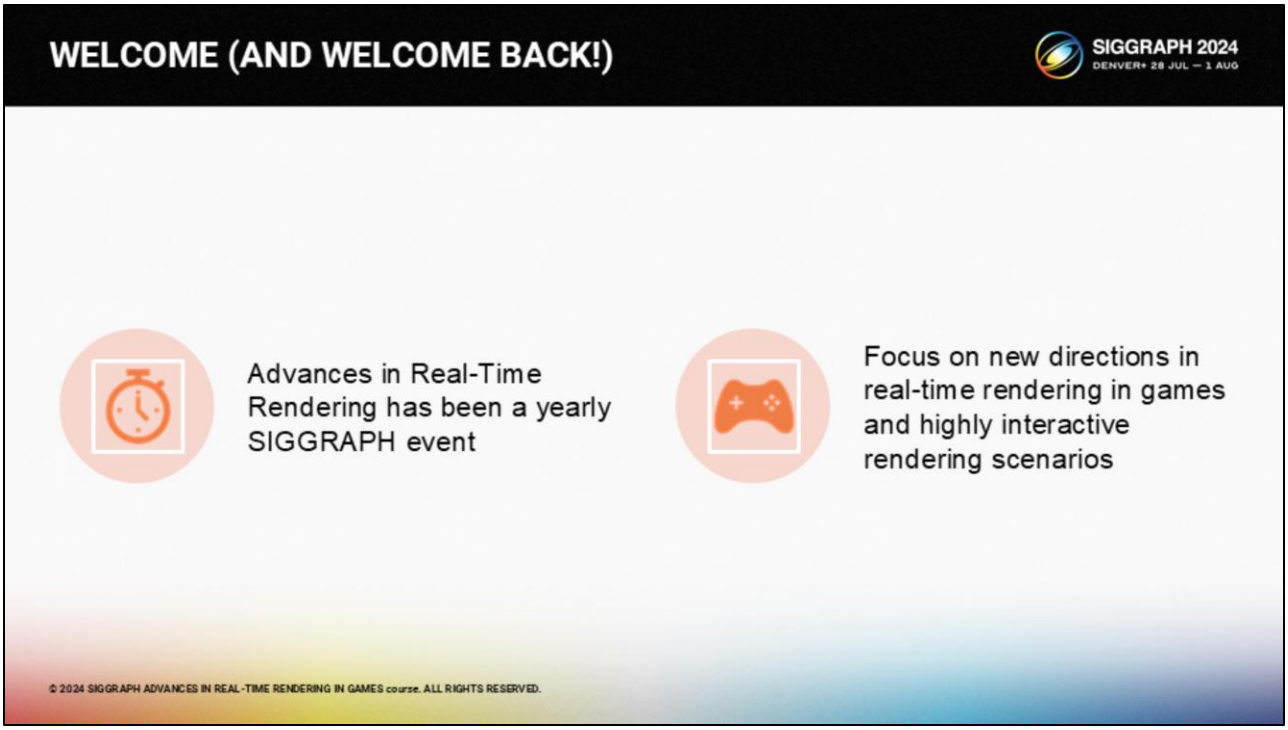
Hello! Welcome to the SIGGRAPH 2024 Advances in Real-Time Rendering in Games course. My name is Natalya Tatarchuk, and I am Chief Technical Officer at Activision.

Today, I'm honored to have organized this course alongside all these incredible speakers, who worked incredibly hard to put together all the materials for the two-part course on the topics of latest and greatest techniques in games and other fast real-time rendering scenarios.

This course wouldn't be possible without their outstanding dedication, and I want to thank them deeply before we begin.

Advances has been a yearly event at SIGGRAPH for close to two decades now - which is a huge privilege and I’m grateful for all the support from the conference. But also - for all the people here, who found our topics interesting and keep coming back!



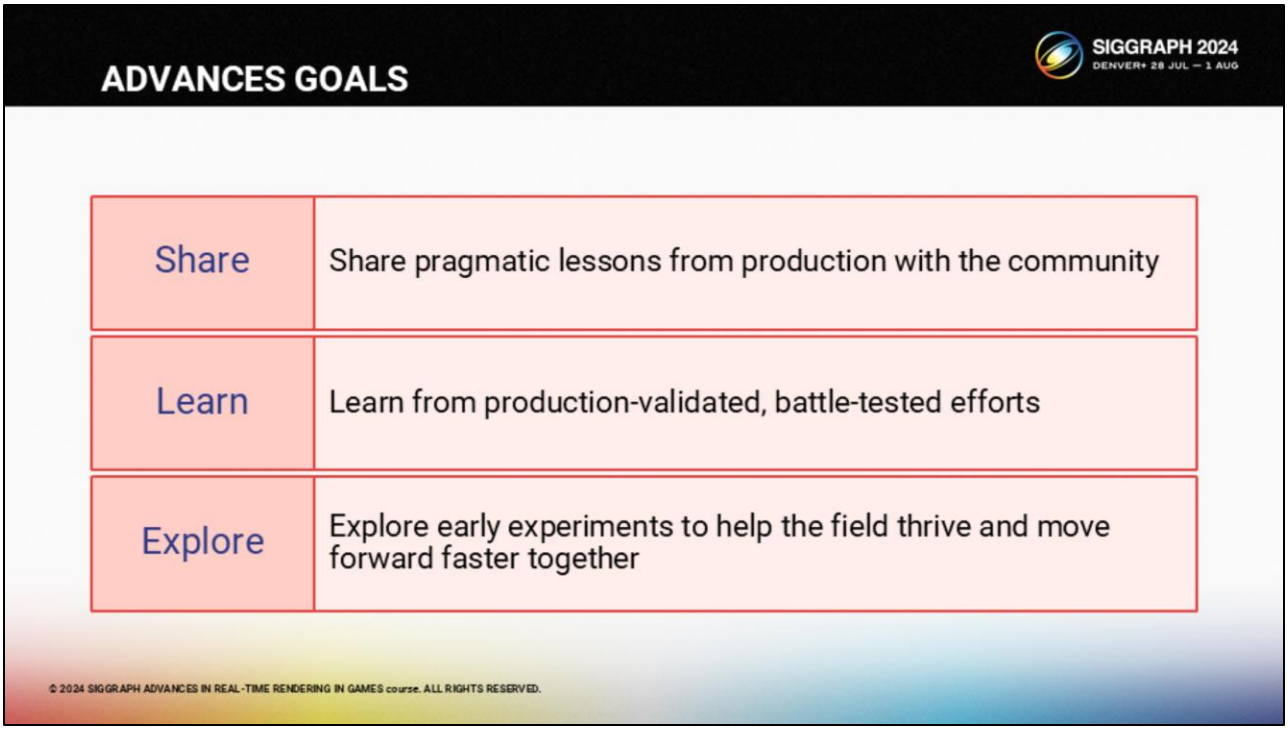
Welcome - and welcome back!

Advances was born in 2006, from my [at the time] frustration of not seeing shared techniques on how to do things in real-time rendering in games, We've come a long way from then, and there are many avenues of learning now, from conference courses, to books on real-time rendering, to so much more – including various people's blogs, which is so wonderful. Still, I am happy that we get to continue to bring new content to Advances this year - new every year. And there even have been a few other Advances spun up - a few years ago there were Advances in animation and rigging; Advances in Neural Rendering, and this year we see Advances in Real-Time Rendering of *Sound!*

Though *this, OG,* Advances has been a yearly event at SIGGRAPH for a number of years, this forum is anchored in exploring new directions in real-time rendering, with a focus on techniques targeted towards practical applications such as video games, and highly interactive (30 - well, now 60+ fps and beyond) rendering on widely available consumer hardware.

In fact, this is our 19th year appearing at SIGGRAPH and a number of fantastic topics have been introduced in this forum, from SSAO, TAA, stochastic SSR, several approaches for dense geometry rendering with GPU-driven compute-based pipelines, compute-based tiling and clustered culling, variety of global illumination approaches and improvements, software VRS, and much much more, throughout the years.

We're grateful to SIGGRAPH for continuously supporting this forum of sharing the lessons from productions and cutting edge experimental techniques w/ our real-time rendering community throughout the years.



The goals for the Advances courses over the years has been simple: foster growth of the real-time graphics community by sharing pragmatic lessons from real-world productions. We also didn't shy away from early experiments, so long as they were designed with practical productions in mind. Our communal knowledge grows faster when learning from such early explorations, building an understanding of what worked, as well as - and sometimes more importantly - what failed to work and why.

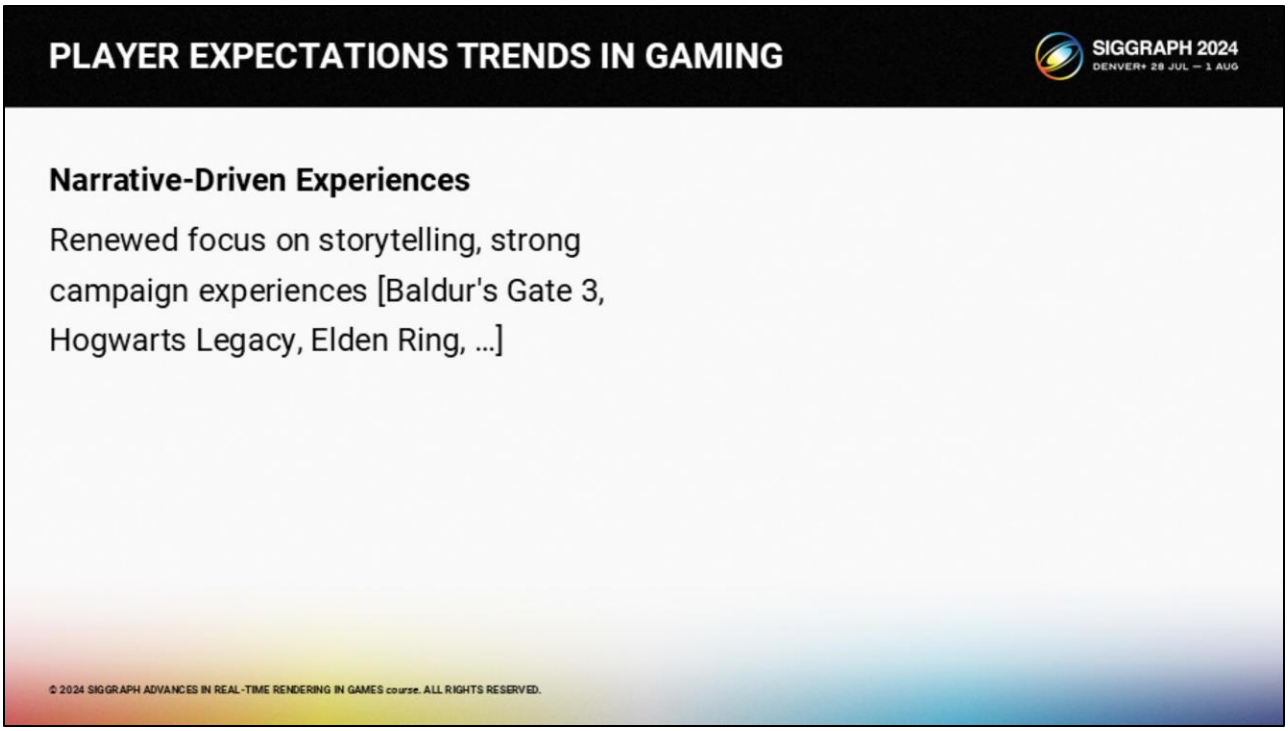
Another key pillar for Advances talks has been to focus on implementation-centric knowledge sharing, to help build better intuition for the deeper core of each technique. We always aim for every talk to provide clear, reproducible algorithms and implementation details. The authors' goals is to share concrete pragmatic considerations for each technique, not only focusing on the algorithmic details, but also on relevant performance considerations, the method's benefits as well as its limitations and constraints.

At the heart of the principles behind the choices for Advances talks is *production-driven innovation*. Each year we strive to deliver completely new content, and this year is no exception - we have a full slate of new content with the latest and greatest techniques

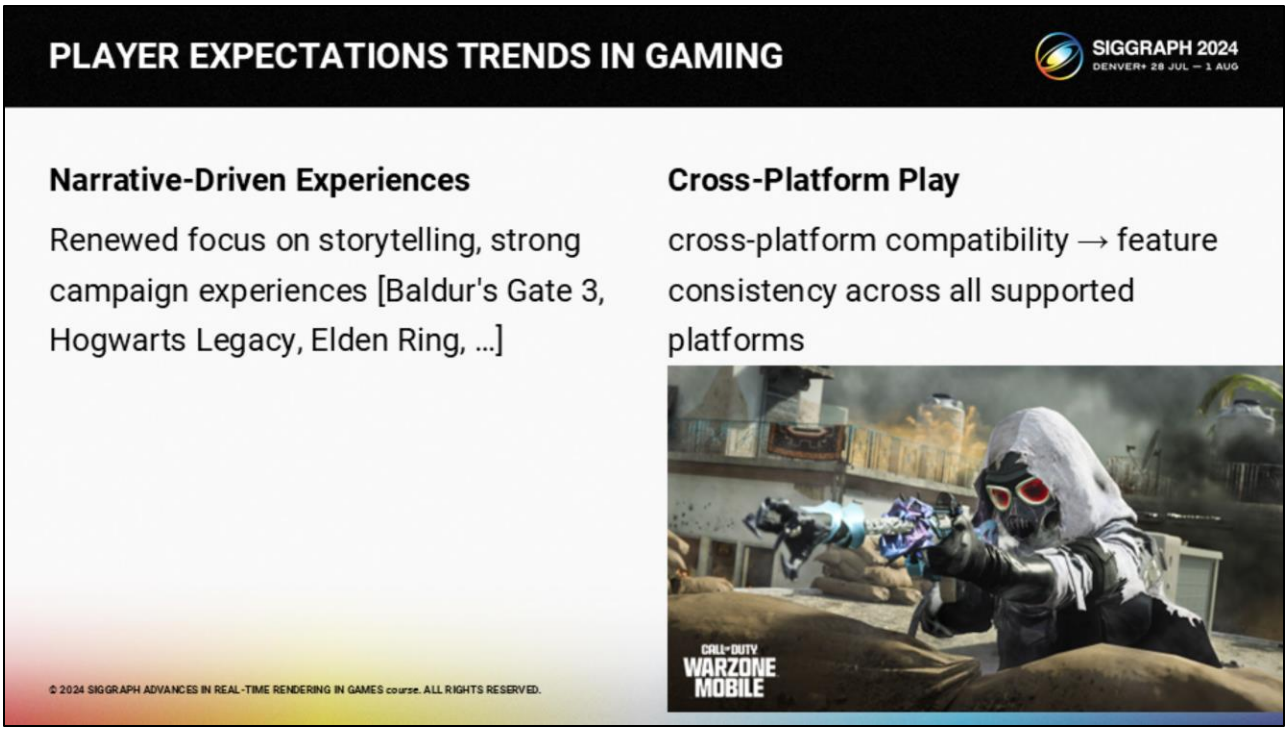
I wanted to share some thoughts on what trends we're seeing, that influenced some of the content you are seeing this year in Advances, as well many conversations with the developers. Now this is my own set of opinions, also fortified and influenced by thoughts and discussions with various folks (who I sampled from generously). These opinions are not intended to represent an official point of view of a company, but, rather, an individual.

I apologize as the next set of slides will be mostly text, as due to the ACM 3P image permission policy. Imagine amazing screenshots of all the games and tech I'll mention - let your brain hallucinate images along with my words :)

I want to start from looking at the expectations that players have from games today.



There’s a renewed focus on storytelling, with games offering deeper, more complex narratives and character development, often with branching storylines based on player choices. Players want compelling stories: narrative and non-PvP experiences are starting to thrive in a post COVID world
And single player came out strong this year: Baldur’s Gate 3, Hogwarts Legacy, Elden Ring, Starfield, Jedi Survivor, Spider-Man 2, Ghost of Tsushima, and many more, bringing with this emphasis on cinematics, and characters rendering in full force, along with powerful vistas.



Many AAA titles are focusing on cross-platform compatibility, allowing players on different consoles and PC to play together, which is increasing player bases and engagement. Players tell us to meet them where they are. They also want to play with their friends wherever they might be. If you want to have inclusive audiences for your games, it's not enough to have appealing content that's inclusive - you also need to run on variety of min-specs, because people come from all socioeconomic backgrounds and this helps people find a place in a game, as well as play with more of their friends.

Cross-play and cross-platform is also important for sustainability of current gaming business models. Games business model is undergoing a significant transition post-COVID and we saw even first-party titles adding support for PC, 3P console (like Sea of Thieves), or mobile in the recent year.

Thus, true cross-platform feature support, and scalability is ever so critical, as well as ability to author content in a scalable “author once” manner to save production costs and testing whenever possible.

We will come back to the need to deliver scalable content at low authoring cost and the connection to a vast platform fragmentation in a bit.

PLAYER EXPECTATIONS TRENDS IN GAMING

SIGGRAPH 2024
DENVER+ 28 JUL — 1 AUG

Narrative-Driven Experiences

Renewed focus on storytelling, strong campaign experiences [Baldur's Gate 3, Hogwarts Legacy, Elden Ring, …]

Live Service Models

Continued releases of live game updates, with seasonal content, regular updates and battle passes

Cross-Platform Play

cross-platform compatibility → feature consistency across all supported platforms

AAA games, and even AA indies, continue to adopt or expand live service models, with regular updates, seasonal content, and battle passes to keep players engaged long-term. Besides that, gaming subscription services are still looking to define themselves: Game Pass, Netflix, PS+, and Nintendo Online - subscription services continue to experiment in their offering and pricing to try and grow while maintaining favorable economics and content offering to the users.

For engineers this means that we have to support evergreen content features, and evolving to add new functionality in a backward compatible way, across the entire ecosystem of platforms.

