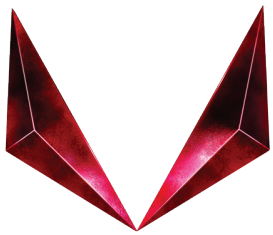


Radeon's next-generation Vega architecture

Contents

1 Introduction

2 "Vega" 10: The first implementation

3 New memory hierarchy and the High-Bandwidth Cache Controller

6 Next-generation geometry engine

8 "Vega" NCU with Rapid Packed Math

9 A revised pixel engine

11 Built for higher clock speeds

12 Tuned for efficiency

13 Display and multimedia improvements

14 A formidable generational shift

Introduction

17 years since the introduction of the first Radeon™, the usage model for graphics processors continues to expand, both within the realm of visual computing and beyond. Our customers are employing GPUs to tackle a diverse set of workloads spanning from machine learning to professional visualization and virtualized hosting—and into new fields like virtual reality. Even traditional gaming constantly pushes the envelope with cutting-edge visual effects and unprecedented levels of visual fidelity in the latest games.

Along the way, the data sets to be processed in these applications have mushroomed in size and complexity. The processing power of GPUs has multiplied to keep pace with the needs of emerging workloads, but unfortunately, GPU memory capacities haven't risen as substantially. Meanwhile, the throughput of nearly all types of high-performance processors has been increasingly gated by power consumption.

With these needs in mind, the Radeon Technologies Group set out to build a new architecture known as "Vega." "Vega" is the most sweeping change to AMD's core graphics technology since the introduction of the first GCN-based chips five years ago. The "Vega" architecture is intended to meet today's needs by embracing several principles: flexible operation, support for large data sets, improved power efficiency, and extremely scalable performance. "Vega" introduces a host of innovative features in pursuit of this vision, which we'll describe in the following pages. This new architecture promises to revolutionize the way GPUs are used in both established and emerging markets by offering developers new levels of control, flexibility, and scalability.

“Vega” 10: The first implementation

The first implementation of the “Vega” architecture is the “Vega” 10 GPU. “Vega” 10 is a relatively large-scale chip meant to serve multiple markets, including high-resolution gaming and VR, the most intensive workstation-class applications, and the GPU computing space, including key vertical markets like HPC and machine learning.
The “Vega” 10 chip is fabricated using 14-nm LPP FinFET process technology, and it packs 12.5 billion transistors into a 486 mm² silicon die. This chip is optimized to take advantage of the inherently lower leakage power of FinFET transistors by operating at much higher clock frequencies than past Radeon™ graphics processors. Radeon™ RX Vega products will ship with boost clocks as high as 1.67GHz, compared to boost clocks around 1GHz for our 28-nm parts of comparable scale.[1]

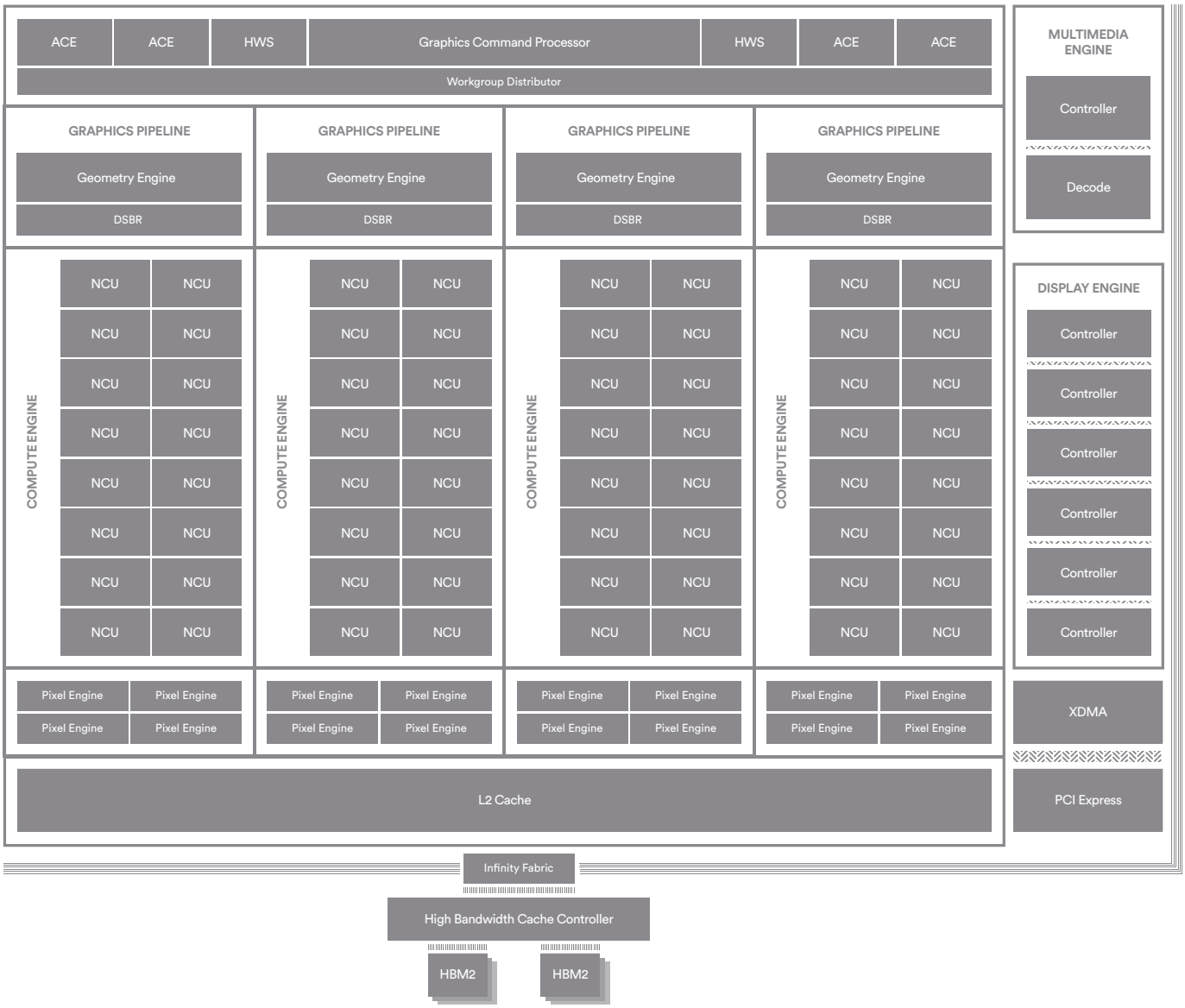


Figure 1: **“Vega” 10 block diagram**

The “Vega” 10 graphics core has 64 next-generation compute units (NCUs), which give it a grand total of 4,096 stream processors.[2] Although this unit count may be familiar from prior Radeon™ GPUs, the combination of higher clock speeds and improvements in the “Vega” architecture can improve instruction throughput substantially. On the Radeon™ RX Vega64 Liquid Cooled Edition, this “Vega” NCU shader array is capable of 13.7 teraflops of single-precision arithmetic throughput. Thanks to its facility for packed 16-bit math, this same shader array can achieve a peak rate of 27.4 teraflops of half-precision arithmetic throughput. A similar dynamic applies to other key graphics rates. For instance, the fixed-function geometry pipeline is capable of four primitives per clock of throughput, but “Vega’s” next-generation geometry path has much higher potential capacity, as we’ll explain in more detail below.

