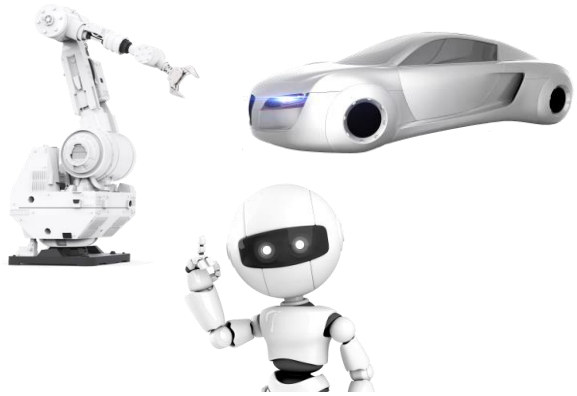
Open Standards: Powering the Future of Embedded Vision

Neil Trevett
President
The Khronos Group

Need for Embedded Acceleration Standards

Increasing Sensor Compute Load

Diverse camera and senor arrays feed sophisticated processing - including inferencing



Cost and time to integrate and utilize sensors, GPUs and processors in diverse markets has become a major constraint on innovation and efficiency

Advanced User Interfaces

High quality 3D graphics, augmented reality, diverse display systems

Open Standard APIs in Embedded Markets

- Enable cross-platform software reusability
- Decouple software and hardware for easier development and integration of new components
- Provide cross-generation reusability
- Facilitate field upgradability

Topics for this Session

Introduction to Khronos and open standard APIs for vision and compute acceleration

Khronos safety-critical APIs including Vulkan SC and the new SYCL SC Exploratory Forum

The new Khronos EMVA Camera Working Group

Details on how to get involved!

Khronos Connects Software to Silicon

Founded in 2000
~ 200 Members | ~ 40% US, 30% Europe, 30% Asia

KHRONOS® GROUP

Open, royalty-free interoperability standards to harness the power of GPU, XR and multiprocessor hardware

3D graphics, augmented and virtual reality, parallel programming, inferencing and vision acceleration