The amount of content continues to increase with every season, every patch, and every offering. So this trend also puts continued pressure on memory and disc footprint, as well as cloud-streaming, to deliver the breadth of all that live game content to the players at speed across the globe.

Provocative question – with this rapid delivery of content to players, does this portend the end of baking?

The need to release content at this rather fast pace necessitate far more fluidity to generate content, and heavy-weight processes such as lengthy baking are at odds with that need. So we should seek improvements that speed up the build iteration time *especially* in the cases where we need to release and patch content at this frequency.

Scale: Tears of the Kingdom, Elden Ring, and Shadow of the Erdtree have shown there is a lot of demand in massive experiences that overwhelm the player. Vistas in all major games continue to be a talking point and I don't think this goes away.

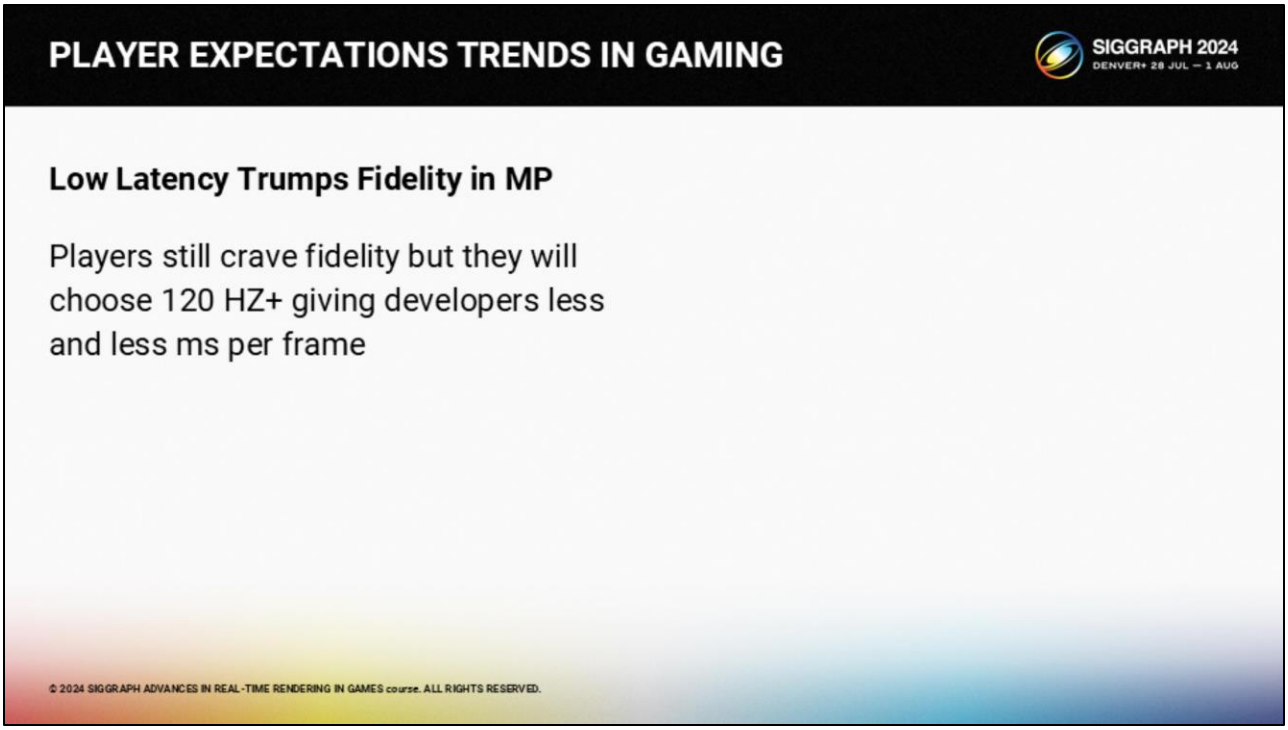
Immersion:
Of course, immersion for players is very much about fidelity, across the board [though see the asterisk on that in a sec]

But also players want to get lost in the game.

They want everything to be possible (e.g., Baldur's Gate III). This is getting more and more important for gamers. How much freedom a player has and how a game reacts to their choices is something that is top of mind and something that players can see and feel. The Mass Effect dialogue wheel isn't going to cut it in the future of gaming. Everything being scripted is not going to cut it. Much interest stems from the AI-driven runtime NPC dialogue systems, whether what Unity shown at SIGGRAPH last year with their Sentis Alien demo

(https://youtu.be/OYf6h0Dn24k?si=BD8T6Ig3tmcBwI3k&t=3403) or the NVIDIA ACE demo or many other offerings from start-up and tools developers.

But we still yet to prove that this can be done performantly on consoles and mobile in game and that it will fit well into the character authoring pipeline, so it’s a great frontier. This also needs higher quality JIT animation generation, from JIT state machines to speech or text to facial animation pipelines.

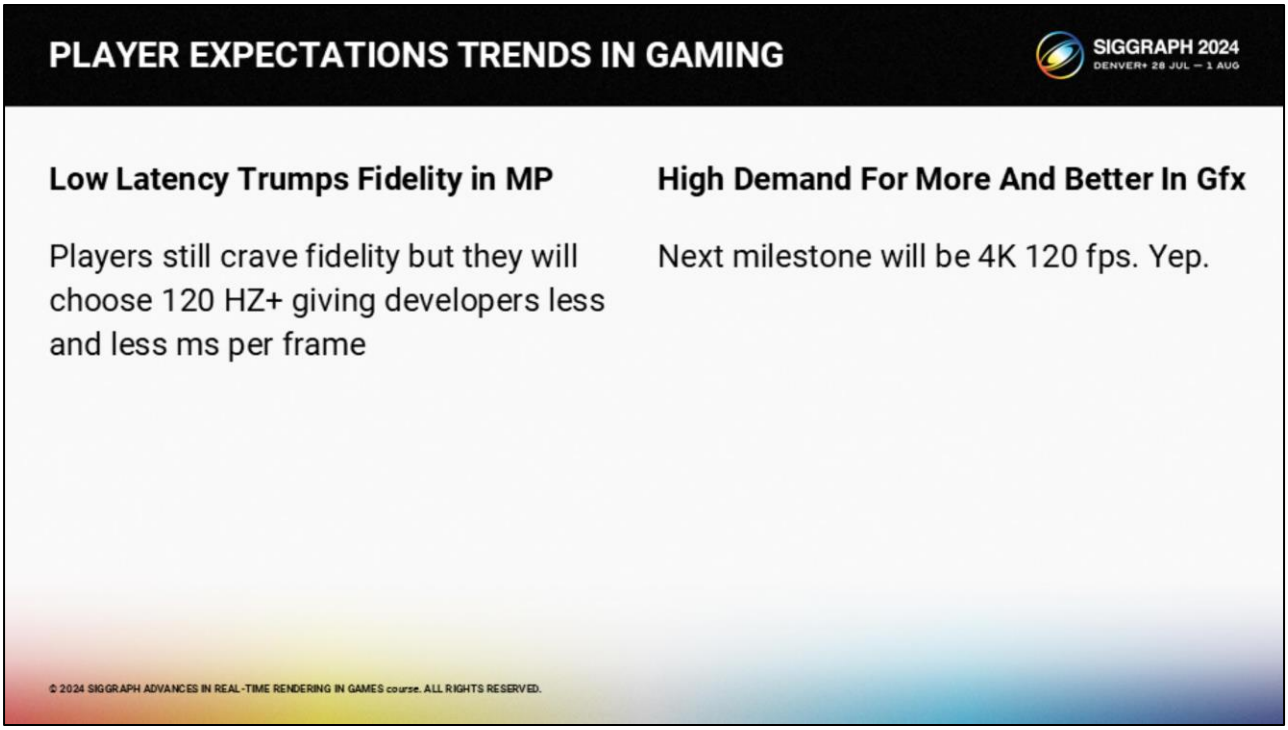


Graphics and fidelity are already in a great place but I think there's probably 2 big pushes we're seeing:

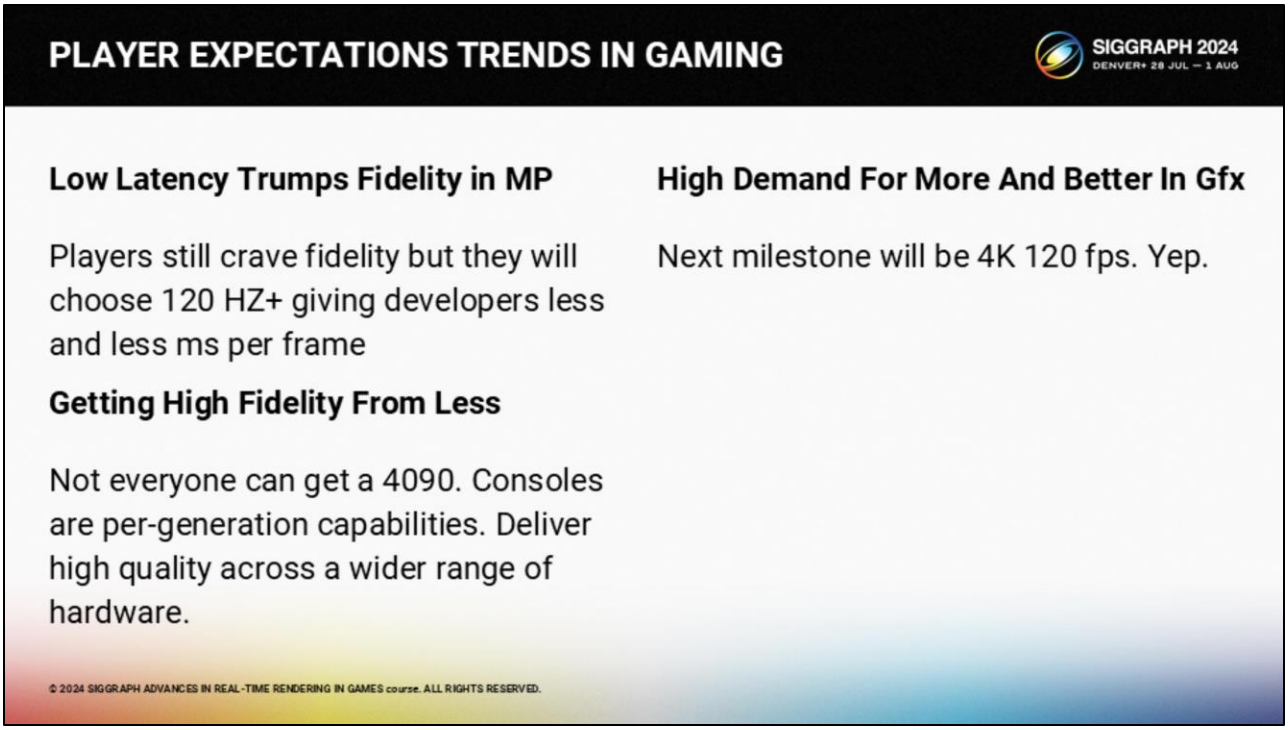
Players care about low latency - they still crave fidelity but they will choose 120 HZ+. Many games are unlocking and offering 300 HZ [Call of Duty, Fortnite, Overwatch, others]. This, of course, means you have less ms for the graphics calculations per frame to deliver that latency, jitter-free.

And the manufacturers continue to push the boundary of what players can experience – for example, ASUS just put out a 480Hz 1440p OLED.

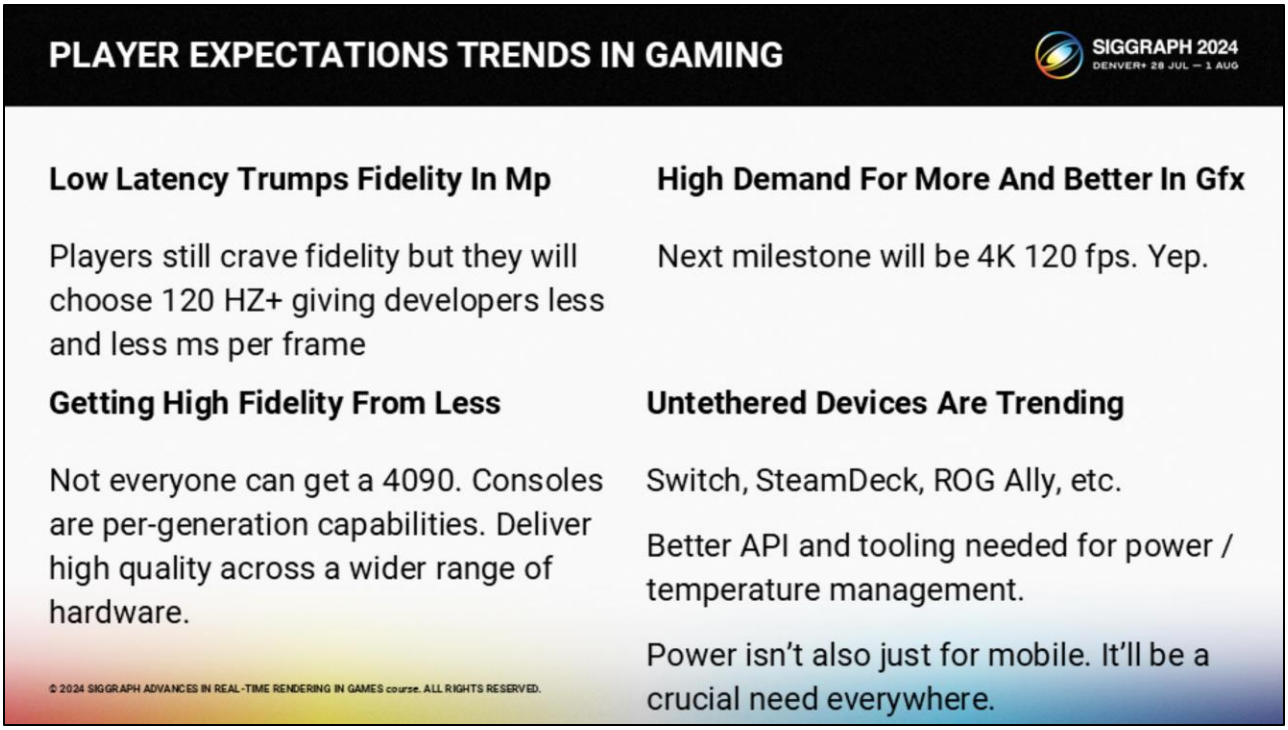
Of course, we could just hallucinate more frames? That’s the future!



Yet there's always a high demand for more and better when it comes to graphics. I believe from the graphical standpoint the next big milestone is likely getting 4k 120 fps+ - consistently. You've got folks at some IHVs trying to drop buzzwords like 8k gaming but really we haven't cracked 4k 120fps as an industry.



Getting high fidelity from less is the other big push. Not everyone can get a 4090. Long console cycles contributes to the need to extend what we can get from a given hardware system. We're starting to see more things like internal 1080p upscaled to 4k (e.g., just-in-time ML upscaling from a cool GDC talk by Sony Santa Monica). And while this helps increase the player base and helps the games business model - this is also about creating equitable experience for inclusivity for games - letting players across variety of backgrounds enjoy the features maximally.



Untethered devices are trending with players everywhere:

- Switch, ofc, SteamDeck and ROG Ally
- Untethered PC tablets
- iPads and other tablets

Yet players expect better quality and performance when docked so reactive and dynamic scalability is important there too

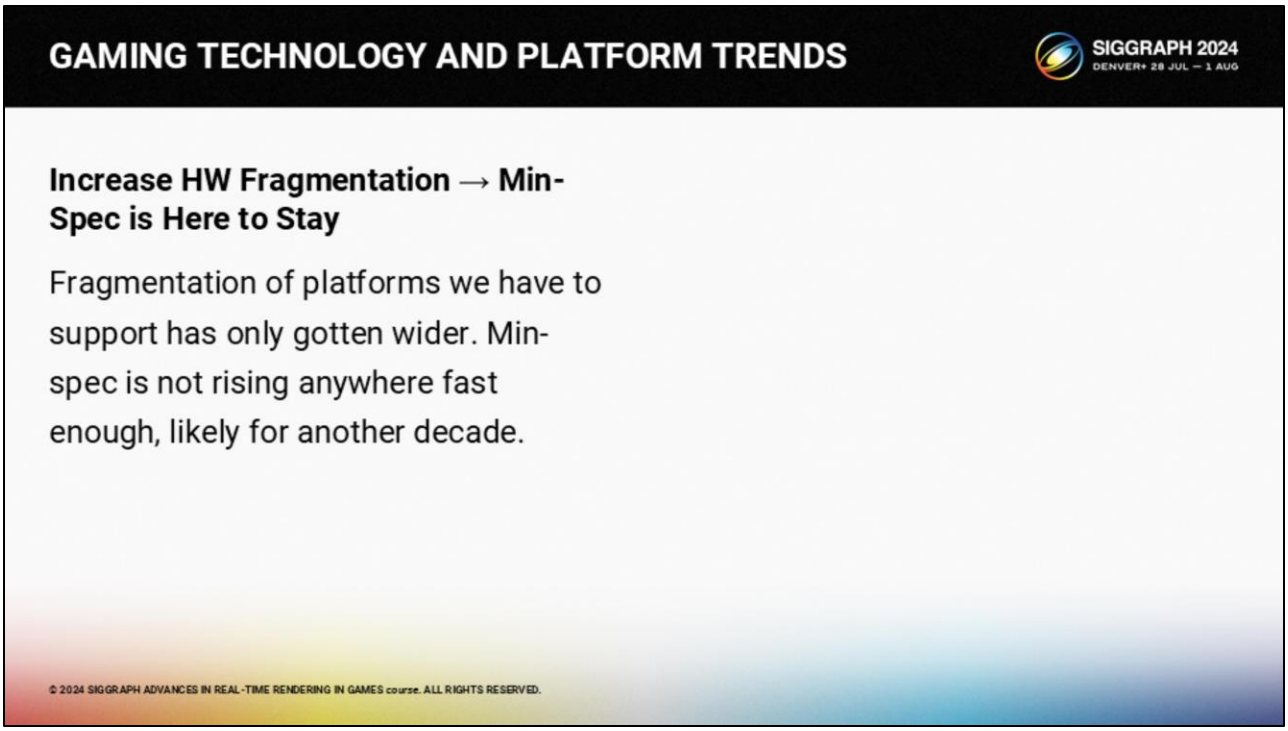
Streaming continues to advance but is a ways away: the promise of playing Call of Duty on a potato is grand (I kid) but tech and infrastructure obstacles continue to plague advancement due to latency and cost considerations.

