PERFORMANCE AND GRAPHICS OPTIMIZATION ON BETA CAE PRODUCTS

Benjamin Coquelle, Demir Ali
AMD, USA

KEYWORDS –
Graphics performance

ABSTRACT –
We will describe how beta cae and AMD improved ANSA rendering engine by using new technology

TECHNICAL PAPER -

1. VBO GRAPHICS TECHNOLOGY AND BETA CAE SOFTWARE

VBO technology

VBO stands for Vertex Buffer Object[1]. It is an OpenGL[2] technology for vertex data (position, normal, color...). Since the storage for buffer objects is allocated by OpenGL, vertex buffer objects are a mechanism for storing vertex data in "fast" memory (i.e. video RAM), thereby allowing for significant increases in vertex throughput between the application and the GPU.

The main difference between the classical OpenGL drawing mechanism, called immediate mode, and VBO is the application doesn't need to send the data each time to the GPU from system memory to GPU memory (from RAM to Video RAM). By doing that the application avoids to send data through PCIe bus which is most of the time the bottleneck on CAD/CAE application

	RAM	PCIe Gen2	Video RAM
Memory bandwith	17GB/s	8GB/s	147.2GB/s

This technology was adopted in OpenGL core 1.5.
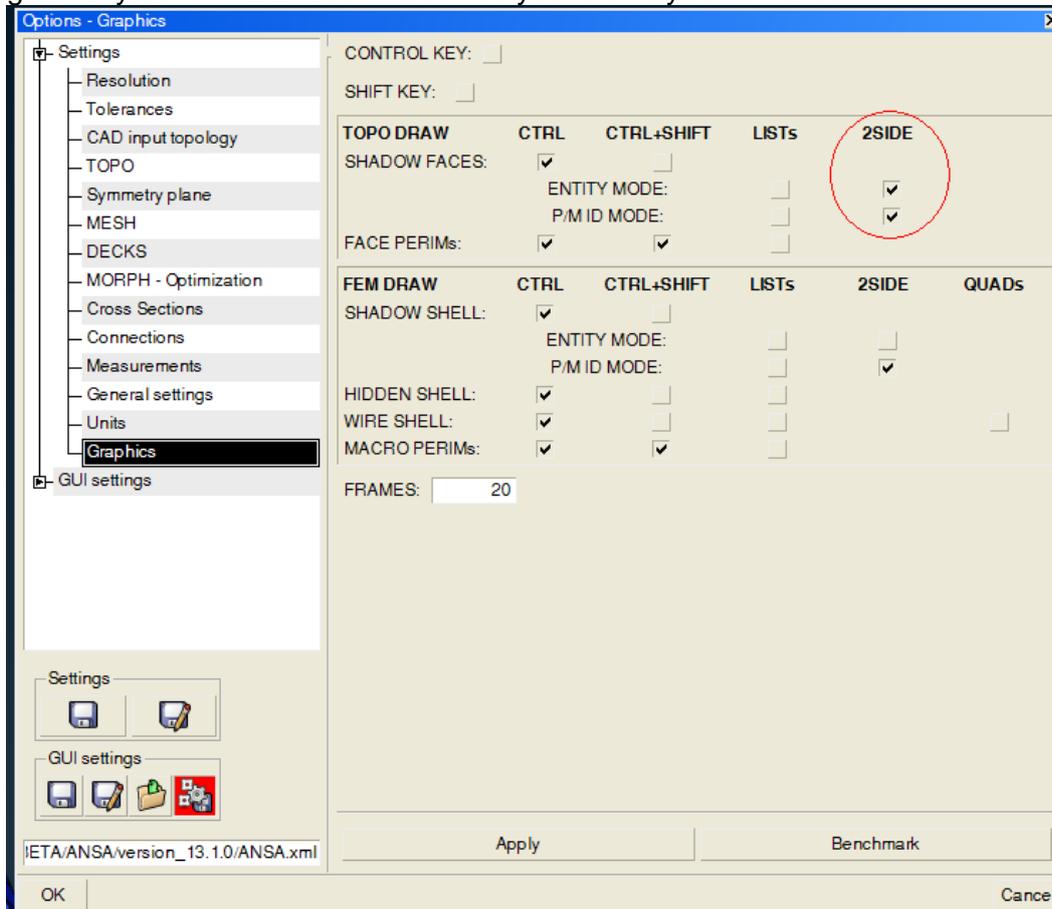
How BETA CAE took advantage of VBO

BETA CAE did a great job during the last year to move from immediate mode to VBO technology. By using VBO you have more constraints since you will have to maintain the geometry on the GPU but you don't have a direct access to it. Moreover you need to create relatively big geometry buffer on the GPU to actually see improvements.
With the release 13.1 in December 2010, BETA CAE introduces VBO for the geometry and it gave a huge boost on Ansa graphics performance.

Ansa has several graphics optimization mode one can set using this option – performance_mode with different option:
- Mode 0 and 1, ANSA will use VBO
- Mode 2 and 3 ANSA will use vertex array, it is an older technology but it helps the driver to send a set of vertices in one time so you can still expect good performance.
- Mode 4 and beyond will use very old opengl which won't give you good performance, but at least it allows everyone to run ANSA on almost every systems

However it is not because you choose to use VBO that ANSA will be able to use them all the time.

For example if you don't select 2 Side (stands for two side lighting), ANSA is not able to use VBO because ANSA will have to check at each frame the normal orientation and maybe reoriented depending on the viewpoint. Because of that ANSA is not able to create static geometry buffer. As a result we advise you to always use 2 Side.

