



**K H R O N O S**<sup>™</sup>  
G R O U P

## Khronos Overview

GDC | March 2016

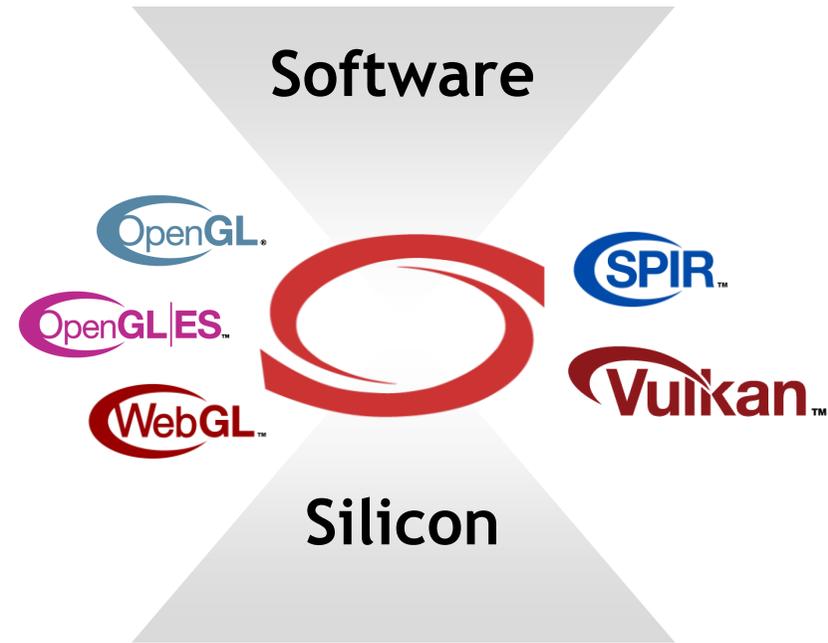
Neil Trevett | Khronos President

NVIDIA Vice President Developer Ecosystem

[ntrevett@nvidia.com](mailto:ntrevett@nvidia.com) | [@neilt3d](https://twitter.com/neilt3d)

# Khronos Gaming API Update

- Vulkan 1.0 launched on 16<sup>th</sup> February 2016
  - The first new generation open standard graphics API since OpenGL - 25 years ago
- WebGL 2.0 is getting close
  - OpenGL ES 3.0 capability to the Web
  - Aiming for finalization in 2Q16
- glTF 1.0 momentum is building
  - Transmitting 3D assets to online Web, AR and VR experiences



Khronos is an Industry Consortium of over 100 companies creating **OPEN STANDARD APIs** to enable software to access hardware acceleration for **graphics, parallel compute and vision**

# The Need for a New Generation GPU API

- **Explicit**
  - Open up the high-level driver abstraction to give direct, low-level GPU control
- **Streamlined**
  - Faster performance, lower overhead, less latency
- **Portable**
  - Cloud, desktop, console, mobile and embedded
- **Extensible**
  - Platform for rapid innovation



OpenGL has evolved over 25 years and continues to meet industry needs - but there is a need for a complementary API approach



GPUs are increasingly programmable and compute capable + platforms are becoming mobile, memory-unified and multi-core



GPUs will accelerate graphics, compute, vision and deep learning across diverse platforms: **FLEXIBILITY** and **PORTABILITY** are key

# Next Generation GPU APIs



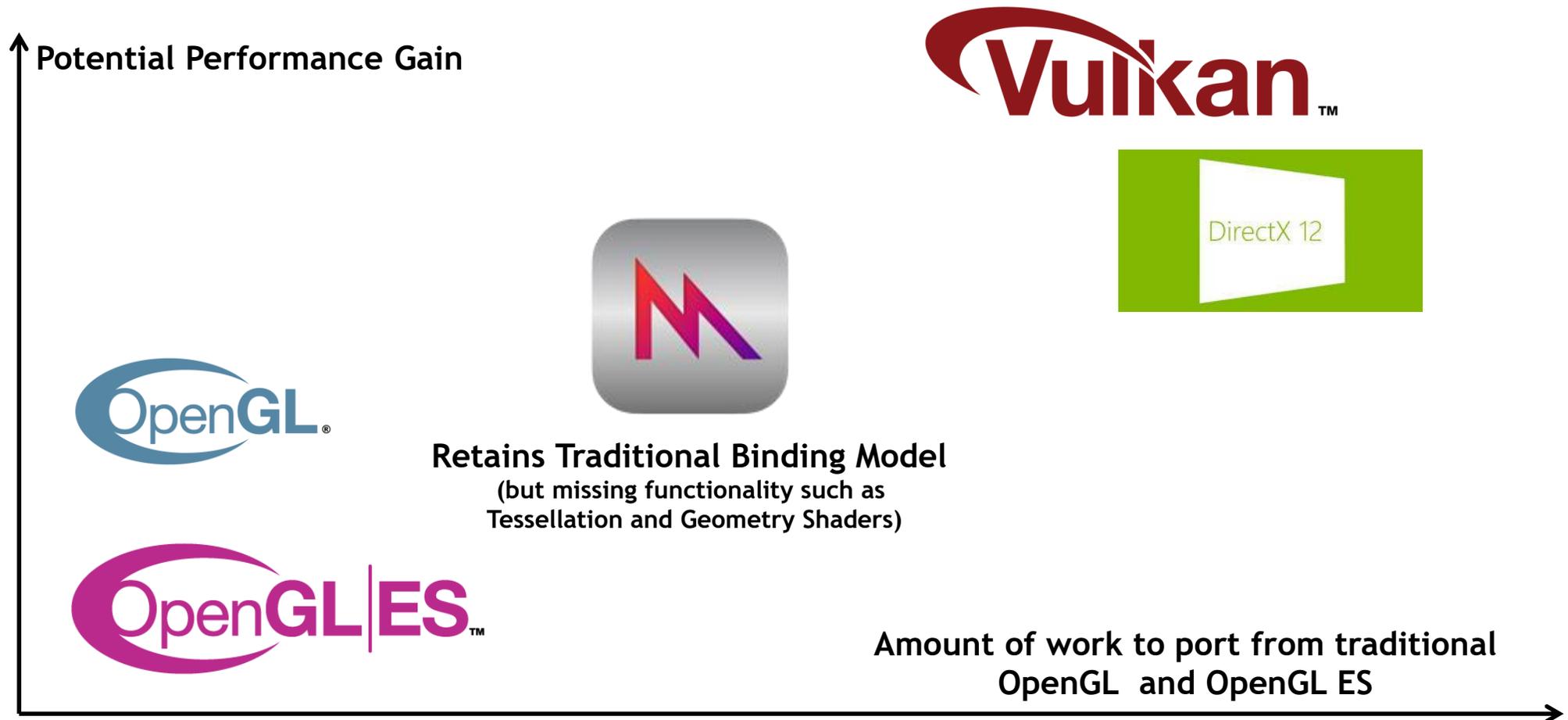
Only Windows 10



Only Apple



# Vulkan - No Compromise Performance



# Vulkan Genesis



Khronos' first API  
'hard launch'  
16Feb16

Khronos members from all segments of the graphics industry agree the need for new generation cross-platform GPU API

Significant proposals, IP contributions and engineering effort from many working group members