And – even more importantly – power isn't just a thing you think for mobile devices only. It is becoming a critical consideration on far more form factors, from untethered PCs, to potentially future generations of other hardware, we also see the requirements there in AR to be quite stringent.

This elevates the demand for better API support and tooling - power

consumption and temperature matters for untethered in undocked state and yet we do not have much available. Some major game production companies write their own, but we need IHVs and ISVs to innovate and deliver in this space! But the pressure is not just on the IHVs – I invite all game developers to share the tools they have developed internally for this with the world so that we can foster a better conversation on how we can innovate faster in this space. Submit your talks to next year's Rendering Engine Conference! https://enginearchitecture.org/

Now, let’s shift gears and look at the gaming technology as well as platform trends

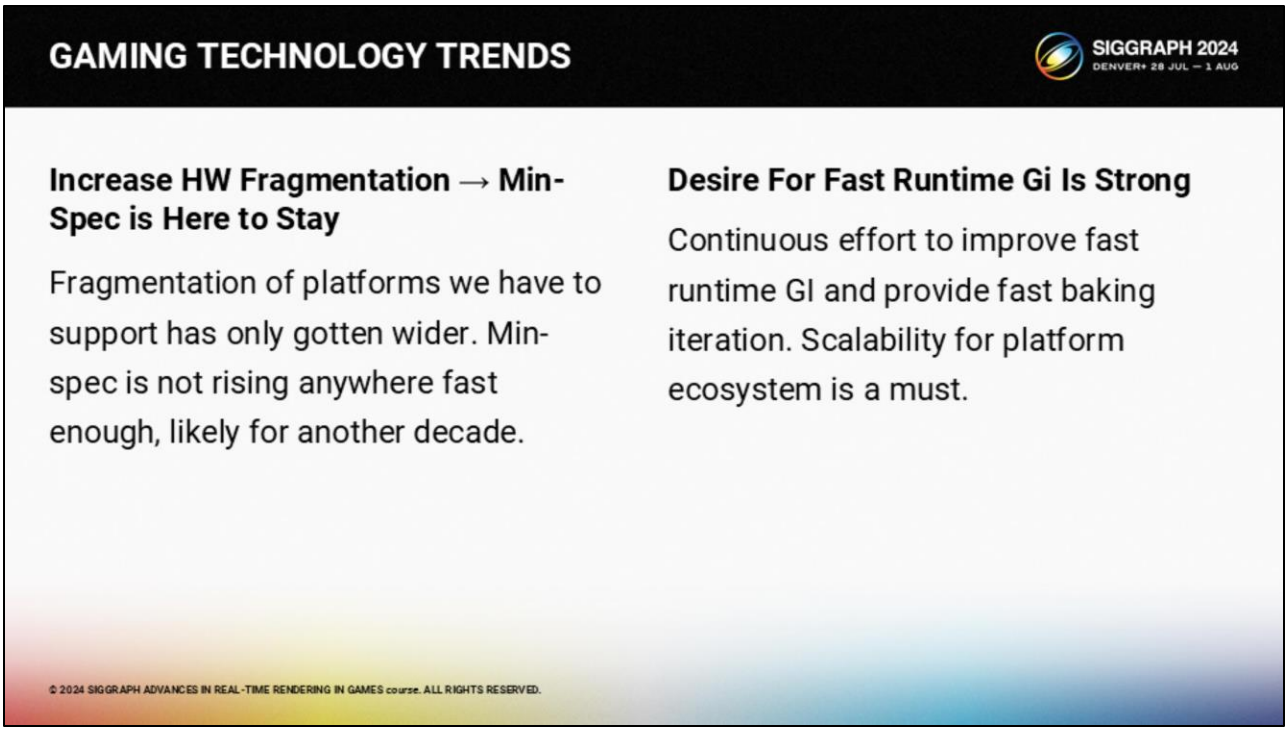


Fragmentation of platforms we have to support has only gotten worse. Hardware divide between console and high-end PC is bigger than it has been in a while injecting more complexity for gaming technology stack. The distance between the high-end PC and low-end PC, or high-end PC and mobile or low-end console is striking and increasing rapidly.

Let me say it a different way - the min-spec is not changing. Min-spec is not rising anywhere fast enough. It's here to stay, likely for another decade. PlayStation 4 and Xbox One released 10 years ago and are still getting new titles. And even when developers will stop shipping on those consoles, they'll have untethered devices [ROG Ally + N years? Next Switch speculations? Mobile / Tablets?] that will be effectively targeting that feature and performance generation.

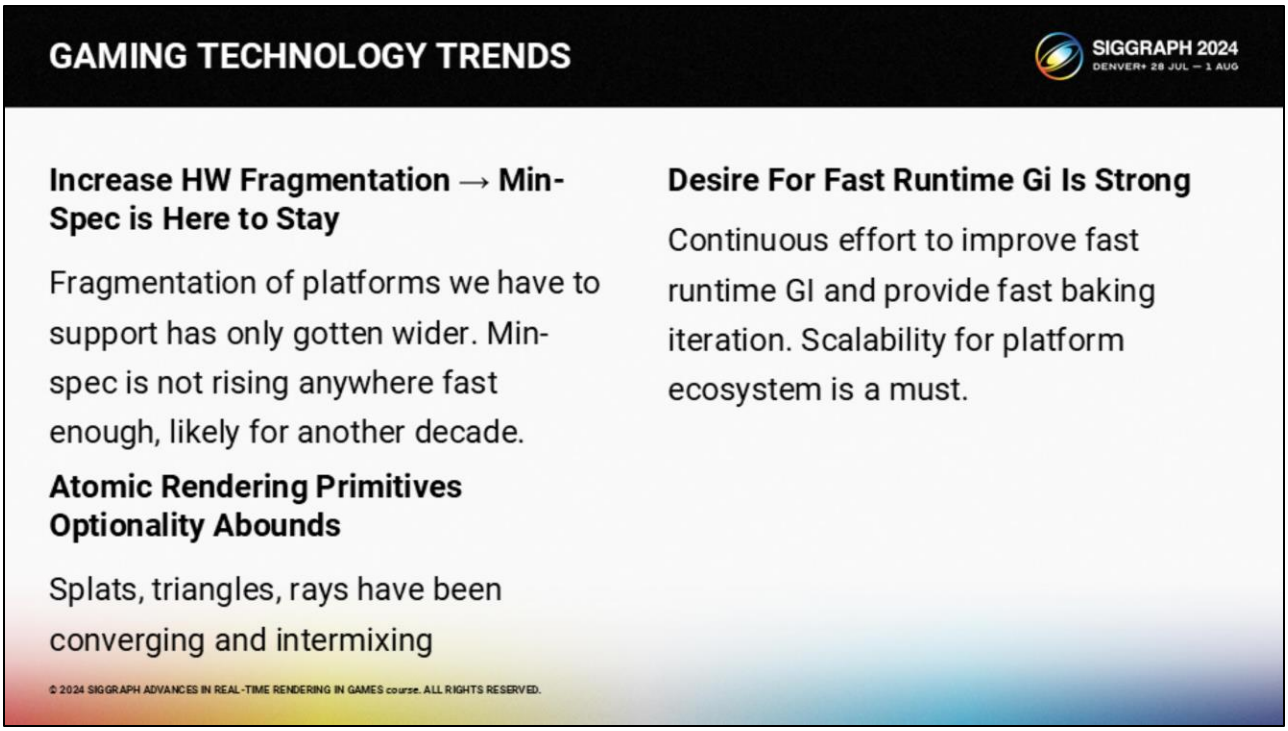
Additionally, while older cards are becoming outdated (like GeForce 1050 which is about a PS4) we have new mobile-ish devices (like Qualcomm notebooks, Windows on ARM) that are right around PlayStation 4 performance and will be relevant for the next 10 years. So while new hardware is continuing the trajectory of improvement, the min spec is just not moving, and we as developers need to figure out how to innovate while supporting existing devices.

And this fragmentation, of course, also makes the criticality of scalability and authoring once more so.

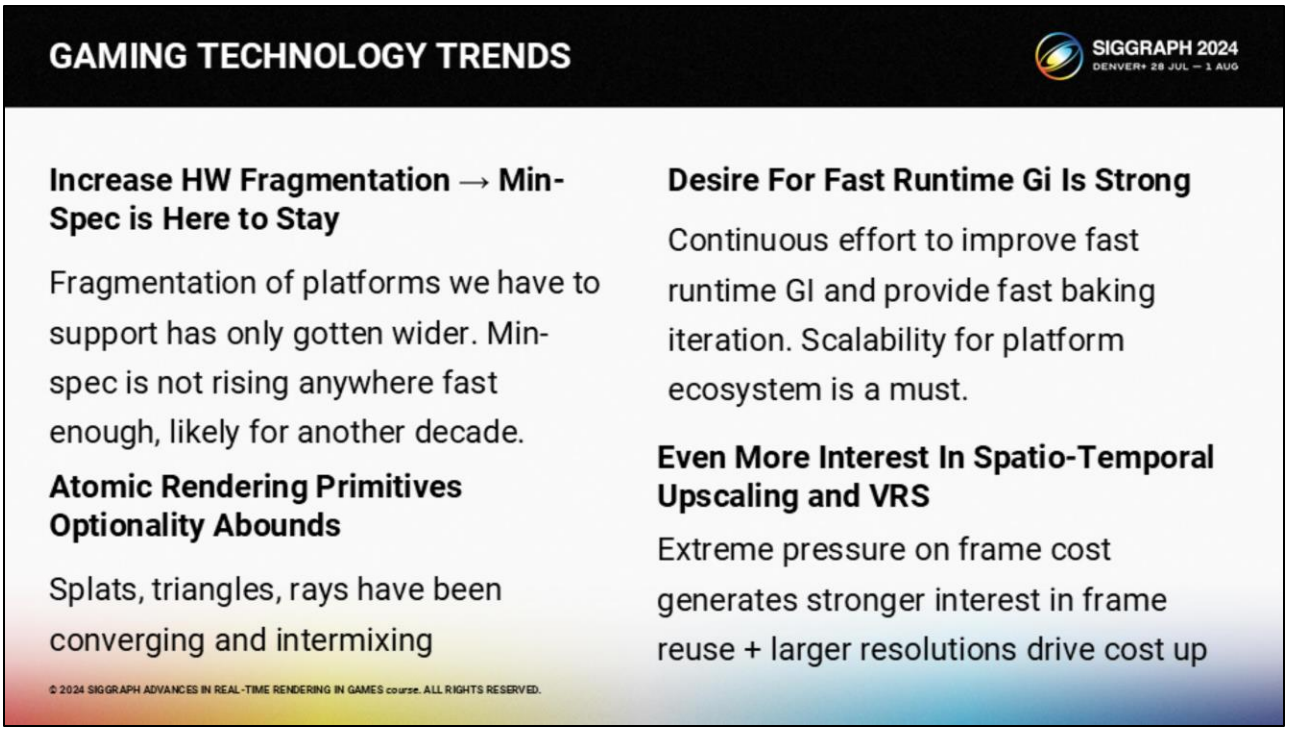


At the same time there is the holy grail (one of the) of real-time CG - the quest for realtime GI: There is continuous effort here to improve current solution (new encoding basis for lighting, use more raytracing), new world representation method (SDF, GPU open AMD bricks etc...) - AND if not for runtime there is a push to get real time preview of lighting in tool for the artists. There will be plenty of cases where there will not be a desire to pay for real-time GI at runtime, but iteration time for baking and memory considerations remain of paramount importance.

You'll see a # of talks related to GI for games in today's course. There is still plenty of room to innovate for solutions here. And, as sizes of the worlds continue to expand, we need to be able to generate GI for larger scenes in a fraction of memory, while maintaining quality for denser scenes.



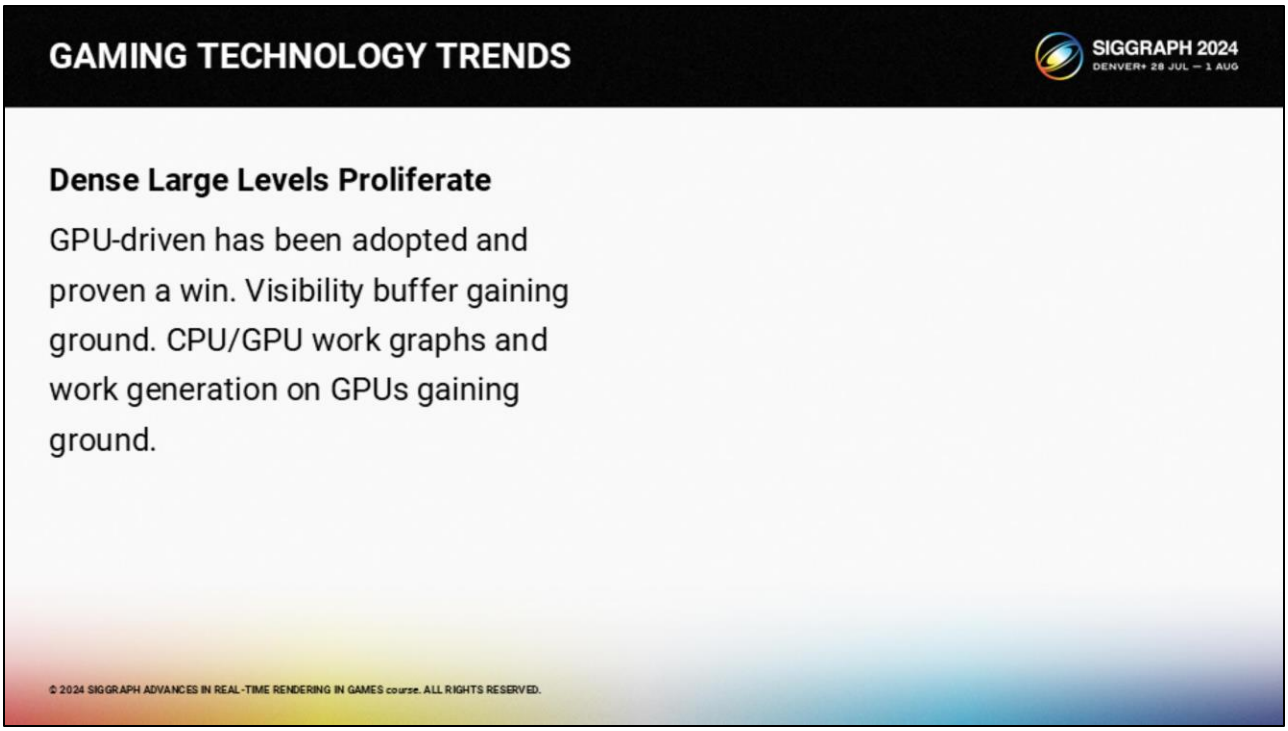
Another interesting observation is that we are starting to become unshackled from the atomic rendering primitive - we don't really care anymore about what is the atomic rendering primitive.
Splats, triangles, rays have been converging and intermixing, and more and more algorithms is interchangeable among them.
It's interesting to find parallels and discover cross reuse. It'll be interesting to see how this impacts our rendering pipeline and even integrates with new ways to generate content from photogrammetry or even AI-driven workflows.



At the same time, we see renewed - even more significant - interest in spatio-temporal upscaling.
Unity's STP (https://mastodon.gamedev.place/@aras/111617623163596044) is a good example of that.

We had this trend for a while, to split computations over multiple frames, then we did more and more upscaling - now we do all this to some extremes, we generate intermediate frames. We're trying to find a balance between proactive rendering and reconstructing pixels on screen [who knew that our 2007 paper on reprojection (https://gfx.cs.princeton.edu/pubs/Nehab_2007_ARS/NehEtAl07.pdf) would have so much legroom?] I'm curious to see where we're at in say 5 years - will we have enough machinery to render say at 720p/30Hz and get a 4k@120Hz with some super smart reconstruction? Are we going to start seeing frame interpolation for real?

VRS techniques also have been yielding great results, even though they often rely on SW VRS implementation, combined with consistent improvements in upscaling. This has been a huge driver for higher frame rates capabilities on a wider platform selection.



We now see that dense large levels proliferate in game worlds. Dense meshes are becoming common place - with photogrammetry and DCC workflows they are incredibly easy to generate. This crossed over AAA, into indies that look incredibly high fidelity (Bodycam in Unreal Engine 5?? https://store.steampowered.com/app/2406770/Bodycam/)

We're seeing GPU-driven has been adopted and proven a win. Starting to see proliferation of high mesh resolution and virtual geometry - we see this in engines support across, as well as on mobile - see the upcoming talk in this Part I course.

We are also seeing new approaches that scale to mobile - see the talk this year.
And there more efforts on CPU/GPU interactions with work graphs and work generation on GPUs.

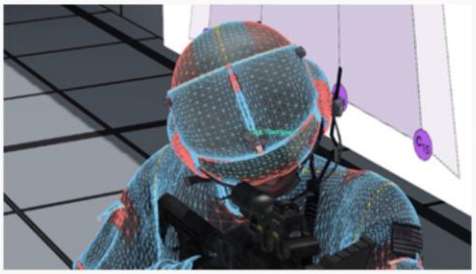
But what that means is that this only increases the importance of compression suitable for runtime as well as build times for these dense geometry preprocessing. And content creators definitely raise the latter issue as a concern. Build times for billion poly levels are many hours, not free. Iteration for content creators will be a key consideration once we have harnessed performance for these high density levels. And, not the least, these extremely dense worlds only make the pressure on footprint that much higher.

