Vulkan (API)

From Wikipedia, the free encyclopedia

Vulkan is a low-overhead, cross-platform 3D graphics and compute API first announced at GDC 2015 by the Khronos Group.[10][11][12] The Vulkan API was initially referred to as the "next generation OpenGL initiative" by Khronos, but use of those names were discontinued once the Vulkan name was announced.[13] Vulkan is derived from and built upon components of AMD's Mantle API, which was donated by AMD to Khronos with the intent of giving Khronos a foundation on which to begin developing a low-level API that they could standardize across the industry, much like OpenGL.[3][10][14][15][16][17][18]

Like OpenGL, Vulkan targets high-performance realtime 3D graphics applications such as videogames and interactive media across all platforms, and offers higher performance and lower CPU usage, much like Direct3D 12 and Mantle. In addition to its lower CPU usage, Vulkan is also able to better distribute work amongst multiple CPU cores.[19]

Vulkan

Developer(s)	Khronos Group, contributions by AMD[1][2][3][4][5]
Initial release	16 February 2016[6]
Stable release	1.0.26 / 6 September 2016[7]
Development status	Active
Operating system	Android, Linux, Tizen, Microsoft Windows[8][9]
Platform	Compatible hardware
Type	3D graphics and compute API[10]
Website	Khronos.org/Vulkan (https://www.khronos.org/vulkan)
As of	7 June 2016

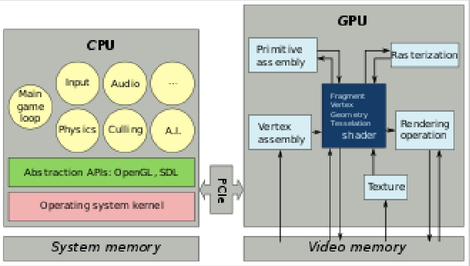
Contents

Features

Vulkan is intended to provide a variety of advantages over other APIs as well as its spiritual predecessor, OpenGL. Vulkan offers lower overhead, more direct control over the GPU, and lower CPU usage.[12] Intended advantages include:

- Vulkan API is well suited for high-end graphics cards as well as for graphics solution present on mobile devices (OpenGL had a specific sub-set for mobile devices called OpenGL ES).

- In contrast to DirectX 12, Vulkan is available on multiple modern operating systems; like OpenGL, the Vulkan API is not locked to a single OS or device form factor. As of release, Vulkan runs on Windows 7, Windows 8, Windows 10, Tizen, Linux, and Android.
- Reduced driver overhead, reducing CPU workloads.[20]
- Reduced load on CPUs through the use of batching, leaving the CPU free to do additional computation or rendering than otherwise.[21]
- Better scaling on multi-core CPUs. Direct3D 11 and OpenGL 4 were initially designed for use with single-core CPUs and only received augmentation to be executed on multi-cores. Even when application developers use the augmentations, the API regularly do not scale well on multi-cores.[22]
- OpenGL uses the high-level language GLSL for writing shaders which forces each OpenGL driver to implement its own compiler for GLSL that executes at application runtime to translate the program's shaders into the GPU's machine code. Vulkan drivers are supposed to ingest instead shaders already translated into an intermediate binary format called SPIR-V (Standard Portable Intermediate Representation), analogous to the binary format that HLSL shaders are compiled into in Direct3D. By allowing shader pre-compilation, application initialization speed improved and a larger variety of shaders can be used per scene. A Vulkan driver only needs to do GPU specific optimization and code generation, resulting in easier driver maintenance, and smaller driver packages in theory (as GPU vendors still have to include OpenGL/CL).[23]
- Unified management of compute kernels and graphical shaders, eliminating the need to use a separate compute API in conjunction with a graphics API.



OpenGL and Vulkan are both rendering APIs, in both cases the GPU executes shaders while the CPU executes everything else. Vulkan has a simpler API which reduces the amount of computation the CPU has to do and it replaces the monolithic global state machine with a much more programmable approach. This enables a fine-grained programming of tasks on the GPU.