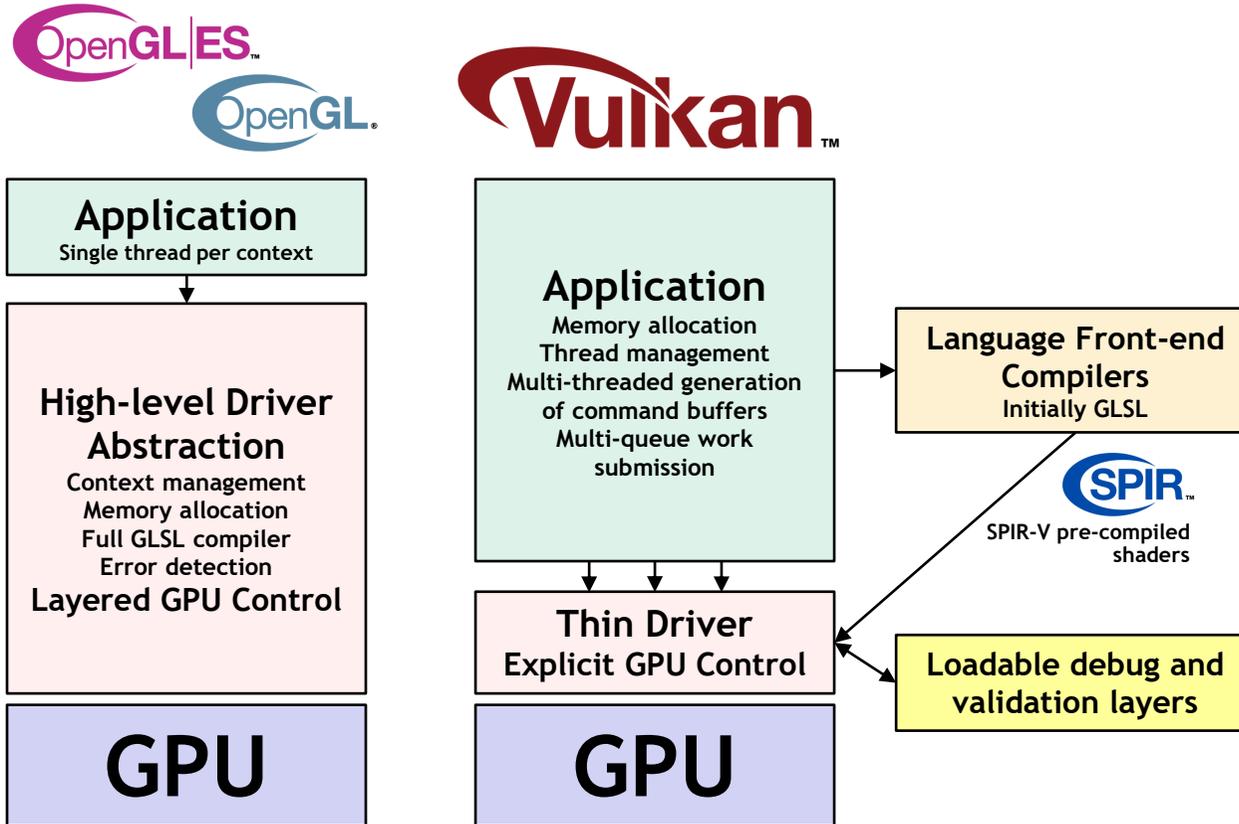
Including an unprecedented level of participation from game engine developers

**18 months**  
A high-energy working group effort

Specification, Conformance Tests, SDKs - all open source...  
Reference Materials, Compiler front-ends, Samples...  
Multiple Conformant Drivers on multiple OS



# Vulkan Explicit GPU Control

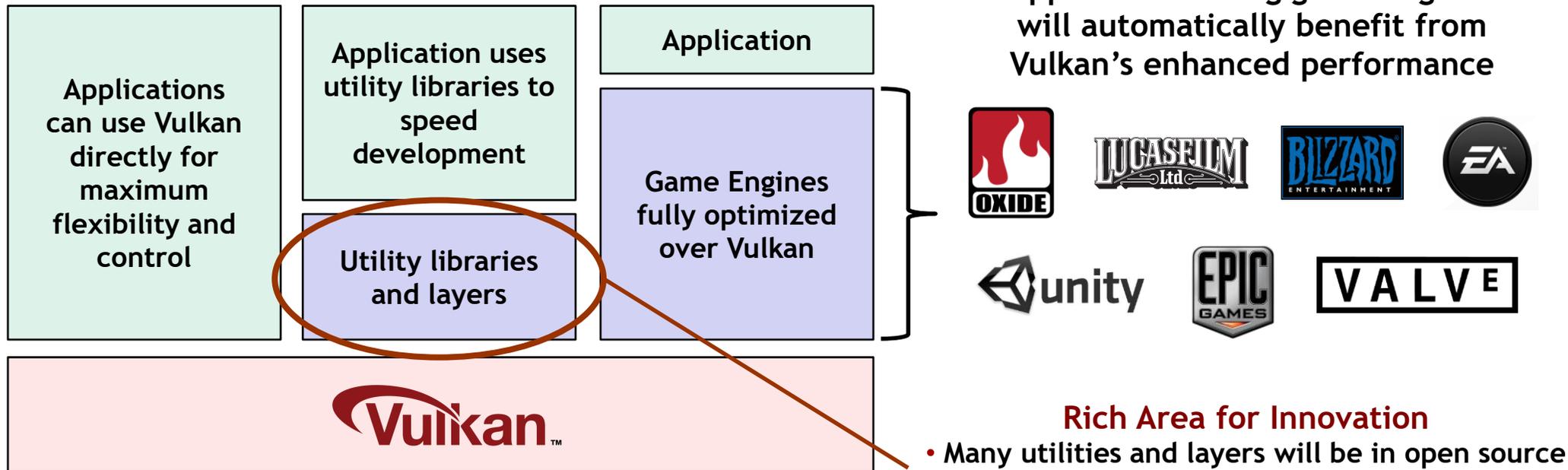


Vulkan 1.0 provides access to OpenGL ES 3.1 / OpenGL 4.X-class GPU functionality but with increased performance and flexibility

## Vulkan Benefits

- Simpler drivers:**  
 Improved efficiency/performance  
 Reduced CPU bottlenecks  
 Lower latency  
 Increased portability
- Resource management in app code:**  
 Less hitches and surprises
- Command Buffers:**  
 Command creation can be multi-threaded  
 Multiple CPU cores increase performance
- Graphics, compute and DMA queues:**  
 Work dispatch flexibility
- SPIR-V Pre-compiled Shaders:**  
 No front-end compiler in driver  
 Future shading language flexibility
- Loadable Layers**  
 No error handling overhead in production code

# The Power of a Three Layer Ecosystem

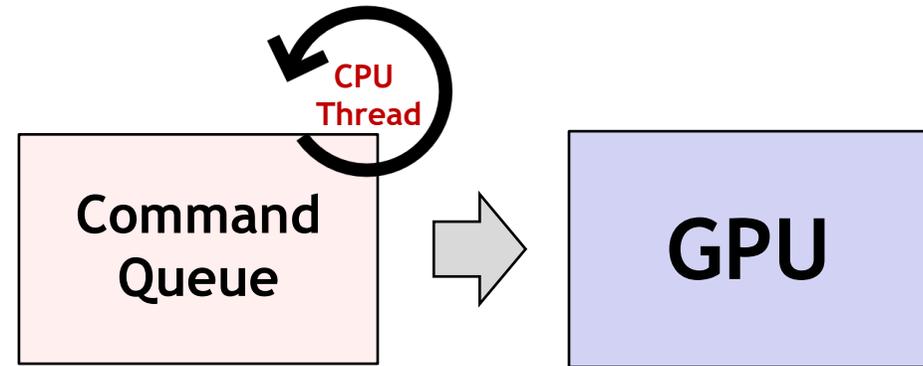
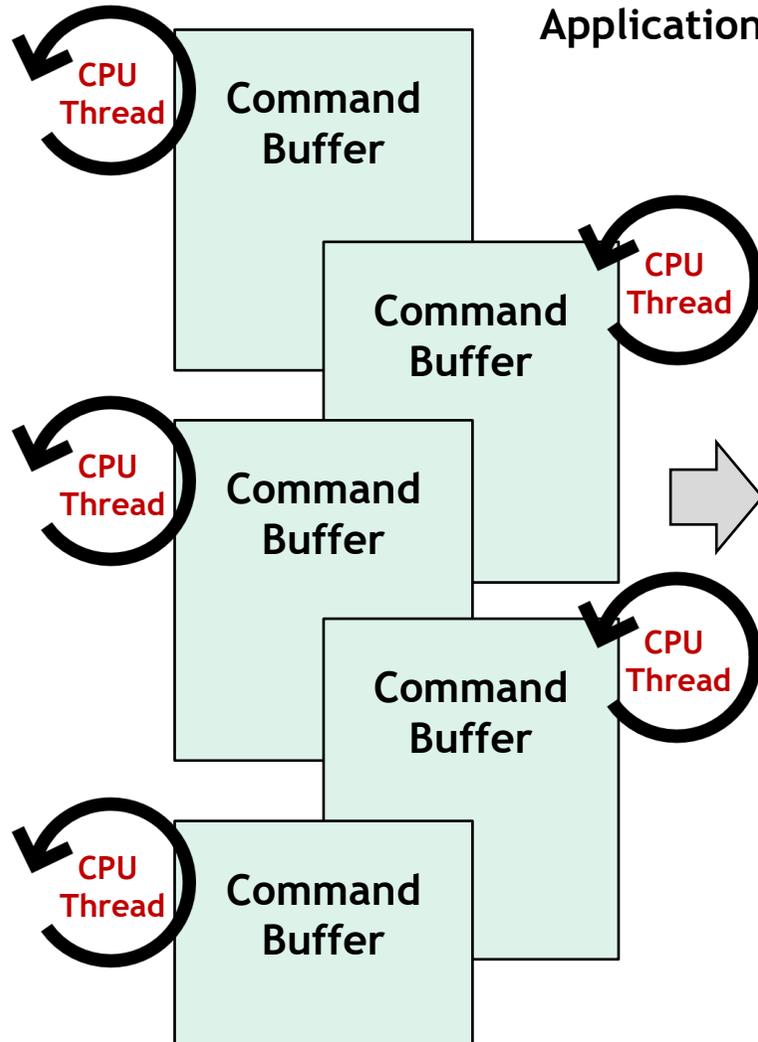