GAMING TECHNOLOGY TRENDS

SIGGRAPH 2024
DENVER+ 28 JUL — 1 AUG

Dense Large Levels Proliferate

GPU-driven has been adopted and proven a win. Visibility buffer gaining ground. CPU/GPU work graphs and work generation on GPUs gaining ground.



Brainerd: Tessellation in *Call of Duty: Ghosts* (SIGGRAPH Advances 2014)

While it hasn't become commonplace, the next frontier will be Compute-based GPU tessellation - though I have to shout out to Wade Brainerd's presentation from 2014 on compute shader based tessellation.
https://advances.realtimerendering.com/s2014/index.html

GAMING TECHNOLOGY TRENDS

SIGGRAPH 2024
DENVER+ 28 JUL — 1 AUG

Dense Large Levels Proliferate

GPU-driven has been adopted and proven a win. Visibility buffer gaining ground. CPU/GPU work graphs and work generation on GPUs gaining ground.

YET! Disc footprint is becoming an existential challenge

Sidebar - and as a contrast / additional challenge - Disc space is becoming an existential challenge for AAA - and that's because it's the storage footprint that matters. For players, this may mean that if two huge AAA games get released, they must make a choice of installing only ONE of them on their hard-drive, and thus the fundamental need to drive the storage footprint is a critical quality of life need for our players.

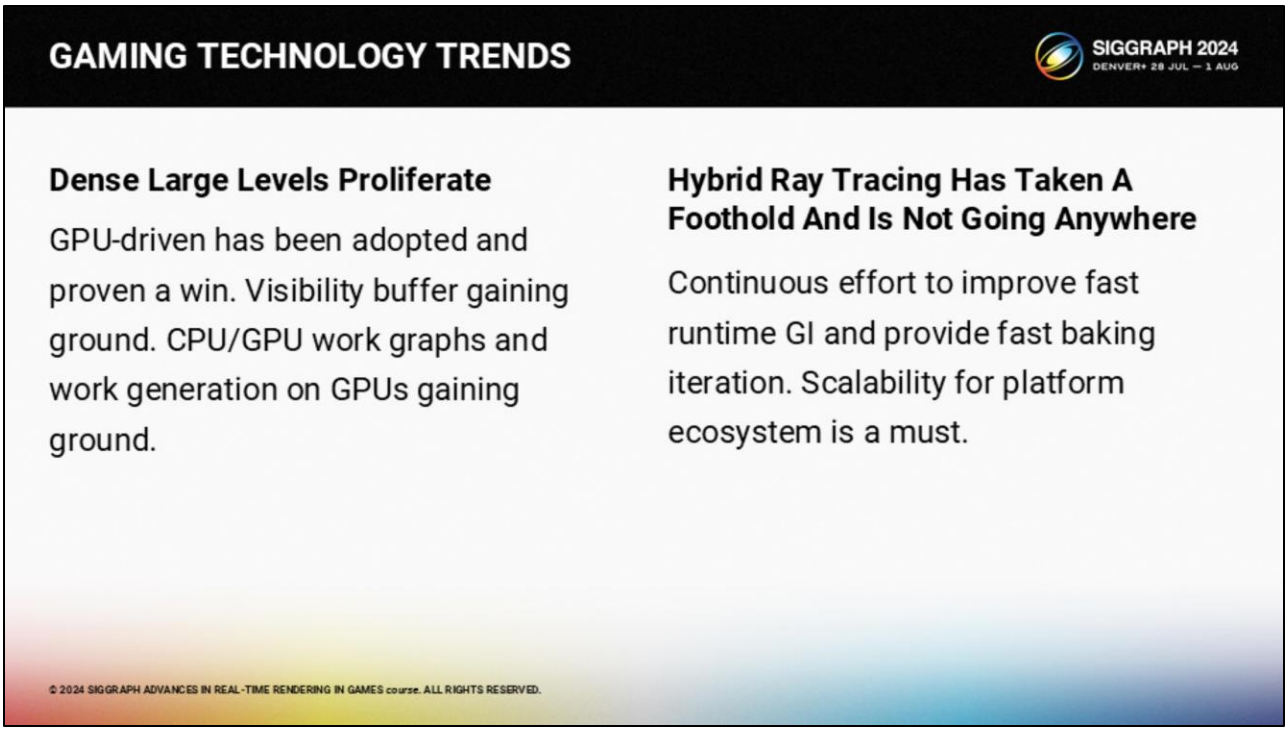
On mobile, we also see a huge cliff in engagement with download size increase per MB.

We are already using, for example, CDN and compression and virtual geometry and whole bunch of other elements - and many are as well. But it's not enough.

payloads and compression, memory in general... memory/storage isn't scaling at same rate as other things. So how do we start thinking differently about approaching solutions in this space?

This is where I wish a lot of the IHV innovation would be occuring for hardware and

software based compression techniques. And given the need to stream in data, we also need to be looking at streaming-friendly algorithms. Michal Iwanicki in his talk on Neural Light Grid will touch on how that was designed to be more streaming friendly than, for example, the global tetrahedral probe data structure Call of Duty used before.



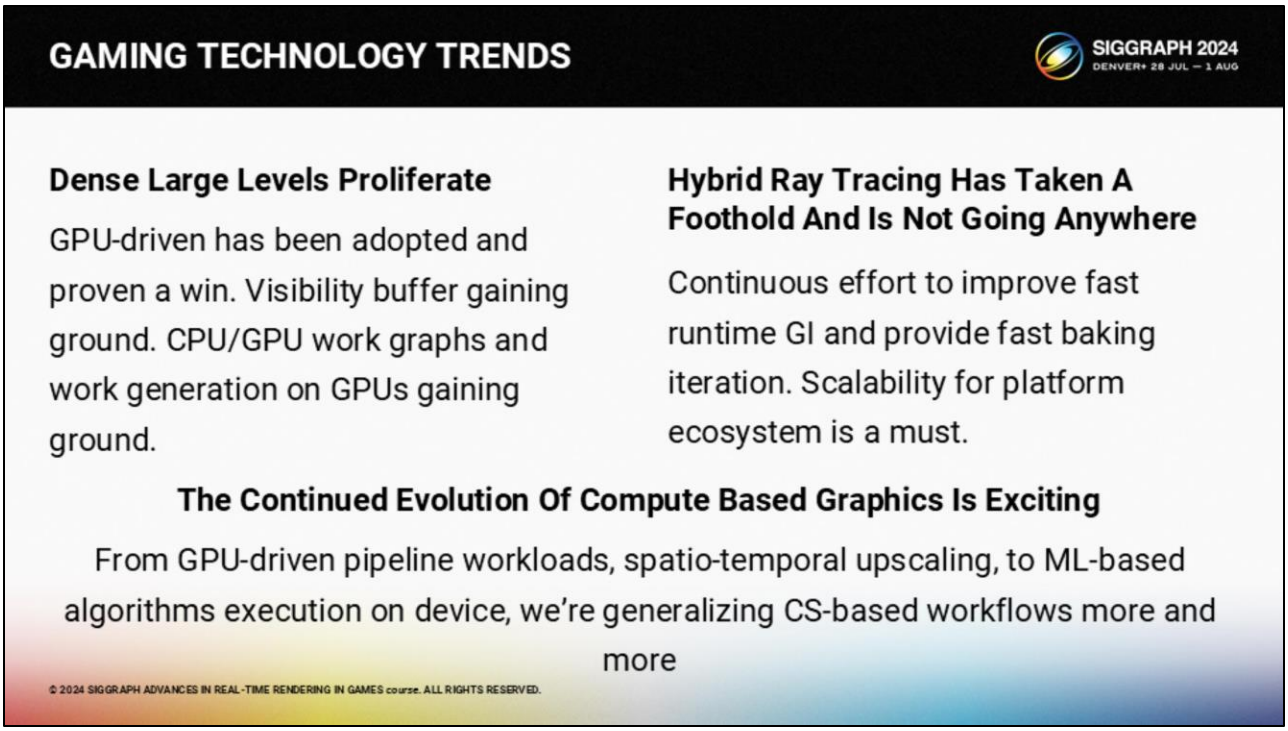
Hybrid ray tracing has taken a foothold and is not going anywhere - in the recent games, we saw impressive recent examples like Alan Wake 2. We can safely say RT will live longer than geometry shaders. :P
There is still a fundamental problem in directions of high geometric complexity versus HW ray tracing which has architecture that isn't quite great yet for high geometric, dynamic complexity - The trend of high density geometry and vast number of shaders still poses challenges for ray tracing

Yet, fully path traced games are still some # years away, I expect that we'll see mostly hybrid uses - with more and more creative uses of the HW - kind of what we're seeing with the compute shaders now. And for ray tracing, in particular with the large gap in the high-end hardware capabilities and features to low-end, introduces enormous complexity for feature support and consistency.
To get there, both for full path tracing and hybrid ray tracing, we still need to solve challenges with

- Dense dynamic deformable geometry
- High divergence in material models
- And on mobile, we need to be considerate of power and memory for ray tracing execution
- And, the largest challenge will be not just performance and support for high-

density meshes but, rather, the real-time denoiser that’s comprehensive, provides suitable quality and performance knobs and easily integrates into the rendering engine

Still the trends are exciting



At the same time, the continued evolution of compute-based graphics is exciting. The graphics pipeline is fairly mature at this point, and the the general compute shaders have become completely embraced and have become a significant part of the graphics frame due to the generality of their interface and ease of use.

From GPU-driven pipeline workloads, spatio-temporal upscaling, to ML-based algorithms execution on device, we're generalizing CS-based workflows more and more - and just like with the spatio-temporal efforts in compute shader, the compute-shader based pipelines have enough performance to yield that we'll see more innovations using this programming model. Ideally, we see further generalization for the CS programming model from the manufacturers, rather than introducing sharded bespoke models for NPUs. :)

And this, along with the programming model for NPUs and ML brings forward the need of evolution of the shading languages which have stagnated over the recent years.

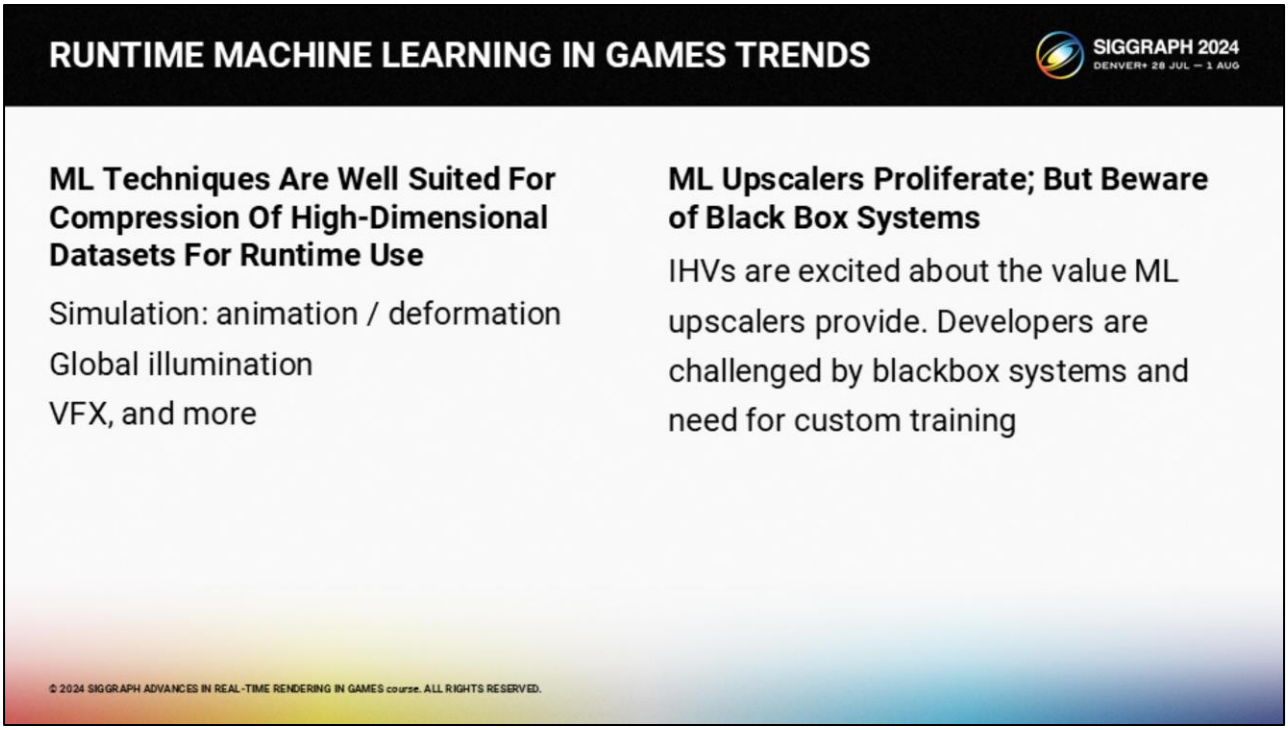
And speaking of machine learning, …

There are many great use cases for machine learning that we can immediately mention as broad categories.

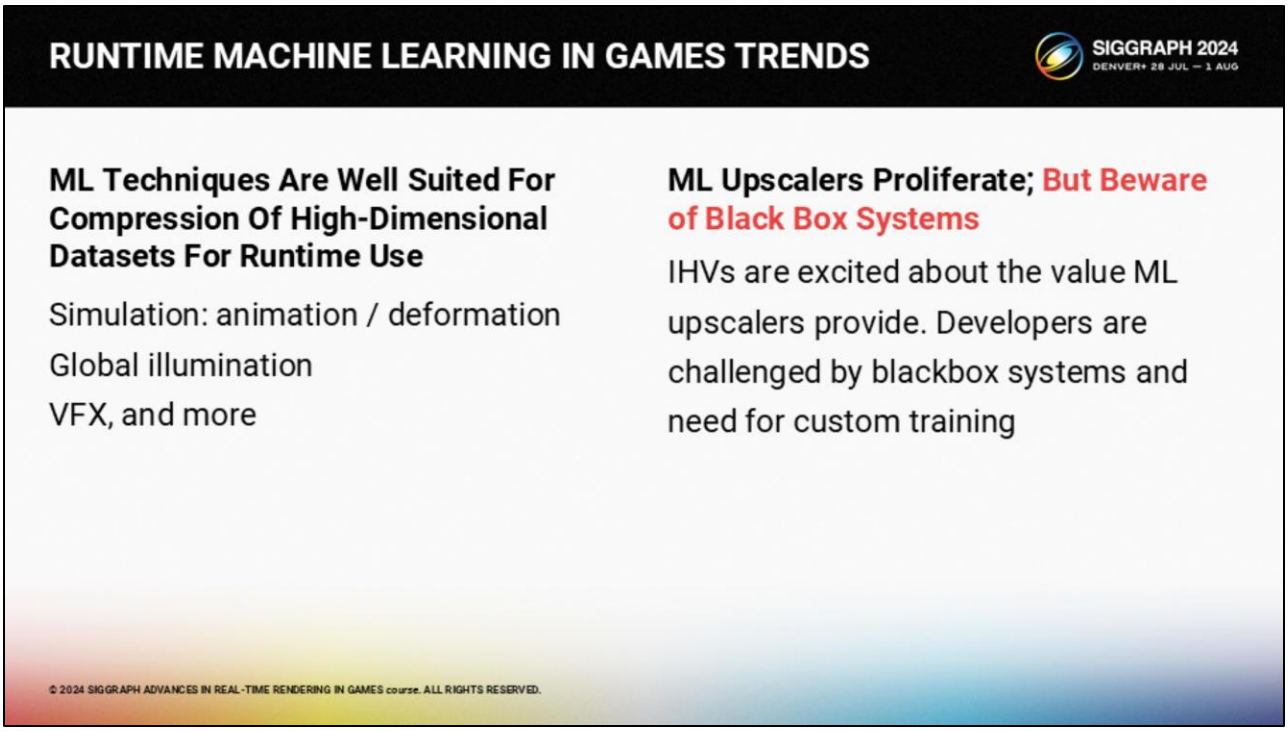
ML techniques are well suited for compression of high-dimensional datasets for runtime use

- Take hundreds of GB data generated by expensive and time-consuming simulation or baking process offline and compress it to a fraction of the cost neural network representation for runtime on-device execution
- Last year we talked at SIGGRAPH about Ziva's ML deformer approach
    - EA is shipping an ML cloth solution
    - Unreal has an ML deformer solution for muscles and clothing and continues to innovate in that space
    - This fall you'll see more high-end AAA games ship with ML deformers as well, even for 60+ fps titles
    - However - the tools to create high fidelity simulation is the bottleneck. Rigging to setup high quality simulation is extraordinarily taxing and much of game industry artists are yet to learn how to do it properly - learning from film colleagues there
    - What are the intermediate leaps without having to pay the full film-quality rigging and simulation setup?

- Lighting is another area suitable for this approach
    - You'll see a talk on that in this course for using neural networks to represent precomputed GI



ML upscalers proliferate - IHVs are excited about the value ML upscalers provide, from DLSS, XeSS and newer entries from AMD.
Also, remember my note about spatio-temporal upscalers - there is still juice in the traditional pipeline [Unity's STP]



But - IHVs are currently excited by the prospect of differentiation by investing into the machine learning solutions specific to their strengths and even when they open source the runtime code, the heart of the training stays black box. They share that the training is done on a variety of titles but the details of the content that's used for the training and the mechanisms for it are bespoke and behind closed doors. Of course, we understand the need for the hardware manufacturers to differentiate.

They also want to plug in the ML upscalers as "blackbox parts" of the rendering pipeline.

