A Vision for OpenCL

Neil Trevett
Vice President Mobile Ecosystem at NVIDIA
President of Khronos and Chair of the OpenCL Working Group

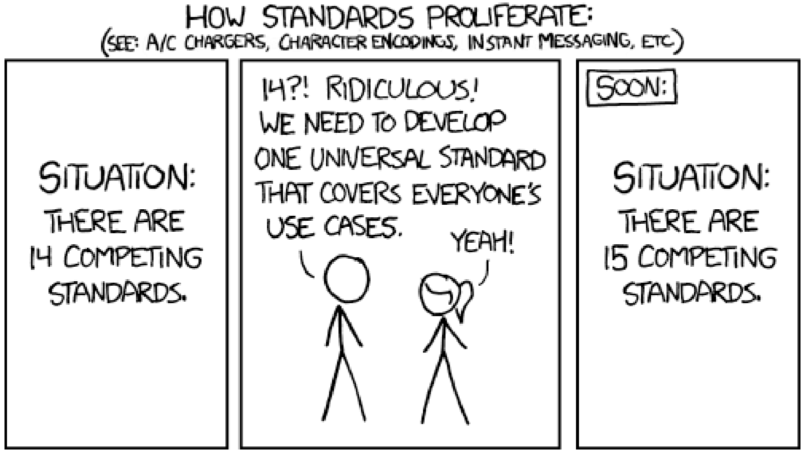
A Vision for OpenCL

- Where are we?
- How did we get here?
- Where might we be going next?

"Know from whence you came. If you know whence you came, there are absolutely no limitations to where you can go."
— James Baldwin

"The best way to predict the future is to invent it."
— Alan Kay

OpenCL has definitely broken out of being 'yet another competing standard for parallel programming.' How will it continue to add value to the industry?



— https://xkcd.com/927/

OpenCL Roadmap

- What markets has OpenCL been aimed at?
- What problems is OpenCL solving?
- How will OpenCL need to adapt in the future?

Discussion
Focus for New
Capabilities

Dec08	Jun10	Nov11	Nov13	
	HPC Desktop Mobile	HPC Desktop Mobile Web	HPC Desktop Mobile Web FPGA	HPC Desktop Mobile Web FPGA Embedded Safety Critical
	3-component vectors Additional image formats Multiple hosts and devices Buffer region operations Enhanced event-driven execution Additional OpenCL C built-ins Improved OpenGL data/event interop	Device partitioning Separate compilation and linking Enhanced image support Built-in kernels / custom devices Enhanced DX and OpenGL Interop	Shared Virtual Memory On-device dispatch Generic Address Space Enhanced Image Support C11 Atomics Pipes Android ICD	
OpenCL 1.0 Specification	OpenCL 1.1 Specification	OpenCL 1.2 Specification	OpenCL 2.0 Specification	

18 months (Dec08 – Jun10); 18 months (Jun10 – Nov11); 24 months (Nov11 – Nov13)

Roadmap Discussions
Binning/Triaging
SW and HW features
Will use Provisional Specs
Some common requests:
- C++ Kernel Language
- Refine and evolve Memory and Execution Models
- Better debug and profiling
- Trans-API Interop

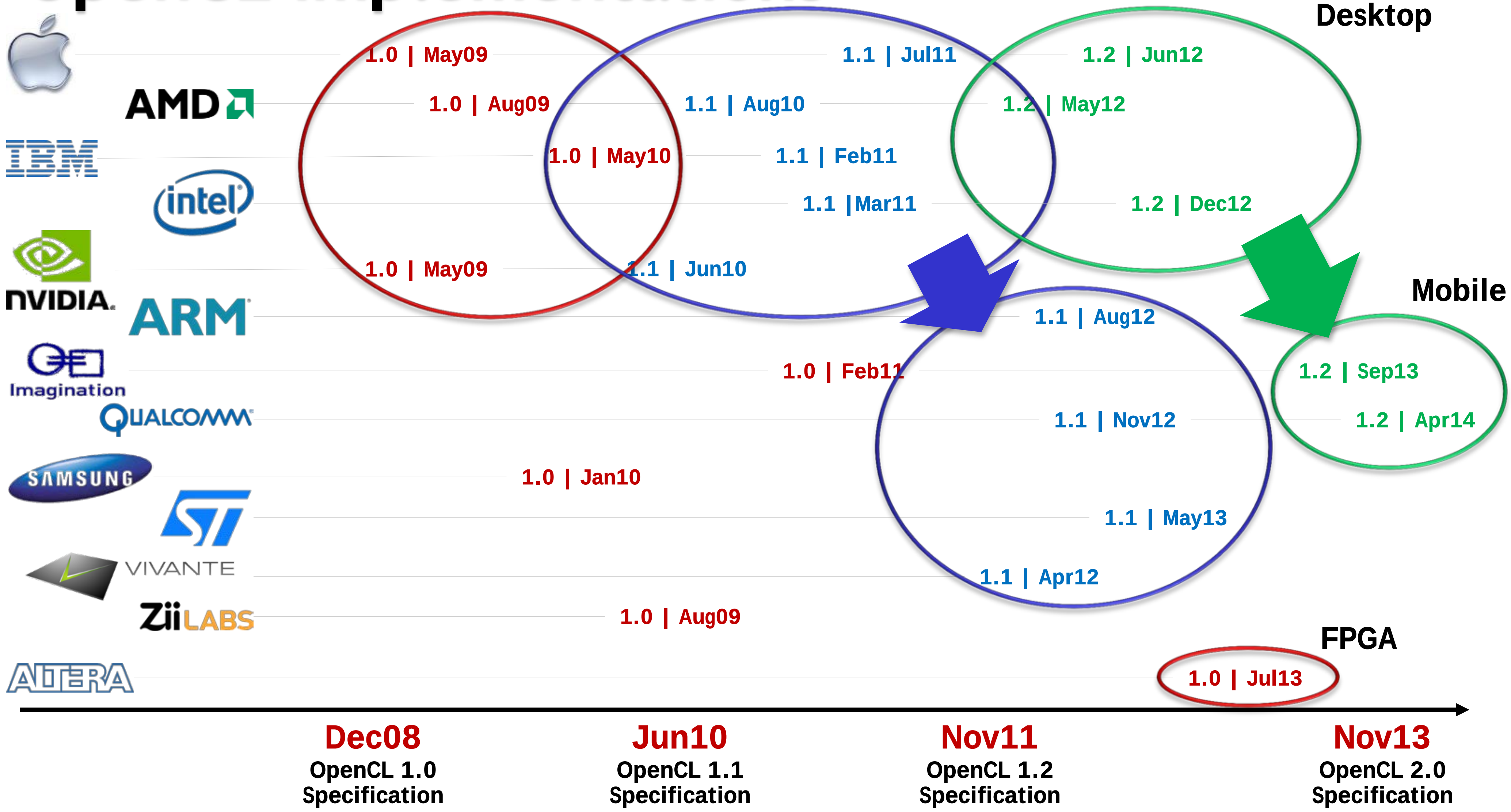
© Copyright Khronos Group 2014

OpenCL Desktop Usage

- **Broad commercial uptake of OpenCL**
  - Mainly imaging, video and vision processing
  - Adobe, Apple, Corel, ArcSoft Etc. Etc.
- **“OpenCL” on Sourceforge, Github, Google Code, Bitbucket finds over 2,000 projects**
  - OpenCL implementations - Beignet, pocl
  - VLC, X264, FFMPEG, Handbrake
  - GIMP, ImageMagick, IrfanView
  - Hadoop, Memcached
  - WinZip, Crypto++ Etc. Etc.
- **Desktop benchmarks use OpenCL**
  - PCMark 8 – video chat and edit
  - Basemark CL, CompuBench Desktop