Similar ecosystem dynamic as WebGL

A widely pervasive, powerful, flexible foundation layer enables diverse middleware tools and libraries

# Vulkan Multi-threading Efficiency

1. Multiple threads can construct Command Buffers in parallel  
Application is responsible for thread management and synch

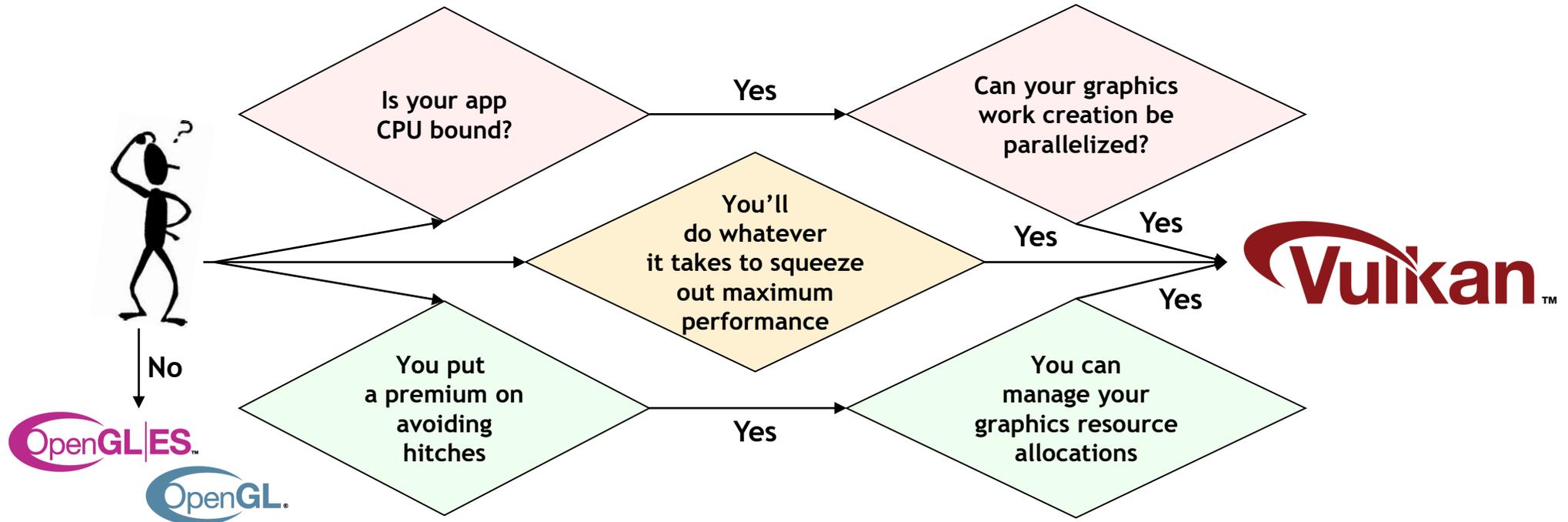


2. Command Buffers placed in Command Queue by separate submission thread

Can create graphics, compute and DMA command buffers with a general queue model that can be extended to more heterogeneous processing in the future

# Which Developers Should Use Vulkan?

- Vulkan puts more work and responsibility into the application
  - Not every developer will need or want to make that extra investment
- For many developers OpenGL and OpenGL ES will remain the most effective API
  - Khronos actively evolving OpenGL and OpenGL ES in parallel with Vulkan



Vulkan provides more choice to developers and can be used to create new classes of end-user experience

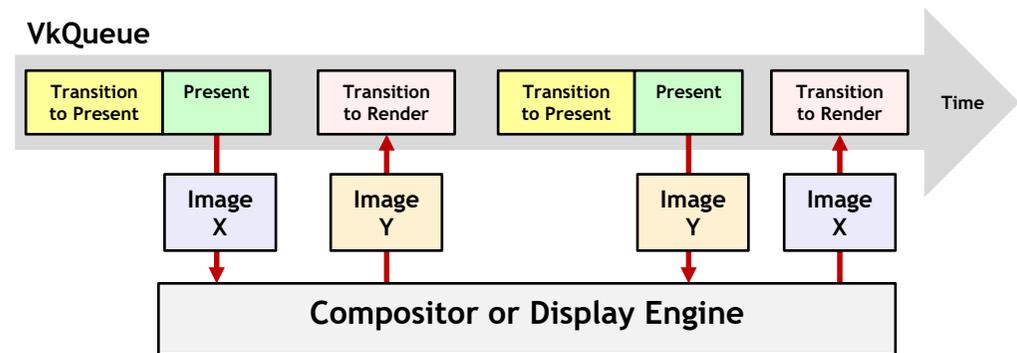
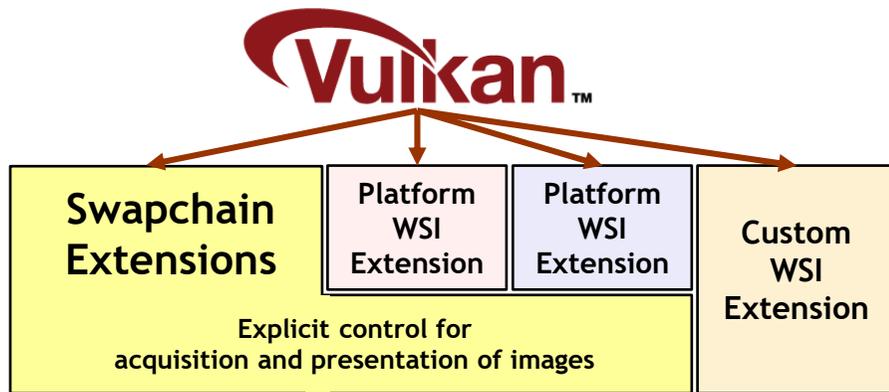
# Vulkan Feature Sets

- **Vulkan supports hardware with a wide range of hardware capabilities**
  - Mobile OpenGL ES 3.1 up to desktop OpenGL 4.5 and beyond
- **One unified API framework for desktop, mobile, console, and embedded**
  - No "Vulkan ES" or "Vulkan Desktop"
- **Vulkan precisely defines a set of "fine-grained features"**
  - Features are specifically enabled at device creation time (similar to extensions)
- **Platform owners define a Feature Set for their platform**
  - Vulkan provides the mechanism but does not mandate policy
  - Khronos will define Feature Sets for platforms where owner is not engaged
- **Khronos will define feature sets for Windows and Linux**
  - After initial developer feedback



# Vulkan Window System Integration (WSI)

- **Explicit control for acquisition and presentation of images**
  - Designed to fit the Vulkan API and today's compositing window systems
  - Cleanly separates device creation from window system
- **Platform provides an array of persistent presentable images = Vulkan Swapchain**
  - Device exposes which queues support presentation
  - Application explicitly controls which image to render and present
- **Standardized extensions - unified API for multiple window systems**
  - Works across Android, Mir, Windows (Vista and up), Wayland and X (with DRI3)
  - Platforms can extend functionality, define custom WSI stack, or have no display at all