“Vega” 10 is the first AMD graphics processor built using the Infinity Fabric interconnect that also underpins our "Zen" microprocessors. This low-latency, SoC-style interconnect provides coherent communication between on-chip logic blocks with built-in quality-of-service and security capabilities. Because it is a standard across our IP portfolio, Infinity Fabric allows us to take a flexible, modular approach to processor design. We can mix and match various IP blocks to create new configurations to serve our customers' needs. In “Vega” 10, Infinity Fabric links the graphics core and the other main logic blocks on the chip, including the memory controller, the PCI Express controller, the display engine, and the video acceleration blocks. Thanks to the Infinity Fabric support built into each of these IP blocks, our future GPUs and APUs will have the option of incorporating elements of the “Vega” architecture at will.

New memory hierarchy and the High-Bandwidth Cache Controller

GPUs are massively parallel processors that require tremendous amounts of data movement to realize peak throughput. They rely on a combination of advanced memory devices and multi-level cache systems to meet this need. In a typical arrangement, registers for the various processing elements pull data from a set of L1 caches,

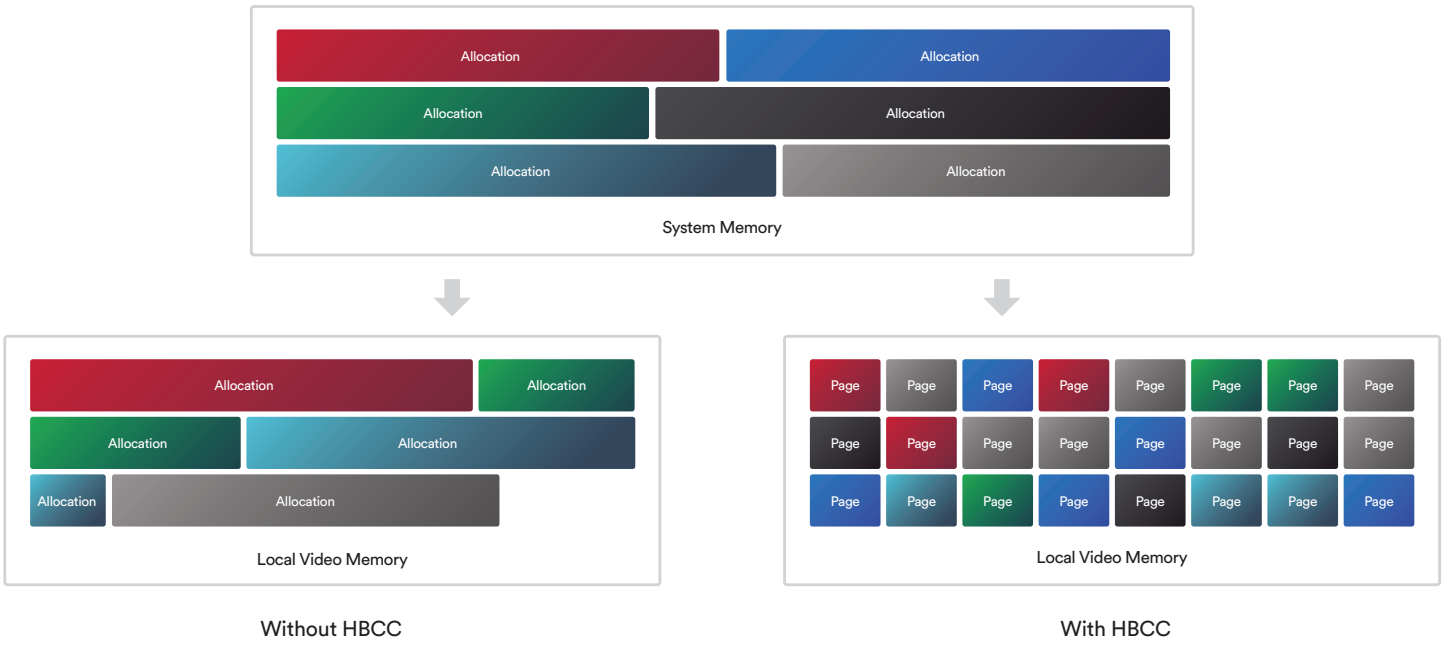


Figure 2: **HBCC vs. Standard Memory Allocation**

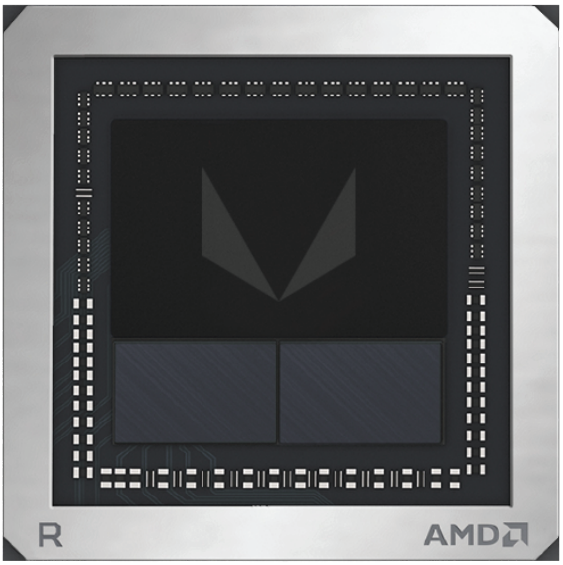
which in turn access a unified, on-chip L2 cache. The L2 cache then provides high-bandwidth, low-latency access to the GPU's dedicated video memory.

A GPU generally requires its entire working set of data and resources to be kept in local video memory, since the alternative (i.e., pulling the data from the host system memory over a PCI Express bus) does not provide sufficiently high bandwidth or low enough latency to keep it running at maximum performance. Developers have experimented with a variety of hacks to work around this issue, but the increased complexity of software memory management poses a daunting challenge. In the meantime, the cost and density of local memory have effectively constrained the maximum size of graphics data sets.

The "Vega" architecture breaks through this limitation by allowing its local video memory to behave like a last-level cache. If the GPU tries to access a piece of data not currently stored in local memory, it can pull just the necessary memory pages across the PCIe bus and store them in the high-bandwidth cache, rather than forcing the GPU to stall while the entire missing resource is copied over. Since pages are typically much smaller than entire textures or other resources, they can be copied over much more quickly. Once the transfer is complete, any subsequent accesses of these memory pages will benefit from lower latency since they are now resident in the cache.

This capability is made possible by an addition to the memory controller logic called the High-Bandwidth Cache Controller (HBCC). It provides a set of features that allow remote memory to behave like local video memory and local video memory to behave like a last-level cache.[3] The HBCC supports 49-bit addressing, providing up to 512 terabytes of virtual address space. This is enough to cover the 48-bit address space accessible by modern CPUs and is several orders of magnitude greater than the few gigabytes of physical video memory typically attached to today's GPUs.

The HBCC is a revolutionary technology for server and professional applications. GPUs based on the "Vega" architecture have the potential to provide such applications with effective memory performance comparable to local video memory while they're processing data sets closer to the capacity of system memory, and in the future, even extended to mass storage devices such as non-volatile storage. Radeon™ Pro SSG products with on-board solid state storage are particularly well-positioned to take advantage of this capability. AMD's technology in the Radeon Pro SSG products significantly reduces latency and CPU overhead for data transfers from a SSD to the GPU; combining this technology with HBCC allows the GPU to behave as if it had terabytes of local video memory.



