camera and light opposing each other
Tessellation vs. Shadowmapping

- Camera and light oppose each other

Camera view, no shadows

Camera view, shadows enabled

Shadowmap view
Tessellation vs. Shadowmapping

- Using main camera tessellation factor
Tessellation vs. Shadowmapping

- Using shadowmap tessellation factor

Camera view, no shadows, wireframe

Camera view, shadows enabled

Shadowmap view
Tessellation vs. Shadowmapping

- Choose the maximum tessellation factor from the main screen factor and shadowmap factor
- Make sure to not generate sub-pixel triangles
Tips and Tricks

- Tessellation vs. Depth Pre-Pass
- Tessellation vs. Shadowmapping
- Tessellation vs. Decals
Tips and Tricks

- Tessellation vs. Depth Pre-Pass
- Tessellation vs. Shadowmapping
- Tessellation vs. Decals
Tessellation vs. Decals

- Tessellated geometry can penetrate through decals
- We had this problem during the development of Metro: Last Light
- Artists had to fix it by tuning the content
Tessellation vs. Decals

- Use always-negative displacement
Tessellation vs. Decals

- Use always-negative displacement
Tessellation vs. Decals

- Use always-negative displacement
Tessellation vs. Decals

- Runways in H.A.W.X. 2
- Modulate based on normal’s vertical component
Tessellation vs. Decals

- Use “screen space decals” technique
- Pope Kim, Screen Space Decals in Warhammer 40,000: Space Marine, Siggraph 2012
Conclusions

- Tessellation can be used to produce spectacular images on all platforms.
- Use your triangles wisely!
- The new paradigm of varying shading rate can bring significant speedup to your effects.
Conclusions

- When adding tessellation to your title, keep these in mind:
  - Tessellation vs. Depth pre-pass
  - Tessellation vs. Shadowmapping
  - Tessellation vs. Decals
References

- Pope Kim, *Screen Space Decals in Warhammer 40,000: Space Marine*, Siggraph 2012
Acknowledgements

Nvidia
Alex Kharlamov
Nick Chirkov
Jon Jansen
Louis Bavoil

4A Games
Oles Shishkovtsov
Yuriy Saschuk
Sergei Karmalsky

Dice
Johan Andersson
Mattias Widmark

Futuremark
Jani Joki
Juha Sjoholm

Ghost Games
Filip Karlsson 2nd

Ubisoft Kiev
Sam Kovalev
Roman Bobel
Kolya Naichuk
Artem Kandinskyi

Ubisoft Sofia
Razvan Eliade
Thanks!