# SPIR-V Transforms the Language Ecosystem

- First multi-API, intermediate language for parallel compute and graphics
  - Native representation for Vulkan shader and OpenCL kernel source languages
  - <https://www.khronos.org/registry/spir-v/papers/WhitePaper.pdf>
- GL\_KHR\_vulkan\_glsl spec released - adds the GLSL features needed for Vulkan
  - Descriptor sets, push constants, specialization constants
  - Separate images/samplers, sub pass input images...
  - Updated front-end open source compiler in Khronos GitHub

## Multiple Developer Advantages

Same front-end compiler for multiple platforms  
Reduces runtime kernel compilation time  
Don't have to ship shader/kernel source code  
Drivers are simpler and more reliable

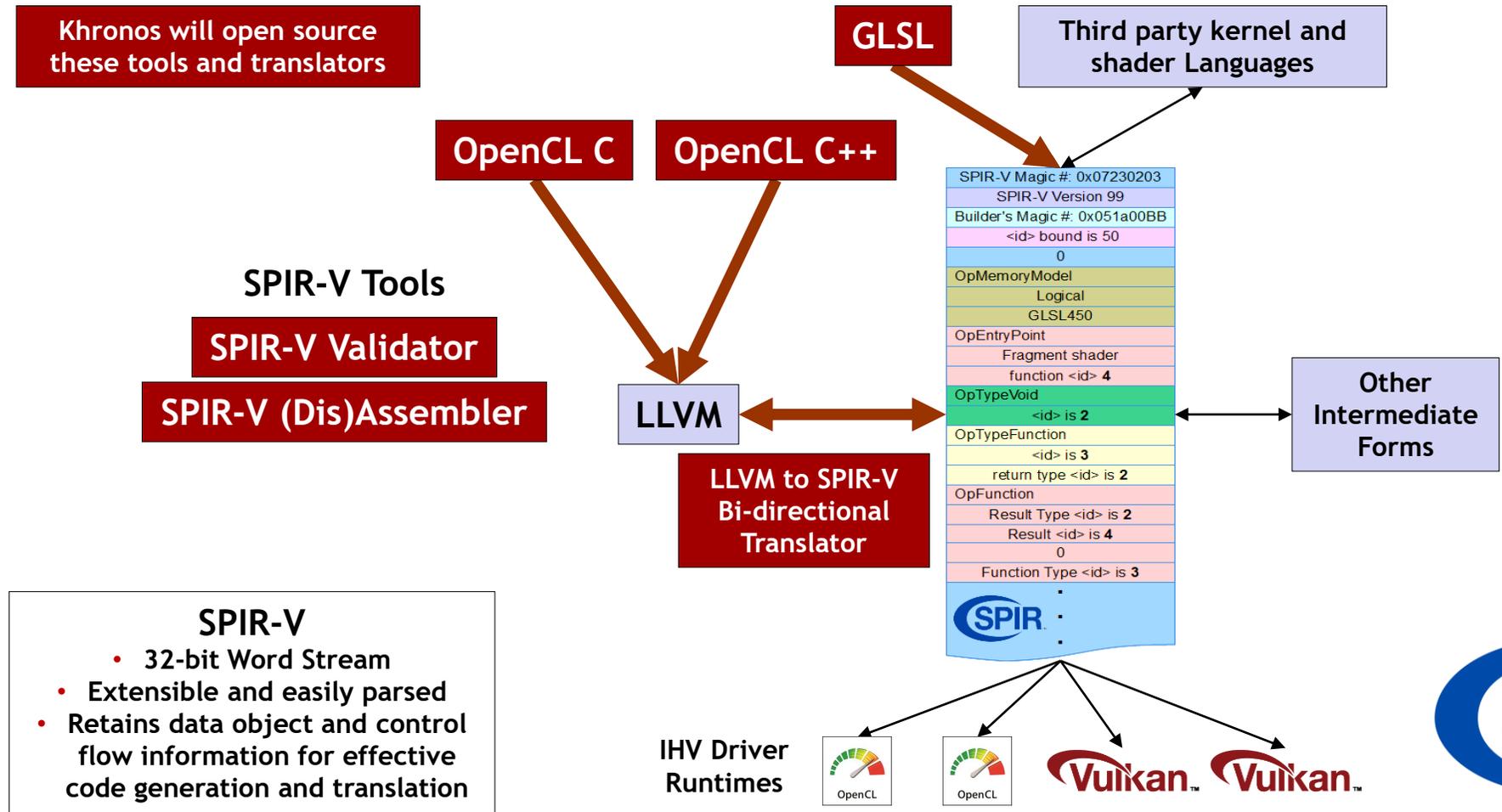


# Evolution of SPIR Family

- SPIR-V is first fully specified Khronos-defined SPIR standard
  - Does not use LLVM to isolate from LLVM roadmap changes
  - Includes full flow control, graphics and parallel constructs beyond LLVM
  - Khronos will open source SPIR-V <-> LLVM conversion tools

	SPIR 1.2	SPIR 2.0	SPIR-V 1.0
LLVM Interaction	Uses LLVM 3.2	Uses LLVM 3.4	100% Khronos defined Round-trip lossless conversion
Compute Constructs	Metadata/Intrinsics	Metadata/Intrinsics	Native
Graphics Constructs	No	No	Native
Supported Language Feature Set	OpenCL C 1.2	OpenCL C 1.2 OpenCL C 2.0	OpenCL C 1.2 / 2.0 OpenCL C++ GLSL
OpenCL Ingestion	OpenCL 1.2 Extension	OpenCL 2.0 Extension	OpenCL 2.1 Core
Vulkan Ingestion	-	-	Vulkan Core

# Driving the SPIR-V Open Source Ecosystem



# Vulkan Developer Resources at Launch



**Khronos.org**  
**Canonical Resources**  
 Specifications, Header Files  
 Feature Set Definitions  
 (Windows and Linux - post developer feedback)  
 Quick Reference and Reference Pages  
 Conformance Test Source and Test Process  
**Materials to Build SDKs and Tools**  
 Compiler toolchain sources  
 Validation Layer Source  
 Loader Source  
 Layers and Loader documentation  
 (open source resources in [github.com/KhronosGroup](https://github.com/KhronosGroup))

**VULKAN DRIVERS**

Behind every great API are the drivers that bring it life on your GPU. Download the latest drivers for your system that now include Vulkan 1.0.



Everything needed to create SDKs for any platform or market

**LunarG**  
 Windows and Linux Installable SDKs  
 Loader and Validation Layer binaries  
 Tools Layers - source and binaries  
 Samples - source and binaries  
 Windows get started guide

**IHV Websites**  
 Drivers and Loader  
 Vendor tools and layers

**Third Party Websites**  
 Layers, Samples etc.

**DEMOS AND SAMPLES**

Download demos and open source samples to take your new Vulkan API for a test drive - and get a heads up on Vulkan resources which will be arriving soon...



**GET INVOLVED! HELP US EVOLVE THE VULKAN ECOSYSTEM**

Khronos has placed an unprecedented amount of materials into open source so you can provide feedback, showcase your work, fix bugs, and extend Vulkan capabilities for the future. Get engaged AND show the world what YOU are doing with Vulkan.