HBCC technology can be leveraged for consumer applications, as well. The key limitation in that space is that most systems won't have the benefit of large amounts of system memory (i.e., greater than 32 GB) or solid-state storage on the graphics card. In this case, HBCC effectively extends the local video memory to include a portion of system memory. Applications will see this storage capacity as one large memory space. If they try to access data not currently stored in the local high-bandwidth memory, the HBCC can cache the pages on demand, while less recently used pages are swapped back into system memory. This unified memory pool is known as the HBCC Memory Segment (HMS).

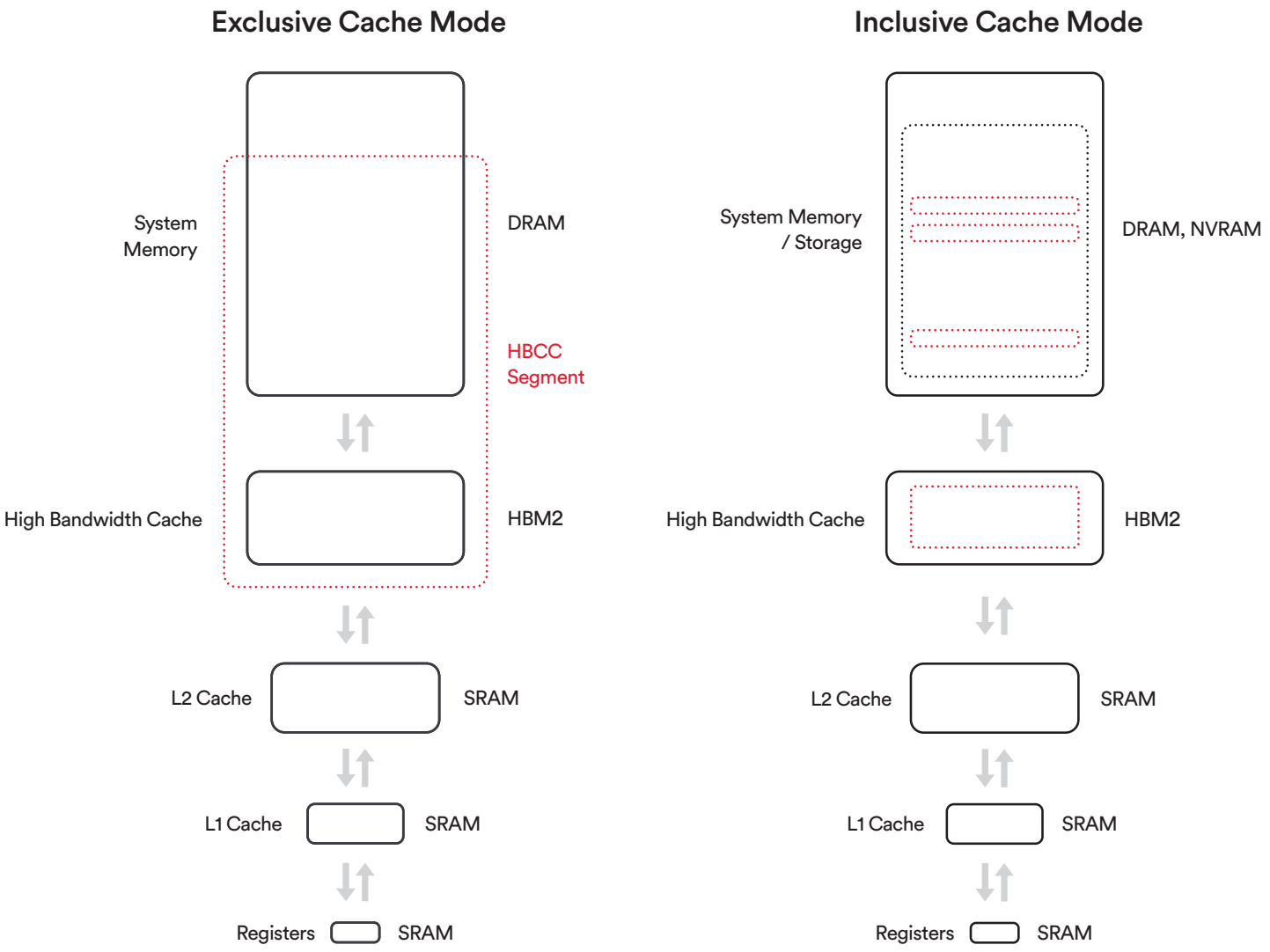


Figure 3: **Memory & cache hierarchy**

The availability of a higher-capacity pool of hardware-managed storage can help enable game developers to create virtual worlds with higher detail, more realistic animations, and more complex lighting effects without having to worry about exceeding traditional GPU memory capacity limitations. In particular, this capability can help remove constraints on level designers, allowing them to realize their visions more fully, with fewer compromises.

To extract maximum benefit from "Vega's" new cache hierarchy, all of the graphics blocks have been made clients of the L2 cache. This arrangement departs from previous GCN-based architectures, where the pixel engines had their own independent caches, and enables greater data re-use. Since the GPU's L2 cache plays a central role in the new memory hierarchy, GPUs based on the "Vega" architecture are designed with generous amounts of it. For example, the

"Vega" 10 GPU features a 4 MB L2 cache, which is twice the size of the L2 cache in previous high-end AMD GPUs.

Treating local video memory like a cache demands the use of the latest memory technology, so HBM2 — second-generation High Bandwidth Memory — is a natural fit. In contrast with the GDDR5 memory devices used on most graphics products today, HBM2 is integrated directly into the GPU package and uses a silicon interposer for the physical interconnect. 3D stacking of memory dice increases density even further, with stacks up to eight devices high and up to eight gigabytes per stack. This arrangement may be used to reduce the total circuit board footprint by more than 75%, enabling extremely compact designs for small-form-factor desktop and notebook systems without sacrificing memory capacity.[4]

HBM2 takes advantage of interposer signal integrity improvements to increase data rates by nearly 2x per pin compared with first-generation HBM while also increasing maximum capacity per stack by a factor of eight.[5] Each stack gets a dedicated 1024-bit memory interface. The wide interfaces allow each device to run at lower clock speeds when providing a given amount of bandwidth, and shorter trace lengths for the interconnects reduce the energy required per bit transferred. These improvements result in much higher power efficiency—over 3.5x the bandwidth per watt versus GDDR5.[6] The combination of lower power consumption and very high peak bandwidth in an extremely compact physical footprint makes HBM2 the clear choice for future.

Next-generation geometry engine

To meet the needs of both professional graphics and gaming applications, the geometry engines in "Vega" have been tuned for higher polygon throughput by adding new fast paths through the hardware and by avoiding unnecessary processing. This next-generation geometry (NGG) path is much more flexible and programmable than before.

To highlight one of the innovations in the new geometry engine, primitive shaders are a key element in its ability to achieve much higher polygon throughput per transistor. Previous hardware mapped quite closely to the standard Direct3D rendering pipeline, with several stages including input assembly, vertex shading, hull shading, tessellation, domain shading, and geometry shading. Given the wide variety of rendering technologies now being implemented by developers, however, including all of these stages isn't always the most efficient way of doing things. Each stage has various restrictions on inputs and outputs that may have been necessary for earlier GPU designs, but such restrictions aren't always needed on today's more flexible hardware.