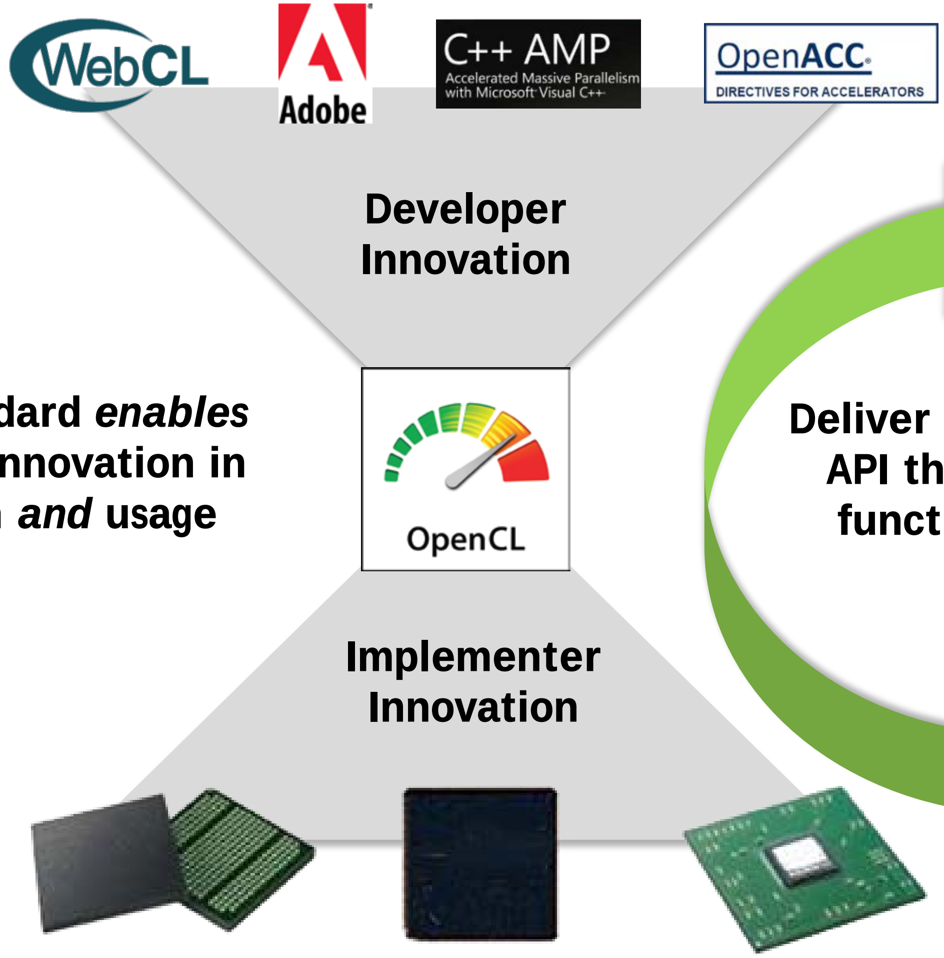
http://streamcomputing.eu/blog/2013-12-28/professional-consumer-media-software-opencl/

OpenCL Implementations



© Copyright Khronos Group 2014 - Page 6

Khronos Foundational APIs



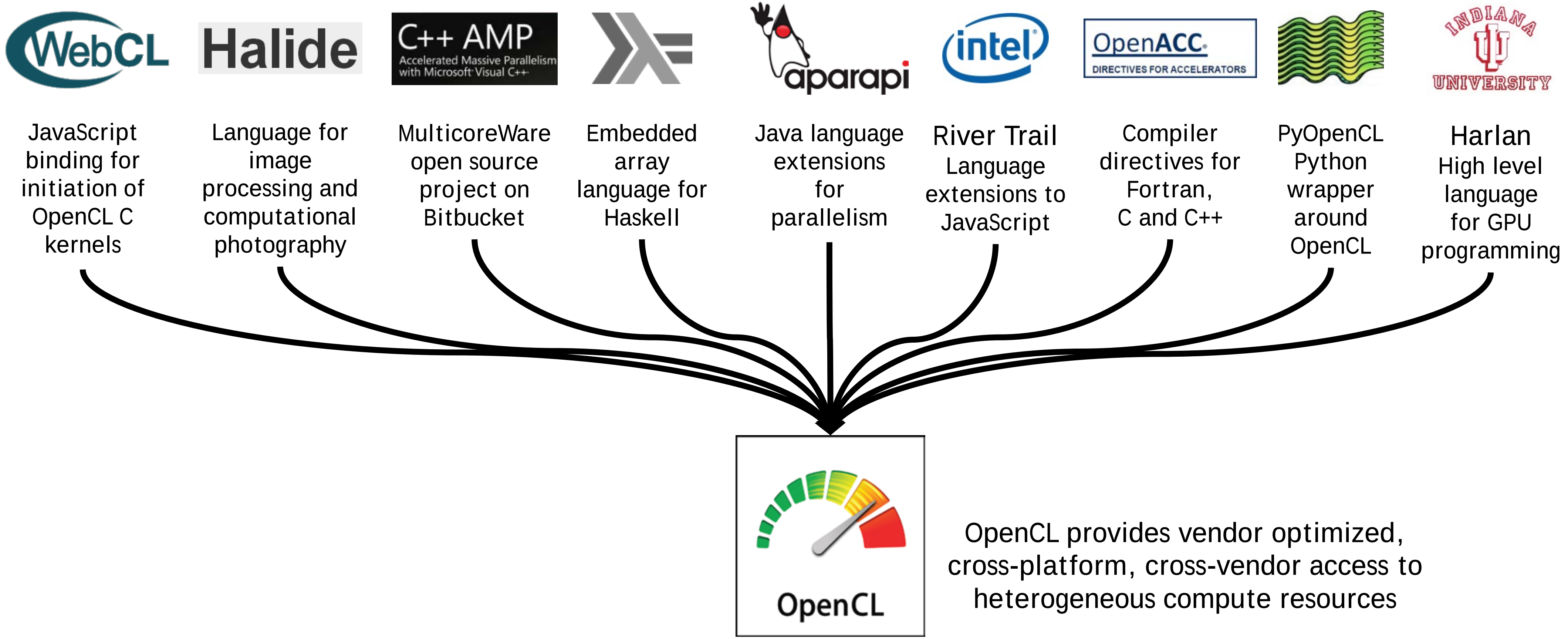
A successful standard *enables* and *encourages* innovation in implementation *and* usage

