Imagination

SHINING A LIGHT ON RAY TRACING

Shining a light on ray tracing

For anyone who knows anything about real-time 3D graphics, something truly exciting is taking place in the industry right now: the proliferation of real-time ray tracing. Often described as the 'holy grail' of computer graphics, ray tracing is where a 3D scene is generated using a technique that mimics how light behaves in the real world, thus providing developers with the tools to make incredibly realistic visuals. In 2016, Imagination introduced a board featuring the world's first dedicated ray tracing accelerator that, for the first time, delivered enough performance for the technology to be used practically, in real time. In 2018, NVIDIA released hardware for the desktop PC market that supported its own version of the technology that it dubbed, 'RTX'.

Before we go on, it's worth understanding why ray tracing is so critical. Ray tracing provides developers with the tools to determine how a scene is structured, delivering a straightforward way to determine inter-object relationships in 3D. An example is reflections. Picture a scene in a game with cars and puddles on the ground with a fire raging, but not visible on the screen. With rasterisation, that fire would not be reflected in the windows and puddles, but with ray tracing the reflections of the offscreen fire would be realistically presented on screen. It's not just to make prettier images – it can also have a fundamental impact on gameplay. Imagine an enemy creeping around an area with windows with the player hiding around the corner. With ray tracing, the player would be able to see the enemy's reflection in the windows, something that would take a lot of manual work with rasterisation.

The natural look ray tracing brings is why it has been used for years by major movie studios in the creation of animated movies and effects. The shiny reflections on Lightning McQueen in the Pixar movie 'Cars', Iron Man's reflective suit and the robots in disguise in Transformers are all made possible thanks to ray tracing.

You may be asking then that if ray tracing is so great, why it has not been standard practice to use it to create games and other 3D images in real time? The reason is that in computational terms it's phenomenally 'expensive' – placing it out of reach of conventional hardware. This is because, as the name implies, when tracing rays, the processor must track all the rays emitted from a light source and calculate how each one interacts with every object and surface in a scene. As each ray hits an object it will, depending on the type of surface, be either absorbed, reflected, refracted, or scattered, resulting in potentially thousands of additional rays that need to be calculated – a process called global illumination. The typical measure of a ray tracer's performance is in millions of rays per second, a theoretical metric similar to GPixels/sec for GPU fillrate and GFLOPS for GPU compute.

In the movie world, extremely, powerful workstations equipped with power-hungry graphics cards are used to create these scenes 'offline' – taking as long as needed to build up the lighting in a scene. However, for playing a game on a portable device or creating the image of a car in a dashboard that won't work – the scene must be created at a nominal target of 60 frames per second (fps) or higher, and all within mobile power constraints.