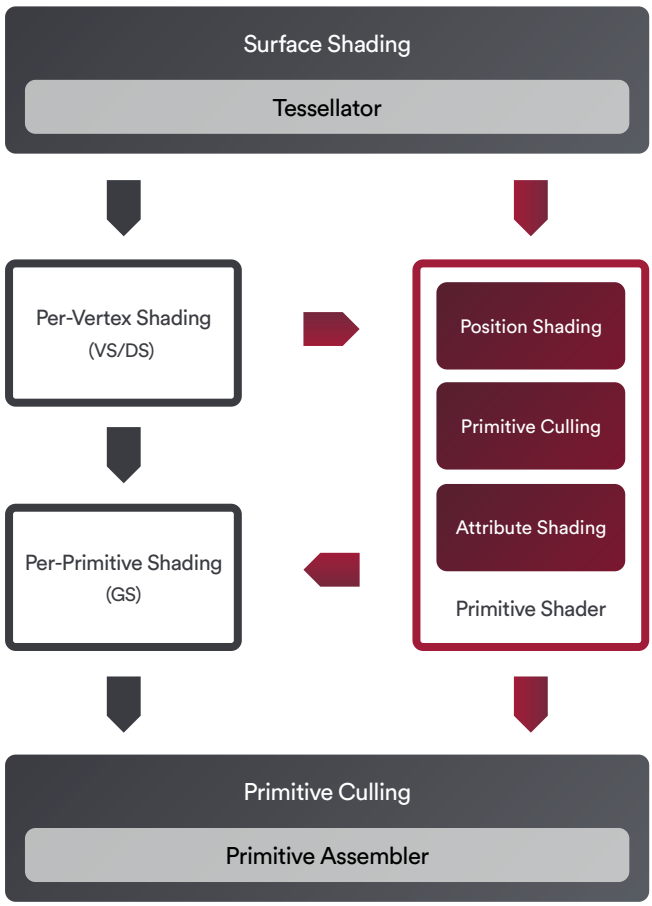


Figure 4
Geometry processing via the traditional DX path (left) and primitive shaders (right)

"Vega's" new primitive shader support allows some parts of the geometry processing pipeline to be combined and replaced with a new, highly efficient shader type. These flexible, general-purpose shaders can be launched very quickly, enabling more than four times the peak primitive cull rate per clock cycle.

In a typical scene, around half of the geometry will be discarded through various techniques such as frustum culling, back-face culling, and small-primitive culling. The faster these primitives are discarded, the faster the GPU can start rendering the visible geometry. Furthermore, traditional geometry pipelines discard primitives after vertex processing is completed, which can waste computing resources and create bottlenecks when storing a large batch of unnecessary attributes. Primitive shaders enable early culling to save those resources.

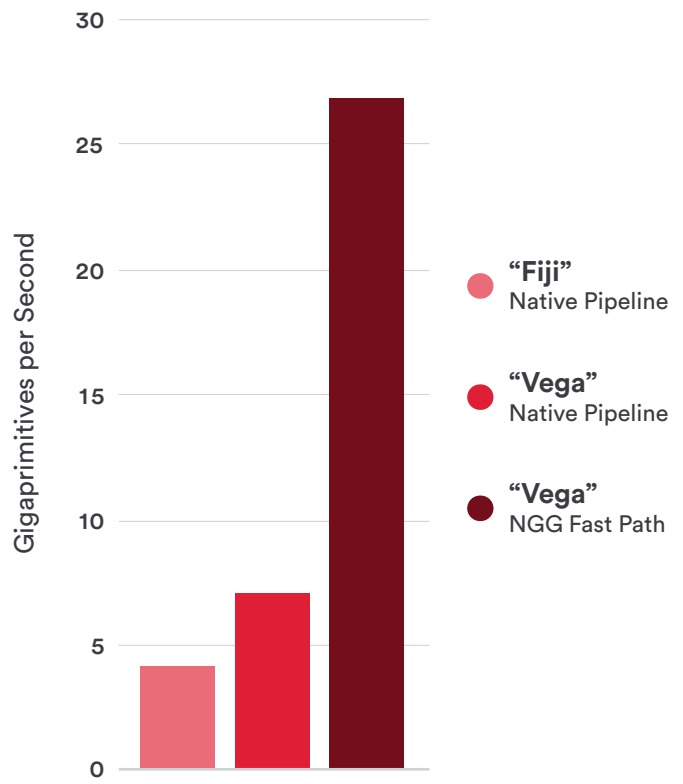


Figure 5: **Peak primitive discard rate for native pipeline versus NGG fast path**

The "Vega" 10 GPU includes four geometry engines which would normally be limited to a maximum throughput of four primitives per clock, but this limit increases to more than 17 primitives per clock when primitive shaders are employed.[7]

Primitive shaders can operate on a variety of different geometric primitives, including individual vertices, polygons, and patch surfaces. When tessellation is enabled, a surface shader is generated to process patches and control points before the surface is tessellated, and the resulting polygons are sent to the primitive shader. In this case, the surface shader combines the vertex shading and hull shading stages of the Direct3D graphics pipeline, while the primitive shader replaces the domain shading and geometry shading stages.

Primitive shaders have many potential uses beyond high-performance geometry culling. Shadow-map rendering is another ubiquitous process in modern engines that could benefit greatly from the reduced processing overhead of primitive shaders. We can envision even more uses for this technology in the future, including deferred vertex attribute computation, multi-view/multi-resolution rendering, depth pre-passes, particle systems, and full-scene graph processing and traversal on the GPU. Primitive shaders will coexist with the standard hardware geometry pipeline rather than replacing it. In keeping with "Vega's" new cache hierarchy, the geometry engine can now use the on-chip L2 cache to store vertex parameter data.

This arrangement complements the dedicated parameter cache, which has doubled in size relative to the prior-generation "Polaris" architecture. This caching setup makes the system highly tunable and allows the graphics driver to choose the optimal path for any use case. Combined with high-speed HBM2 memory, these improvements help to reduce the potential for memory bandwidth to act as a bottleneck for geometry throughput.

Another innovation of "Vega's" NGG is improved load balancing across multiple geometry engines. An intelligent workload distributor (IWD) continually adjusts pipeline settings based on the characteristics of the draw calls it receives in order to maximize utilization.

One factor that can cause geometry engines to idle is context switching. Context switches occur whenever the engine changes from one render state to another, such as when changing from a draw call for one object to that of a different object with different material properties. The amount of data associated with render states can be quite large, and GPU processing can stall if it runs out of available context storage. The IWD seeks to avoid this performance overhead by avoiding context switches whenever possible.

Some draw calls also include many small instances (i.e., they render many similar versions of a simple object). If an instance does not include enough primitives to fill a wavefront of 64 threads, then it cannot take full advantage of the GPU's parallel processing capability, and some proportion of the GPU's capacity goes unused. The IWD can mitigate this effect by packing multiple small instances into a single wavefront, providing a substantial boost to utilization.

"Vega" NCU with Rapid Packed Math

GPUs today often use more mathematical precision than necessary for the calculations they perform. Years ago, GPU hardware was optimized solely for processing the 32-bit floating point operations that had become the standard for 3D graphics. However, as rendering engines have become more sophisticated—and as the range of applications for GPUs has extended beyond graphics processing—the value of data types beyond FP32 has grown.