OpenGL	Vulkan[24]
one single global state machine	object-based with no global state
state is tied to a single context	all state concepts are localized to a command buffer
GPU memory and synchronization are usually hidden	explicit control over memory management and synchronization
extensive error checking	Vulkan drivers do no error checking

Planned features

At SIGGRAPH 2016, Khronos announced that Vulkan would be getting support for automatic multi-GPU features, similar to what are offered by DirectX 12.[25] Multi-GPU support included in-API removes the need for SLi or Crossfire, which requires graphics cards to be of the same model. API multi-GPU removes this constraint, and instead allows the API to intelligently split the workload among two or more completely different GPUs.[26] For example, integrated GPUs included on the CPU can be used in conjunction with a high-end dedicated GPU for a slight performance boost.

Software that uses Vulkan

Games

- *The Talos Principle* – The first game with Vulkan rendering support.[27]
- *Dota 2* – Vulkan support released in May 2016.[28]
- *Doom* – Vulkan support released in July 2016.[29]
- *vkQuake* – A Vulkan *Quake* port using the idTech game engine from *Doom 2016* was released in July 2016.[30][31]

Game engines

- Source 2 – In March 2015, Valve Corporation announced the Source 2 engine, the successor engine to the original Source engine, would support Vulkan.[32][33]
- Serious Engine 4 – In February 2016, Croteam announced that they were supporting Vulkan in their Serious Engine.[34]
- Unreal Engine 4 – In February 2016, Epic Games announced Unreal Engine 4 support for Vulkan at Samsung's Galaxy S7 Unpacked event.[35][36]
- id Tech 6 – In May 2016, id Software announced *Doom*, running the id Tech 6 engine, would support Vulkan.[37]
- CryEngine – Crytek has plans to include Vulkan support in CryEngine.[38]
- Unity – The engine will support Vulkan in an early state in Q3 2016.[39]
- Xenko - Vulkan support was added in July 2016.[40]

OS components

The *Vulkan Window System Interface* (WSI) does for Vulkan what EGL does for OpenGL ES.[41] EGL is used by OpenGL ES programs to interface with the native platform windowing system. EGL handles context management, surface binding and rendering synchronization.

Compatibility

Initial specifications stated that Vulkan will work on hardware that currently supports OpenGL ES 3.1 or OpenGL 4.X and up.[42] As Vulkan support requires new graphics drivers, this does not necessarily imply that every existing device that supports OpenGL ES 3.1 or OpenGL 4.x will have Vulkan drivers available.

Android 7.0 Nougat supports Vulkan.[43] The software was released in August 2016.[44]

Vulkan support for iOS and macOS has not been announced by Apple, but at least one company provides a Vulkan implementation that runs on top of Metal on iOS and macOS devices.[45]