Market Momentum…
Applications, libraries and frameworks that find OpenCL acceleration can deliver a better end-user experience

Deliver the lowest level abstraction possible API that still provides portability - this is functionality needed on *every* platform

Market Momentum..
Many devices competing on performance and power to tap into the value of OpenCL content

OpenCL as Parallel Language Backend



OpenCL provides vendor optimized, cross-platform, cross-vendor access to heterogeneous compute resources

Libraries and Languages using OpenCL

Library Name	Overview	Website
Accelerate	accelerate: An embedded language for accelerated array processing	http://hackage.haskell.org/package/accelerate
amgCL	Simple and generic algebraic multigrid framework	https://github.com/ddemidov/amgcl
Aparapi	API for data parallel Java. Allows suitable code to be executed on GPU via OpenCL.	https://code.google.com/p/aparapi/
ArrayFire	Array-based function library	https://www.accelereyes.com/products/arrayfire
Bolt	Bolt C++ Template Library	https://github.com/HSA-Libraries/Bolt/releases/tag/v1.1GA
Boost.Compute	Boost.Compute is a GPU/parallel-computing library for C++ based on OpenCL.	https://github.com/kylelutz/compute
Bullet Physics	Bullet Physic OpenCL accelerated Rigid Body Pipeline	http://bulletphysics.org/wordpress/?p=381
C++ AMP	CLANG/LLVM based C++AMP 1.2 standard and transforms it into OpenCL-C	https://bitbucket.org/multicoreware/cppamp-driver-ng/wiki/Home
clBLAS	cl BLAS implementation	https://github.com/clMathLibraries/clBLAS
clFFT	OpenCL FFT Libarary	https://github.com/clMathLibraries/clFFT
clMAGMA	clMAGMA 1.1 is an OpenCL port of MAGMA	http://icl.cs.utk.edu/magma/software/view.html?id=190
clpp	OpenCL Data Parallel Primitives Library	https://code.google.com/p/clpp/
clSpMV	Sparse Matrix Solver	http://www.eecs.berkeley.edu/~subrian/clSpMV.html
Clyther	Python just-in-time specialization engine for OpenCL	http://srossross.github.io/Clyther/
Codeplay Math Lib	OpenCL 1.2 Math library	https://www.codeplay.com/products/math/
Concord	C++ Hetrogenous Programing Framework (Support OpenCL 1.2) TBB like	https://github.com/IntelLabs/iHRC/
COPRTHR	CO-PRocessing THReads (COPRTHR) SDK	http://www.browndeertechnology.com/coprthr.htm
DL- Data Layout	DL Enables Optimized Data Layout Across Heterogeneous Processors	http://www.multicorewareinc.com/dl.html
ForOpenCL	Fortran to OpenCL tool	http://sourceforge.net/projects/fortran-parser/files/ForOpenCL/
fortranCL	FortranCL is an OpenCL interface for Fortran 90.	https://code.google.com/p/fortrancl/
FSCL.Compiler	FSharp to OpenCL Compiler	https://github.com/GabrieleCocco/FSCL.Compiler
GATLAS	GPU Automatically Tuned Linear Algebra Software (Project looks stalled)	https://github.com/cjang/GATLAS
GMAC	Global Memory for Accelerators	http://www.multicorewareinc.com/gmac.html
GPULib	Iterative sparse solvers	http://www.txcorp.com/
gpumatrix	A matrix and array library on GPU with interface compatible with Eigen.	https://github.com/rudaoshi/gpumatrix
GPUVerify	GPUVerify is a tool for formal analysis of GPU kernels written in OpenCL	http://multicore.doc.ic.ac.uk/tools/GPUVerify/
Halide	Halide Programming language for high-performance image processing	http://halide-lang.org/
Harlan	Harlan: A Scheme-Based GPU Programming Language	https://github.com/eholk/harlan
HOpenCL	Haskell OpenCL Wrapper API	https://github.com/bgaster/hopencl
libCL	C++ Generic parallel algorithms library	http://www.libcl.org/
Libra SDK	Cross Platform Acceleration API	http://www.gpusystems.com/libra.aspx
M³ Platform	Parallel Framework and Primitive Libraries	http://www.fixstars.com/en/products/m-cubed/
MUMPS	Direct Sparse solver	http://graal.ens-lyon.fr/MUMPS/
Octave	Octave acceleration via OpenCL	http://indico.cern.ch/event/93877/session/13/contribution/89/material/slides/0.pdf