Non-profit, member-driven standards organization, open to any company

Proven multi-company governance and Intellectual Property Framework

Khronos Active Standards

3D Graphics
Desktop, Mobile and Web

Vulkan
ANARI
OpenGL ES
OpenGL
WebGL

3D Assets
Authoring and Delivery

glTF
COLLADA
3DCommerce

Portable XR
Augmented and Virtual Reality

OpenXR

Parallel Computation
Vision, Inferencing, Machine Learning

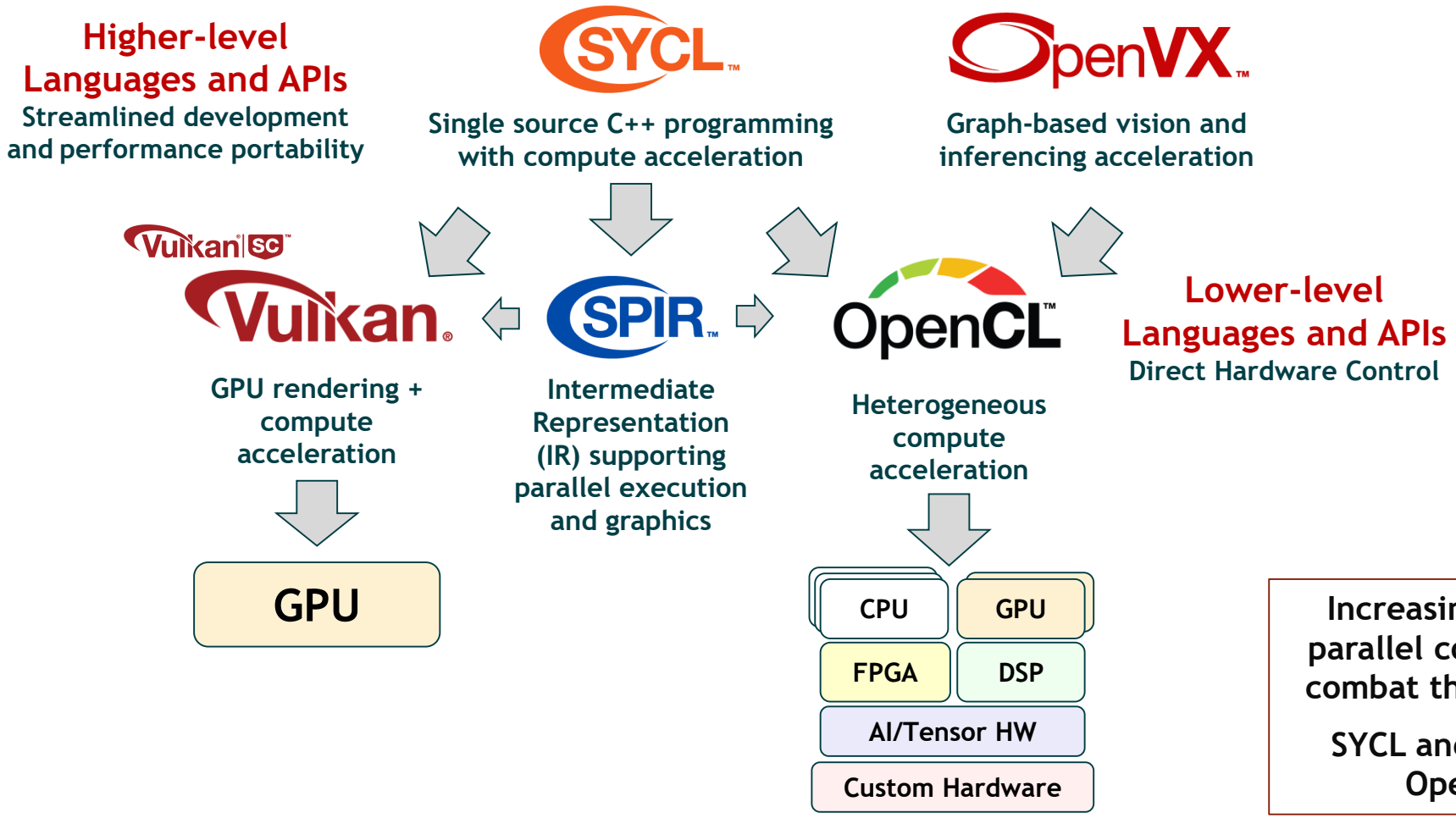
OpenCL
Camera
OpenVX
NNEF
SYCL
SPIR

Safety Critical APIs

OpenGL SC
Vulkan SC

KHRONOS SAFETY CRITICAL ADVISORY FORUM | SC

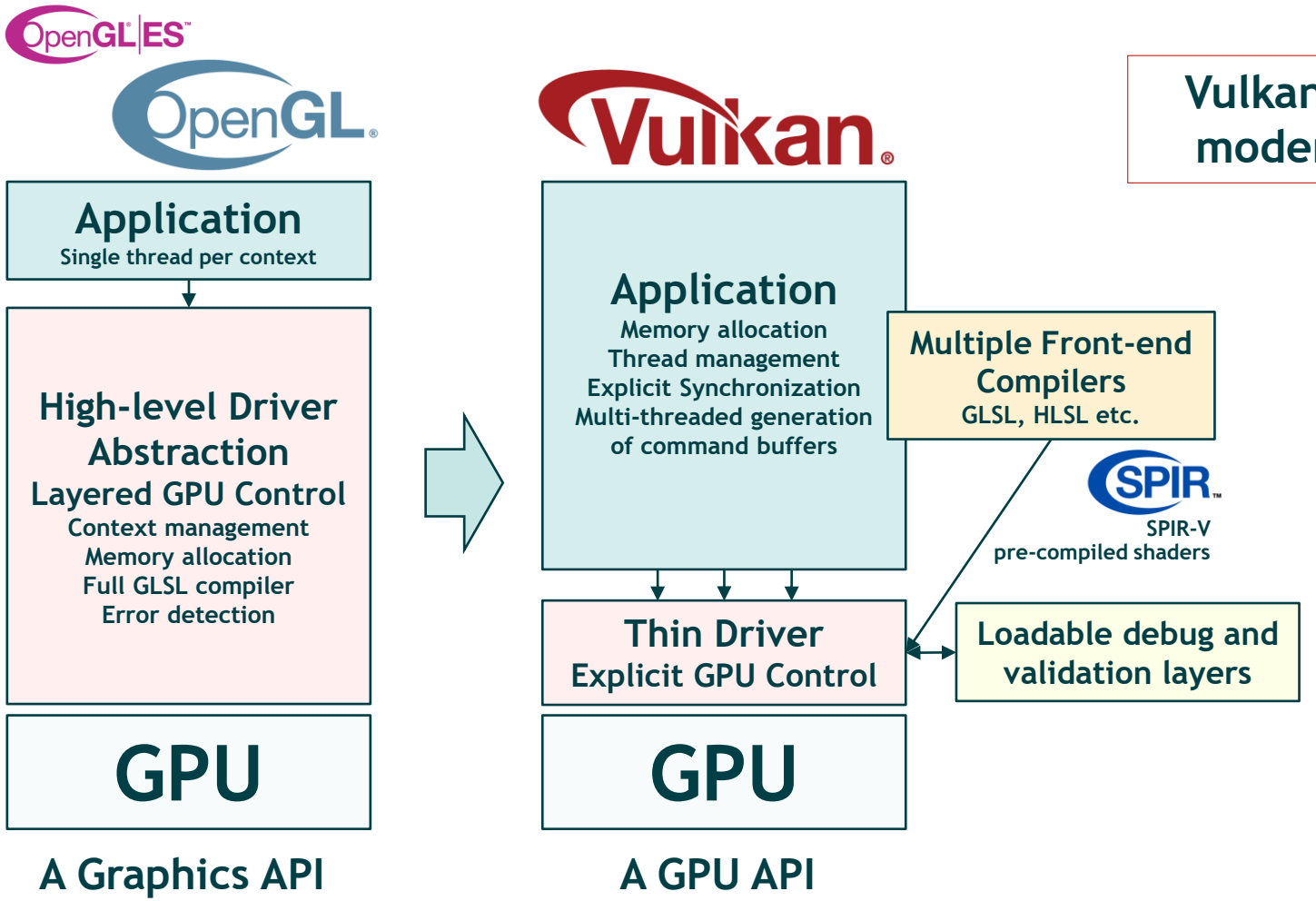
Khronos Compute Acceleration Standards



Increasing industry interest in parallel compute acceleration to combat the 'End of Moore's Law'

SYCL and SPIR were originally OpenCL subprojects

Vulkan Newgen 3D Graphics and Compute



Vulkan is the *only open standard* modern, cross-platform GPU API

Vulkan Advantages

- Simpler drivers
- Direct GPU Control
- Multiple graphics, command and DMA queues
- Multiple offline language compilers