Company	Hardware				Software support: Vulkan 1.0		
	Microarchitecture	Available since	GPUs (chips)	Graphic cards / SoCs	Linux	Microsoft Windows[a]	Android Nougat and later[46]
AMD	GCN 4th	June 2016	Polaris 10, Polaris 11, Vega 10	Radeon Rx 400 Series	AMDGPU PRO (beta, Ubuntu only)[47][48]	Radeon Software[49]	N/A
	GCN 3rd	August 2014	Tonga, Fiji, Carrizo	Radeon R9 Series			
	GCN 2nd	March 2013	Bonaire, Hawaii, Kaveri, Kabini, Temash, Mullins, Beema, Carrizo-L	Radeon HD 7790, PlayStation 4, Xbox One	Work in Progress		
	GCN 1st	February 2012	Oland, Cape Verde, Pitcairn, Tahiti	Radeon HD 77xx–7900 Series			
	TeraScale 3 "Northern Islands"	December 2010	Aruba (Trinity/Richland), Barts, Turks, Caicos, Cayman	Radeon HD 7xxx–76xx Series, Radeon HD 6000 Series	not supported		
	TeraScale 2 "Evergreens"	September 2009	Cedar, Cypress, Juniper, Redwood, Palm, Sumo	Radeon HD 5000 Series			
	TeraScale 1	May 2007	R600, RV630, RV610, RV790, RV770, …	Radeon HD 2000 Series, HD 3000, HD 4000			
Nvidia	Pascal	May 2016	GP100, GP104, GP106, GP107	GeForce 10 series, Tegra X2	Nvidia GeForce driver[50][51]	Nvidia GeForce driver[51]	Yes
	Maxwell 2.0	September 2014	GM200, GM204, GM206, GM20B	GeForce 900 series, Tegra X1			
	Maxwell 1.0	February 2014	GM107, GM108	GeForce GTX 750 Ti, GTX 750, GTX 860M			
	Kepler	March 2012	GK110, GK104, GK106, GK107, GK208	GeForce 600 series, GeForce 700 series, Tegra K1			
	Fermi	March 2010	GF100, GF104, GF106, GF108, GF110, GF114, GF116, GF119	GeForce 400 series, GeForce 500 series	not supported		
	Tesla 2.0	June 2008	GT200, GT215, GT216, GT218,	GeForce 200 series, GeForce 300 series			
	Tesla 1.0	November 2006	G80, G84, G86, G92, G94, G96, G98	GeForce 8 series, GeForce 9 series, GeForce 100 series			
	Kaby Lake	2016			Mesa[52][53]	Planned in 15.40.20 (https://software.intel.com/en-us/blogs/2016/03/14/new-intel-vulkan-beta-1540204404-graphics-driver-for-windows-78110-	
	Skylake	August 2015		Core i3-/i5-/i7-6000, Pentium G4xxx, Celeron G39xx			

Intel						1540)	N/A
	Broadwell	June 2015		Core i3-/i5-/i7-5000		not supported	
	Haswell	September 2012		Core i3-/i5-/i7-4000, Pentium G3xxx, Celeron G18xx			
	Ivy Bridge	April 2012		Core i3-/i5-/i7-3000, Pentium G2xxx, Celeron G16xx			
	Sandy Bridge	January 2011		Core i3-/i5-/i7-2000, Pentium Gxxx, Celeron Gxxx	not supported		
	Westmere	January 2010		Core i3-/i5-/i7-xxx, Pentium G69xx, Celeron G1101			
Imagination Technologies	PowerVR Series 8XE	February 2016	GE8200, GE8300		PowerVR Graphics SDK v4.1[54]		
	PowerVR Series 7XT	November 2014	GT7200, GT7400, GT7600, GT7800, GT7900	Apple A9, Helio X30 (MT679?)			
	PowerVR Series 7XE	November 2014	GE7400, GE7800				
	PowerVR Series 6XT	January 2014	GX6240, GX6250, GX6450, GX6650	Apple A8, A8X, MediaTek MT8173, MT8176,			
	PowerVR Series 6XE	January 2014	G6050, G6060, G6100 (XE), G6110	RK3368			
	PowerVR Series 6 (Rogue)	January 2012	G6100, G6200, G6230, G6400, G6430, G6630	MediaTek MT6595M, MT6595T, MT6595M, MT6795, MT8135, Helio X10 (MT6795), LG H13, Apple A7, Atom Z3460, Z3480, Z3530, Z3560, Z3570, Z3580			
	PowerVR Series 5XT	January 2009	SGX543, SGX544, SGX554	Apple S1, A5, A5X, A6, A6X, NovaThor L8540, L8580, L9540, TI OMAP 4470, 5430, 5432, MediaTek MT5327, MT6589M, MT6589T, MT6589, MT8117, MT8121, MT8125, MT8389, Atom Z2460, Z2520, Z2560, Z2580,	not supported		