Courtesy: AMD

Libraries and Languages using OpenCL #2

Open Fortran Parser	ANTLR-based parsing tools that support the Fortran 2008 standard	http://fortran-parser.sourceforge.net/
OpenACC to OpenCL Compiler	Rose based OpenACC to OpenCL Compiler.	https://github.com/tristanvdb/OpenACC-to-OpenCL-Compiler
OpenCL.jl	Julia OpenCL 1.2 bindings	https://github.com/jakebolewski/OpenCL.jl
OpenCLIPP	OpenCL Integrated Performance Primitives - A library of optimized OpenCL image processing functions	https://github.com/CRVI/OpenCLIPP
OpenCLLink	Mathematica to use the OpenCL parallel computing language	http://reference.wolfram.com/mathematica/OpenCLLink/guide/OpenCLLink.html
OpenClooVision	Computer vision framework based on OpenCL and C#	http://opencloovision.codeplex.com/
OpenCV-CL	OpenCL accelerated OpenCV	http://amd-dev.wpengine.netdna-cdn.com/wordpress/media/2013/07/opencv-cl_instructions-246.pdf
OpenHMPP	Directive-based OpenACC and OpenHMPP Source to OpenCL compiler	http://www.caps-entreprise.com/products/caps-compilers/
Paralution	C++ sparse iterative solvers and preconditioners library with OpenCL support	http://www.paralution.com/
Pardiso	Direct Sparse solver	http://www.pardiso-project.org/
Pencil	PENCIL to be a suitable target language for the compilation of domain-specific languages (DSLs).	https://github.com/carpproject/pencil
PETSc	Portable, Extensible Toolkit for Scientific Computation	http://www.mcs.anl.gov/petsc/
PyOpenCL	OpenCL parallel computation API from Python	http://mathema.tician.de/software/pyopencl/
QT with OpenCL	Using OpenCL with QT	http://doc.qt.digia.com/opencl-snapshot/
RaijinCL	library for matrix operations for OpenCL	http://www.raijincl.org/
Rivertrail	JavaScript which supports Data Parallelism via OpenCL	https://github.com/rivertrail/rivertrail/wiki
RNG	Random number generation for parallel computations	http://www.iro.umontreal.ca/~lecuyer/
ROpenCL	Parallel Computing for R Using OpenCL	http://repos.openanalytics.eu/html/ROpenCL.html
Rose Compiler	Rose Compiler with OpenCL Support	http://rosecompiler.org/
Rust-OpenCl	OpenCL bindings for Rust.	https://github.com/luqmana/rust-opencl
ScalaCL	Scala support of OpenCL	https://github.com/ochafik/ScalaCL
SkelCL	SkelCL is a library providing high-level abstractions for alleviated programming of modern parallel heterogenec	https://github.com/skelcl/skelcl
SnuCL	SnuCL naturally extends the original OpenCL semantics to the heterogeneous cluster	http://snucl.snu.ac.kr/
SpeedIT 2.4	OpenCl based OpenFoam acceleration library	http://vratis.com/index.php?option=com_content&view=category&layout=blog&id=49&Itemid=88&lar
streamscan	StreamScan: Fast Scan Algorithms for GPUs without Global Barrier Synchronization-	https://code.google.com/p/streamscan/
SuperLU	Direct Sparse solver	http://crd-legacy.lbl.gov/~xiaoye/SuperLU/
TM-Task Management	Heterogeneous Task Scheduling and Management	http://www.multicorewareinc.com/tm.html
Trilinos	Building blocks for the development of scientific applications; constructing and using sparse and dense matrice	http://trilinos.sandia.gov/
VexCL	VexCL is a C++ vector expression template library for OpenCL/CUDA	http://ddemidov.github.io/vexcl
ViennaCL	open-source linear algebra library for computations on many-core architectures (GPUs, MIC) and multi-core CPU	http://viennacl.sourceforge.net/
VirtualCL	VirtualCL (VCL) cluster platform is a wrapper for OpenCL™	http://www.mosix.cs.huji.ac.il/txt_vcl.html
VOBLA	Vehicle for Optimized Basic Linear Algebra - Optimized Basic Linear Algebra DSL	https://github.com/carpproject/vobla
VOCL	Virtualized OpenCL enviornment	http://www.mcs.anl.gov/~thakur/papers/xiao-vocl-inpar12.pdf
VSI/Pro®	VSIPL implementation in OpenCL	http://www.techsource.com/press/pdfs/Run_Time-TechSource_press_release.pdf
WAMS	Algebraic Multigrid Solver using state-of-the-art wavelet preconditioners- solver for sparse linear equations	http://www.newengland-scientific.com/

Courtesy: AMD

Widening OpenCL Ecosystem

OpenCL C Kernel Source

Alternative Language for Kernels

Apps and Frameworks

SPIR Generator
(e.g. patched Clang)

https://github.com/KhronosGroup/SPIR

SYCL

SPIR is easier compiler target than C

SPIR

OpenCL run-time can consume SPIR

OpenCL C Runtime

Device X

Device Y

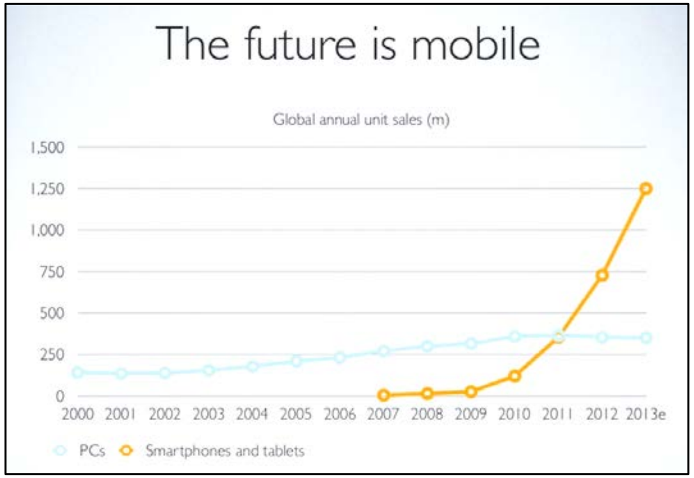
Device Z

SPIR
Standard Portable Intermediate Representation
CLOSE COOPERATION WITH LLVM COMMUNITY
SPIR 1.2 Released January 2014
(uses LLVM 3.2)

SYCL
Programming abstraction that combines portability and efficiency of OpenCL with ease of use and flexibility of C++
SYCL 1.0 Provisional Released March 2014

The Future is Mobile

- **Mobile SOCs now beginning to need more than just 'GPU Compute'**
  - Multi-core CPUs, GPUs, DSPs, ISPs, specialized hardware blocks
- **OpenCL can provide a single programming framework for all processors on a SOC**
  - OpenCL 1.2 Built-in Kernels for custom HW
  - How should OpenCL embrace DSPs?
  - Appropriate precision/resource demands?
- **What are the key mobile applications that will drive use of heterogeneous SOCs?**
  - What will it mean for compute APIs?



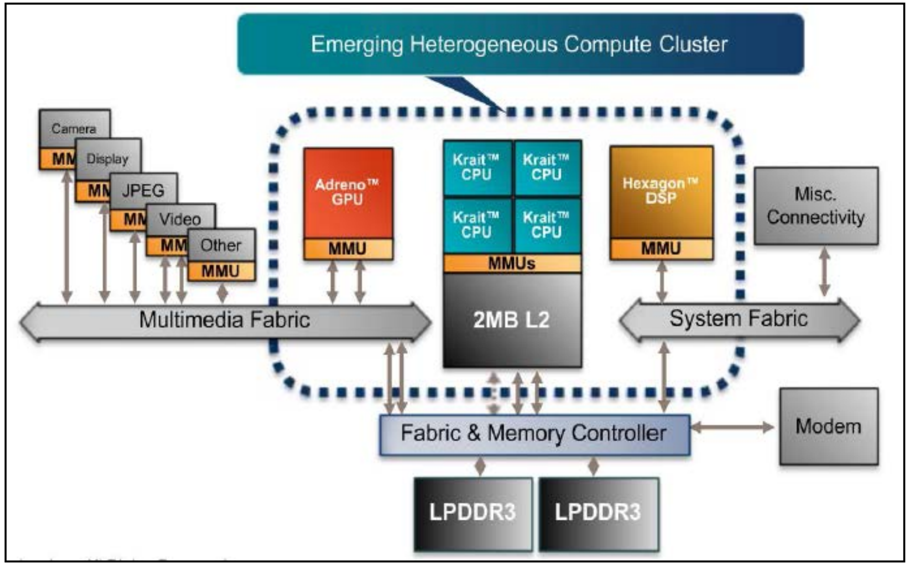
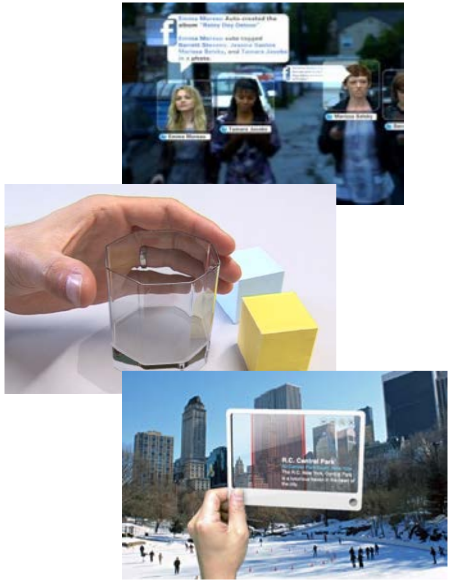