Vulkan resources on Github

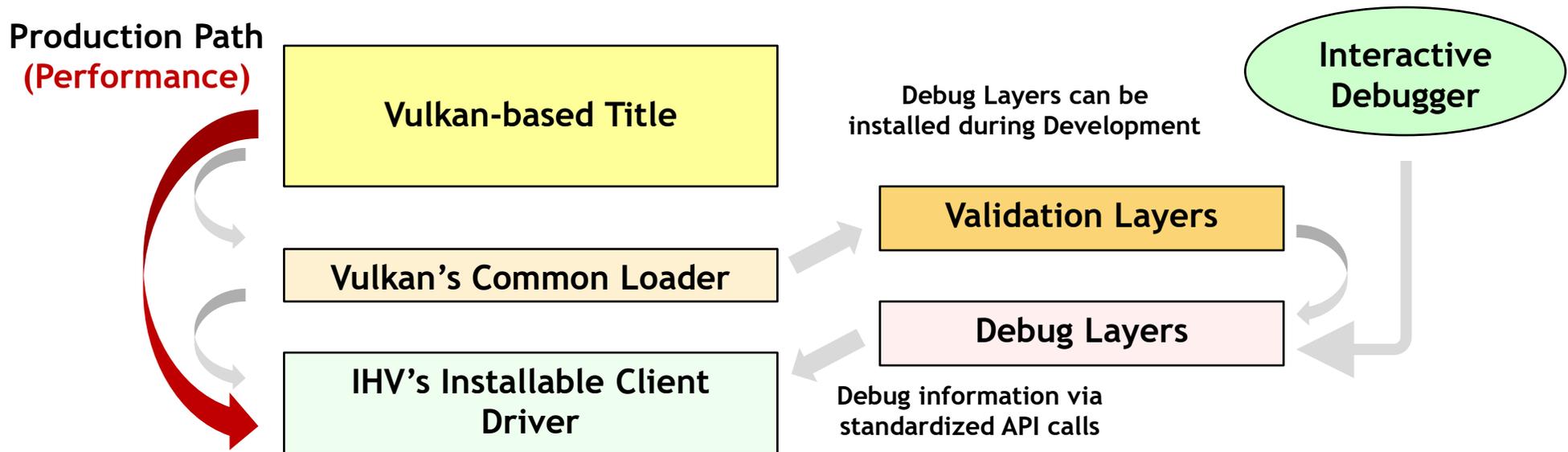
**Discussions**

Issue Trackers:

- Vulkan Specification, Reference Pages, and API Registry
- Vulkan CTS
- Vulkan Loader and Validation Tools
- Vulkan Sample Code
- Data Format specification

# Vulkan Tools Architecture

- Layered design for cross-vendor tools innovation and flexibility
  - IHVs plug into a common, extensible architecture for code validation, debugging and profiling during development without impacting production performance
- Khronos Open Source Loader enables use of tools layers during debug
  - Finds and loads drivers, dispatches API calls to correct driver and layers



# LunarG SDK for Vulkan

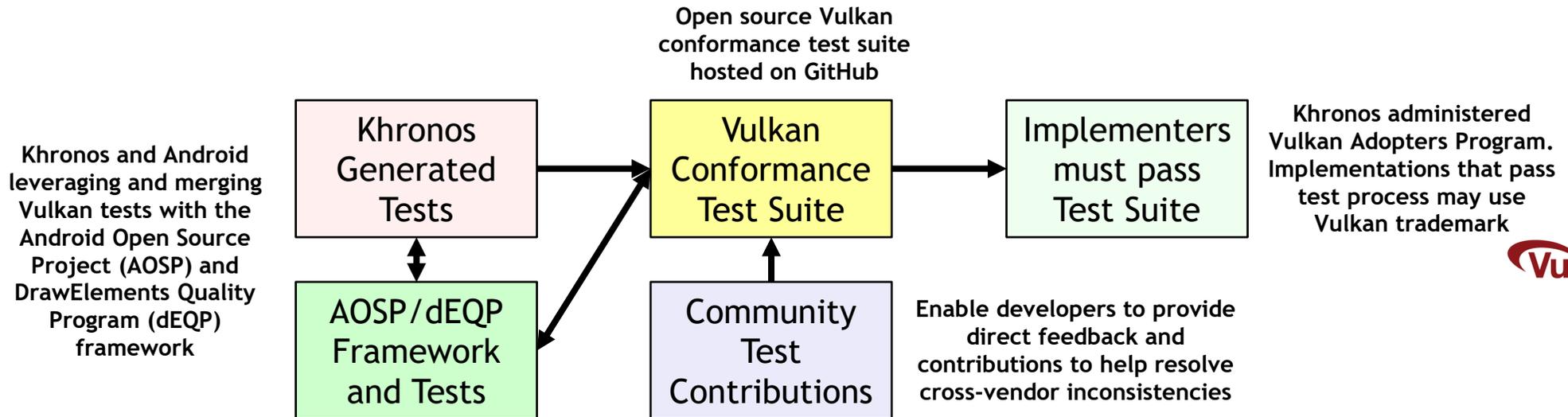
- Valve sponsored LunarG to develop a free, open source SDK for Vulkan
  - Utilities, samples, debugging tools, documentation
  - For Windows and Linux on launch - Android coming soon
- Validation Layer - checks many aspects of Vulkan code:
  - Device limits, draw state, parameter values
  - Multi-thread object access rules, texture and render target formats
  - Object Tracker, Memory Tracker
- Other SDK Tools
  - Trace and replay tools
  - GLSL Validator
  - SPIR-V Disassembler and Assembler
- RenderDoc Graphics debugger
  - Free and open source
  - Adding Vulkan support
  - <https://github.com/baldurk/renderdoc>

[vulkan.lunarg.com](http://vulkan.lunarg.com)



# Conformant Vulkan Drivers

- 30 Driver submissions passed conformance at Vulkan 1.0 launch
  - [www.khronos.org/conformance/adopters/conformant-products](http://www.khronos.org/conformance/adopters/conformant-products)
  - AMD: Linux and Windows
  - ARM: Linux
  - Imagination Technologies: Linux
  - Intel: Linux, Windows
  - NVIDIA: Android 6.0, Linux (desktop and embedded), Windows 7-10
  - Qualcomm: Android 6.0



# Vulkan Ecosystem Active at Launch



THE BRENWILL WORKSHOP  
Graphics Technology Expertise

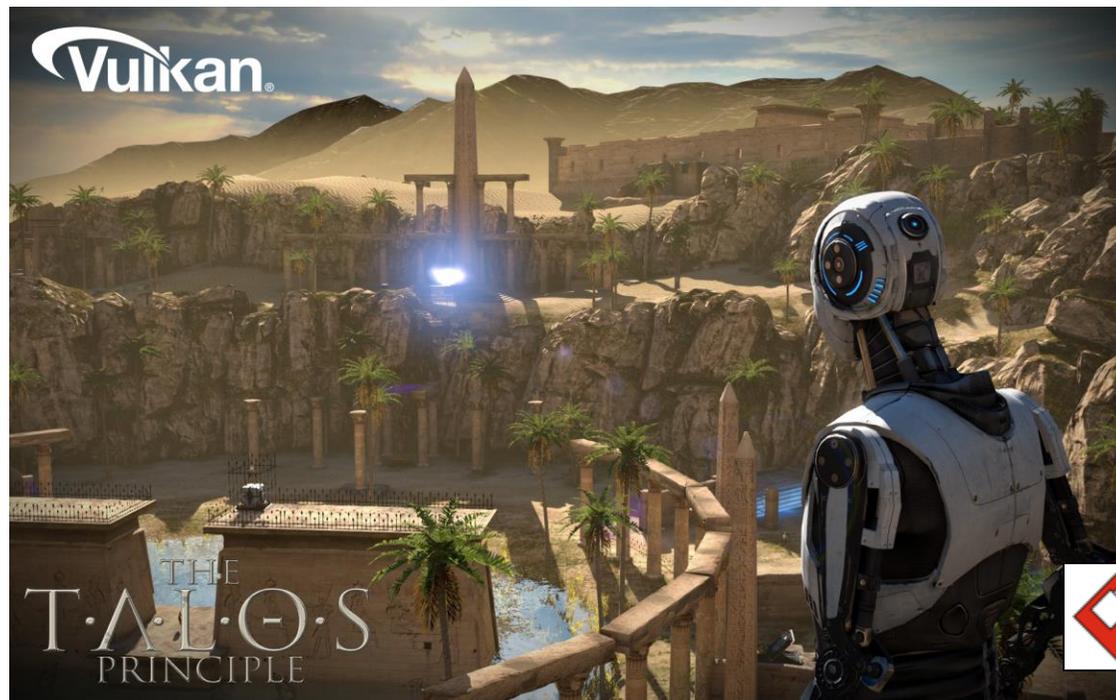
“By building your application or game using the Vulkan API, you can run your modern graphics application or game unchanged across an entire industry of platforms and development tools”  
Brenwill Workshop



Vulkan and OpenGL ES  
over Metal - in development

“Vulkan has a huge potential! We're only scratching the surface of what can be done with it, and porting The Talos Principle to Vulkan should be seen as a proof of concept,” said Dean Sekulic graphics engine specialist at Croteam.  
“Vulkan in just one sentence? The endless war between performance and portability is finally over!”