OpenCL 3.0 Adoption and Evolution

Programming and Runtime Framework for Application Acceleration

Offload compute-intensive kernels onto parallel heterogeneous processors

CPUs, GPUs, DSPs, FPGAs, Tensor Processors

OpenCL C and C++ kernel languages

OpenCL 3.0 is cleanly extensible baseline

C++ for OpenCL compiler for C++17 kernels

Vulkan/OpenCL Interop (provisional)

Asynchronous DMA for embedded platforms

Example future OpenCL 3.0 extensions under consideration at Khronos

Command Buffer Record/Replay

Unified Shared Memory

Floating Point Atomics

Image Tiling Controls

YUV Multi-planar Images

Cross-workgroup Barriers

External Memory Export

Cooperative Matrices

Timeline Semaphores

Generalized Image from buffer

32 and 64-length vectors

Indirect Dispatch

OpenCL™

OpenCL 3.0 Adopters

Product Conformance Status

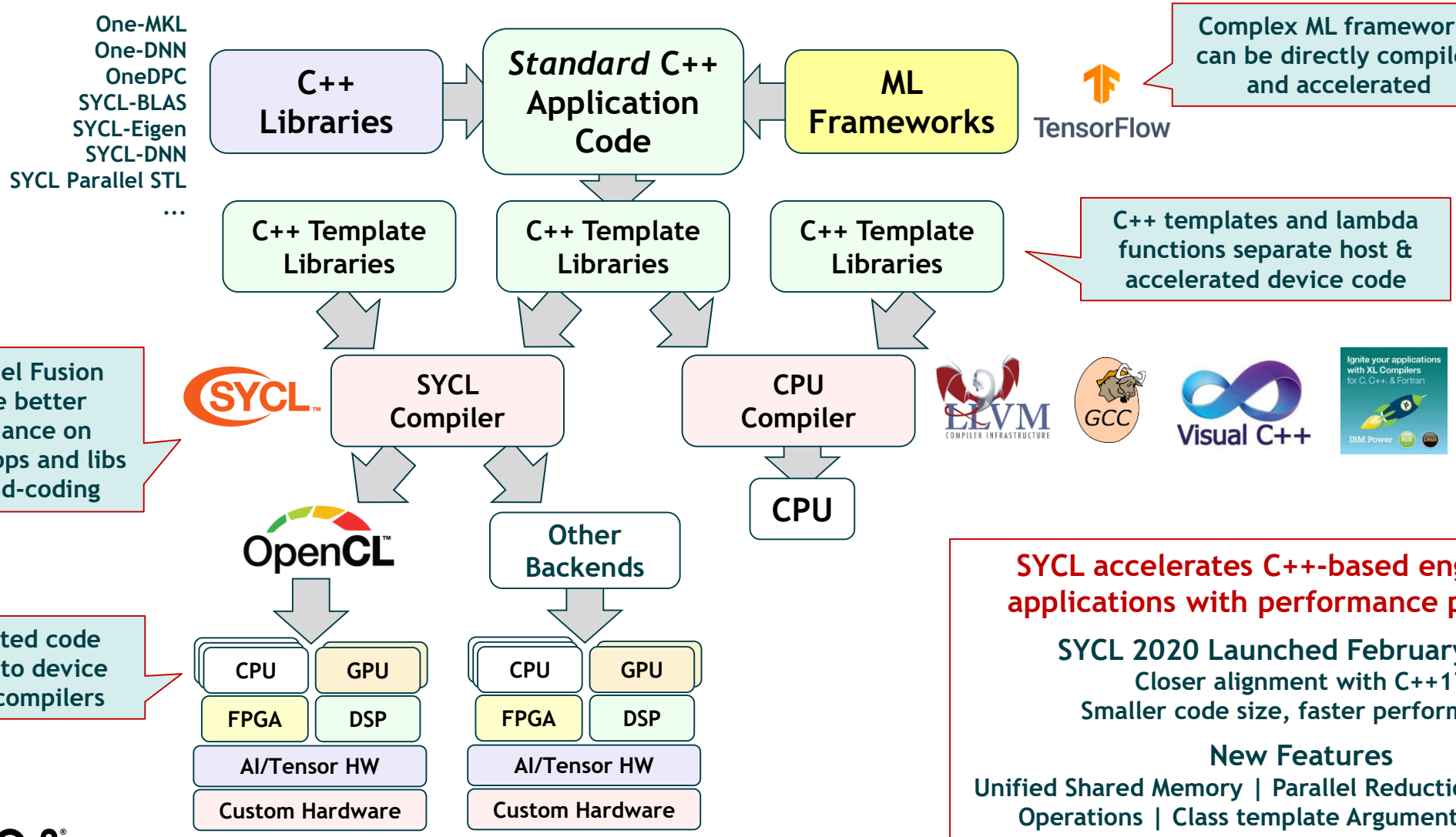
✓ OpenCL 3.0 Adopters Shipping Conformant Implementations

arm ✓ Google ✓ intel ✓ NVIDIA ✓ QUALCOMM ✓

codeplay Imagination ✓ Microsoft QNX VeriSilicon ✓

KHRONOS GROUP

SYCL Single Source C++ Parallel Programming



SYCL accelerates C++-based engines and applications with performance portability

SYCL 2020 Launched February 2021
Closer alignment with C++17
Smaller code size, faster performance

New Features
Unified Shared Memory | Parallel Reductions | Subgroup Operations | Class template Argument Deduction