Image Courtesy Qualcomm

© Copyright Khronos Group 2014 - Page 12

Mobile Visual Computing

- Compute acceleration is most useful when a LOT of data needs to be processed
- On Mobile - where will you get a LOT of data?   The camera!



Computational Photography and Videography

Face, Body and Gesture Tracking

3D Scene/Object Reconstruction

Augmented Reality

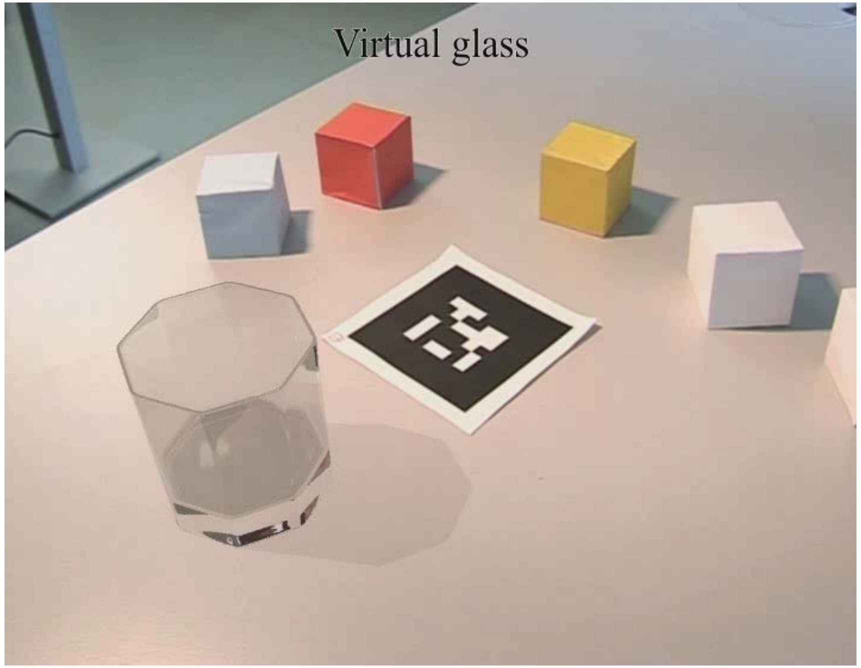
Hyper Realistic AR Using Visual Compute

Augmented Reality before compute acceleration

Augmentations are not convincingly integrated into the scene in terms of positioning or lighting

Courtesy Metaio
http://www.youtube.com/watch?v=xw3M-TNOo44&feature=related

After - Real-time demo on CUDA laptop



Significant ray-casting and light field reconstruction processing enables augmentations to appear realistically in the scene

High-Quality Reflections, Refractions, and Caustics in Augmented Reality and their Contribution to Visual Coherence
P. Kán, H. Kaufmann, Institute of Software Technology and Interactive Systems, Vienna University of Technology, Vienna, Austria
https://www.youtube.com/watch?v=i2MEwVZzDaA

APIs for Mobile Compute

GPU Compute Shaders (OpenGL 4.4 and OpenGL ES 3.1)
Pervasively available on almost any mobile device or OS
Easy integration into graphics apps – no API interop needed
Program in GLSL not C
Limited to acceleration on a single GPU

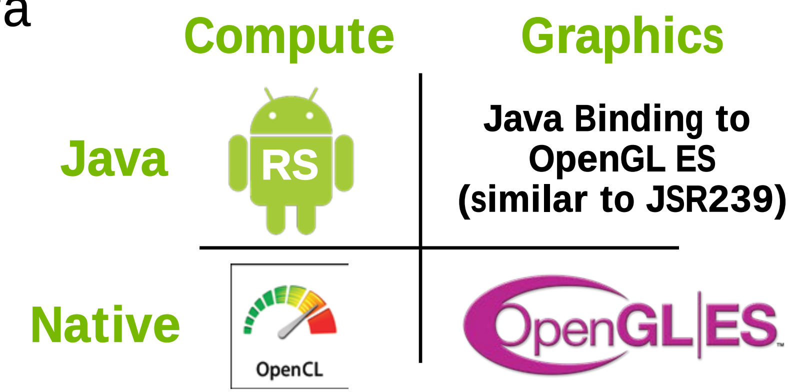
C/C++ Language Integrated GPU Compute
Easy programmability and low level access to GPU: Unified Memory, Virtual Addressing,
Mature and optimized tools and performance
Extensive compute and imaging libraries available (NPP, cuFFT, cuBLAS, cuda-gdb, nvprof etc.)
NVIDIA only, GPU only

General Purpose Heterogeneous Programming Framework
Flexible, low-level access to any devices with OpenCL compiler
Open standard for any device or OS – being used as backend by many languages and frameworks
Single programming and run-time framework for CPUs, GPUs, DSPs, hardware
Needs full compiler stack and IEEE precision

Easy, High-level Compute Offload from Java
C99 based kernel language for simple offload from Java apps to CPU and GPU
JIT Compilation provide host and device portability
Android only
Limited control over acceleration configuration

RenderScript and OpenCL

- **RenderScript and OpenCL do not directly compete**
  - RS addressing very different needs to OpenCL – at a different level in the stack
- **RenderScript designed for 99% of Android developers - using Java**
  - Code critical sections as native C - automatic offload to CPU/GPU
  - Programmer Simplicity and Portability across 1,000's Android handsets
  - Future - Dynamic load balancing through integration with Android instrumentation and power management systems
- **BUT - other types of developer *need* OpenCL-class control in native code**
  - Middleware engines: Unity, Epic Unreal, metaio AR, Bullet Physics …
  - Leading edge apps: real-time video/vision/camera
  - OEM functionality: e.g. camera pipeline
  - These are the developers/apps/engines that hardware vendors want for differentiation