				Z2760, Exynos 5410	
Qualcomm	Adreno 500 series		Adreno 510, Adreno 530	Snapdragon 430, 625, 650, 652, 820	**1.0**[55]
	Adreno 400 series			Snapdragon 415, 615, 616, 617, 805, 808, 810	
	Adreno 300 series			Snapdragon 200, 208, 210, 212, 400, 410, 412, 600, 800, 801	not supported
ARM	Bifrost[56]	June 2016	Mali-G71, …	TBA	**1.0**[57]
	Midgard 4th	Q4 2015	Mali-T860, Mali-T830, Mali-T880	Exynos 8890, Exynos 7870, Kirin 950, 955, MediaTek MT6738, MT6750, Helio X20 (MT6797), X25 (MT6797T), P10 (MT6755), P20 (MT6757)	
	Midgard 3rd	October 2013	Mali-T760, …	Exynos 7420, Exynos 5433, MT6752, MT6732, RK3288	
	Midgard 2nd	August 2012	Mali-T600 series, T720	Exynos 5250, 5260, 5410, 5420, 5422, 5430, 5800, 7580, Mediatek MT6735, MT6753, Kirin 920, 925, 930, 935	not supported

History

The Khronos Group began a project to create a next generation graphics API in July 2014 with a kickoff meeting at Valve Corporation.[58] At SIGGRAPH 2014, the project was publicly announced with a call for participants.[10]

According to the US Patent and Trademark Office, the trademark for Vulkan was filed on February 19, 2015.[59]

Vulkan was formally named and announced at Game Developers Conference 2015, although speculation and rumors centered around a new API existed beforehand and referred to it as "**glNext**".[60]

In early 2015, LunarG (funded by Valve) developed and showcased a Linux driver for Intel which enabled Vulkan compatibility on the HD 4000 series integrated graphics, despite the open source Mesa drivers not being fully compatible with OpenGL 4.0 until later that year.[61][62] There is still the possibility[63] of Sandy Bridge support, since it supports compute through Direct3D11.

On August 10, 2015, Google announced that future versions of Android would support Vulkan.[64] Android Nougat launched support for Vulkan on August 22, 2016.

On December 18, 2015, the Khronos Group announced that the 1.0 version of the Vulkan specification was nearly complete and would be released when conformant drivers were available.[12] The specification and the open-source Vulkan SDK were released on February 16, 2016.[6]

See also

- OpenGL – Another graphics API by the Khronos Group
- OpenCL – Another compute API by the Khronos Group
- Mantle – A low-level graphics and compute API from AMD, the foundation of Vulkan
- Direct3D – Windows and Xbox-only graphics API. Version 12 is a low-level API similar to Vulkan
- Metal – A low-level graphics and compute API for iOS and macOS
- AMDGPU – AMD's fully open-source unified graphics driver for Linux

Notes

a. Drivers so far have supported Windows 7 and later.