Why we've been using rasterisation

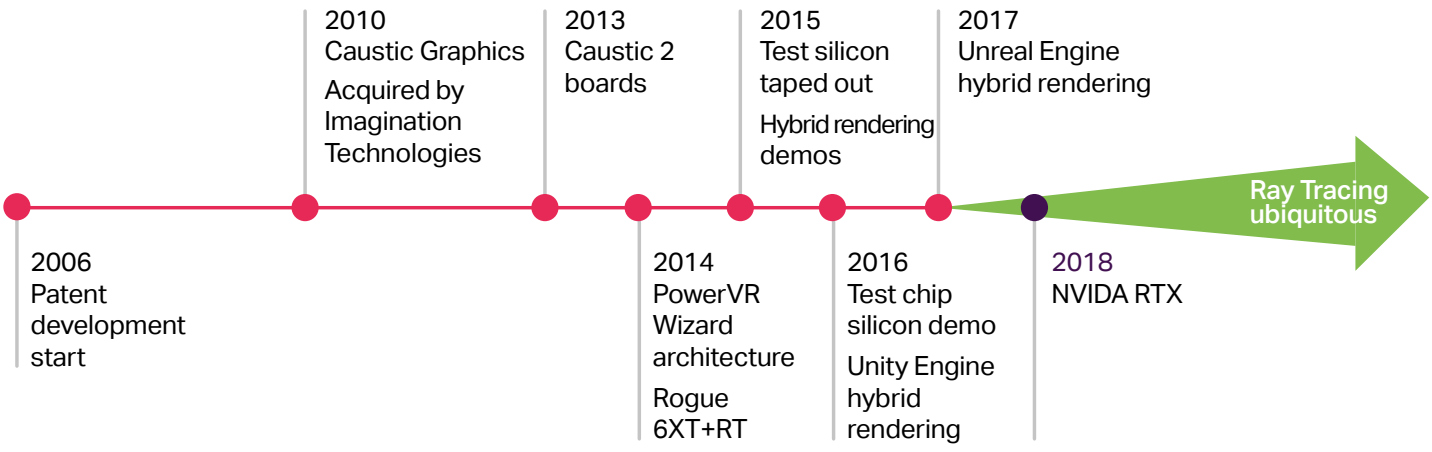
Due to these limitations, the standard for 3D graphics has been a technique called rasterisation, where 3D objects are created using a mesh of triangles. These are then mapped onto a 2D plane and then textured and shaded. This is computationally far 'cheaper' than ray tracing and with their highly parallel nature, today's GPUs are highly tuned to efficiently analyse and shade triangles.

However, the shaders that determine how each pixel should look can only 'fake' real world lighting, which limits realism. While there are rasterisation-based techniques that can be used to emulate ray tracing, such as high shader or compute workloads, many bandwidth-hungry off-screen render targets, and overall bit power consumption cost, these add such complexity and/or are so inefficient that they reduce the benefit of using rasterisation in the first place.

Compared to rasterisation ray tracing is an inherently elegant solution. A single rendering algorithm can be used to create effects without needing to rely on tools such as shadow maps or other lighting passes. To give just one example, in rasterisation dynamic cube maps are frequently used to simulate reflections. However, these are not only computationally and bandwidth expensive but they can also suffer from latency when updating and pixelation due to resolution limitations. Similarly, cascaded shadow maps, with percentage closer filtering to determine soft shadows, requires extra geometry processing, compute, pixel shading and a lot of bandwidth – issues which are trivial with ray tracing.

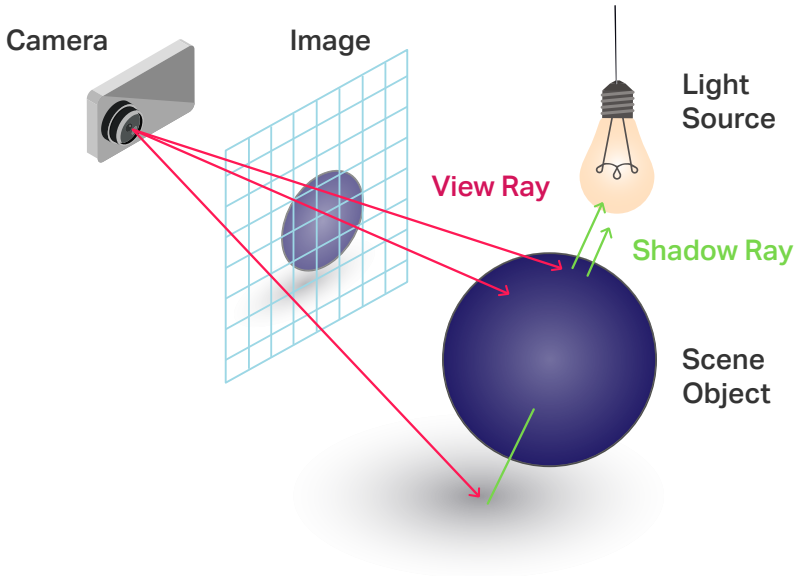
That's not to say that ray tracing does not involve complexity and require expertise to make use of it efficiently, but once it's understood and built into workflows it enables developers to be more productive, freeing them up to enhance their work with additional content or to concentrate on other aspects of the application.