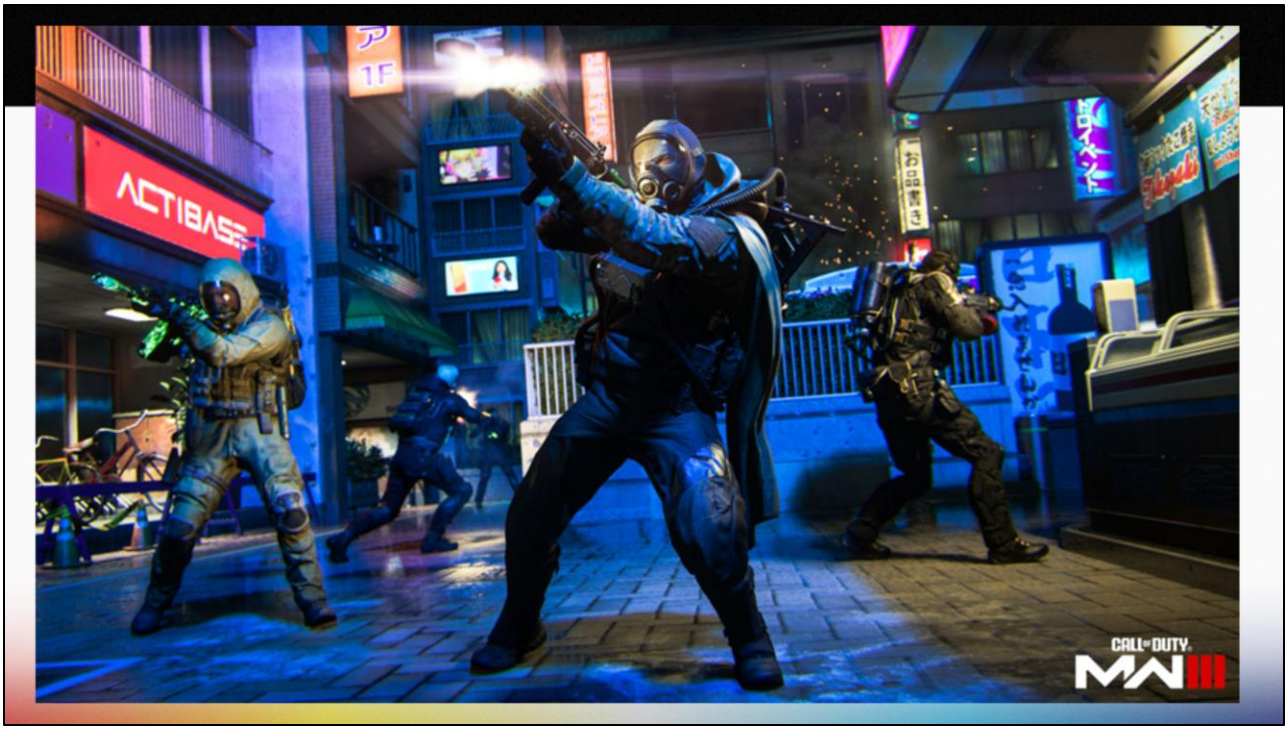
What are the drawbacks of that approach that concern game developers?
For the latter, for example, many engines have quite diverse postprocessing pipelines, as well as various storage / update points for frame buffer information. A blackbox postprocessing module brings a sizable amount of pain on integration and maintenance burden, as it also locks the evolution of the engine's postprocessing pipeline for future changes, once integrated. It can also introduce significant performance issues, when a data element that's not generated otherwise is required.

But that's a smaller issue as compared to the burden of blackbox training.

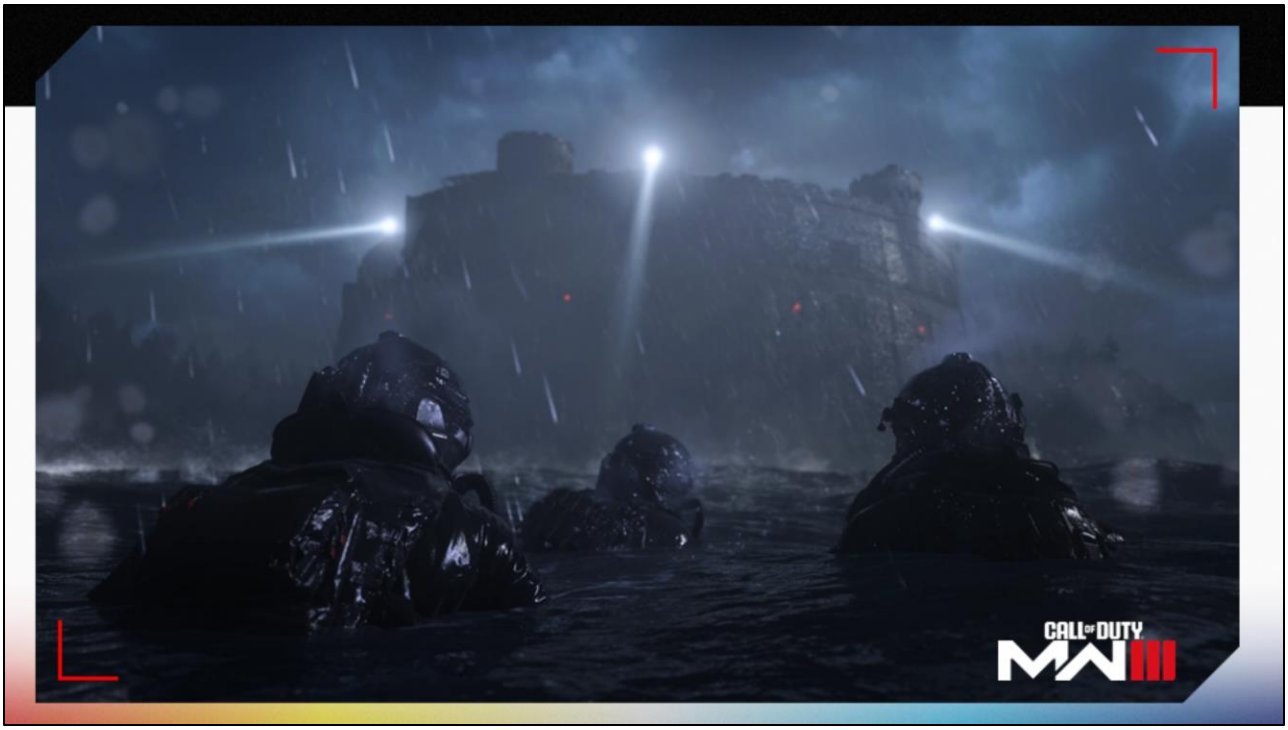
When the manufacturers provide training, they train behind closed doors. They do specify that some key developers can provide datasets, but the provenance of the training data isn't clear, and the training cadence is not specified.

Yet we talked a few minutes ago about the rapid development and delivery for live games. With that also comes a fair amount of experimentation and exploration of art styles. But if the manufacturers train on a set of images for a given game, they make implicit assumptions about the appearance for this game.

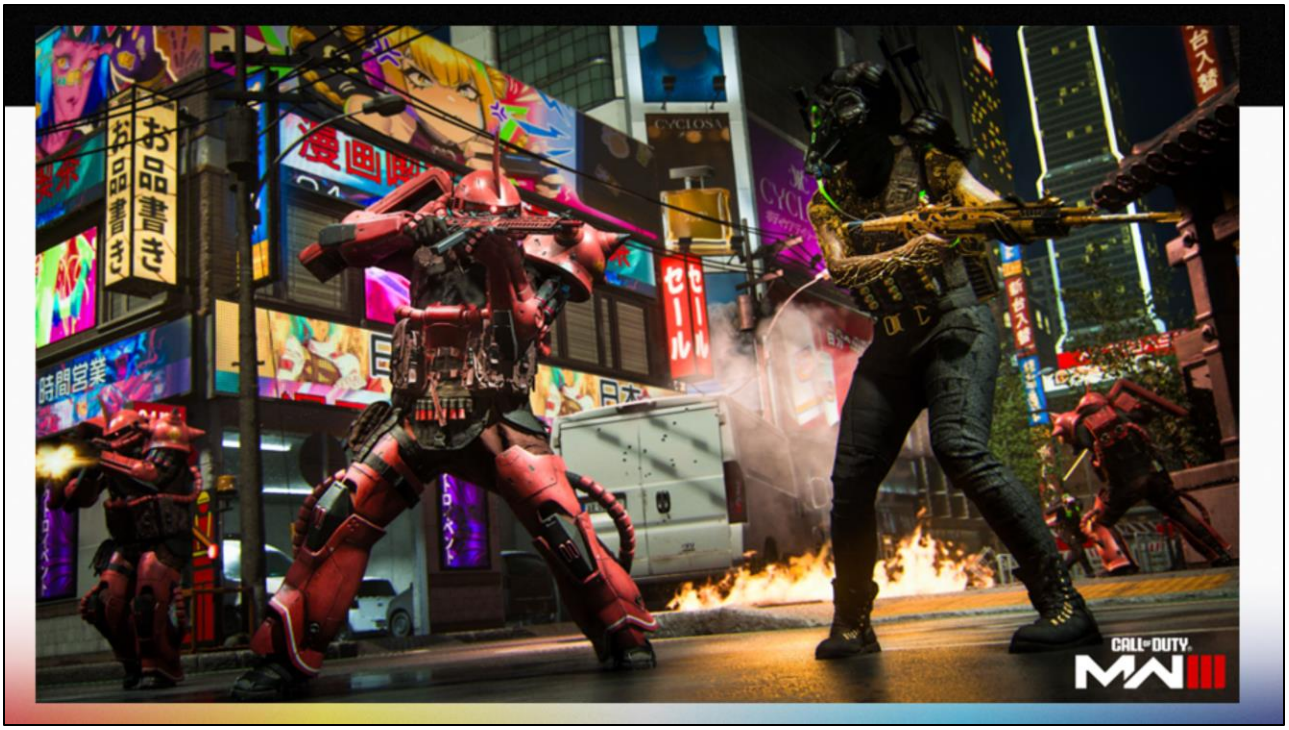
We don’t have to go far to explore this. Let’s take at our game, Call of Duty.



What if the ML upscaler was trained on realistic content like you see from Call of Duty: Modern Warfare III which had a very photorealistic look when shipped, like this frame, for example (which, although from a later season, is representative).

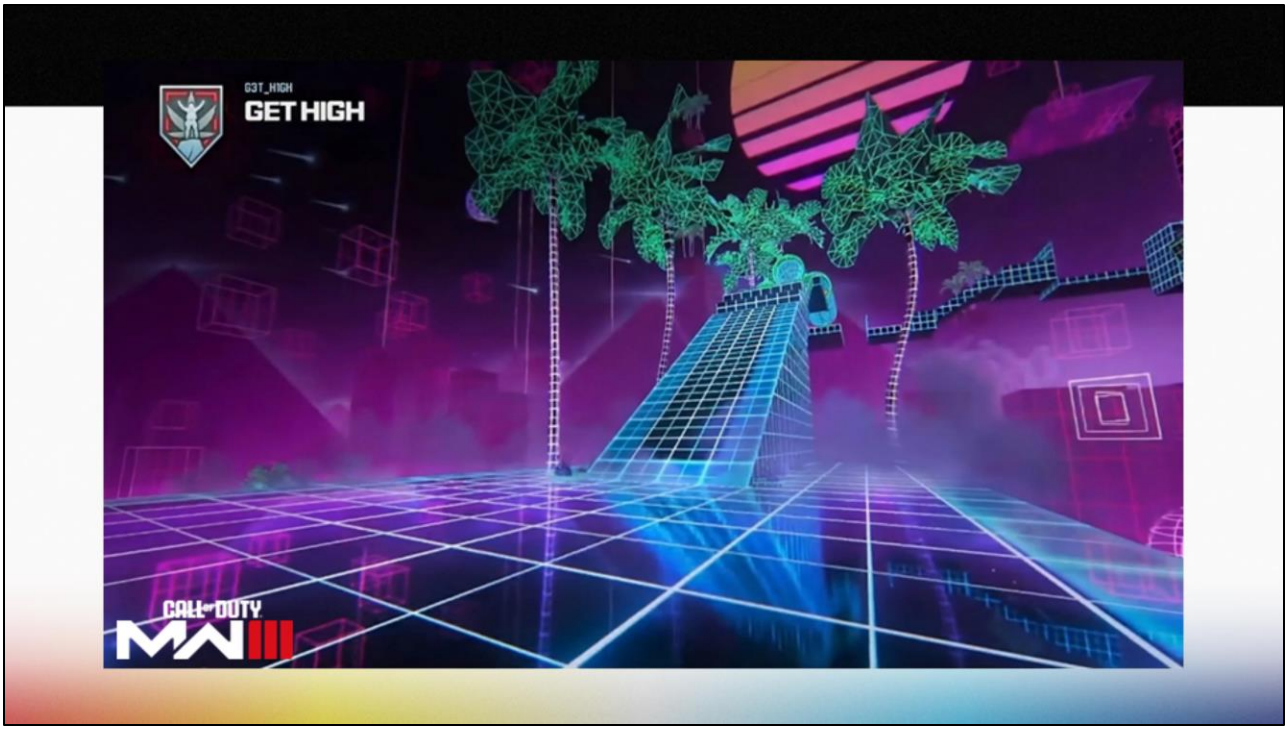


Or like this

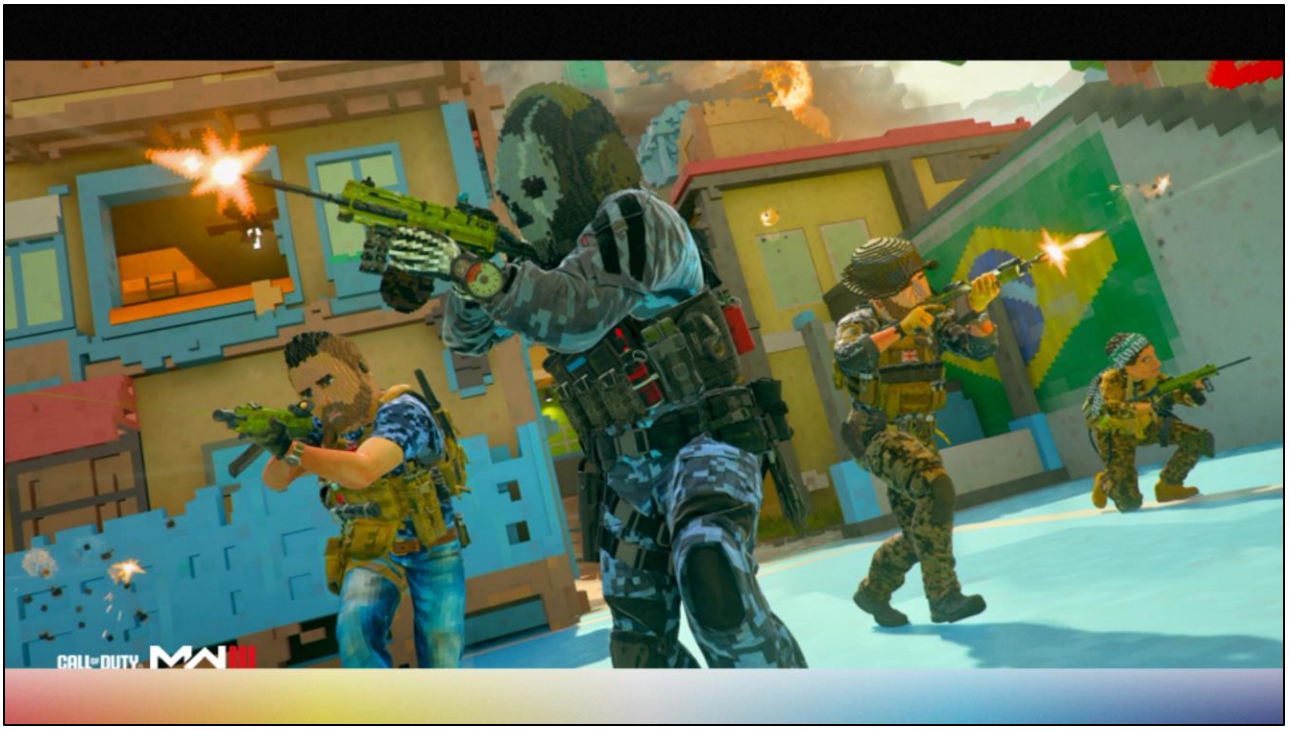


Or this.

But then your designers and artists wanted to offer a cool new mode that’s different to the players?



Or this (a new mode shipped in Call of Duty Season 4) - as you can see - entirely different art style, with a Tron-like level style.



Or this (the Bit Party mode that shipped with Call of Duty MWIII Season 4).

Would the trained blackbox module still work even with these rather divergent art styles? What if we encountered an error, what would be the process to incorporate the new art style in our tight season release schedule?

What's the flow for having 3P authored ML-trained system for the developers? What's the turnaround time? These elements matter and we should be discussing this to arrive at usable solutions together with the IHVs and API providers.

With high level services we are often not provided with the ability to train against our own data set, reducing effectiveness as well as exposing our data to the training infrastructure. While many developers are not able to afford their own training, we believe these costs will come down significantly over time and that the ability to train tailored workloads will be very important to producing best-in-class results. This may be possible with parameter tuning but it will dramatically improve our ability to build solutions if we are able to train our own data for running inference on these devices.

Note that it's important to consider, that, with live service games or frequent

releases, such as our consideration, training must be accounted for not as one-off event at the point of the ship of the original title, but, rather, with every season, new significant batch of MTX content, or even with significant content patches, i.e. aligned to the release models of the title updates. Thus, a critical consideration for any ML techniques is to be able to have the utmost flexibility for rapid training and testing based on the content changes and pressures of the game needs.