References

1. Hruska, Joel. "Not dead yet: AMD's Mantle powers new Vulkan API, VR efforts". *Extreme Tech*. Retrieved 26 June 2015.
2. Kirsch, Nathan. "Is AMD Mantle Dead As We Have Known It? Vulcan API Uses Mantle Technology for OpenGL". *Legit Reviews*. Retrieved 25 June 2015.
3. Shilov, Anton. "AMD: Vulkan absorbed 'best and brightest' parts of Mantle". *KitGuru*. Retrieved 26 June 2015.
4. Koduri, Raja (4 March 2015). "On APIs and the future of Mantle". AMD. Retrieved 19 May 2015. "...(T)he Khronos Group has selected Mantle to serve as the foundation for Vulkan..."
5. Michaud, Scott (3 March 2015). "GDC 15: Khronos Acknowledges Mantle's Start of Vulkan". PC Perspective. Retrieved 19 May 2015.
6. https://www.khronos.org/news/press/khronos-releases-vulkan-1-0-specification
7. https://github.com/KhronosGroup/Vulkan-Docs/blob/1.0/ChangeLog.txt
8. Bright, Peter; Walton, Mark (16 February 2016). "Vulkan now official, with 1.0 API release and AMD driver [Updated]". United Kingdom: Ars Technica. Retrieved 18 February 2016.
9. Valich, Theo (17 February 2016). "Mantle Cycle is Complete as Khronos Releases Vulkan 1.0". VR World. Retrieved 19 February 2016.
10. "More on Vulkan and SPIR - V: The future of high-performance graphics" (PDF). Khronos Group. p. 10. Retrieved 27 June 2015. "Thanks AMD!"
11. "Vulkan : Graphics and compute Belong Together" (PDF). Khronos.org. March 2015. Retrieved 2015-03-05.
12. "Vulkan - Graphics and compute belong together". Khronos.org. Retrieved 2015-03-05.
13. Batchelor, James (3 March 2015). "glNext revealed as Vulkan graphics API". *develop-online.net*.
14. Mah Ung, Gordon (6 March 2015). "Mantle is a Vulkan: AMD's dead graphics API rises from the ashes in OpenGL's successor". PCWorld.
15. "AMD Gaming: One of Mantle's Futures: Vulkan | AMD Blogs". Community.amd.com. Retrieved 2015-03-05.
16. Hruska, Joel (4 March 2015). "Not dead yet: AMD's Mantle powers new Vulkan API, VR efforts". ExtremeTech. Retrieved 2015-03-05.
17. "AMD's Mantle Lives On In Vulkan - Lays The Foundation For The Next OpenGL". Wccftech.com. 2014-06-20. Retrieved 2015-03-05.
18. Kirsch, Nathan. "Is AMD Mantle Dead As We Have Known It? Vulcan API Uses Mantle Technology for OpenGL". Legit Reviews. Retrieved 2015-03-05.
19. Hruska, Joel. "Next-generation Vulkan API could be Valve's killer advantage in battling Microsoft". *Extreme Tech*. Retrieved 26 June 2015.

