Advances in Real-Time Rendering in 3D Graphics and Games course (I & II)

Siggraph 2010

Los Angeles, CA

Introduction: Graphics Feature Development for Games

Natalya Tatarchuk

Bungie

More Than Pretty Pixels

- Visual bar for computer games has risen sharply
  - It's assumed that a AAA title is going to have outstanding visuals
- Focus on the whole package
  - Treat game as a cohesive collection of features
  - Make smart choices of what to focus on

We often hear people ask: "is graphics a solved problem"? "Are we done now"? Well, no, far from it.

In previous generations it was enough to focus on normal mapping or some other technique that made the pretty pixels prettier. It was enough of a challenge just to get it to work. While there's a lot of innovation still happening in those areas, in pushing the hardware to the limit;

Now that's not enough – we have to do more! (with often less frame and memory budget..)

Rather than any one or two key graphics features

Even if graphics is crucial to the game (like Halo Reach, which you see in this image), you still have to distribute your budget among your features for a given frame. Graphics is not the only thing that's competing for the performance budget (33 or 16 ms) - but certainly one of the most performance-critical aspects of the game. If we break down the frame you see here, you'll see elements of a complex world, many hero vehicles and characters at once, effects, destructible environments, detailed lighting, terrain, far away elements of the environment, and that's just the tip of the iceberg!

So we have to be smart about what features we choose.

How Will You Stand Out From The Crowd?

- Focus on what matters to the end user

Differentiation is important. Many games are hitting the photoreal bar right now and even more will in the future

So, how will you stand out from the crowd: focus on what matters to the end user

> Example: RPG versus first-person shooter – different priorities

> Whatever details that the user can perceive for the game genre your're working on

> What is the quality that they can feel during game play

> Sometimes it's not trivial to know what really matters to the end user

> Many times we, as graphics programmers, get hung up on perfect shadows, etc., but it's possible that the audience for your game won't care about it.

Starting in Halo3 we had user-created screenshots. When we looked at the shots that users created, and what they felt excited about – we noticed that it wasn't necessarily correlated to what we as graphics programmers spent most time on. For examples, players always loved shots of big explosions, nice shiny armor shots, and pick out little things that we didn't they would pick out. That tells us that what a graphics programmer sees and what the end user sees is not always the same!. (though many times they compliment each other nicely).

How Will You Stand Out From The Crowd?

- Focus on what matters to the end user
  - It's all about player experience!
  - In the *player's living room*!

Make sure you're looking at the results the same way that the player would be looking at. Must author your content for the same visual target and resolution as you expect your player to view it.

How Will You Stand Out From The Crowd?

- Focus on what matters to the end user
- Use graphics as a story-telling element
- Graphics is a realization of artistic vision

Use graphics as a story-telling agent. Making stories more compelling, making worlds more immersive. Setting the mood: atmospheres, time of day, lighting, shadows, shading: what conveys the sense of surroundings and the sense of the story. Graphics is the vehicle that delivers that to the player and makes them feel the story that the game designers had in mind. For example, Anton Kaplanyan will talk about how Crytek used their deferred renderer to make the next-gen environments of Crysis 2 come alive.

Use the concept for your game to drive what graphics features are important for your implementation. In the end, it's about artistry, and not about fancy technology in isolation (Artistry – not technology for technology's sake). A good example of that is the method that the guys from Avalanche used to translate Pixar's vision into the medium of games, or how the artists' vision for characters was executed in Uncharted 2.

Games Are a Unique Medium

- Games are about pixels in motion

Dynamic techniques are crucial

   Games are not a static medium

Games are 4D: time is a very important dimension!

   It's just about screenshots!

   Don't overlook the time axis

Temporal coherence! A fair amount of time for SSAO, lighting, shadow techniques goes into removing artifacts that are tolerable in a static image but become very aggravating to the player when moving around (for example, dancing aliasing in shadows, or the shower door effect in SSAO implementations). That's just an example of a common theme in many talks today.

Games Are a Unique Medium

- Games are about pixels in motion
- Don't forget about input lag
- Design and gameplay needs to define what gets exposed to the player

Input lag requirements vary for each game genre

How does your feature influence the input lag

The design of your game controls what you need to expose for your features- and what you should focus on. Alex Vlachos will talk in the later part of the day how design of Portal 2 influenced the design of his water flow system – and how the water flow system influenced the game play.

It's All About the Pipeline

- Content longevity and content production efficiency are key
- Polish == iteration
- Good IP is alive for ~5 years; great IP – 10+!
  - This means all assets are likely to be around for just as long
  - Can't go through and rework them for big budget titles for every new release

Successful games are not cheap to produce

> Great part of that cost is content generation

Techniques that improve content workflow are crucial

> Including offline toolchain techniques

Titles often live for 3-4 years (through out their development cycle). Good IP will be alive for 5 years, great IP – 10! Now the asserts that were created for a particular technique

What Makes A Technique Successful in Games?

- Integration with the artist workflow and the engine / tools pipeline
  - 30% : 70%

A common theme in the talks throughout the course is how does a graphics feature go through the pipeline. For example, Per Einarsson and Sam Martin will discuss how they integrated global illumination into Frostbite's engine and their choices along the way. Robert Kihl will talk about how they improved the artist workflow for generation of destructible environments.

What Makes A Technique Successful in Games?

- Integration with the artist workflow and the engine / tools pipeline

- Does it effect gameplay positively?
  - Examples: water flow / GI / thousands of lights in a deferred renderer / shadows, and many more…

Not Just About Performance

- What are the artist controls?
- How does it deal with other elements in the game? (effects, leaf systems, etc.)
- Multi-platform support

A good technique is judged not just on performance alone

We Know We Can Optimize Shaders…

How the heck do we get this thing into memory? Runtime storage, load times, runtime bandwidth considerations… Can it be stored in an efficient manner on the GPU?

What Are the Goals for This Course?

- Bring you the latest techniques from the game development community
- With focus on practical, pipeline-friendly techniques
- Share common challenges encountered during game development (wrt graphics techniques)
  - With the eye toward the future
- Inspire new research directions!

Advances in Real-Time Rendering in 3D Graphics and Games I

Wednesday, 28 July | 9:00 AM - 12:15 PM | Room 515 AB

Time	Session
9:00 am	Tatarchuk **Introduction**
9:05 am	Ownby, Hall and Hall **Rendering techniques in *Toy Story 3***
9:55 am	Einarsson, Martin **A Real-Time Radiosity Architecture for Video Games**
10:45 am	Yang, McKee **Real-Time Order Independent Transparency and Indirect Illumination using Direct3D 11**
11:30 am	Kaplayan ***CryENGINE 3*: Reaching the Speed of Light**

Advances in Real-Time Rendering in 3D Graphics and Games II

Wednesday, 28 July | 2:00 PM - 5:15 PM | Room 515 AB

2:00 pm	Lauritzen **Sample Distribution Shadow Maps**	
2:25 pm	Salvi **Adaptive Volumetric Shadow Maps**	
2:55 pm	Hable *Uncharted 2*: **Character Lighting and Shading**	
3:45 pm	Kihl **Destruction Masking in *Frostbite* 2 using Volume Distance Fields**	
4:10 pm	Vlachos **Water Flow in *Portal 2***	
5:00 pm	**General Q&A**	