VBO limitation with μeta

μeta is mostly use in simulation as a result there are a lot of animations running on μeta. In those case the geometry changes very often and it is very hard to create VBO. One can argue video games are handling that, but in video games animation are predefined while in application like μeta it depends on real life physics so you cannot predefine your geometry. In that case vertex array will be use to optimize the performance

New technology upcoming

With the problem describe above, AMD has thought about another way of using the GPU to draw.

We are introducing the "pinned buffer" technology which will allow the GPU to read directly from the system memory to draw. This way you reduce the memory transfer, the drawing pattern can change from :

- *CPUmemory->GPU memory->vertex processing*

To :

- *CPUmemory ->vertex processing*

Which will help in animation process, when application have a streaming of geometry information like μeta.

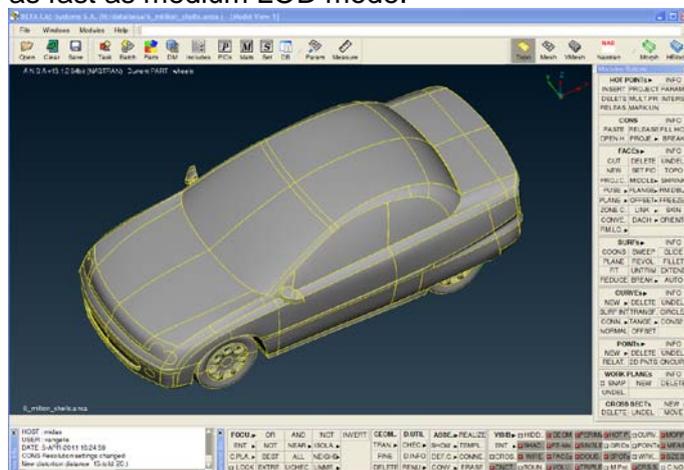As a result the source code will be easier to maintain and will be faster.

2. PERFORMANCE ENHANCEMENT IN DIFFERENT RENDERING MODE

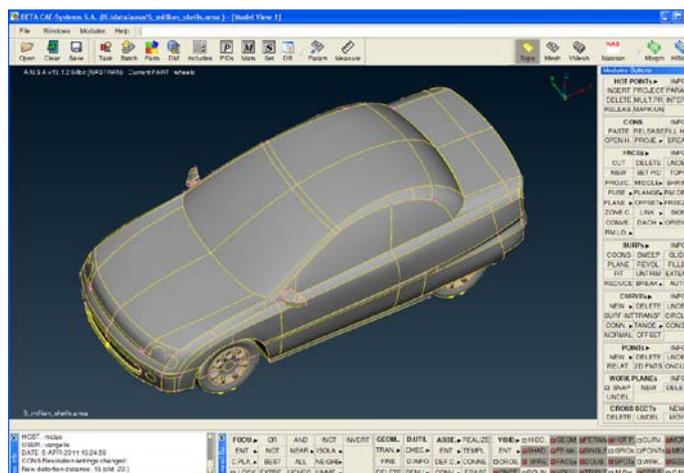|	ANSA 13.0+V7750	ANSA13.1+V7800	ANSA 13.2+V7900
CUSTOMER DATA	2 FPS	7 FPS	8 FPS
CUSTOMER DATA - FULL LOD	<1 FPS	2 FPS	7 FPS
BETA CAE DATA	10 FPS	14 FPS	15 FPS

The array above shows the performance of specific dataset across different ANSA release and HW.
As you can see the introduction of VBO and new hardware with ANSA 13.1 gave a huge boost in performance. However we were still slow in Full LOD mode because ANSA were using massively immediate mode OpenGL which requires a lot of CPU time so the graphics card was not fully used.

With introduction of ANSA 13.2, BETA CAE did a incredible job to use VBO on markers so full LOD can be hardware accelerated too. As you can see FULL-LOD mode is now almost as fast as medium LOD mode.

Medium LOD

Full LOD

3. AMD COMMITMENT TO BETA CAE

Q&A

BETA CAE products are key to AMD and their partner, that is why we want to be sure the end user can have the best experience has possible by running BETA CAE product on AMD graphics card.
Before each FirePro driver release we run manual test to ensure the performance and the reliability of our products with BETA CAE application.

BETA CAE gets access to early technology

The other aspect of this commitment is to give a early access to new hardware and software technology to BETA CAE.

They have received our new generation card (FirePro V7900) last May and we are sending them beta driver to test our latest work as soon as possible so they can report us any problem

The pinned buffer drawing technology has been presented to BETA CAE this spring and we hope we will be able to work together to implement this technology to µeta as we think it will bring new performance enhancement.

Our big next step is to work together on OpenCl with our new APU Hardware technology (fusion[3]).
This technology allows AMD to regroup on the same DIE (chip) a CPU and a GPU. From then on we can have system configuration with :
- a APU to work on the compute side
- a discrete GPU to draw

The GPU can access quickly the system memory (used by the APU) thanks to our new pinned buffer technology so we can transfer data very easily between the two devices.

4. CONCLUSIONS

We started working more closely with BETA CAE a year ago and BETA CAE has already made a tremendous job by taking into account our advice. As a result the end user is now working with a fast and reliable application which helps him to work faster.
We will continue to work closely with BETA CAE to make sure end-users benefit from the best applications as possible which take the most of the current hardware.

REFERENCES

(1)   http://www.opengl.org/wiki/Vertex_Buffer_Object

(2)   http://www.opengl.org/

(3)   http://sites.amd.com/us/fusion/apu/Pages/fusion.aspx