*Talos Principle on Steam has beta Vulkan back-end*



# One Week After Launch of Vulkan 1.0

<http://www.pccworld.com/article/3035020/linux/valves-steam-os-now-supports-vulkan-the-cross-platform-alternative-to-directx-12.html>

## Valve's SteamOS now supports Vulkan, the cross-platform alternative to DirectX 12

Nvidia leads the pack, with Intel and AMD not far behind, signalling a new dawn for Linux gaming.



Credit: Scott Robinson via Flickr/Creative Commons

<http://blogs.nvidia.com/blog/2016/02/22/shield-marshmallow-vulkan/>



SHIELD continues to set the standard for delivering product updates to consumers. With our latest update, we're delivering Android 6.0 Marshmallow to SHIELD Android TV. We're also making it the first Android consumer platform to ship with support for the new-generation Vulkan graphics API. Announced during CES last month, our latest over-the-air software upgrade, available now for SHIELD Android TV, brings Android 6.0 Marshmallow and a host of improvements and upgrades. In addition to delivering Marshmallow to the living room, it also continues NVIDIA's rollout of Vulkan drivers across multiple platforms, including Windows 7-10, Linux and now Android. The Vulkan API provides highly efficient, low-level access to modern graphics hardware, such as the Maxwell GPU in SHIELD Android TV. Vulkan has been created by experts from across the industry working together at the Khronos Group — an open standards consortium. Today, NVIDIA is equipping Android developers with fully conformant Vulkan drivers just one week after the specifications launched. In addition, Google has announced that Vulkan will be a core platform API in a future version of Android to drive new classes of interactive user experiences. More information is available on the Vulkan developer hub, where Vulkan Android samples can be downloaded.



SHIELD Android TV is fully conformant to both Khronos Vulkan CTS and Google Certification.

<http://www.androidcentral.com/vulkan-samsung-galaxy-s7-potentially-very-big-deal>

## Imagination introduces PowerVR Series8XE GPUs, brings OpenGL ES 3.2 and Vulkan to the ultra-affordable market

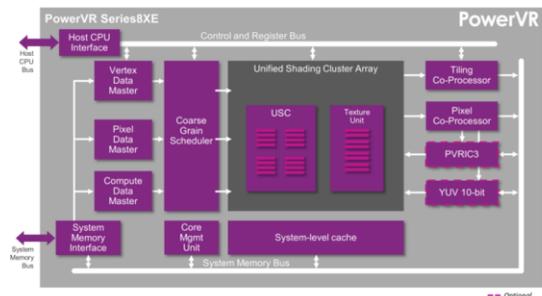
by PRIYAM NATH  
FEBRUARY 22, 2016

296  
Shares

f 261

Twitter icon

g+ 35



<http://www.androidauthority.com/powervr-series8xe-673991/>

## SOFTPEDIA DESKTOP MOBILE WEB NEWS

≡ Softpedia > News > Linux

### Ubuntu 16.04 LTS to Ship with Full Support for Vulkan in Mir Display Server

Canonical is jumping on the Vulkan train

Feb 17, 2016 18:49 GMT · By Silviu Stahie

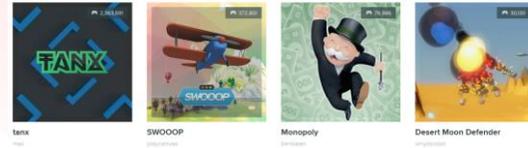
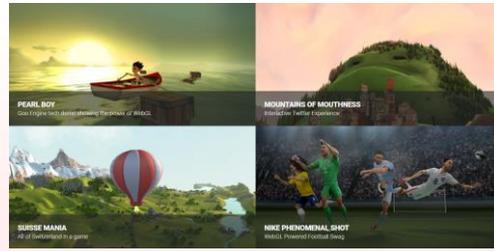
Ubuntu 16.04 LTS (Xenial Xerus) is going to integrate full support in Mir for the latest Vulkan 1.0 specifications.

Vulkan is stealing all the headlines in the Linux world and with good reason. It's an incredible leap forward for the open source platform, even if Vulkan is technically aimed at all the major operating systems, including Windows, Android, and even Tizen.

<http://news.softpedia.com/news/ubuntu-16-04-lts-to-ship-with-full-support-for-vulkan-in-mir-display-server-500543.shtml>

<https://www.youtube.com/watch?v=FnKu7MLB7VQ>

# WebGL Games Ecosystem



>1 Billion desktop and mobile browsers  
Including Facebook social gaming

PEX



xeoEngine



CopperLight



three.js

# WebGL 2.0

Fixed function Pipeline



Programmable Vertex and fragment shaders



32-bit integers and floats  
NPOT, 3D/depth textures  
Texture arrays  
Multiple Render Targets



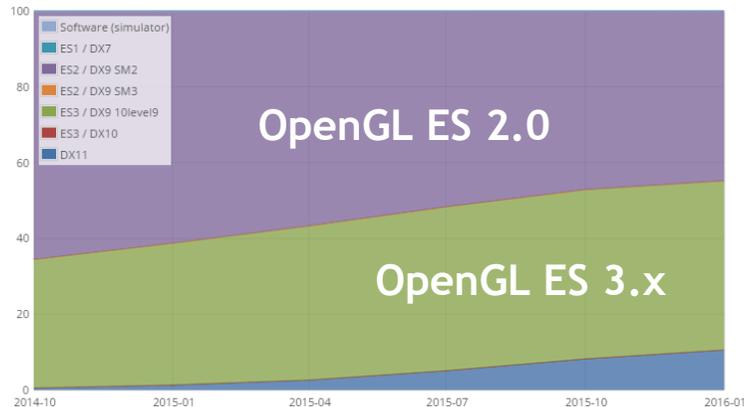
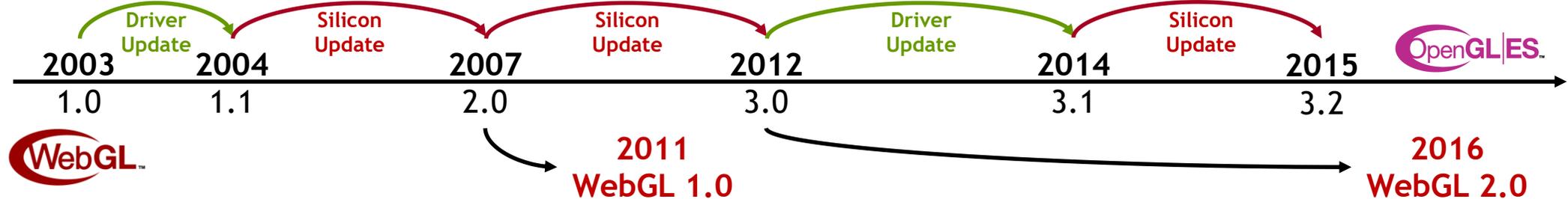
Compute Shaders



Tessellation and geometry shaders  
ASTC Texture Compression  
Floating point render targets  
Debug and robustness for security



Epic's Rivalry demo using full Unreal Engine 4  
<https://www.youtube.com/watch?v=jRr-G95GdaM>



<http://hwstats.unity3d.com/mobile/gpu.html>

## WebGL 2.0

Enhanced visual quality, performance

- Instancing | Multiple render targets | Uniform buffers
- Transform feedback | Multi-sampled Renderbuffers | 3D textures
- NPOT textures | More texture formats | Occlusion queries
- Vertex array objects | Sampler objects | Sync objects
- Fragment depth | Primitive restart | ...

# 3D Needs a Transmission Format!

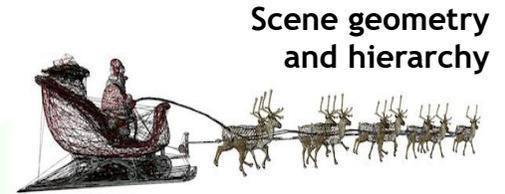
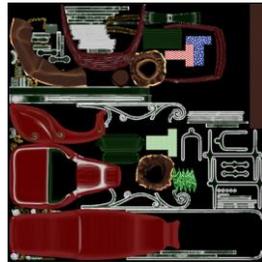
- Efficient Format to define, transmit and use 3D Assets
  - Rich scene/object representation with animations, materials - OBJ, STL etc. too limited
  - Common format for tools, engines to import and export
  - Publishing format for content repositories and services

Audio	Video	Images	3D
MP3	H.264	JPEG	?
			!