OpenVX Cross-Vendor Vision and Inferencing

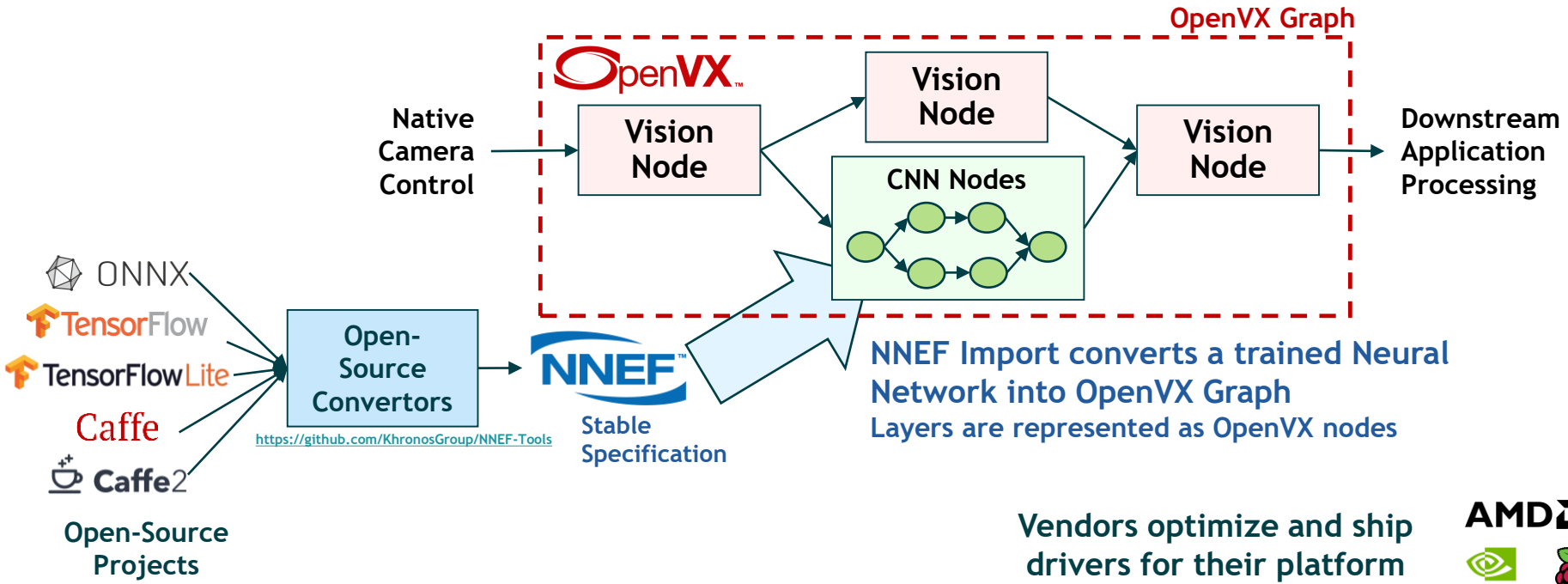
High-level graph-based abstraction for portable, efficient vision processing

Optimized OpenVX drivers created, optimized and shipped by processor vendors

Implementable on almost any hardware or processor with performance portability

Graph can contain vision processing and NN nodes for global optimization

Run-time graph execution need very little host CPU interaction



Open-Source Projects

Vendors optimize and ship drivers for their platform

Full list of conformant OpenVX implementations here:
https://www.khronos.org/conformance/adopters/conformant-products/openvx

AMD cadence Imagination intel NVIDIA Qualcomm socionext SYNOPSYS Texas Instruments eSilicon ETRI VeriSilicon

Open Source OpenVX & Samples

Open Source OpenVX 1.3

Fully Conformant on Raspberry Pi

Raspberry Pi 3 and 4 Model B with Raspbian OS

Memory access optimization via tiling/chaining

Highly optimized kernels on multimedia instruction set

Automatic parallelization for multicore CPUs and GPUs

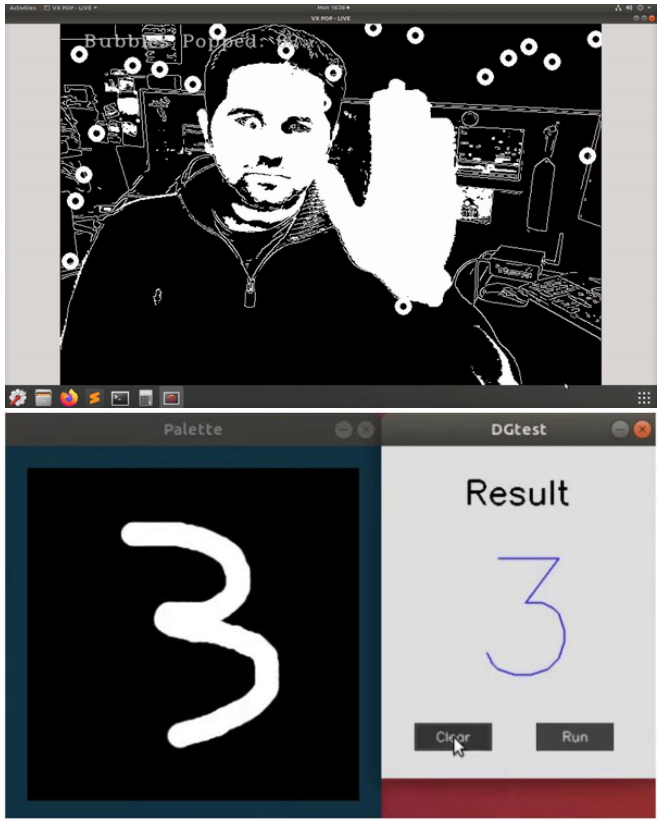
Automatic merging of common kernel sequences