RUNTIME MACHINE LEARNING IN GAMES TRENDS

SIGGRAPH 2024
DENVER+ 28 JUL — 1 AUG

ML Techniques Are Well Suited For Compression Of High-Dimensional Datasets For Runtime Use

Simulation: animation / deformation

Global illumination

VFX, and more

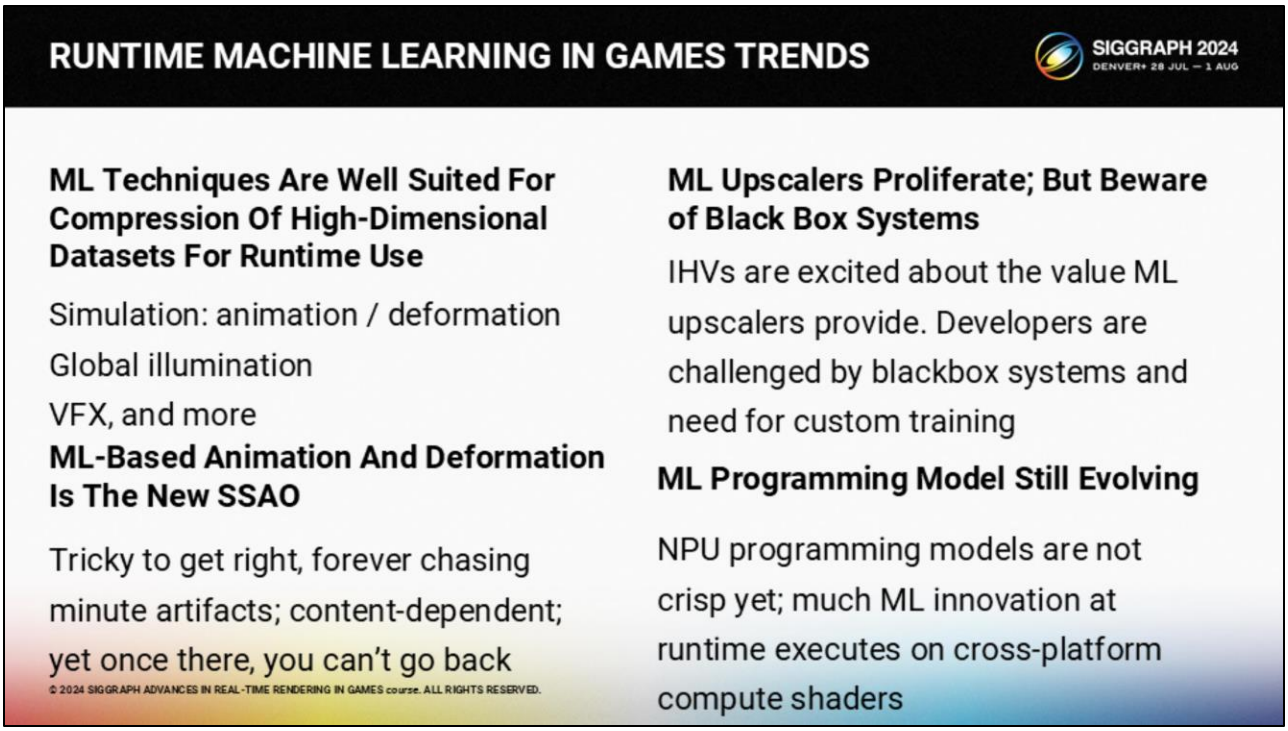
ML-Based Animation And Deformation Is The New SSAO

Tricky to get right, forever chasing minute artifacts; content-dependent; yet once there, you can't go back

ML Upscalers Proliferate; But Beware of Black Box Systems

IHVs are excited about the value ML upscalers provide. Developers are challenged by blackbox systems and need for custom training

ML-based Animation and deformation is the next frontier - it will be the new SSAO
Tricky to get right, forever chasing minute artifacts; content-dependent; yet once there, you can't go back



We're at the start of the ubiquity of ML usage for rendering pipeline - talks like Michal's will be fundamental at unlocking further thinking in how to leverage this space. We will also see the shakeout between programming model (GPU? More generallized compute shaders? NPU with separate sharded memory? UMA for GPU and NPU and CPU access? To be defined)
What's the programming model? ML on GPU using compute shaders for modern engines - convenient, cross platform, and uses the same resources (COD compute shaders, etc. Unity's sentis,)
NPU programming models are not crisp yet; much ML innovation at runtime executes on cross-platform compute shaders.

We think that ISA support for matrix multiplication and accumulate support for tensor data, in varying levels of precision, is an acceptable abstraction. Moreover, leveraging fixed functionality for work such as the sparsity of the underlying tensor data represents a useful analog to compressed textures and fixed function filtering. We also recommend a focus on efficient memory models including a push towards unified memory architectures rather than building expectations of manual memory movement for discrete parts.

Additionally, scheduling independent command streams between discrete processors creates non-trivial programming complexity with regards to

resource synchronization and fencing. Witness difficulties with managing DMA engines and interleaved I/O decompression. Thus, retaining most of this functionality on the GPU fundamentally simplifies the programming model on the CPU as well.

And of course, when thinking about NPU programming model versus GPU ML execution, having stable, reliable, familiar debugging and performance analysis tools is such a huge plus. Just looking at the evolution of PIX on Windows, I hope that we can continue to take advantage of what the current tools already offer when thinking about new programming units.

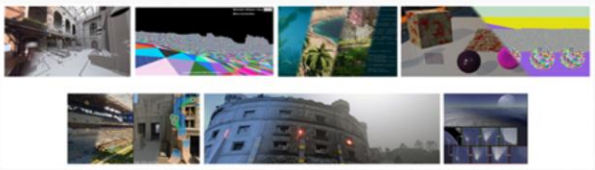
Last, we ask IHVs to deeply consider the problems explored in the last twenty years with GPU shader compilers and analysis tools, and not risk resetting that progress in the NPU space.

So as you can see, there is a lot going on. And with that - with these trends in mind - let's turn to our talks today.

PART I

SIGGRAPH 2024
DENVER+ 28 JUL – 1 AUG

Welcome and Introduction to Part I
Natalya Tatarchuk (Activision)



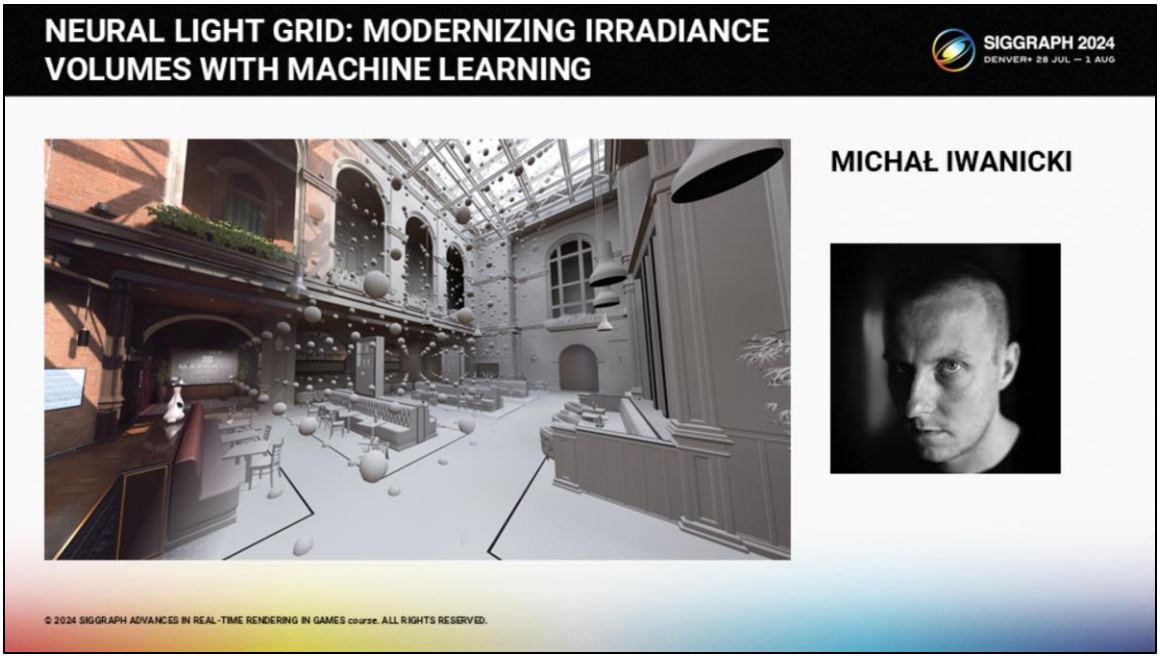
Neural Light Grid: Modernizing Irradiance Volumes with Machine Learning
Michał Iwanicki (Activision)

Seamless Rendering on Mobile: The Magic of Adaptive LOD Pipeline
Shun Cao (Tencent Games)

Flexible and Extensible Shader Authoring in Frostbite with Serac
Simon Taylor (EA | Frostbite)

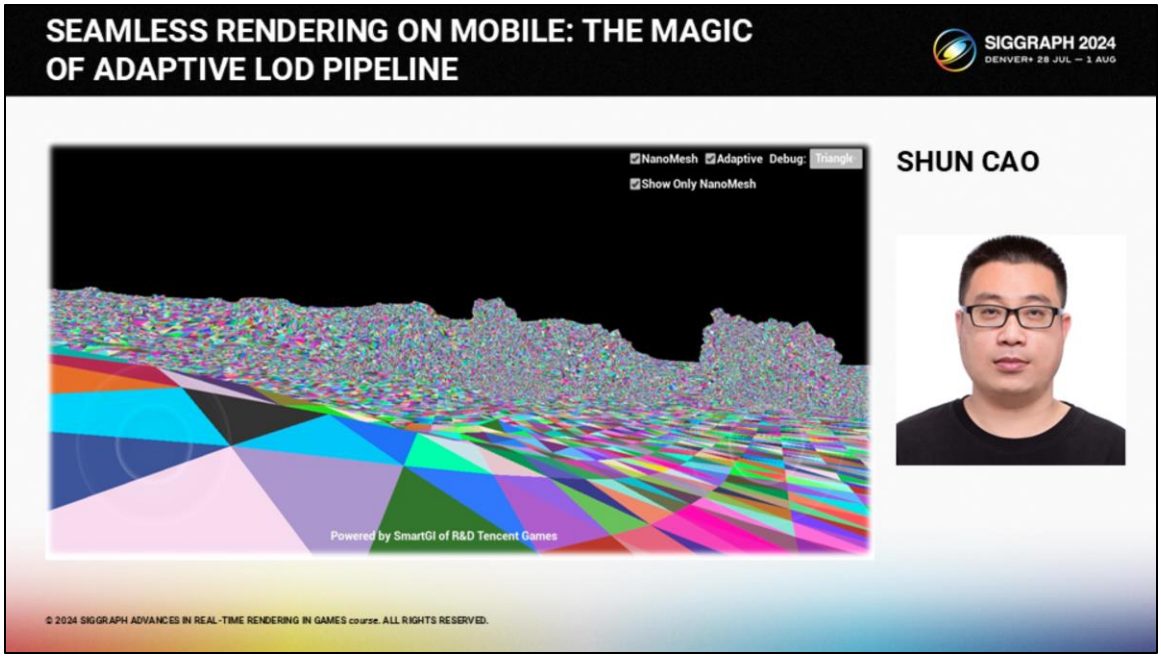
Closing Notes for Part I
Natalya Tatarchuk (Activision)

The first part of the course – the morning section will start from..



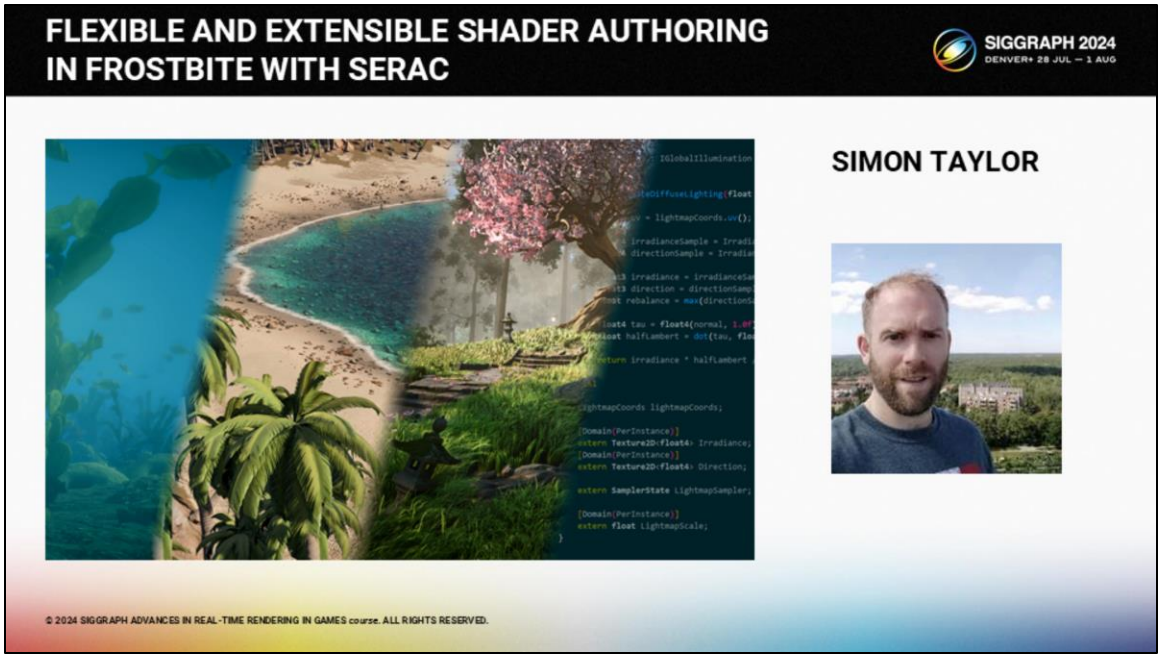
A talk by Michal Iwanicki where he will explore how modern machine learning techniques can address artifacts that often plague irradiance volumes though they have been used in video games for years maintaining acceptable performance on constrained platforms, ultimately leading to a solution implemented in millions of copies of Call of Duty: Warzone and Modern Warfare 3.

https://advances.realtimerendering.com/s2024/index.html#neural_light_grid



In the next talk in Part I, Shun Cao from Tencent Games will introduce a seamless and adaptive rendering solution, featuring mobile-friendly lightweight LOD generation with low IO overhead and high-performance rendering on mobile platforms.

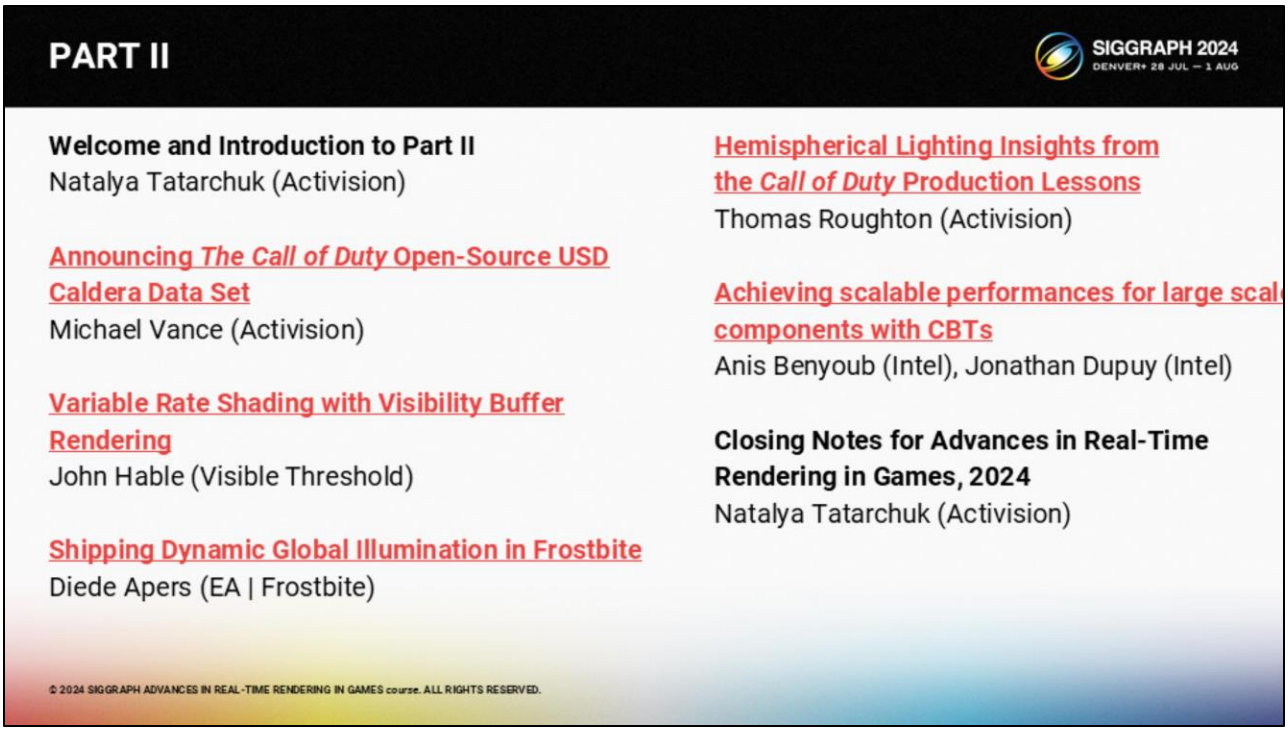
https://advances.realtimerendering.com/s2024/index.html#tencent



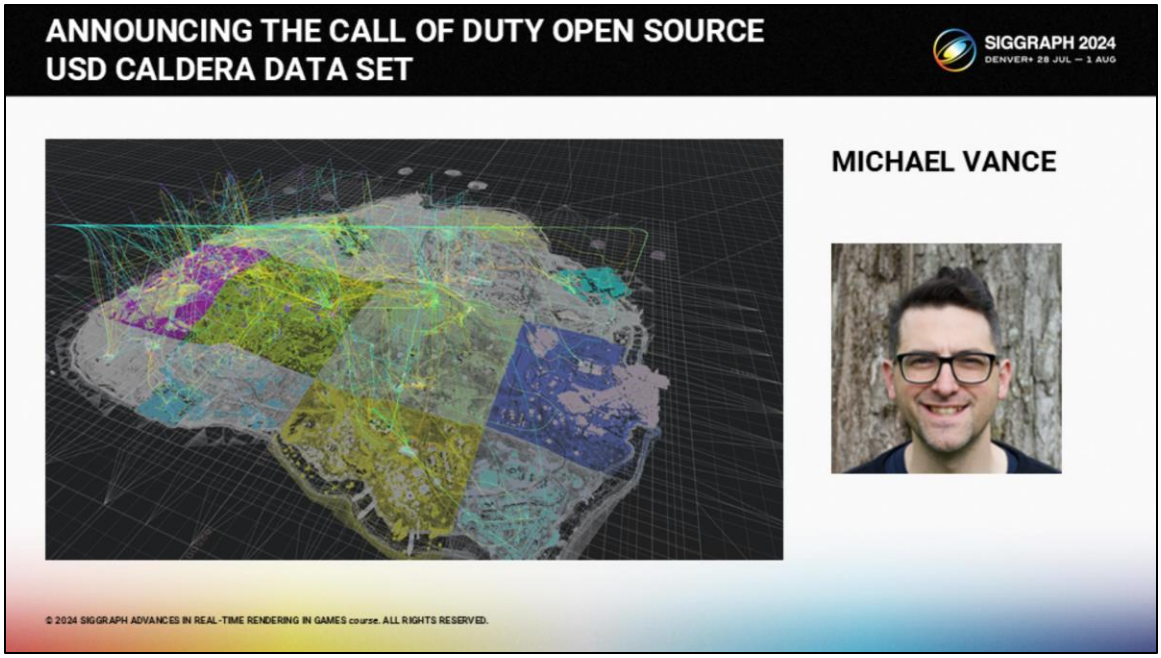
In the next topic we'll talk about shader languages, with Simon sharing the lessons learned in Frostbite, which, like other large game engines, faces challenges with shader code due to the trade-off between performance and flexibility in modern shading languages. To address this, the team developed Serac, a domain-specific language wrapper around HLSL that enhances shader authoring workflows without sacrificing performance; this talk will cover Serac's implementation, integration, and lessons learned from its deployment across the engine and titles.