To take graphics to the next level then, it makes sense to turn to ray tracing and it is this that Imagination did with its ray tracing architecture, first announced first demonstrated in 2012.



PowerVR Ray Tracing History

Coming at it from a different angle



Ray racing shoots rays into a scene to build up the algorithm

With its ray tracing architecture Imagination did two groundbreaking things. First, we pioneered 'hybrid' rendering, where traditional rasterisation is used for most of the scene and ray tracing used for the parts where it makes the most difference, namely reflections and shadows, thus greatly reducing the computational performance and bandwidth required.

The second approach is to use a technique called 'backwards ray tracing'. This reverses the concept of rays emitted from a light source bouncing around the scene to eventually reach the eye. Instead, rays are cast from the viewer or 'camera', into the scene and when it hits surfaces it is then traced back to the light source. If the light cannot reach the light source it doesn't need to be calculated, vastly reducing computational complexity.

NVIDIA has taken a similar approach with its new Turing architecture-based cards, where conventional rasterisation hardware is combined with dedicated silicon designed specifically to accelerate some ray tracing calculations. Hardware is, of course, no use without supporting APIs and software titles and NVIDIA has worked with Microsoft so that the Direct X 12 API supports ray tracing. With PowerVR Ray Tracing, however, developers can access it through open standards using extensions to OpenGL ES™ and Vulkan®.

With an update patch to its game 'Battlefield 5' to provide RTX support, the developers DICE brought a ray traced software title to the mainstream for the first time. It should be noted that for performance reasons, ray tracing is only applied in the game to reflective surfaces such as car panels and puddles; shadows are still rasterised. Nevertheless, the significant performance hit caused by enabling ray tracing in the game led to criticism by some product reviewers, particularly in light of the fact that the cards cost over £1,200 to purchase – the highest ever amount for a consumer graphics card.

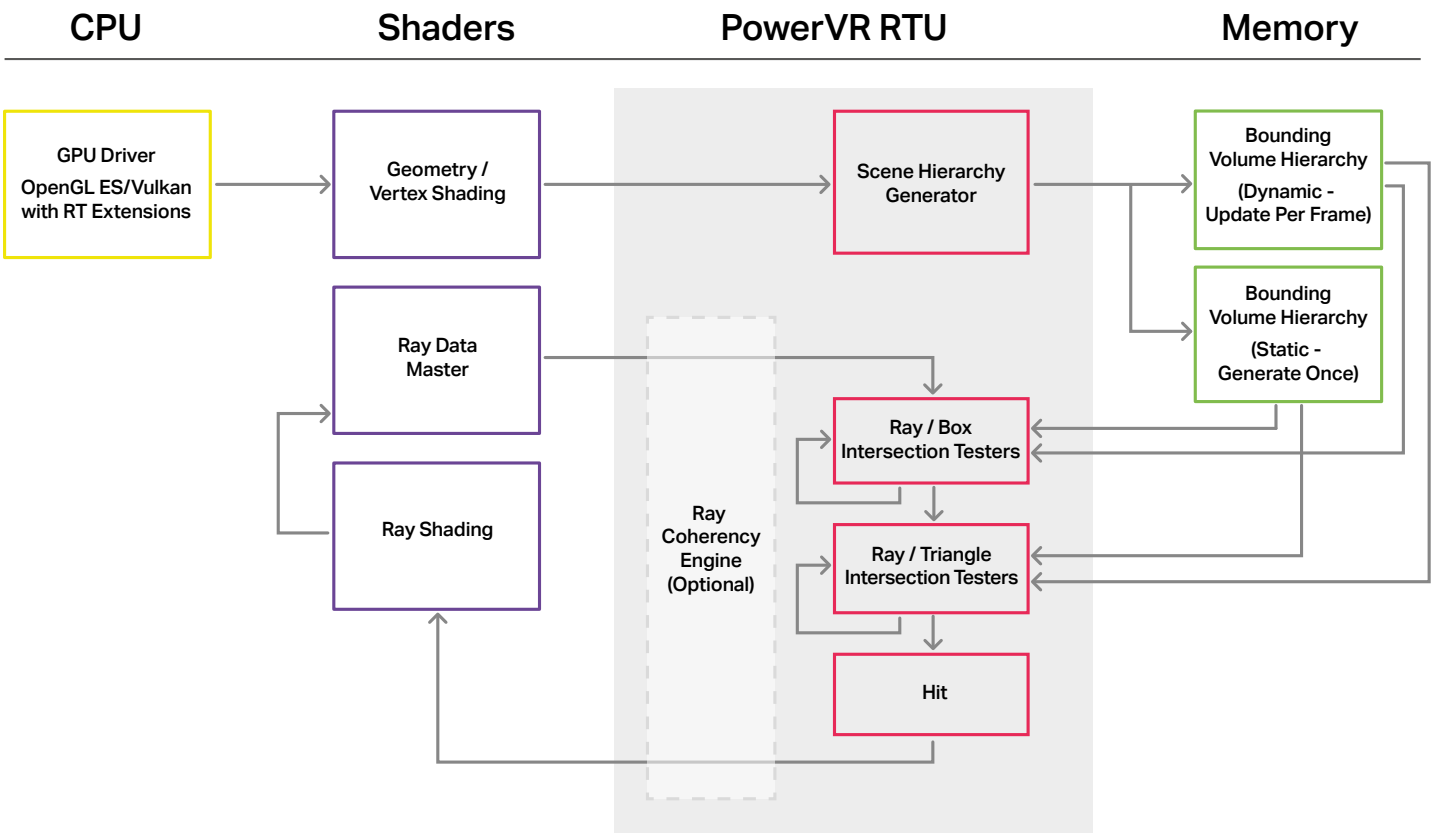
It's worth noting that Battlefield 5 is the first publicly available game supporting RTX and several more are promised, while Epic has announced support in its widely used Unreal engine. It will be interesting to see how these developments impact the market.