Supports NNEF Import Feature Set

> "Raspberry Pi is excited to bring the Khronos OpenVX 1.3 API to our line of single-board computers. Many of the most exciting commercial and hobbyist applications of our products involve computer vision, and we hope that the availability of OpenVX will help lower barriers to entry for newcomers to the field."
>
> Eben Upton
>
> ***Chief Executive Raspberry Pi Trading***

Open Source OpenVX Tutorial and Code Samples

https://github.com/rgiduthuri/openvx_tutorial
https://github.com/KhronosGroup/openvx-samples
https://github.com/kiritigowda/OpenVX/tree/master/digitClassification#digit-classification



Growing Need for APIs for Functional Safety

Demand for advanced GPU-accelerated graphics and compute is growing in an increasing number of industries where safety is paramount, such as automotive, autonomy, avionics, medical, industrial, and energy

1990s
Avionics

2010s
Automotive

2020s…
Everywhere

Safety-critical APIs are designed to reduce system-level safety-critical certification effort and costs

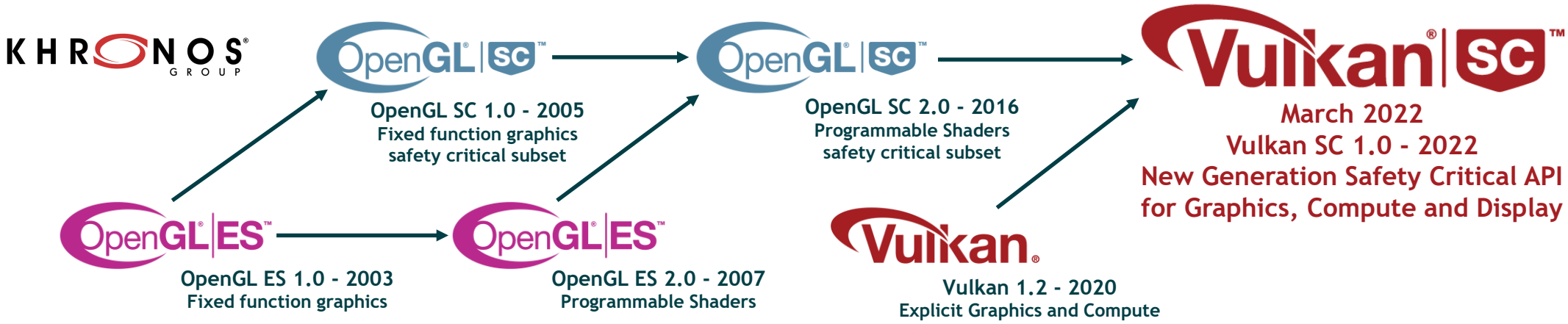
1) Streamlined to reduce documentation and testing surface area
2) Deterministic behavior to simplify system design and testing
3) Robust and unambiguous fault handling

ISO 26262

DO-178C

Industry safety-critical standards include
RTCA DO-178C Level A / EASA ED-12C Level A (avionics)
ISO 26262 ASIL D (automotive)
IEC 61508 (industrial)
IEC 62304 (medical)

Khronos Safety Critical GPU API Evolution



Khronos has 20 years experience in adapting mainstream APIs for safety-critical markets

Leveraging proven mainstream APIs with shipping silicon implementations and developer tooling and familiarity

Vulkan SC has significantly higher performance and flexibility than OpenGL SC

OpenGL SC will continue to be supported by Khronos, but new developments will focus on Vulkan SC

Vulkan SC targets any systems requiring safety critical graphics and/or compute

Enabling new safety-critical markets

Vulkan SC 1.0 Design Philosophy

Vulkan

Vulkan 1.2 is a compelling starting point

Widely adopted, royalty-free open standard

Explicit control of device scheduling, synchronization and resource management

Smaller surface area than OpenGL

Not burdened by runtime debug functionality

Very little internal state

Well-defined thread behavior

Ingests SPIR-V IR - no runtime front-end compiler

Vulkan SC reduces cost and effort for to produce evidence packages for system certification

Vulkan SC can be invaluable for real-time embedded applications, even if not formally safety-certified

Vulkan SC

Streamlined

Remove non-essential runtime functionality

Sparse memory

Descriptor update templates

Certain types of object deleters

Deterministic

Predictable execution times and results

Offline compilation of pipelines

Static memory allocation

Robust

Removing Ambiguity

No ignored parameters or undefined behaviors

Enhanced fault handling and reporting functionality

Rigorous conformance test suite

MISRA C alignment

MISRA

Testable

Open-source Conformance Test Suite

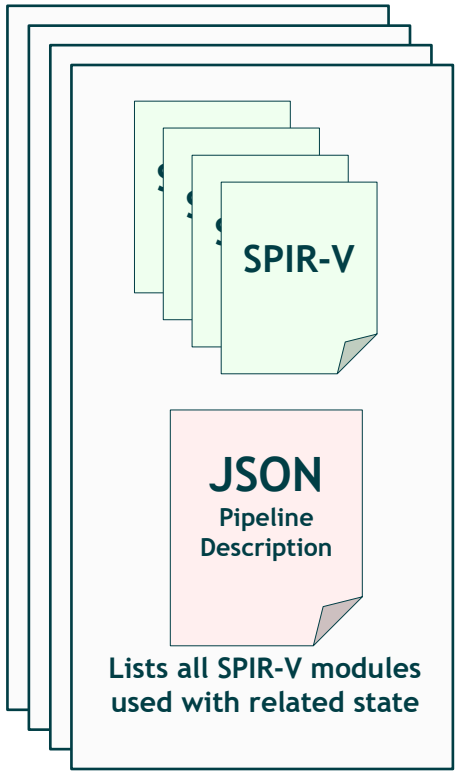
Freely available under Apache 2.0 open-source license