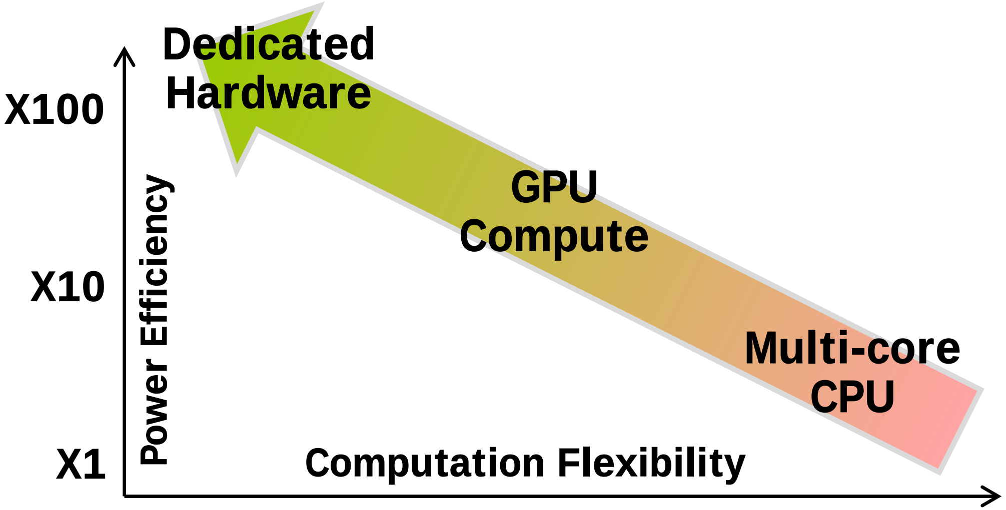


OpenCL on Android can enable specialized access to native acceleration and be an effective platform for RenderScript innovation

Vision Processing Power Efficiency

- **GPUs are more power efficient than CPUs at vision acceleration**
  - When exploiting data parallelism can be x10 as efficient – and getting better!
- **SOCs have space for transistors - but can't turn on at same time!**
  - Would exceed Thermal Design Point of mobile devices
- **Dedicated units can further increase locality and parallelism for efficiency**
  - Dark Silicon - specialized hardware – only turned on when needed
- **Ultra-low power vision scanners will become essential for wearables**
  - Sensor and visual awareness to trigger full vision acceleration subsystems

Enabling new mobile vision-based experiences requires pushing computation onto GPUs and dedicated hardware



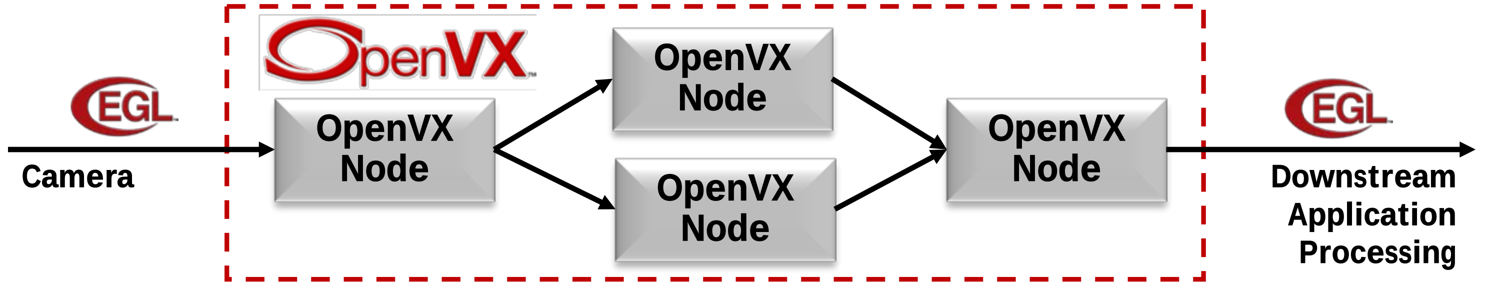
© Copyright Khronos Group 2014 - Page 17

OpenVX – Efficient Vision Acceleration

- **Khronos open-standard - out-of-the-Box vision framework**
  - Focus on low-power, real-time mobile and embedded vision acceleration
- **Performance portability across diverse processor architectures**
  - ISPs, Dedicated vision blocks, DSPs and DSP arrays, GPUs, Multi-core CPUs
- **Suited for low-power, always-on acceleration**
  - Can run on dedicated hardware – no compiler, CPUs or GPUs required
- **Complementary to OpenCV**
  - Which is great for prototyping

Directed graphs of vision operators provide opportunity for optimizing performance and power

Each Node can be implemented in software or accelerated hardware
Nodes may be fused to eliminate memory transfers
Processing can be tiled to keep data entirely in local memory/cache



Example OpenVX Graph

OpenVX and OpenCL are Complementary

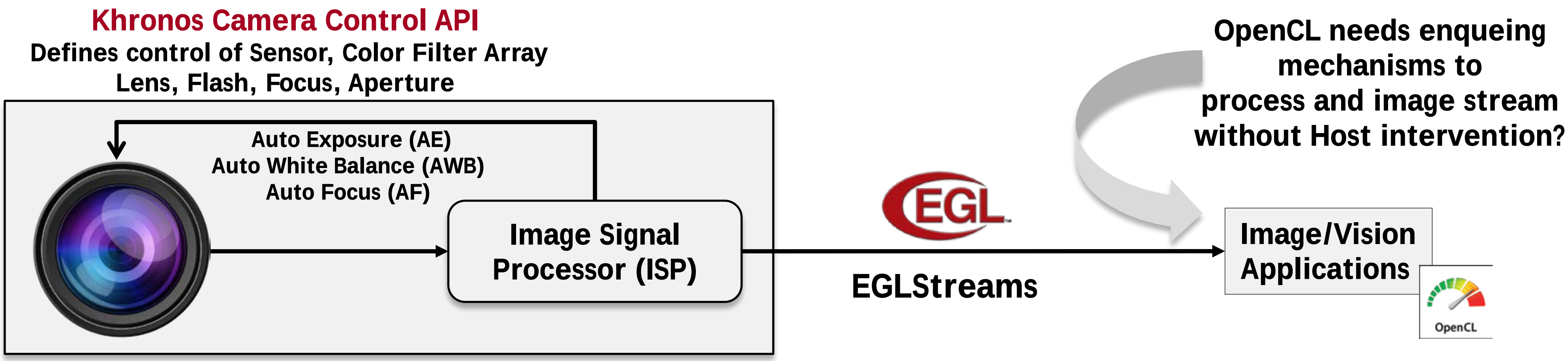
	OpenCL	OpenVX
Use Case	General Heterogeneous programming	Domain targeted Vision processing
Ease of Use	General-purpose math libraries with no built-in vision functions	Fully implemented vision operators and framework 'out of the box'
Architecture	Language-based – needs online compilation	Library-based - no online compiler required
Target Hardware	'Exposed' architected memory model – can impact performance portability	Abstracted node and memory model - diverse implementations can be optimized for power and performance
Precision	Full IEEE floating point mandated	Minimal floating point requirements – optimized for vision operators

It is possible to use OpenCL to build OpenVX Nodes on programmable devices
BUT - do we need definition of an efficient, vision-capable OpenCL Device?
Precisely defined precision for image and vision operations?