Despite the issues surround the launch of the NVIDIA RTX cards and the in-game performance, making real-time ray tracing a reality is a notable achievement that should be admired by anyone that cares about graphics.

How do the architectures compare?

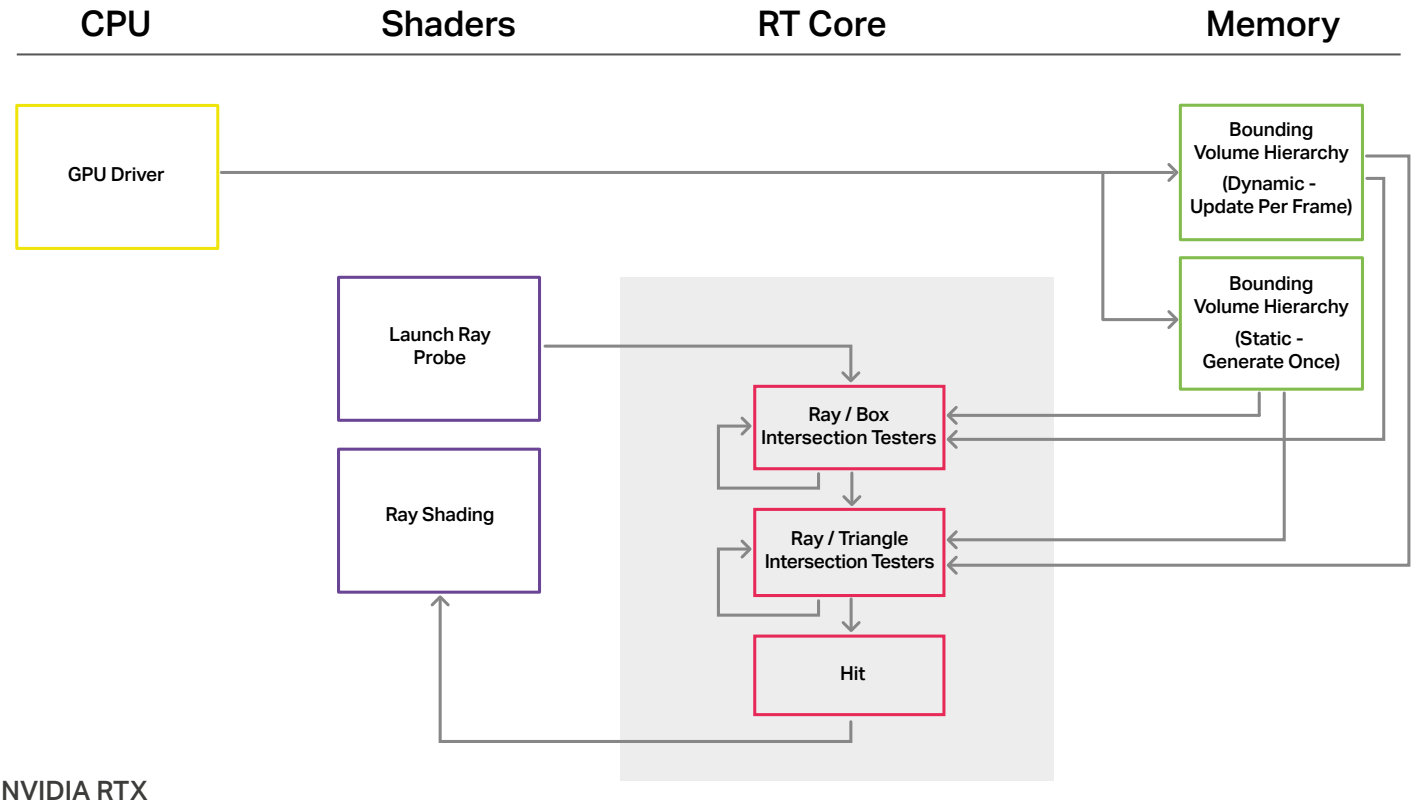
Imagination's original 'Wizard' ray tracing GPU combined a PowerVR Series6 GPU with a Ray Tracing Unit (RTU), a block dedicated to accelerating ray tracing in real time within smartphone power envelopes (using a 28nm process technology, which is much older than today's state-of-the-art 7nm that would deliver higher density and lower power consumption). Our hardware was able to do this more than 100x faster than a conventional GPU and it was this that made ray tracing a practical reality for the first time. Imagination was ahead of the market in this regard and by 2016 had produced working silicon integrated into a PCIe evaluation board designed for demonstration and development purposes. PowerVR Ray Tracing's proven silicon has over 220 associated patents to date, either granted or pending.

It's interesting to compare Imagination's architecture with the NVIDIA solution. PowerVR Ray Tracing features a Scene Hierarchy Generator (SHG) in hardware. The SHG generates a Bounding Volume Hierarchy data structure which is designed to greatly improve the efficiency of detecting which triangles intersect with which rays. Using a brute force approach would require testing every single ray with every triangle in the world which has computationally always been too expensive to perform in real time.



PowerVR Ray Tracing

How do the architectures compare?



NVIDIA RTX

The PowerVR Scene Hierarchy Generator splits the scene into a hierarchical tree of bounding boxes: essentially a large box containing the scene which then is split hierarchically into ever smaller bounding boxes until the lowest level contains triangles. This hierarchical approach cuts down the number of tests by checking ray box intersections and then drilling down inside it until the correct triangle is located.

While both Imagination's and NVIDIA's core contain the Ray/Box Intersection in hardware and both use a Bounding Volume Hierarchy (BVH) data structure only Imagination has the Scene Hierarchy Generator (SGH) in hardware, which means we can support dynamic geometry (e.g. animated characters in a game) much more efficiently.