The programmable compute units at the heart of "Vega" GPUs have been designed to address this changing landscape with the addition of a feature called Rapid Packed Math. Support for 16-bit packed math doubles peak floating-point and integer rates relative to 32-bit operations. It also halves the register space as well as the data movement required to process a given number of operations. The new instruction set includes a rich mix of 16-bit floating point and integer instructions, including FMA, MUL, ADD, MIN/MAX/MED, bit shifts, packing operations, and many more.

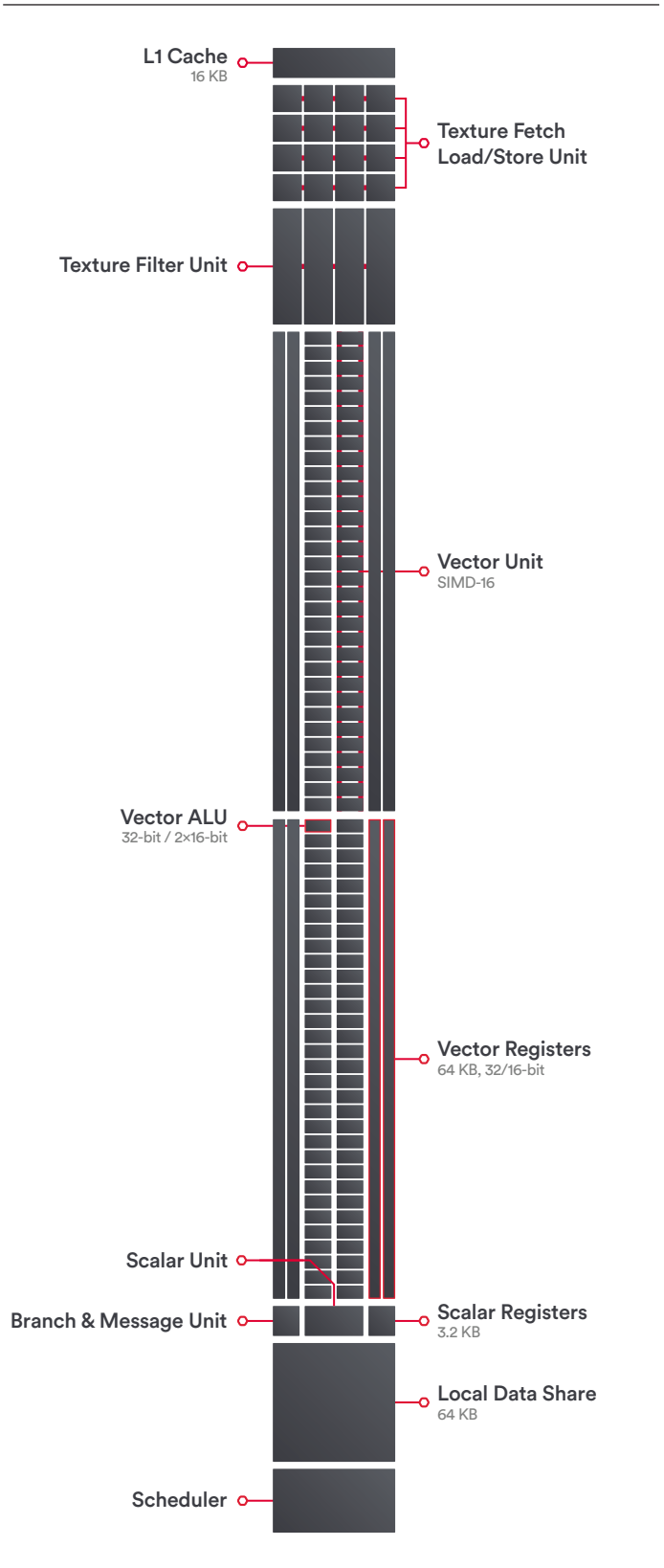


Figure 6: The "Vega" NCU

For applications that can leverage this capability, Rapid Packed Math can provide a substantial improvement in compute throughput and energy efficiency. In the case of specialized applications like machine learning and training, video processing, and computer vision, 16-bit data types are a natural fit, but there are benefits to be had for more traditional rendering operations, as well. Modern games, for example, use a wide range of data types in addition to the standard FP32. Normal/direction vectors, lighting values, HDR color values, and blend factors are some examples of where 16-bit operations can be used.

With mixed-precision support, "Vega" can accelerate the operations that don't benefit from higher precision while maintaining full precision for the ones that do. Thus, the resulting performance increases need not come at the expense of image quality.

In addition to Rapid Packed Math, the NCU introduces a variety of new 32-bit integer operations that can improve performance and efficiency in specific scenarios. These include a set of eight instructions to accelerate memory address generation and hashing functions (commonly used in cryptographic processing and cryptocurrency mining), as well as new ADD/SUB instructions designed to minimize register usage.

The NCU also supports a set of 8-bit integer SAD (Sum of Absolute Differences) operations. These operations are important for a wide range of video and image processing algorithms, including image classification for machine learning, motion detection, gesture recognition, stereo depth extraction, and computer vision. The QSAD instruction can evaluate 16 4x4-pixel tiles per NCU per clock cycle and accumulate the results in 32-bit or 16-bit registers. A maskable version (MQSAD) can provide further optimization by ignoring background pixels and focusing computation on areas of interest in an image.

The potent combination of innovations like Rapid Packed Math and the increased clock speeds of the NCU deliver a major boost in peak math throughput compared with earlier products, with a single "Vega" 10 GPU capable of exceeding 27 teraflops or 55 trillion integer ops per second.

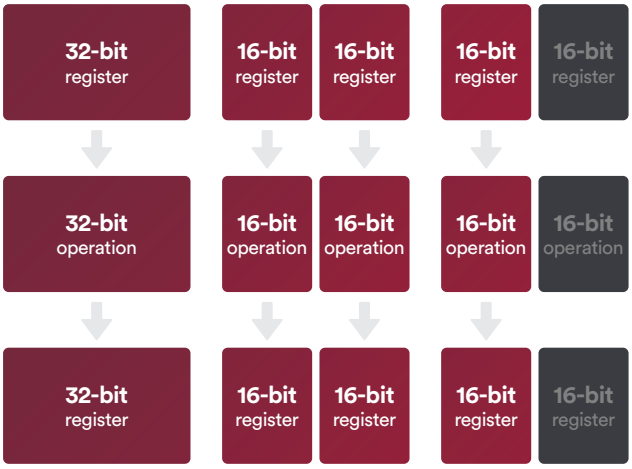


Figure 7: **Rapid Packed Math illustration**

A revised pixel engine

As ultra-high resolution and high-refresh displays become more widespread, maximizing pixel throughput is becoming more important. Monitors with 4K, 5K, and 8K resolutions and refresh rates up to 240Hz are dramatically increasing the demands on today's GPUs. Meanwhile, VR headsets present a novel challenge to pixel rates. The pixel engines in the "Vega" architecture are built to tackle these demands with an array of new features.

The Draw-Stream Binning Rasterizer (DSBR) is an important innovation to highlight. It has been designed to reduce unnecessary processing and data transfer on the GPU, which helps both to boost performance and to reduce power consumption. The idea was to combine the benefits of a technique already widely used in handheld graphics products (tiled rendering) with the benefits of immediate-mode rendering used high-performance PC graphics.