© Copyright Khronos Group 2014 - Page 19

Need for Camera Control API

- **We have choice of APIs for image and vision image processing**
  - BUT no open standard API for camera control to FEED these APIs!
- **Need advanced control of ISP and camera subsystem**
  - Generate sophisticated image stream for advanced imaging & vision apps
- **Khronos Camera Control API in development!**
  - Advanced, high-frequency burst control of camera and sensor operation
- **EGLStreams provides efficient streaming of images between APIs**
  - OpenCL needs efficient handling of EGLStreams



Mixamo - Avatar Videoconferencing

- Real time facial animation capture on mobile - ported directly from PC
- Animate an avatar while conferencing
- Full GPU acceleration of vision processing using OpenCL

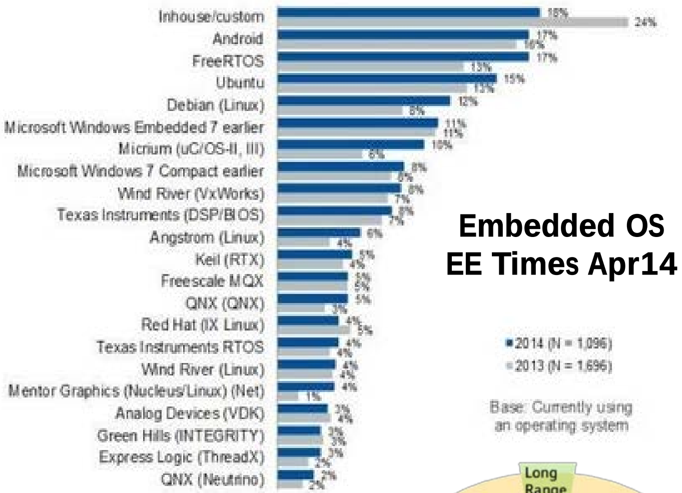
mixamo

NVIDIA Tegra K1 Development Board

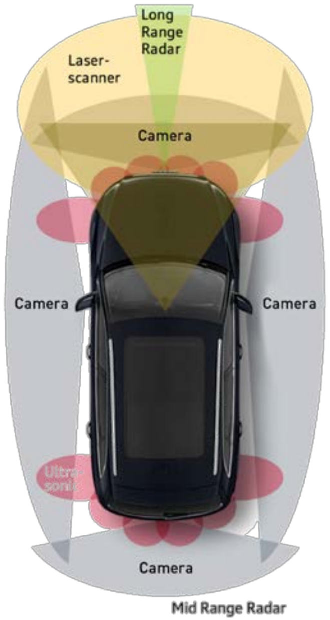
KHRONOS GROUP

Embedded Compute

- **Embedded computing will be everywhere**
  - Automotive - ADAS
  - Sensor processing – IOT
  - Etc. etc…
- **Android/Linux increasingly popular embedded OS**
  - Already have OpenCL support
- **Many embedded processors will be tiny**
  - But will need heterogeneous processing
  - But won’t need IEEE floating point
- **Security will be key for many embedded apps**
  - Life Critical apps
- **Software Certification**
  - Safety Critical subset - like OpenGL SC?



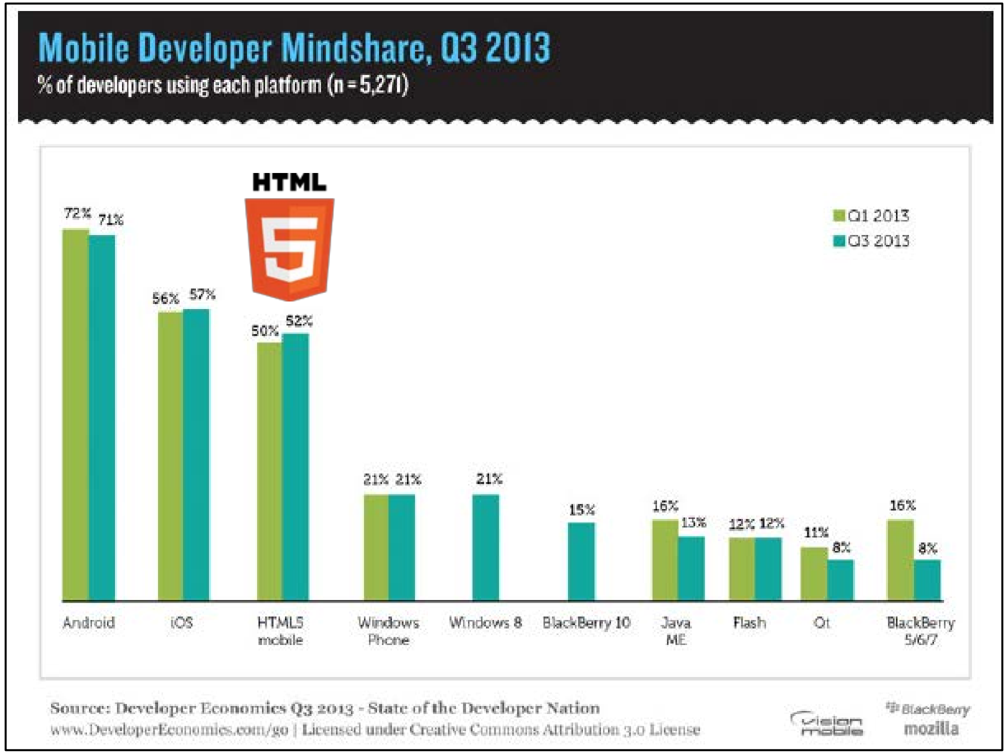
Embedded OS
EE Times Apr14



Will OpenCL reach down into these new resource constrained markets?