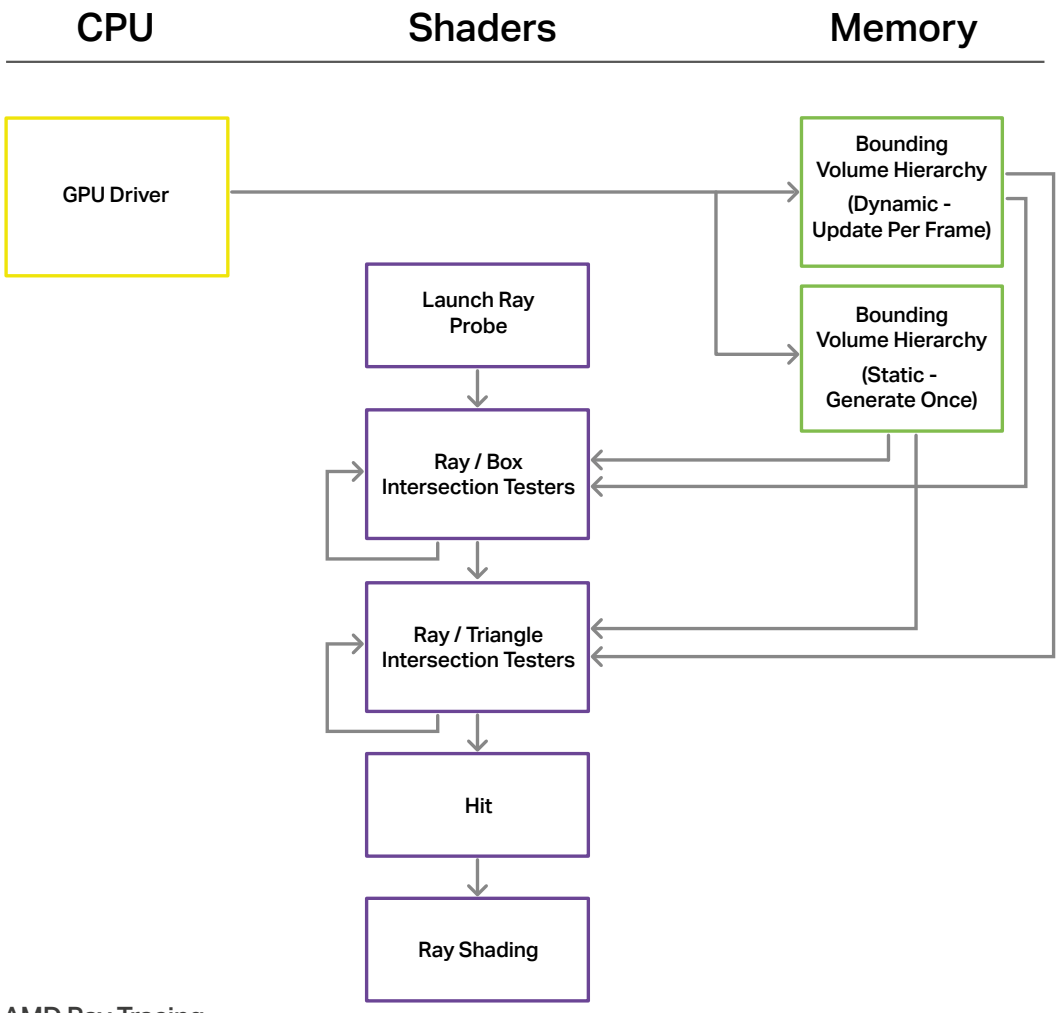
Another key differentiator in PowerVR is an optional block called the Ray Coherency Engine. When rays hit most natural type of materials in a 3D scene, they tend to scatter randomly and are therefore unlikely to be coherent. This random ray behaviour means that as rays are processed they hit and go off in different directions and thus intersect different boxes/triangles. This greatly decreases memory access efficiency and as a result, reduces performance. The Coherency Engine finds commonality between rays and then groups them together, increasing efficiency but at a silicon cost.

AMD currently does not have any dedicated hardware for ray tracing and uses the shaders to perform all ray tracing computation, resulting in much lower performance.

How do the architectures compare?