Leverages 1 million+ test Vulkan test suite with added SC-specific tests

Confirms and documents Vulkan SC implementation compatibility

Vulkan SC Offline Compiled Pipelines

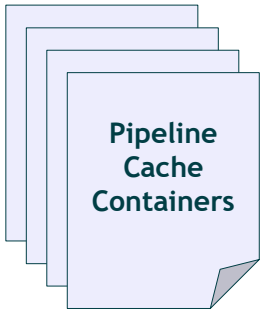
A Vulkan Pipeline defines how the GPU processes data



Lists all SPIR-V modules used with related state

Implementation-Specific Pipeline Cache Compiler (PCC)

Vulkan SC



Pipeline Cache Utility

Extracts information from pipeline cache files to analyze dataflow and the amount of memory used by the processing in the pipeline

Application

Memory for pipelines is reserved at device at device creation time as fixed size pools. Similarly sized pipelines can be assigned to the same pool to minimize memory size and fragmentation

Vulkan SC Precompiled Pipelines enable run-time determinism

Eliminates need for runtime memory allocation

Offline **Runtime**

OpenVX SC Profile

Minimizes Run-time Surface Area and Implementation Size

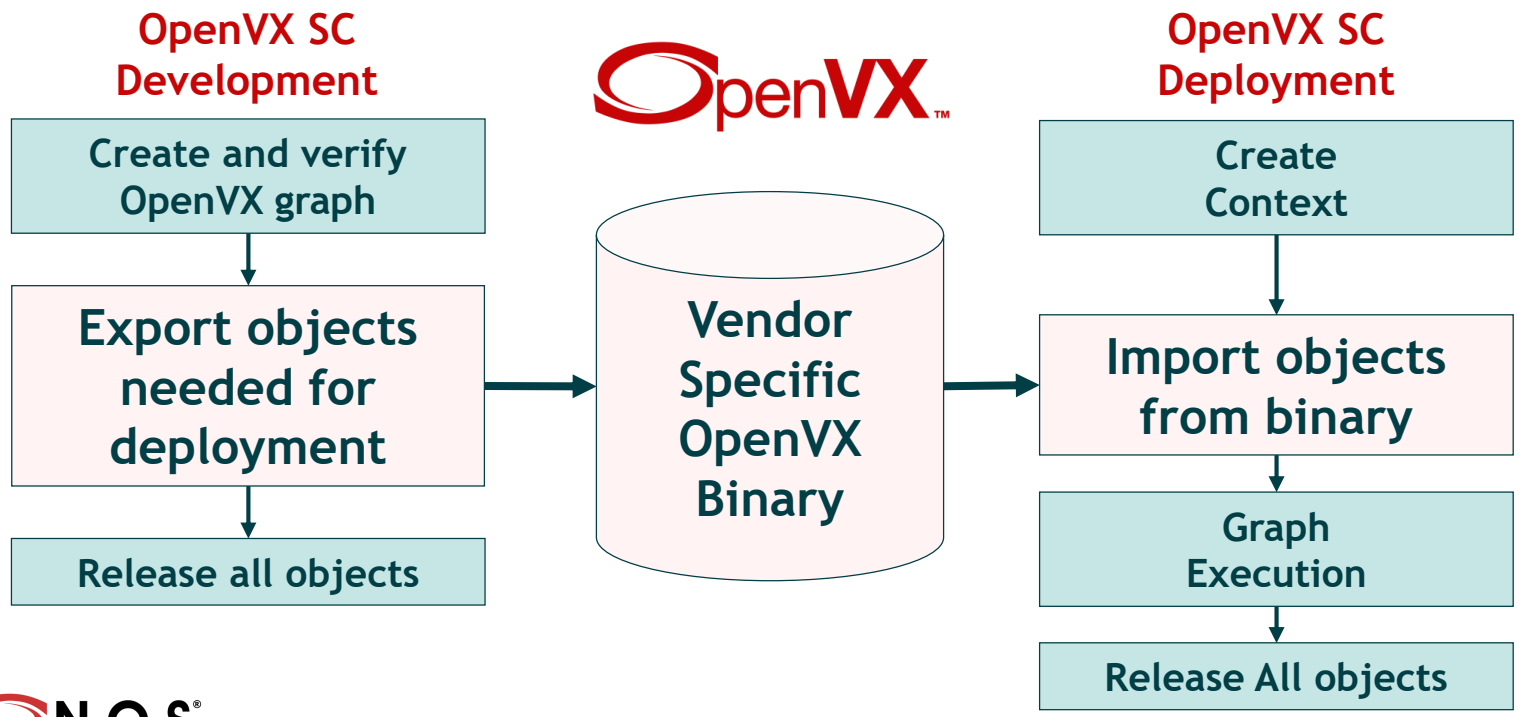
Eases system-level safety certification

Separated Development and Deployment environments

Robust Specification

Annotated specification with Functional Requirement tag numbers

MISRA-C compliant headers



The OpenVX SC profile combined with ingestion of trained Neural Networks enables OpenVX as a cross-platform inferencing engine for safety critical markets