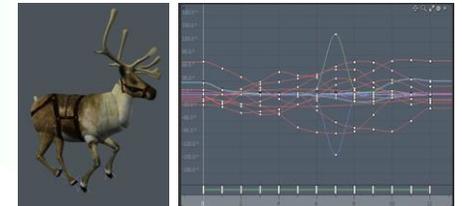
A widely adopted media format ignites previously untapped commercial opportunities



Materials and textures



Scene geometry and hierarchy



Animations and skins

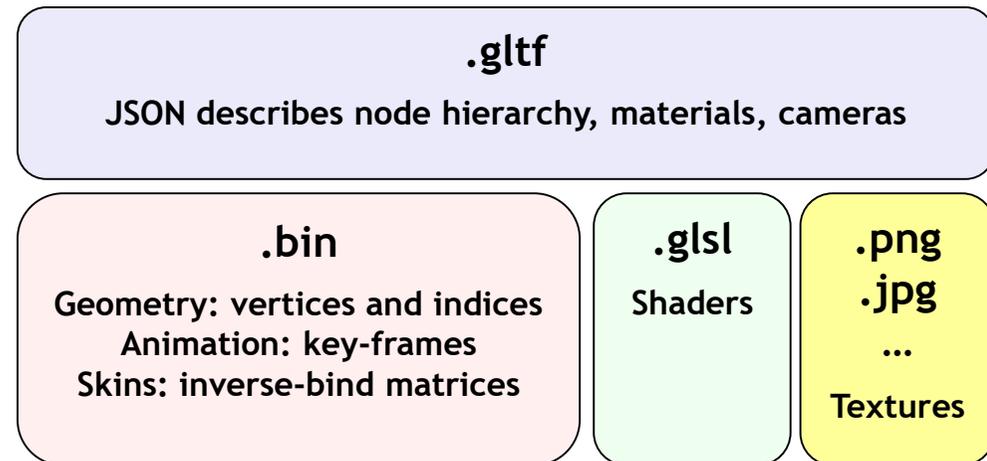
# glTF = “JPEG for 3D”

- ‘GL Transmission Format’
  - Runtime asset format for WebGL, OpenGL ES, and OpenGL applications
- **Compact representation for download efficiency**
  - Binary mesh and animation data
- **Loads quickly into memory**
  - GL native data types require no additional parsing
- **Full-featured**
  - 3D constructs (node hierarchy, materials, animation, cameras, lights)
- **Runtime Neutral**
  - Can be created and used by any tool, app, or runtime
- **Flexible Extensibility**
  - E.g. payloads with compression and streaming

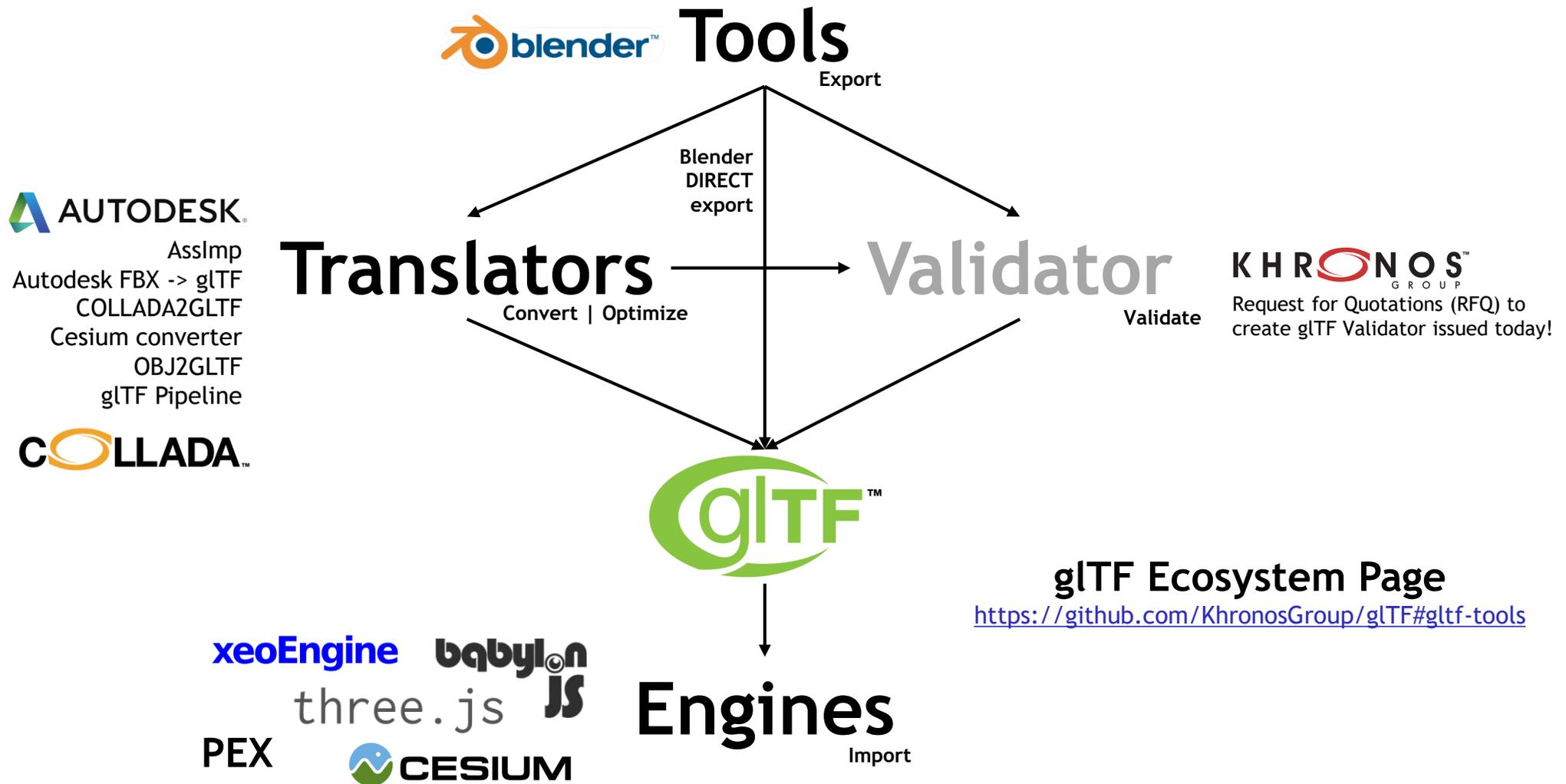


# glTF Internals

- **JSON describes node hierarchy**
  - Includes cameras
  - References geometry, animations, skins, shaders, textures
- **Vertices**
  - Uses native typed array format
  - Includes key-frame animations and skinning
- **Shaders**
  - With extensions for materials
- **Textures**
  - Use existing standard image compression formats e.g. JPEG
- **Extras**
  - For app-specific data (metadata)