Standard immediate-mode rendering works by rasterizing each polygon as it is submitted until the whole scene is complete, whereas tiled rendering works by dividing the screen into a grid of tiles and then rendering each tile independently.

The DSBR works by first dividing the image to be rendered into a grid of bins or tiles in screen space and then collecting a batch of primitives to be rasterized in the scan converter. The bin and batch sizes can be adjusted dynamically to optimize for the content being rendered. The DSBR then traverses the batched primitives one bin at a time, determining which ones are fully or partially covered by the bin. Geometry is processed once, requiring one clock cycle per primitive in the pipeline. There are no restrictions on when binning can be enabled, and it is fully compatible with tessellation and geometry shading. ("Vega" 10 has four front-ends in all, each with its own rasterizer.)

This design economizes off-chip memory bandwidth by keeping all the data necessary to rasterize geometry for a bin in fast on-chip memory (i.e., the L2 cache). The data in off-chip memory only needs to be accessed once and can then re-used before moving on to the next bin. "Vega" uses a relatively small number of tiles, and it operates on primitive batches of limited size compared with those used in previous tile-based rendering architectures. This setup keeps the costs associated with clipping and sorting manageable for complex scenes while delivering most of the performance and efficiency benefits.

Pixel shading can also be deferred until an entire batch has been processed, so that only visible foreground pixels need to be shaded. This deferred step can be disabled selectively for batches that contain polygons with transparency. Deferred shading reduces unnecessary work by reducing overdraw (i.e., cases where pixel shaders are executed multiple times when different polygons overlap a single screen pixel).

Deferred pixel processing works by using a scoreboard for color samples prior to executing pixel shaders on them. If a later sample occludes or overwrites an earlier sample, the earlier sample can be discarded before any pixel shading is done on it. The scoreboard has limited depth, so it is most powerful when used in conjunction with binning.

These optimizations can significantly reduce off-chip memory traffic, boosting performance in memory-bound scenarios and reducing total graphics power consumption. In the case of "Vega" 10, we observed up to 10% higher frame rates and memory bandwidth reductions of up to 33% when the DSBR is enabled for existing game applications, with no increase in power consumption.[8]

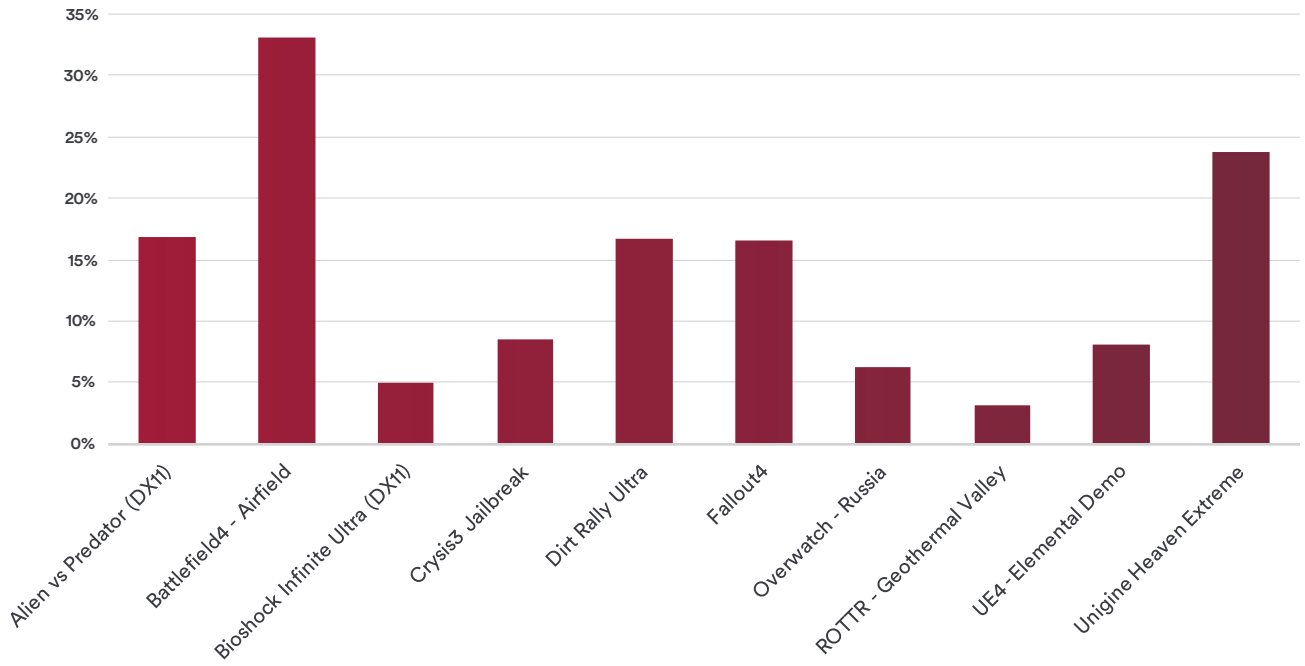


Figure 8: **HBCC vs. Standard Memory Allocation**

Even larger performance improvements are possible when developers submit geometry in a way that maintains screen space locality or in cases where many large overlapping polygons need to be rendered. For example, performance more than doubles in one professional graphics workload, the energy01 test in SPECviewperf ® 12, thanks to the DSBR.[9]

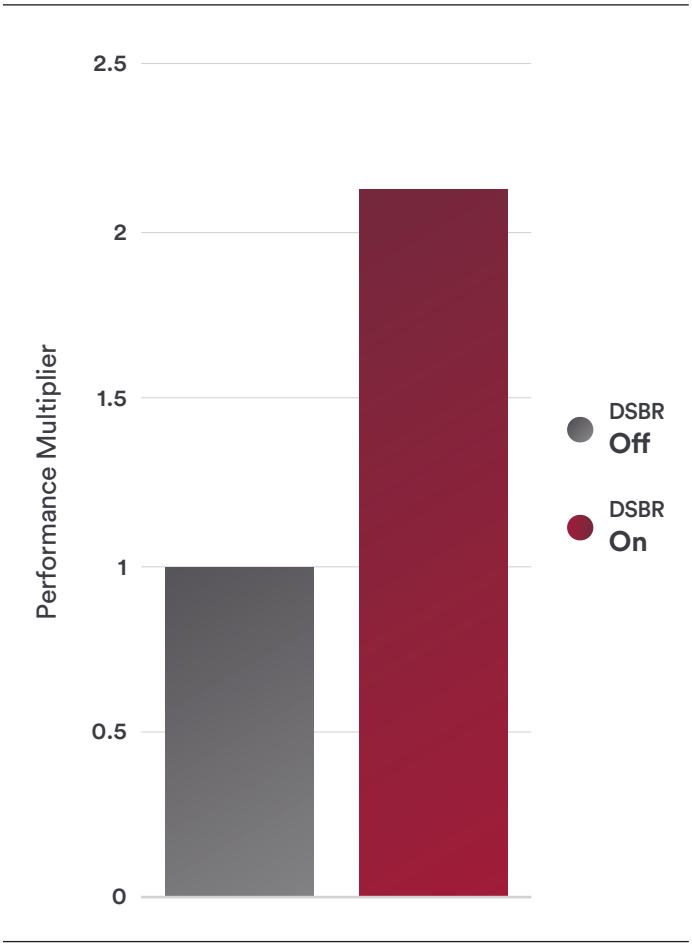


Figure 9: **SPECviewperf® 12 energy01 performance with and without DSBR**

"Vega's" new pixel engine includes more than just performance and efficiency optimizations. It also incorporates the most feature-complete implementation of DirectX® 12 (feature level 12_1) support in any GPU released to date, as well as extensive support for the Vulkan® 1.0 API.