That will conclude Part I and you all will be free to grab lunch and discuss what you learned.. :)

https://advances.realtimerendering.com/s2024/index.html#serac



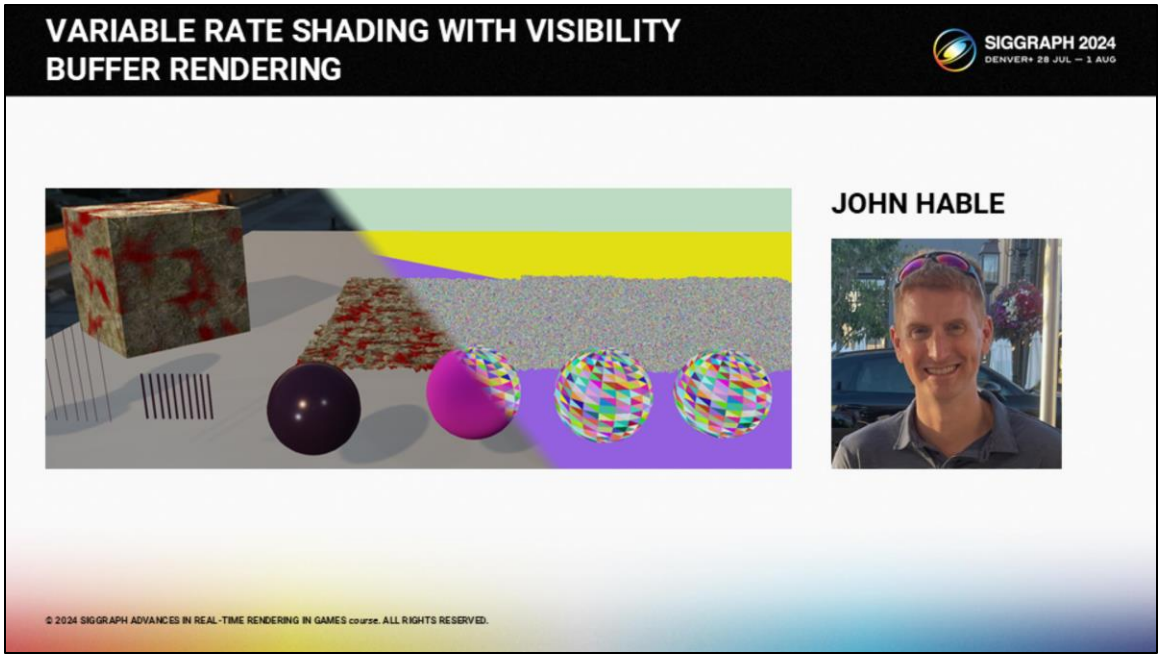
And then at 2pm mountain time come back for Advances in Real-Time Rendering in Games Part II for a very packed slate of talks



After my welcome, we have a surprise for you - Michael Vance will share an announcement introducing the Call of Duty Caldera Open Source USD Data Set, featuring production assets from Call of Duty: Warzone, licensed for academic research and non-commercial use. This geometrically complex data set in OpenUSD format aims to inspire innovative solutions to challenges in environmental geometry and we're excited to share it with the academia and industry for the first time ever!

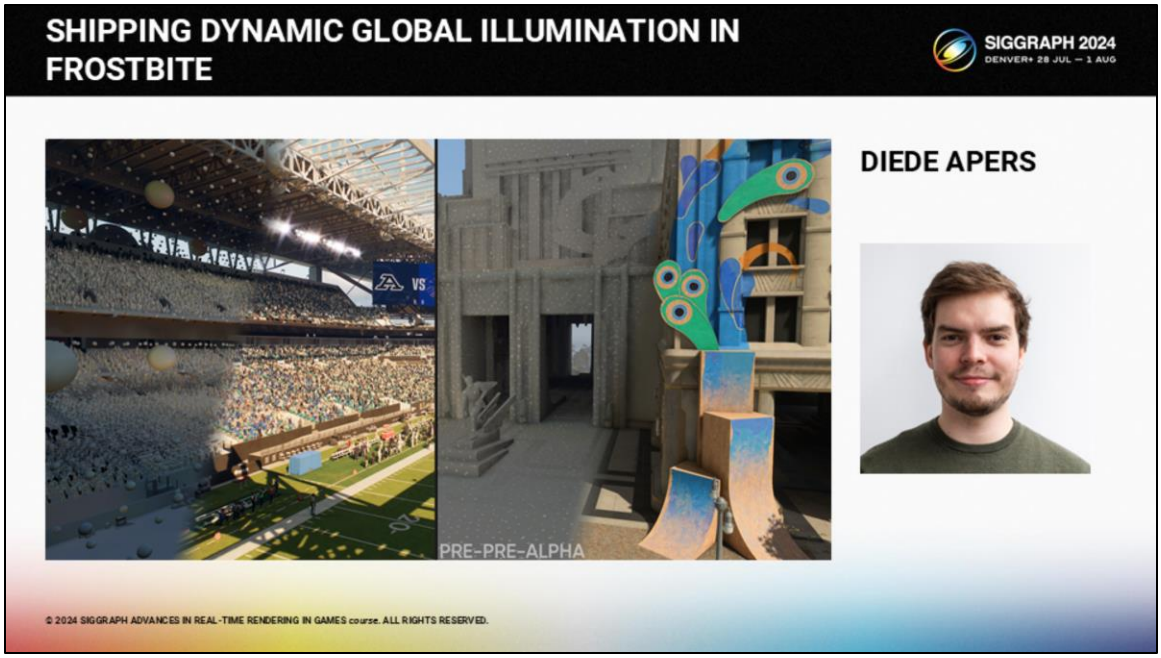
https://advances.realtimerendering.com/s2024/index.html#caldera

You can also read about it at *Activision Releases Call of Duty®: Warzone Caldera Data Set for Academic Use blog, and* ***Download the Call of Duty Data Set Now***



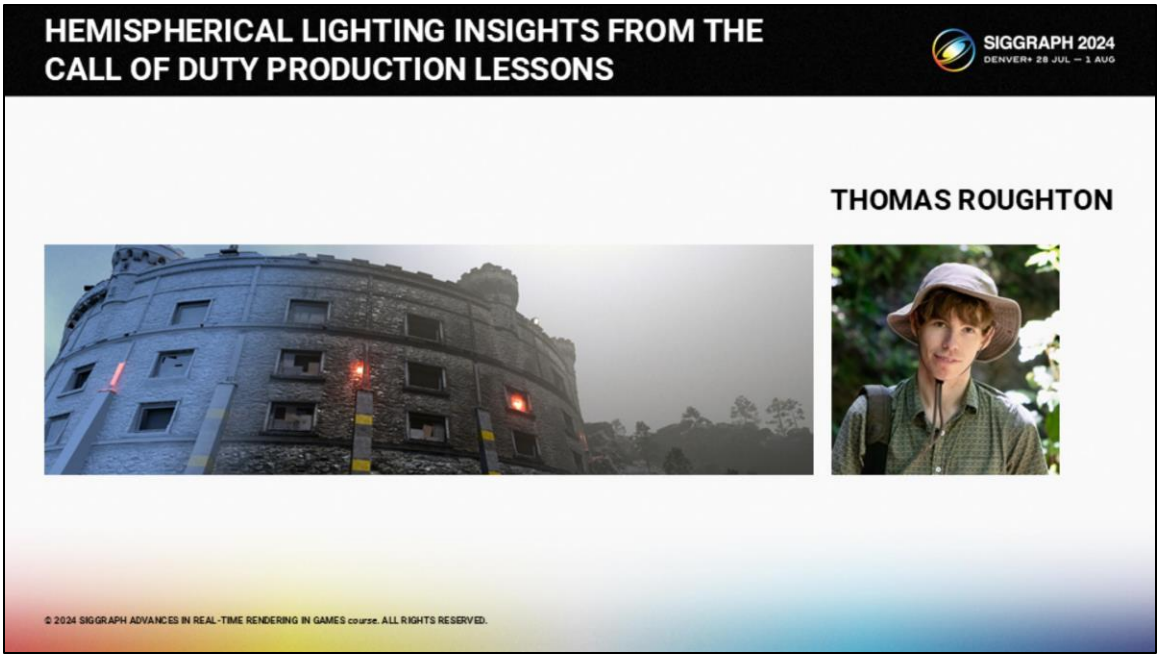
Michael will be followed by John Hable from Visible Threshold, who will talk about visibility Buffer rendering which offers a unique alternative to GBuffer and Forward rendering, particularly in decoupling shading rate from native rendering resolution. This talk will explore a method to reduce the number of shaded pixels while preserving visual fidelity, along with the associated advantages and disadvantages.

https://advances.realtimerendering.com/s2024/index.html#hable



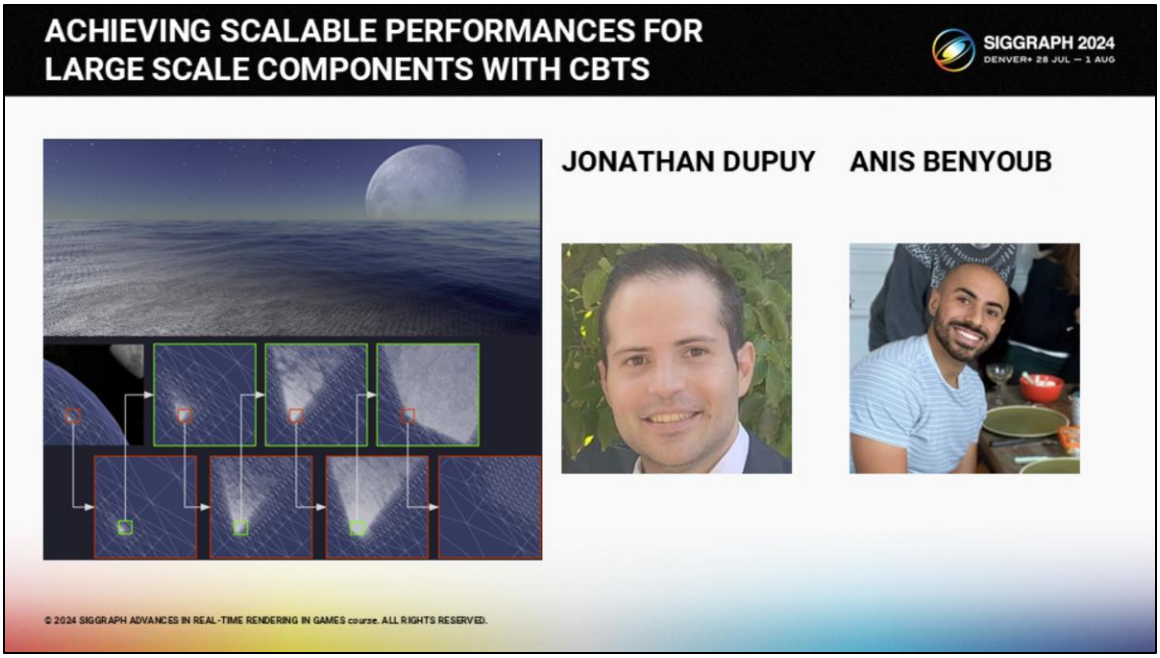
Diede Apers from EA Frotstbite team then will talk about the evolution of the GIBS dynamic global illumination system, which was first presented in Advances in 2021, and has since evolve to become production-validated in shipping Frostbite games. This talk will share more details about that process, and discuss several practical learnings, improvements, and optimizations that were crucial to achieve truly dynamic global illumination at 60fps on current generation consoles.

https://advances.realtimerendering.com/s2024/index.html#gibs2



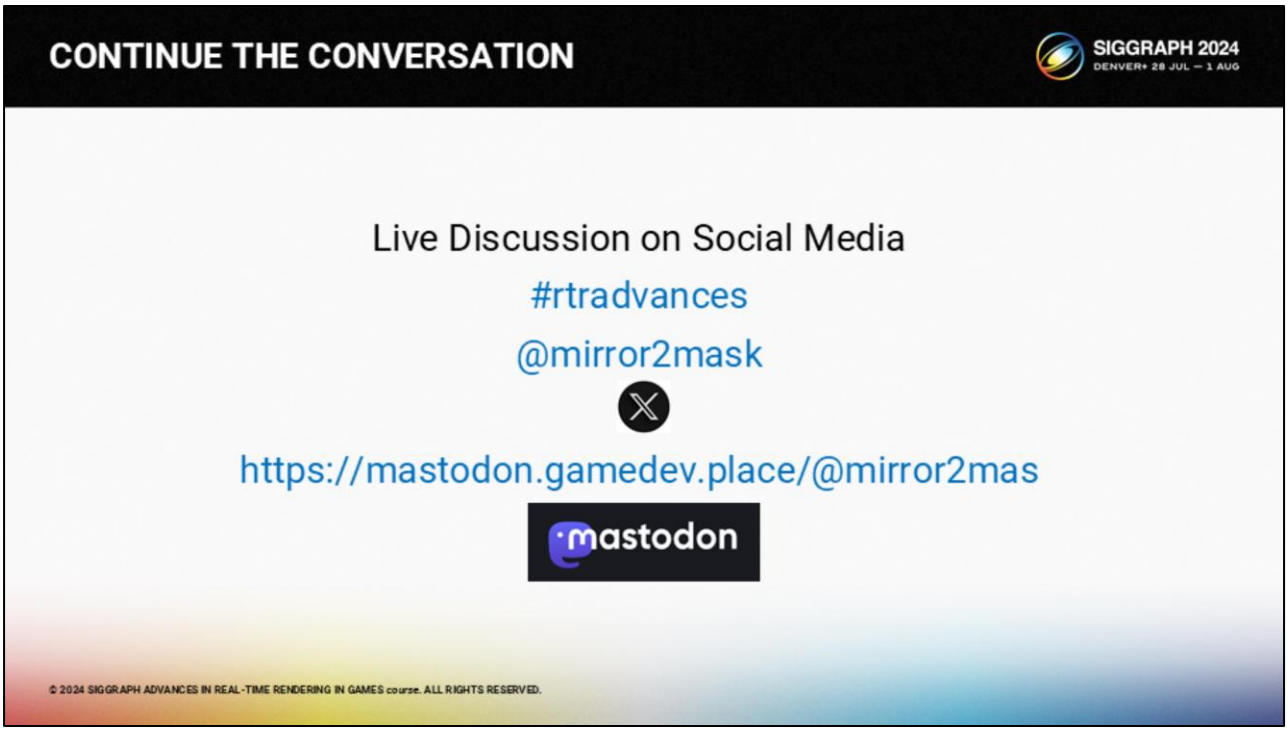
We will continue the theme of global illumination topics with a talk from Thomas Roughton from Activision who will discuss the use of precomputed lighting in Activision's Call of Duty to achieve high visual fidelity and smooth framerates, focusing on hemispherical occlusion and the efficient computation from visibility cones to enhance runtime AO.