Due to the limit on the number of rays that can be placed into a scene, image denoising is required in order to produce high-quality results at acceptable frame rates. NVIDIA used the Tensor Cores to do this while Imagination has its PowerVR neural network accelerator (NNA), which offers up to 10 million tera operations per second on a single core, to perform this important post-processing step.

Our silicon-proven solution was demonstrably very power efficient. The Wizard SoC operated at just two watts and the demo board, built using older 28nm process technology, was drawing around 10 watts and at a typical 600MHz generated a peak rate of 300MRay/sec. NVIDIA claims 8GRay/sec maximum for Turing at 1.5GHz, which is 2.5 times the clock, while consuming 225W. At 2.5x the clock PowerVR Ray Tracing would produce 750MRay/sec, so if scaled to match the 8GRay/sec figure, it's reasonable to conclude that our solution in a modern SoC implementation would be much more power efficient than NVIDIA's offering.



AMD Ray Tracing

Imagine the possibilities

Now that the first taste of ray tracing has arrived in the mainstream consumers are going to want to see more. Once gamers start to appreciate the benefits that ray tracing can bring they will want the same experience on their mobile devices, their game consoles and even in their car. Ray tracing can also add huge value to many visual experiences. Kitchen retailers create 3D renders to enable their clients to visualise their new purchase and ray tracing could take this to another level.

Ray tracing could be used to enhance the realism of dials in digital dashboards and for the 3D car model in surround view. It could take data from the cameras to accurately reflect the environment onto the 3D car model so the driver could judge distances more accurately.

And while AR and VR are yet to break into the mainstream there is still a lot of belief out there that they will eventually do so. When it comes to VR, to keep everything smooth, techniques such as variable sample rates and foveated rendering need to be employed and with our hybrid ray tracing these are easier to achieve.

Imagine the possibilities

While the NVIDIA solution is designed for gaming PCs where the power consumption is of little concern, our patented, 'mobile-first' ray tracing technology is an ideal fit that can scale from battery-powered devices, such as smartphones to portable or permanently powered game consoles.

Cloud-based gaming is also widely predicted to supplement, and eventually replace local hardware with high-powered gaming racks using the internet to deliver high-quality gaming to players with low-end hardware. However, power consumption costs and heat management are key challenges faced by server farms. Ray tracing is highly beneficial in this scenario and can, in fact, help deliver widely scalable server-farm-based gaming. As every player in a map is occupying the same 'world' the geometry can be updated once and then sent to every player on the server, with ray tracing used to generate each player's unique view – rather than the geometry having to be processed for each viewpoint as it would be with rasterisation. This would enable a server architecture that offers lower power consumption while also delivering stunning visual effects quality.

Ray tracing is a disruptive technology that promises to revolutionise 3D graphics. Imagination's PowerVR Ray Tracing is available as a widely usable licensable architecture capable of enabling stand-alone ray-tracing processors or hybrid ray tracing/rasterisation devices.

If you want to create products capable of displaying state-of-the-art real-time graphics, in a cost-effective and power efficient manner, then you should talk to Imagination today.

Industry esperts' thoughts

Jon Peddie
President and founder of Jon Peddie Research

I think ray tracing is the big topic this year. As the famous saying goes; a picture is worth 1,000 words – and now it can be truly realised with a perfectly rendered ray tracing image. Where physically accurate photo-realistic representation of objects is important – such as for concept design and virtual prototyping for industries such as automotive, architecture, fashion as well as for gaming – it is rapidly evolving from a 'nice to have' to a 'must have'. In 2012, I first saw real-time hybrid ray tracing from Imagination Technologies and since then it has developed a dedicated ray tracing engine offered in the form of IP, and with good reason."

Jon Peddie Research is a technical research and consulting firm. Based in Tiburon, California, JPR provides specialized services to companies in high-tech fields including graphics hardware development, multimedia for professional applications and consumer electronics, entertainment technology, high-end computing, and Internet access product development.

www.imaginationtech.com