SYCL Safety-Critical Exploratory Forum

Exploring real-world industry requirements for *open* and *royalty-free* high-level compute APIs suitable for safety-critical markets

Proven Khronos Exploratory Process to ensure industry requirements are fully understood before starting standardization initiatives

Any company is welcome to join

No cost or IP Licensing obligations

Project NDA to cover Exploratory Forum Discussions

Khronos SYCL Safety-Critical Exploratory Forum

Online discussion forum and weekly Zoom calls

No detailed design activity to protect participants IP

Explore if consensus can be built around an agreed **Scope of Work** document

Discuss what standardization activities can best execute actions in the Scope of Work

Scope of Work Document

Agreed SOW document released from NDA and made public

Initiation of Khronos Working Group to execute the SOW

More information and signup instructions

https://www.khronos.org/syclsc

Growing Need for Camera API Standard

Increasing Sensor Diversity

Including camera arrays and depth sensors such as Lidar

Multiple Sensors Per System

Synchronization and coordination become essential

Cost and time to integrate and utilize sensors in diverse markets has become a major constraint on innovation and efficiency

Increasing Sensor Processing Demands

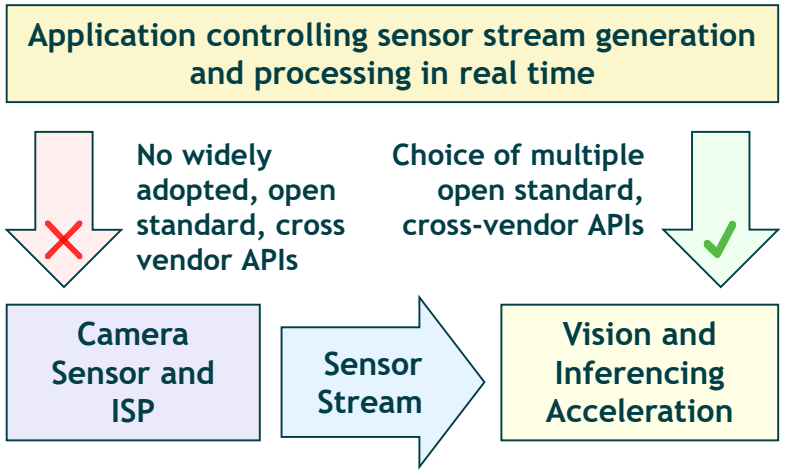
Including inferencing. Sensor outputs need to be flexibly and efficiently generated and streamed into acceleration processors

Proprietary APIs Hinder Innovation

Vendor-specific APIs to control cameras, sensors and close-to-sensor ISPs prevent rapid integration of new technologies

Industry Need for Embedded Camera API

The industry lacks an open, cross-vendor standard to control embedded camera systems



A cross-platform camera API would have many benefits

Cross-vendor portability of camera/sensor code for easier system integration of new sensors

Preservation of application code across multiple generations of cameras and sensors

Sophisticated control over sensor stream generation for effective downstream accelerated processing

Khronos Camera Working Group announced in February 2022

EMVA and Khronos have cooperated since 2020 to understand need for a new API for standard camera and image capture control

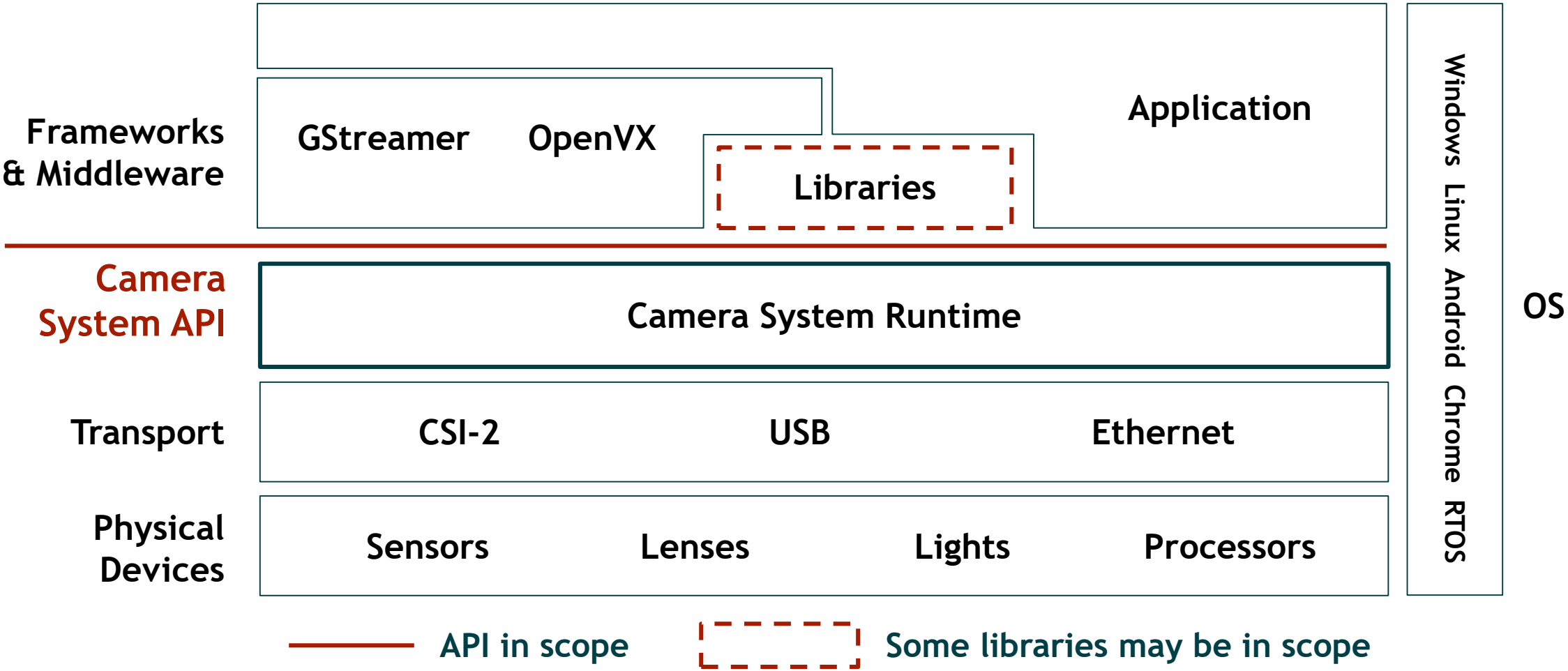
Over 70 companies met at the Khronos Camera Exploratory Group through 2021 to create consensus on a Scope of Work document