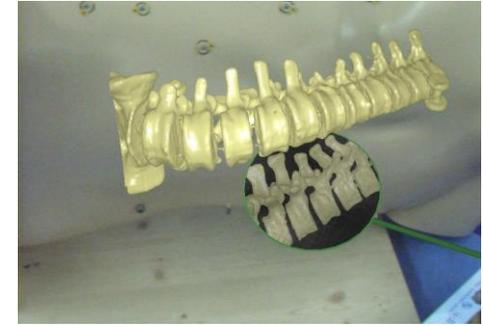
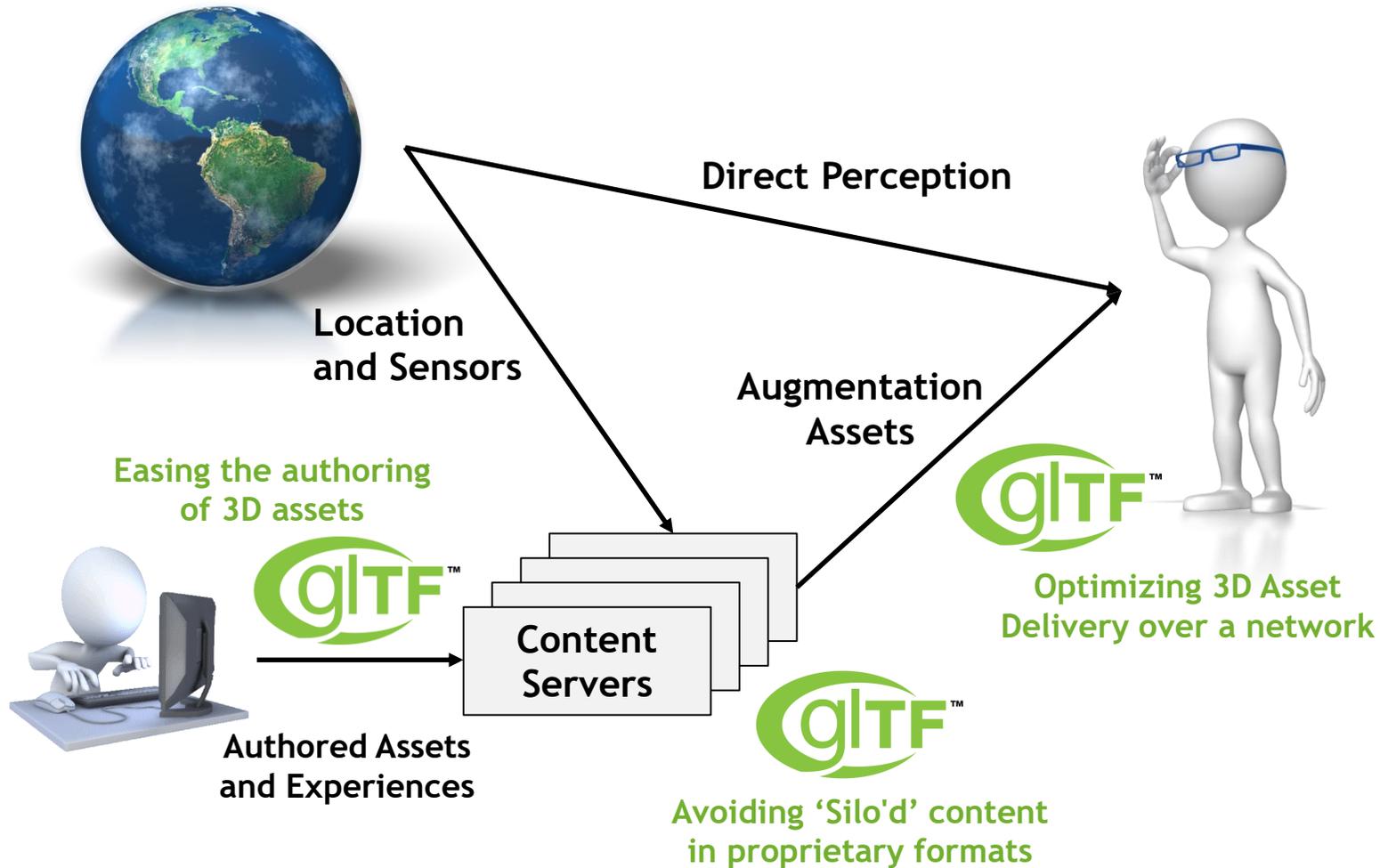


# glTF Ecosystem



glTF Ecosystem Page  
<https://github.com/KhronosGroup/glTF#gltf-tools>

# glTF for VR and AR



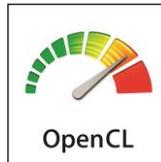
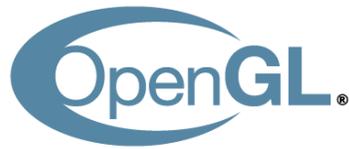
# glTF Validator RFQ! Consider Bidding!!

- **Validate glTF assets: geometry, material, animation, and skins**
  - Ensure compliance with spec and schema
- **Augment glTF sample models to cover corner cases**
- **Optional features**
  - Binary glTF, REST service, drag-and-drop validator, drag-and-drop three.js viewer
- **Schedule**
  - **March 15** - Khronos Released RFQ
  - **March 31** - RFQ responses received by Khronos
  - **April 15** - Contractor selected and notified
  - **April 29** - Contract executed and start of work
- <https://www.khronos.org/rfq>



# Khronos Roadmap Discussions

SPIR-V Ingestion for OpenGL and OpenGL ES for shading language flexibility

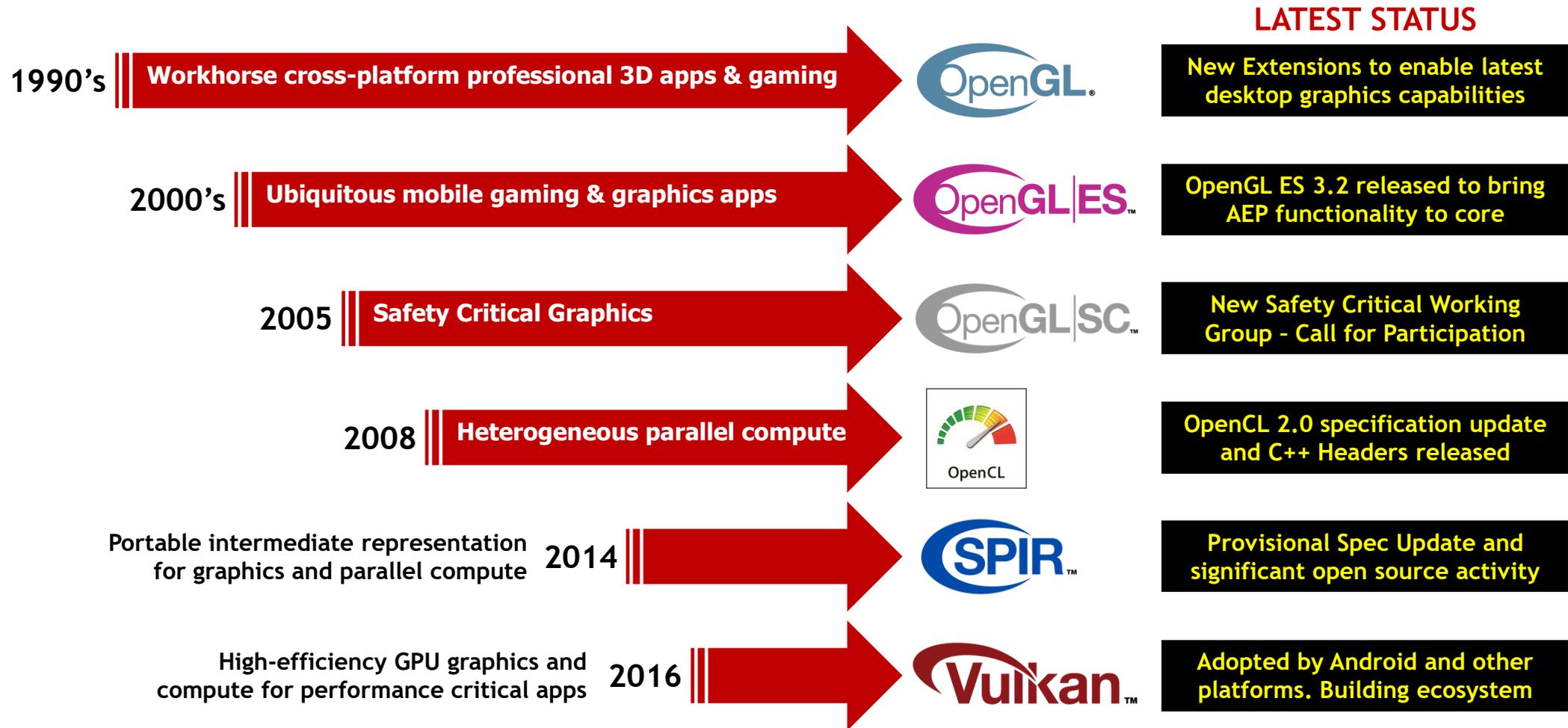


*Khronos members decide how to evolve and mix and match a rich set of APIs and technologies to meet market needs*

Bring new generation API capabilities to WebGL once they become pervasive

**OpenCL-class Heterogeneous Compute to Vulkan runtime:**  
C++ Shading Language  
Tiered precision  
Shared virtual memory  
Dynamic parallelism...

# Khronos Open Standards for Graphics and Compute



## LATEST STATUS

New Extensions to enable latest desktop graphics capabilities

OpenGL ES 3.2 released to bring AEP functionality to core

New Safety Critical Working Group - Call for Participation

OpenCL 2.0 specification update and C++ Headers released

Provisional Spec Update and significant open source activity

Adopted by Android and other platforms. Building ecosystem