This fact makes "Vega" 10 the ideal platform for advanced graphics development and for experimenting with new techniques.

Feature	"Polaris"	"Vega"
DirectX Feature Level	12_0	12_1
Resource Binding *(Tier 1-3)*	Tier 3	Tier 3
Tiled Resources *(Tier 1-3)*	Tier 2	Tier 3
Resource Heap *(Tier 1-2)*	Tier 2	Tier 2
Raster Ordered Views	No	Yes
Conservative Rasterization *(Tier 1-3)*	No	Tier 3
Pixel Shader Stencil Reference	Yes	Yes
Standard Swizzle	No	Yes
High Performance Concurrent Async Compute and Graphics	Yes	Yes
Shader Intrinsics	SM6.0+	SM6.0+

Figure 10: **DirectX® feature support**

Built for higher clock speeds

One of the key goals for the "Vega" architecture was achieving higher operating clock speeds than any prior Radeon™ GPU. Put simply, this effort required the design teams to close on higher frequency targets. The simplicity of that statement belies the scope of the task, though. Meeting "Vega's" substantially tighter timing targets required some level of design effort for virtually every portion of the chip.

In some units—for instance, in the texture decompression data path of the L1 cache—the teams added more stages to the pipeline, reducing the amount of work done in each clock cycle in order to meet "Vega's" tighter timing targets.

Adding stages is a common means of improving the frequency tolerance of a design, but those additional stages can contribute more latency to the pipeline, potentially impacting performance. In many cases, these impacts can be minor. In our texture decompression example, the additional latency might add up to two clock cycles out of the hundreds required for a typical texture fetch—a negligible effect.

In other instances, on more performance-critical paths, the "Vega" project required creative design solutions to better balance frequency tolerance with per-clock performance. Take, for example, the case of the "Vega" NCU. The design team made major changes to the compute unit in order to improve its frequency tolerance without compromising its core performance.

First, team changed the fundamental floorplan of the compute unit. In prior GCN architectures with less aggressive frequency targets, the presence of wired connections of a certain length was acceptable because signals could travel the full distance in a single clock cycle. For this architecture, some of those wire lengths had to be reduced so signals could traverse them within the span of "Vega's" much shorter clock cycles. This change required a new physical layout for the "Vega" NCU with a floorplan optimized to enable shorter wire lengths.

This layout change alone wasn't sufficient, though. Key internal units, like the instruction fetch and decode logic, were rebuilt with the express goal of meeting "Vega's" tighter timing targets. At the same time, the team worked very hard to avoid adding stages to the most performance-critical paths. Ultimately, they were able to close on a design that maintains the four-stage depth of the main ALU pipeline and still meets "Vega's" timing targets.

"Vega" 10 also leverages high-performance custom SRAMs originally developed by the "Zen" CPU team. These SRAMs, modified for use in the in general-purpose registers of the Vega NCU, offer improvements on multiple fronts, with 8% less delay, an 18% savings in die area, and a 43% reduction in power use versus standard compiled memories.[10]

Tuned for efficiency

The "Vega" architecture includes a host of provisions intended to improve the GPU's power efficiency in various ways, some of them new and others extensions or refinements of existing technologies.

Among the changes in "Vega" 10 is a new, more capable power-management microcontroller. Thanks to its additional performance headroom, the new microcontroller allows for the implementation of more complex power-control algorithms, and the addition of a floating-point unit allows for higher-precision calculations to be a part of that mix.

Accordingly, "Vega" 10 carries over the Adaptive Voltage and Frequency Scaling (AVFS) technology from prior AMD GPUs. A network of AVFS sensor modules distributed across the chip allows the silicon to self-tune, in real time, for optimal voltage levels at the present temperature and clock frequency.

Meanwhile, an improved "deep sleep" state allows "Vega" 10 to scale down its clock speeds dramatically at idle in order to achieve substantially lower power consumption. In deep sleep, a clock divider can reduce clock frequencies by a factor of 32, reducing operating speeds across the entire clock tree. At the same time, the "Vega" graphics core can shift from its default PLL clock source to a lower speed (~100MHz) static clock. That new clock source is then modified by the clock divider, yielding exceptionally low clock speeds for the graphics core in deep sleep.

This deeper sleep state is complemented by the addition of a low-power memory state. In addition to three active memory states, "Vega" 10 incorporates a low-frequency state for its HBM2 memory, with memory clocks as low as 167MHz. The GPU drops its memory into this state when the graphics engine is completely idle, such as when displaying a static screen.

"Vega" 10 also allows for finer-grained control over operating frequencies with the addition of a third clock

domain, beyond the graphics core and memory domains, for its Infinity Fabric SoC-level interconnect. The Infinity Fabric logic links the graphics core to other on-chip units like the multimedia, display, and I/O blocks. In "Vega" 10, this fabric is clocked separately from the graphics core. As a result, the GPU can maintain high clock speeds in the Infinity Fabric domain in order to facilitate fast DMA data transfers in workloads that feature little to no graphics activity, such as video transcoding. Meanwhile, the GPU can keep the graphics core clocked down, saving power without compromising performance in the task at hand.

To take advantage of this sort of dynamic power-tuning capability, "Vega" adds a new feature known as active workload identification. The driver software can identify certain workloads with specific needs—such as full-screen gaming, compute, video, or VR—and notify the power management microcontroller than such a workload is active. The microcontroller can then tune the chip's various domains appropriately to extract the best mix of performance and power consumption.

Display and multimedia improvements

"Vega" 10 offers robust support for the latest display standards, carrying over the deep versatility of the "Polaris" architecture and adding several new capabilities to the mix.

The GPU supports the DisplayPort™ 1.4 standard with HBR3, multi-stream transport (MST), HDR, and high-precision color formats. It also supports HDMI® 2.0 at up to 4K/60Hz, 12 bits per color channel, and 4:2:0 encoding. The HDCP content protection standard is supported for both HDMI® and DisplayPort™ outputs.

Of course, Radeon™ FreeSync™ technology is supported for variable-refresh-rate gaming and, with FreeSync 2™, low-latency tone-mapping of HDR content to the attached display.

Like "Polaris," "Vega" can drive up to six simultaneously attached displays, but it expands support for multiple displays at high bit depths, resolutions, and refresh rates.

Below is a table showing the differences in modes supported compared to the prior generation.