https://advances.realtimerendering.com/s2024/index.html#roughton

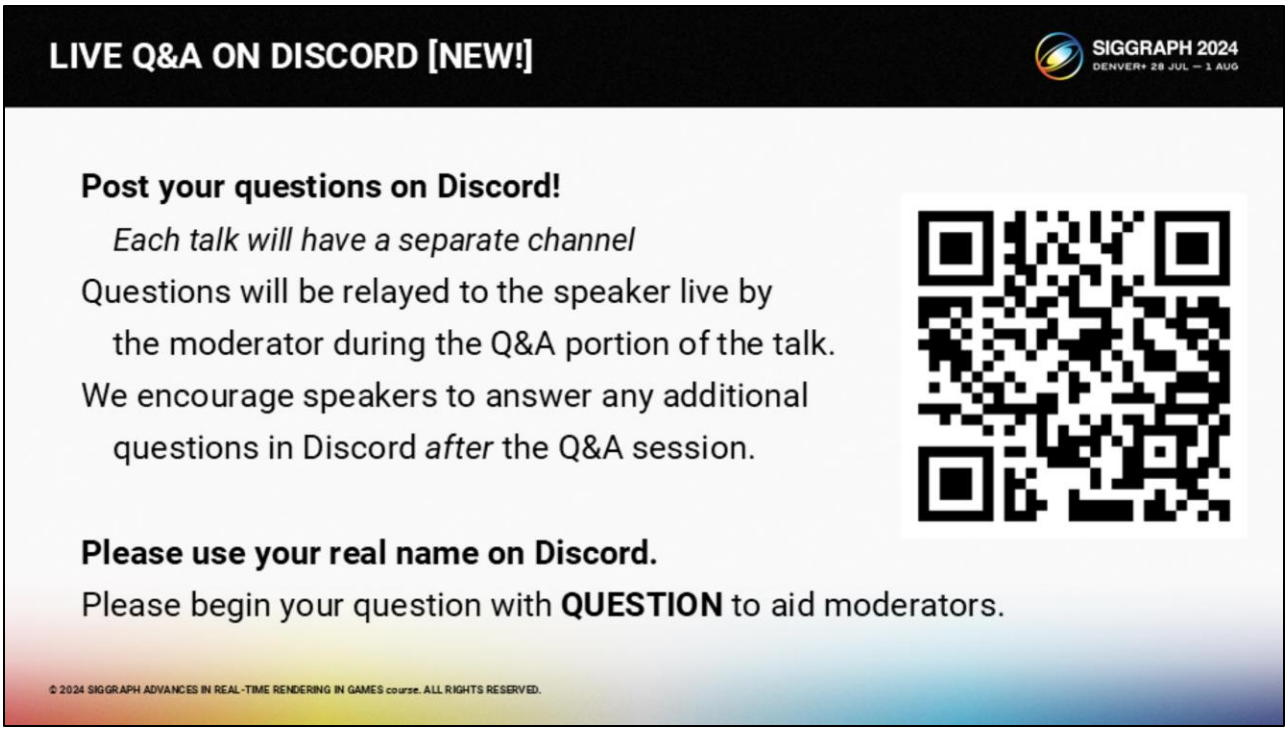


In the last technical talk for the day, Anis Benyoub and Jonathan Dupuy from Intel Corporation will explore the benefits of the concurrent binary tree (CBT) for GPU-friendly terrain tessellations, focusing on its application to arbitrary polygon meshes and overcoming limitations of low subdivision levels.

https://advances.realtimerendering.com/s2024/index.html#cbt2



We'd love to know what you thought about the course and the talks in each session! Share them on social media or via the evaluations. Please share your thoughts using these tags



And we have something new this year to share with you - we have a Discord server for Advances [thank you, Michael Vance, for setting that up]! For each talk, please post your questions on Discord throughout the talk - questions will be relayed to the speaker live by the moderator, and after the course.
Please use your real name on Discord.
Please begin your question with QUESTION to aid moderators.

https://discord.gg/Ns2K8arfHA - the other benefit is that you'll be able to access the discussion even well past the conference.

CONTINUE THE CONVERSATION: CONTENT

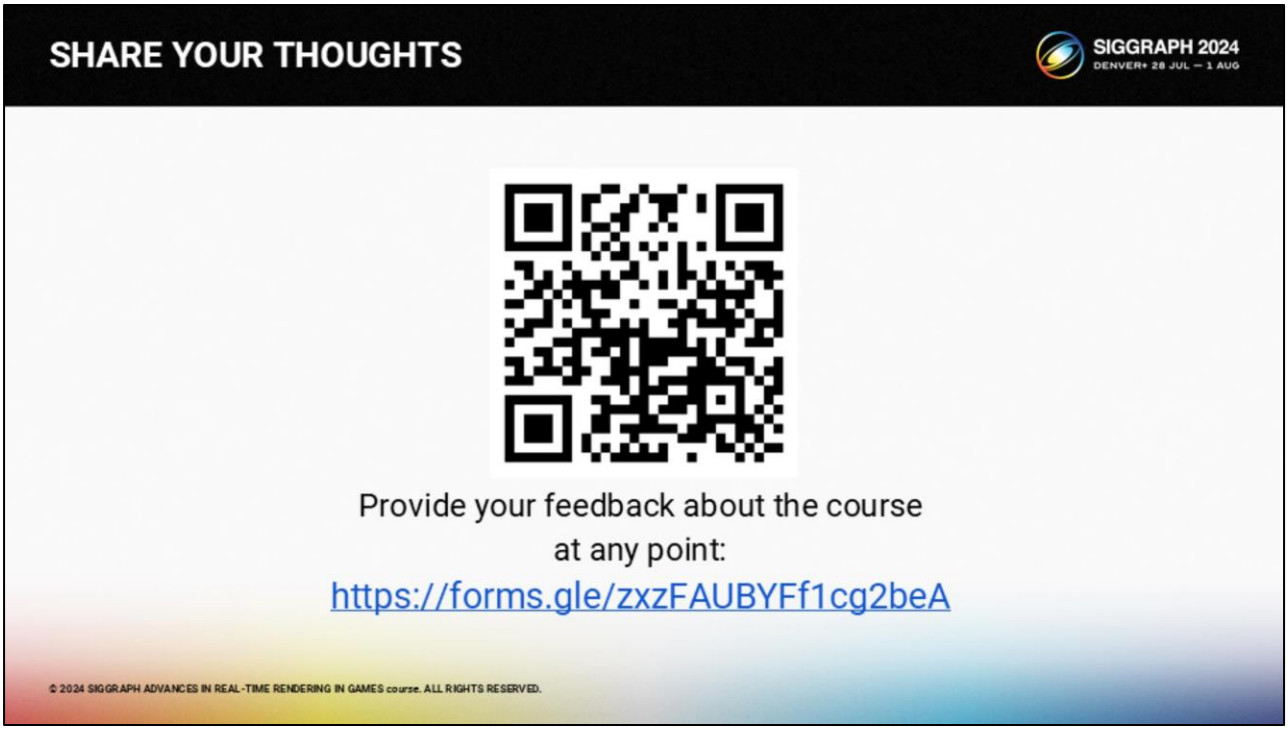
SIGGRAPH 2024
DENVER+ 28 JUL — 1 AUG

Slides and videos will be posted on our website and YouTube channel as soon as possible.

Course Web Page and Slides
http://advances.realtimerendering.com

Advances YouTube Channel

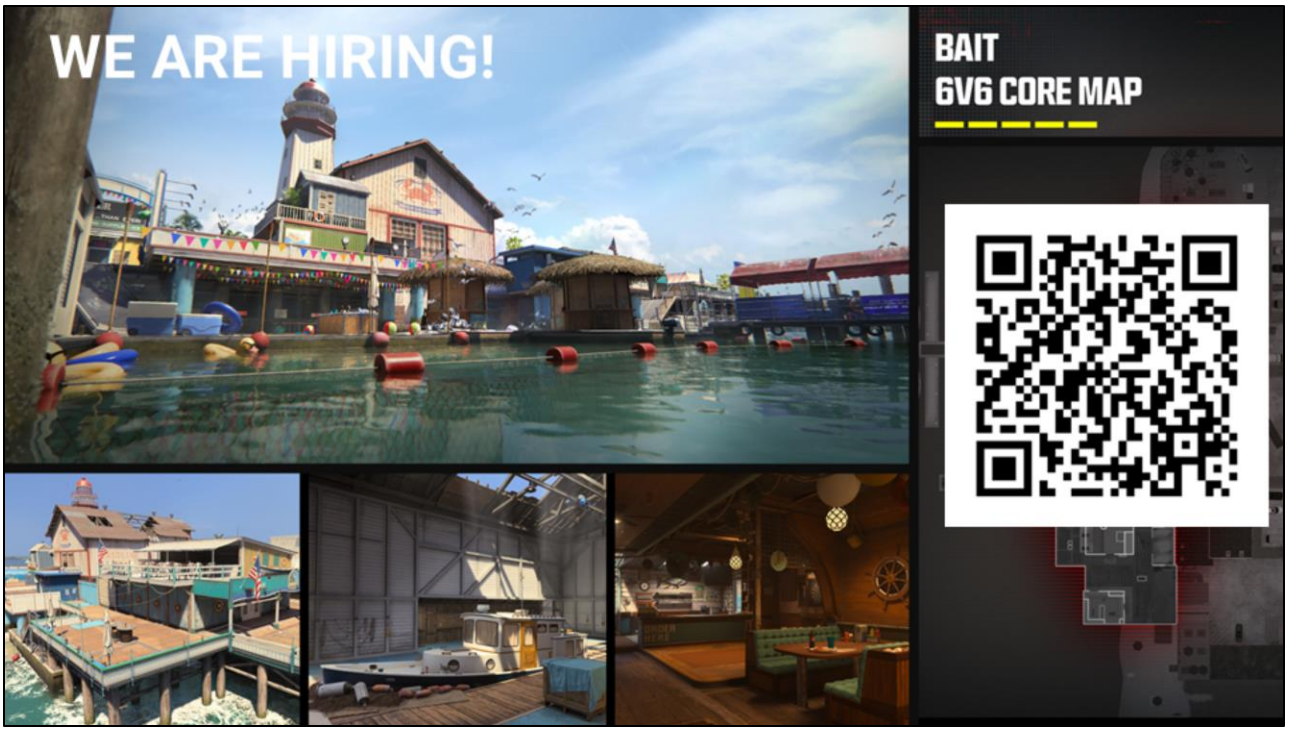
We have a packed program today. But don’t worry - after the conference, we will post all the content on Advances website - QR code on the left. And we are hoping to receive the videos down the line from SIGGRAPH and post them on the Advances YouTube as well.



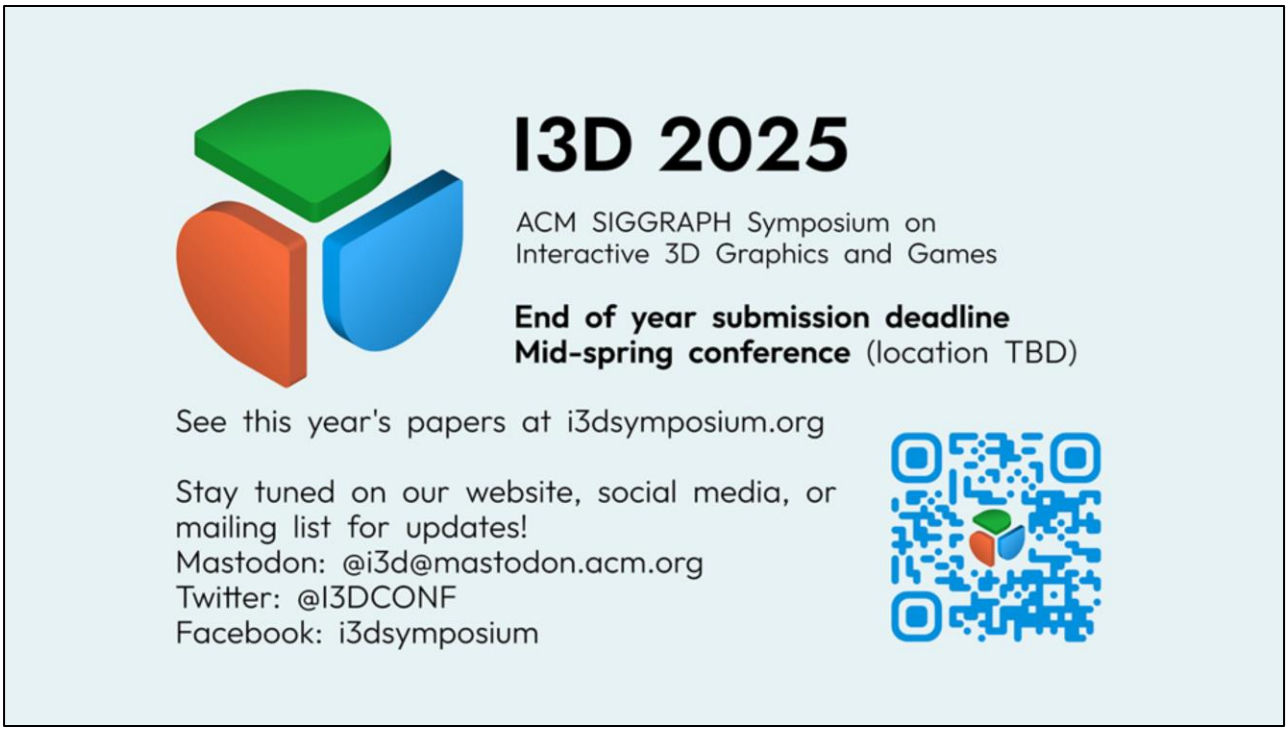
And this year we have an evaluation form that you can fill out at any point - please do. Your thoughts about the course and how it could evolve in the future, what works, what could be fortified help shape it over the years - please let us know!

Also this is year I want to extend my gratitude to mentors to our speakers who helped with shaping of several of the talks in the course.

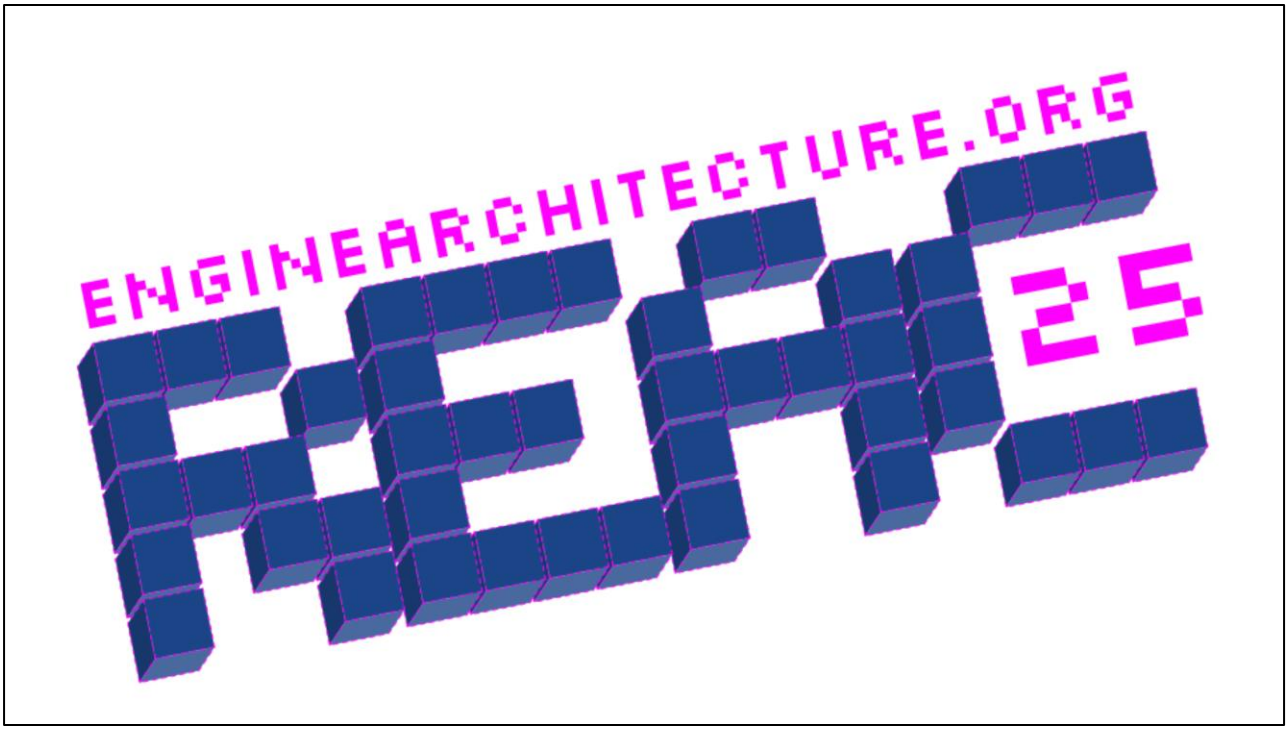
The trends that I shared today were strengthened by the discussion w/ these fine folks who generously shared a great deal of their thoughtful ideas with me on these.



And, not the least – if you're interested in joining Activision, we are hiring – reach out!



And I also wanted to mention that ACM SIGGRAPH I3D, a conference dedicated to real-time rendering, is going to have its call for submission soon - keep it in mind and submit your work there, especially if you are in games! And I3D reviewers - bring people from the games industry to the programming committee, so that we can have a better cross-polination of relevant peer review perspectives!



I also want to mention the virtual conference on Rendering Engine architecture that myself and Michael Vance from Activision along with Steve McAuley from Sony Santa Monica and Angelo Pesce from Roblox have been organizing for a few years - we will be doing it in the 2025 year. So please start thinking about talks you'd want to submit or reach out to us for ideas you are having. The goal for that conference is to build a better conversation about the pipeline and architecture for practical real-time engines for games and other similar projects. Stay tuned for more information but you also can take a look at the conference from 2024 by going to enginearchitecture.org

ACKNOWLEDGEMENTS

- Andres Burbano, Ruth West and Leona Caffey
- Eric Haines and Tomas Akenine-Möller
- Michael Vance

Last but not least. This course was supported by Leona Caffey and Andres Burbano (this year's chair) and Ruth West (this year's courses' chair) from the 2024 SIGGRAPH organizing committee and I'm grateful for their support and help.

I also want to thank, Eric Haines and Thomas Akenine-Moller for hosting the advances site for almost two decades and Michael Vance for his help with the Advances Discord server organization.

Many many thanks!

So, without further ado – let's get into the talks!