Web Acceleration APIs

- **Khronos and W3C liaison for Web APIs**
  - Leverage proven native APIs
  - Fast API development/deployment
  - Designed by hardware community
  - Familiar foundation reduces developer learning curve



	WebGL	WebCL	HTML Canvas / SVG	WebVX? Vision Processing	WebStream? Sensor Fusion	WebCAM(!) Camera control and video processing
JavaScript	WebGL	WebCL	HTML Canvas, SVG	WebVX? Vision Processing	WebStream? Sensor Fusion	WebCAM(!) Camera control and video processing
Native	OpenGL ES	OpenCL	OpenGL Path Rendering	OpenVX	StreamInput	Camera Control

Native APIs shipping or Khronos working group

JavaScript API shipping, acceleration being developed or work underway

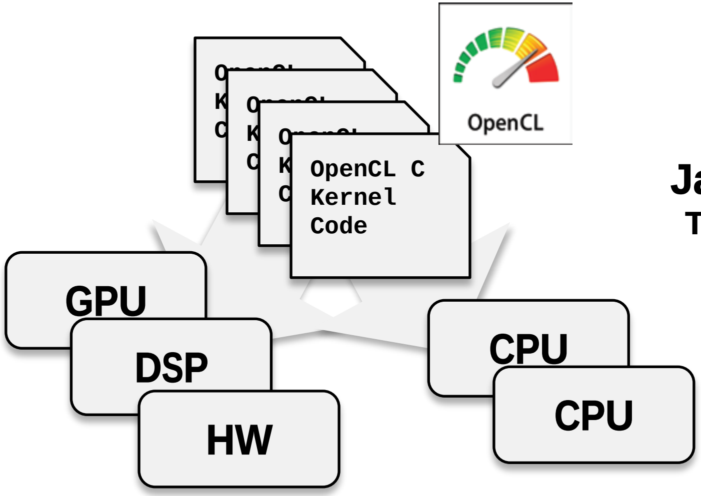
Possible future JavaScript APIs or acceleration

WebCL: Heterogeneous Computing for the Web

- **WebCL 1.0 specification officially finalized at GDC March 2014**
  - https://www.khronos.org/webcl
- **WebCL defines JavaScript binding to the OpenCL APIs**
  - Enables initiation of Kernels written in OpenCL C within the browser
- **Typical Use Cases**
  - 3D asset codecs, video codecs and processing, imaging and vision processing
  - Physics for WebGL games, Online data visualization, Augmented Reality

JavaScript Platform API

To query, select and initialize compute devices



JavaScript Runtime API

To build and execute kernels across multiple devices

WebGL/WebCL Ecosystem

Low-level APIs provide a powerful foundation for a rich JavaScript middleware ecosystem

Content downloaded from the Web

Content
JavaScript, HTML, CSS, ...

Middleware can make WebGL and WebCL accessible to non-expert programmers
E.g. three.js library: http://threejs.org/ used by majority of WebGL content

JavaScript Middleware

Browser provides WebGL and WebCL
Alongside other HTML5 technologies
No plug-in required

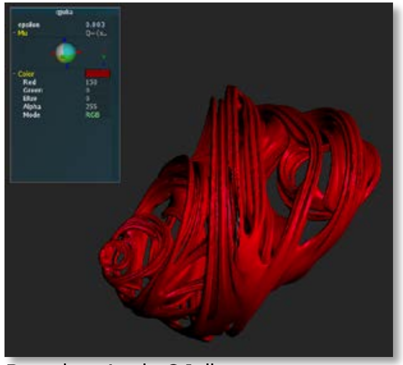
WebGL | **WebCL**
JavaScript | **HTML5 / CSS**

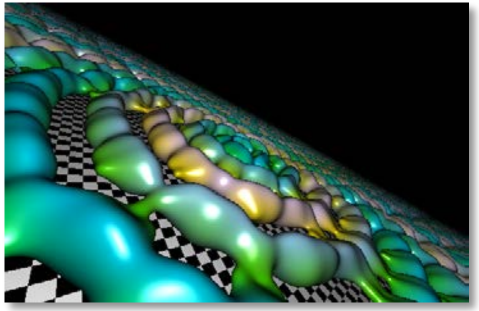
OS Provided Drivers
WebGL uses OpenGL ES 2.0 or Angle for OpenGL ES 2.0 over DX9
WebCL uses OpenCL 1.X

OpenGL ES | OpenGL | OpenCL

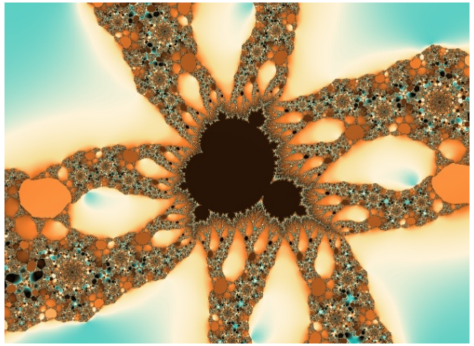
Open Source Implementations and Resources

- **WebCL Conformance Framework and Test Suite (contributed by Samsung)**
  - https://github.com/KhronosGroup/WebCL-conformance/
- **Nokia - Firefox build with integrated WebCL**
  - Firefox extension, open sourced May 2011 (Mozilla Public License 2.0)
  - https://github.com/toaarnio/webcl-firefox
- **Samsung - uses WebKit, open sourced June 2011 (BSD)**
  - https://github.com/SRA-SiliconValley/webkit-webcl
- **Motorola Mobility - uses Node.js, open sourced April 2012 (BSD)**
  - https://github.com/Motorola-Mobility/node-webcl
- **AMD – uses Chromium (open source)**
  - https://github.com/amd/Chromium-WebCL


Based on Apple QJulia


Based on Iñigo Quilez, Shader Toy


Based on Iñigo Quilez, Shader Toy


http://fract.ured.me/

© Copyright Khronos Group 2014 -

WebCL - Parallel Computation for the Web

http://www.youtube.com/user/SamsungSISA#p/a/u/1/9Ttux1A-Nuc

WebCL - Designed-in Architectural Security

- **Leverages OpenCL 1.2 robustness/security extensions**
  - Context Termination: to prevent DoS from long running kernels
  - Memory Initialization: no leakage from out of bounds memory access
- **API and Language Restrictions**
  - Not supported: structures as Kernel arguments, Kernel names>256 characters, mapping of CL memory objects into host memory, program binaries, some OpenCL API calls and built-ins
- **WebCL Kernel Validator** https://github.com/KhronosGroup/webcl-validator
  - Open source - provided as a "library API" for easy integration into browsers
  - Parses and validates kernel code against specification
  - Initializes local/private memory if underlying OpenCL implementation does not
  - Tracks memory allocations and traces valid ranges for reads and writes
  - Run time checks to make all memory accesses safe

Wrap Up - Broadening OpenCL's Impact

Now is a great time for input and feedback
We do read the forums!
Or join Khronos and have a voice at the working group!

SPIR and SYCL
Evolve C++ support into core
Converge SPIR and LLVM

Foundation API
Evolve memory
and Execution Models
for all markets

Lots of apps, languages and frameworks
Better debug, profiling
and trans-API Interop

Lots of OpenCL Devices
Drive availability of
latest API version.
Minimize extensions and
optional features

Heterogeneous Mobile SOCS
Drive mobile platform adoption
Fully Embrace DSPs and
Dedicated Hardware

WebCL
Needs pervasive OpenCL
including Android

Vision Processing
Vision friendly precision
and stream handling

Embedded
Precision and
resource handling for
small processors

Security
Memory and execution
security mechanisms

Safety Critical
Functionality
subset for
certification