20. http://www.tomshardware.com/news/khronos-group-vulkan-graphics-api,28678.html
21. http://blog.imgtec.com/powervr/vulkan-high-efficiency-on-mobile
22. http://blog.imgtec.com/powervr/vulkan-scaling-to-multiple-threads
23. Kessenich, John. "An Introduction to SPIR-V" (PDF). Khronos Group. Retrieved 5 March 2015.
24. "FOSDEM 2016 - Vulkan in Open-Source". *fosdem.org*. Retrieved 2016-02-27.
25. http://www.pcworld.com/article/3101843/software-games/vulkan-next-will-bring-better-support-for-vr-and-multiple-gpus.html
26. http://www.anandtech.com/show/10067/ashes-of-the-singularity-revisited-beta/4
27. Williams, Daniel; Smith, Ryan (17 February 2016). "Quick Look: Vulkan Performance on The Talos Principle". Anandtech. Retrieved 19 February 2016.
28. "Dota 2 Update - May 23rd 2016". *store.steampowered.com*. Valve. Retrieved 24 May 2016.
29. "DOOM – Vulkan Support Now Live". *bethesda.net*. Retrieved 2016-07-11.
30. Larabel, Michael (25 July 2016). "Quake 1 Ported To Run On Vulkan". *Phoronix*. Phoronix Media. Retrieved 26 August 2016.
31. Papadopoulos, John (21 July 2016). "Here is your first look at the first Quake being recreated in Unreal Engine 4 (fan project)". Dark Side of Gaming. Retrieved 26 August 2016.
32. Kollar, Philip (3 March 2015). "Valve announces Source 2 engine, free for developers". Retrieved 3 March 2015.
33. Mahardy, Mike (3 March 2015). "GDC 2015: Valve Announces Source 2 Engine". IGN. Retrieved 3 March 2015.
34. "The Talos Principle will support Vulkan". 4 Feb 2016.
35. "Epic Games adds Vulkan support to Unreal Engine 4". *bit-tech*. Retrieved 2016-02-24.
36. "Epic Games Unveils ProtoStar at Samsung Galaxy Unpacked". *www.unrealengine.com*. Retrieved 2016-02-24.
37. "id announces Doom running on Vulkan". 7 May 2016.
38. http://www.tweaktown.com/news/53506/directx-12-vulkan-support-way-cryengine/index.html
39. https://blogs.unity3d.com/2016/05/19/make-your-daydreams-a-reality/
40. "Xenko 1.7β released!". 1 Jul 2016.
41. "Vulkan 1.0 specification released with day-one support for Wayland". 2016-02-16.
42. "Vulkan Overview" (PDF). Khronos Group. June 2015. Retrieved 18 August 2015. p. 19 "Vulkan Status"
43. http://arstechnica.com/gadgets/2016/04/android-ns-second-preview-build-supports-vulkan-and-new-emoji/
44. "Final Developer Preview before Android 7.0 Nougat begins rolling out". 2016-07-19.
45. "MoltenVK". *Molten*. Retrieved 5 April 2016.
46. "The Android platform includes an Android-specific implementation of the Vulkan API specification from the Khronos Group". 2016-04-05.
47. "AMD Radeon Software AMD GPU-PRO Beta Driver Linux for Vulkan". 2016-03-18.
48. https://community.amd.com/community/gaming/blog/2016/02/16/radeon-gpus-are-ready-for-the-vulkan-graphics-api
49. "AMD Radeon Software version 16.15.1009 supports Vulkan". 2016-02-16.
50. http://www.nvidia.com/download/driverResults.aspx/101818/en-us
51. "Vulkan Driver Support". *NVIDIA Developer*. Nvidia. Retrieved 2016-04-04.
52. "Open-source Vulkan drivers for Intel hardware". 2016-02-16.
53. Larabel, Michael (8 July 2016). "Mesa 12.0 Released With OpenGL 4.3 Support, Intel Vulkan & Many Other Features". *Phoronix*. Retrieved 9 July 2016.
54. "Imagination announces Vulkan SDK for PowerVR Rogue GPUs". *Imagination Blog*. PowerVR Developer Technology Team. 16 March 2016. Retrieved 27 April 2016.
55. "Qualcomm announces Vulkan API support for Adreno 5xx and 4xx GPUs".
56. "ARM Bifrost GPU Architecture". 2016-05-30.
57. "ARM® Mali™ GPUs with Vulkan Conformance".
58. *SIGGRAPH 2015: 3D Graphics API State of the Union* (Video). SIGGRAPH 2015: Khronos Group. 16 September 2015. Event occurs at 57:24. Retrieved 12 November 2015.
59. "US Patent and Trademark Office". Retrieved 2015-03-07.
60. Batchelor, James. "glNext revealed as Vulkan graphics API | Latest news from the game development industry | Develop". Develop-online.net. Retrieved 2015-03-05.
61. "Valve Developed An Intel Linux Vulkan GPU Driver - Phoronix". *www.phoronix.com*. Retrieved 2015-07-22 .

62. "Learning More About The Intel Vulkan Driver, Linux Vulkan Plans - Phoronix". *www.phoronix.com*. Retrieved 2015-07-22.
63. "Evan Odabashian on Twitter". Retrieved 2015-07-22.
64. Woods, Shannon (12 August 2015). "Low-overhead rendering with Vulkan". *Android Developers Blog*.

External links

- Vulkan 1.0 specification (https://www.khronos.org/registry/vulkan/specs/1.0/xhtml/vkspec.html)

Retrieved from "https://en.wikipedia.org/w/index.php?title=Vulkan_(API)&oldid=739449229"

Categories: 2015 software | 3D graphics APIs | Cross-platform software | Graphics libraries | Graphics standards | Video game development | Virtual reality | Vulkan (API)

- This page was last modified on 14 September 2016, at 19:12.
- Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.