Working Group is open to all Khronos members and meeting weekly to execute the Scope of Work

emva
european machine vision association

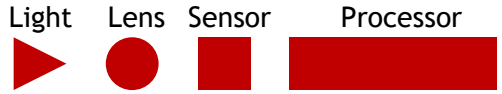
KHRONOS GROUP
CONNECTING SOFTWARE TO SILICON

Typical Software Stack using Camera API

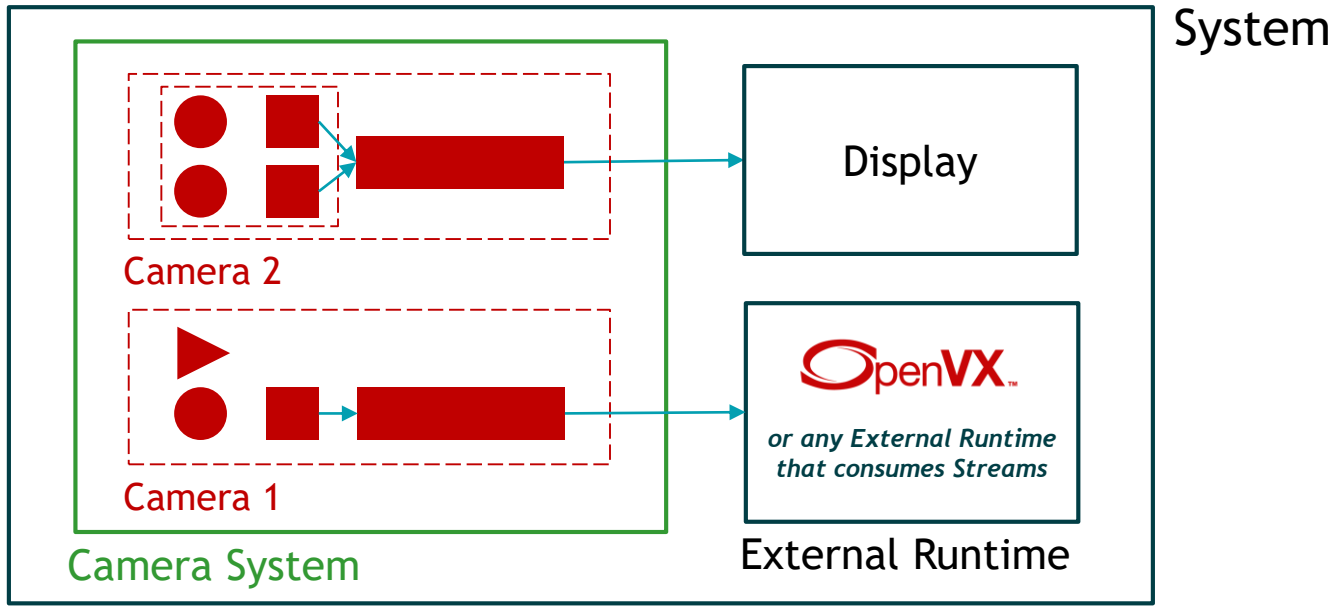


Named transport layers, frameworks and operating systems are illustrative examples

Camera API Terminology



Physical Devices = queryable and controllable via a Device ID:
Logical Device = set of Devices queried and controlled via a single Device ID
Frame = Image + Metadata accessed via Frame ID
Stream = sequence of Frames
Camera = a Logical Device that exports one or more Streams from the Camera System



Get Involved!

Khronos is developing a growing family of open, royalty-free API standards relevant to embedded and safety-critical markets

Any company is welcome to join Khronos to influence standards development!

All Khronos members can participate in the new Camera Working Group!

Get involved in the new SYCL SC Exploratory Forum at zero cost!

KHRONOS® GROUP

Additional Resources

Khronos Resources

Joining Khronos
https://www.khronos.org/members/
memberservices@khronosgroup.org

OpenCL, OpenVX, Vulkan SC
https://www.khronos.org/opencl/
https://www.khronos.org/openvx/
https://www.khronos.org/vulkansc/

Camera Working Group
https://www.khronos.org/camera

SYCL SC Exploratory Group
https://www.khronos.org/camera

2022 Embedded Vision Summit

Come talk to Khronos at the
Exhibition Booth: #525