Pixel Format	Display Mode	Simultaneous Displays Supported	
		"Polaris" 10	"Vega" 10
32 bit	4K 60Hz[1]	6	6
	4K 120Hz[1,2]	1	2
	5K 60Hz[1,2]	3 *(dual cable)* 1 *(single cable)*	3 *(dual cable)* 3 *(single cable)*
	8K 30Hz[1,2]	1	3
	8K 60Hz[1,2,3]	1	1
64 bit HDR	4K 60Hz[1]	1	3
	4K 120Hz[1,2]	–	1
	5K 60Hz[1,2]	–	1

1. System and board-level support may vary from GPU's maximum supported capabilities
2. DisplayPort only; not supported on HDMI
3. Via dual DisplayPort cables

For HDR displays, the operating system and applications may choose to use a higher-precision color storage format with 16 bits of floating-point precision per color channel—or 64 bits per pixel, in total. Due to the added bandwidth required, these modes have tighter restrictions on the number of display heads that can be connected at the same time. "Vega" 10 triples the number of heads supported at 4K 60Hz with 64-bit pixel formats and adds support for two additional display modes not possible with "Polaris" 10.

While "Polaris" 10 can support 4K displays well for SDR content and some HDR content, "Vega" 10 adds support for all HDR material using 64-bit pixel formats at 4K and 120Hz.

"Vega" 10 naturally includes the latest versions of AMD's video encode and decode acceleration engines, as well. Like "Polaris," "Vega" offers hardware-based decode of

HEVC/H.265 main10 profile videos at resolutions up to 3840x2160 at 60Hz, with 10-bit color for HDR content. Dedicated decoding of the H.264 format is also supported at up to 4K and 60Hz. “Vega” can also decode the VP9 format at resolutions up to 3840x2160 using a hybrid approach where the video and shader engines collaborate to offload work from the CPU.

“Vega’s” video encode accelerator also supports today's most popular formats. It can encode HEVC/H.265 at 1080p240, 1440p120, and 2160p60. Encoding H.264 video is also supported at 1080p120, 1440p60, and 2160p60. “Vega’s” ability to encode the H.264 format at 3840x2160 at up to 60Hz is an upgrade from “Polaris,” which tops out at 2160p30.

Furthermore, “Vega” 10 extends its video encoding acceleration to an exciting new use case. Radeon™ GPUs uniquely provide robust hardware support for SR-IOV virtualization, allowing the GPU to be shared between multiple user sessions in a virtualized environment. “Vega” 10 adds a crucial new piece to this puzzle: sharing the hardware video encoding and decoding acceleration capabilities built into the GPU. With Radeon Virtualized Encoding, “Vega” 10 GPUs can provide hardware-encoding acceleration for up to 16 simultaneous user sessions. This capability should make “Vega” 10 especially well-suited to hosting sessions in multi-user virtualized applications with graphically intensive workloads, such as enterprise remote workstations and cloud gaming.

A formidable generational shift

In all, the “Vega” architecture delivers improvements in performance on multiple fronts alongside ground-breaking new capabilities. The combination of architectural innovation, process technology improvements, and clock speed uplift allows “Vega” to surpass the performance of our prior-generation Fiji GPU by substantial margins. Below is a brief look at the deltas in key performance specs and rates from the prior generation to this new one.

	R9 Fury X	RX Vega[64] Liquid Cooled Edition
Peak Clock Frequency *(MHz)*	1050	1677
L2 Cache Size *(MB)*	2	4
Peak Shader Throughput *(FP32 Teraflops)*	8.6	13.7
Peak Shader Throughput *(FP16 Teraflops)*	8.6	27.4
Peak Texture Filtering *(Gigatexels/s)*	269	429
Peak Render Back-End Throughput *(Gigapixels/s)*	67	107
Peak Memory Bandwidth *(GB/s)*	512	484
Peak Memory Capacity	4GB VRAM	8GB HBC*

*The Radeon™ Vega Frontier Edition has 16 GB of HBC

Although the improvements are considerable, the numbers above don't fully account for the gains in delivered performance made possible by features like the Draw-Stream Binning Rasterizer and the Next-Generation Geometry engine. Meanwhile, the shift from the traditional VRAM model to High-Bandwidth Cache is a revolutionary change that can't be accounted for by numbers alone.

Thanks to its newfound flexibility and improved programming model, the “Vega” architecture has the potential to deliver even larger improvements in efficiency and performance over time as developers begin to take full advantage of its capabilities.

©2017 Advanced Micro Devices, Inc. All rights reserved. AMD, the AMD Arrow logo, Radeon, FreeSync, LiquidVR, CrossFire and combinations thereof are trademarks of Advanced Micro Devices, Inc. Other product names used in this publicat ion are for identification purposes only and may be trademarks of their respective companies.

The terms HDMI and HDMI High-Definition Multimedia Interface, and the HDMI Logo are trademarks or registered trademarks of HDMI Licensing LLC in the United States and other countries.

All other trademarks and copyrights are the property of their respective owners. All rights reserved.

PCIe® is a registered trademark of PCI-SIG Corporation.

DirectX® is a registered trademark of Microsoft Corporation in the US and other jurisdictions.

OpenCL and the OpenCL logo are trademarks of Apple Inc. used by permission by Khronos.

OpenGL® and the oval logo are trademarks or registered trademarks of Silicon Graphics, Inc. in the United States and/or other countries worldwide.

DisplayPort™ is a trademark of the Video Electronics Standards Association (VESA).

SPEC® and SPECviewperf® are registered trademarks of Standard Performance Evaluation Corporation. Learn more about SPECviewperf at *https://www.spec.org/gwpg/gpc.static/vp12info.html*

[1] Boost clock speed based on Radeon RX Vega 64 Liquid Cooled edition. Fiji clock speed in based on Radeon Fury X (1050 MHz).

[2] Discrete AMD Radeon™ and FirePro™ GPUs based on the Graphics Core Next architecture consist of multiple discrete execution engines known as a Compute Unit (“CU”). Each CU contains 64 shaders (“Stream Processors”) working together. GD-78

[3] This feature is still in development and may be better utilized in future releases of Radeon Software, SDKs available via GPUOpen, or updates from the owners of 3D graphics APIs.

[4] 75% smaller footprint is based on Vega 10 package size with HBM2 (47.5 mm x 47.5 mm) vs. total PCB footprint of R9 290X GPU package + GDDR5 memory devices and interconnects (110 mm x 90 mm).

[5] 8x capacity per stack is based on maximum of 8 GB per stack for HBM2 vs. 1 GB per stack for GDDR5.

[6] 3.5x power efficiency is based on measured memory device + interface power consumption for R9 390X (GDDR5) vs. RX Vega 64 (HBM2).

[7] Peak discard rate: Data based on AMD Internal testing of a prototype RX Vega sample with a prototype branch of driver 17.320. Results may vary for final product, and performance may vary based on use of latest available drivers.

[8] Bytes per frame savings for DSBR: Data based on AMD Internal testing of the Radeon Vega Frontier Edition using an Intel Core i7-5960X CPU with 16 GB DDR4 RAM, Windows 10 64 bit, AMD Radeon Software driver 17.20.

[9] SPECviewperf® performance for DSBR: Data based on AMD Internal testing of an early Radeon Pro WX 9100 sample using an Intel Xeon E5-1650 v3 CPU with 16 GB DDR3 RAM, Windows 10 64 bit, AMD Radeon Software driver 17.30. Using SPECviewperf® 12.1.1 energy-01 subtest, the estimated scores were 8.80 with DSBR off and 18.96 with DSBR on. Results may vary for final product, and performance may vary based on use of latest available drivers.

[10] Comparison based on AMD engineering simulation between standard compiled SRAMs and a new custom SRAM designed by Zen team.

Please Recycle