Systems 2030: The Extended Reality Case

Sarita Adve

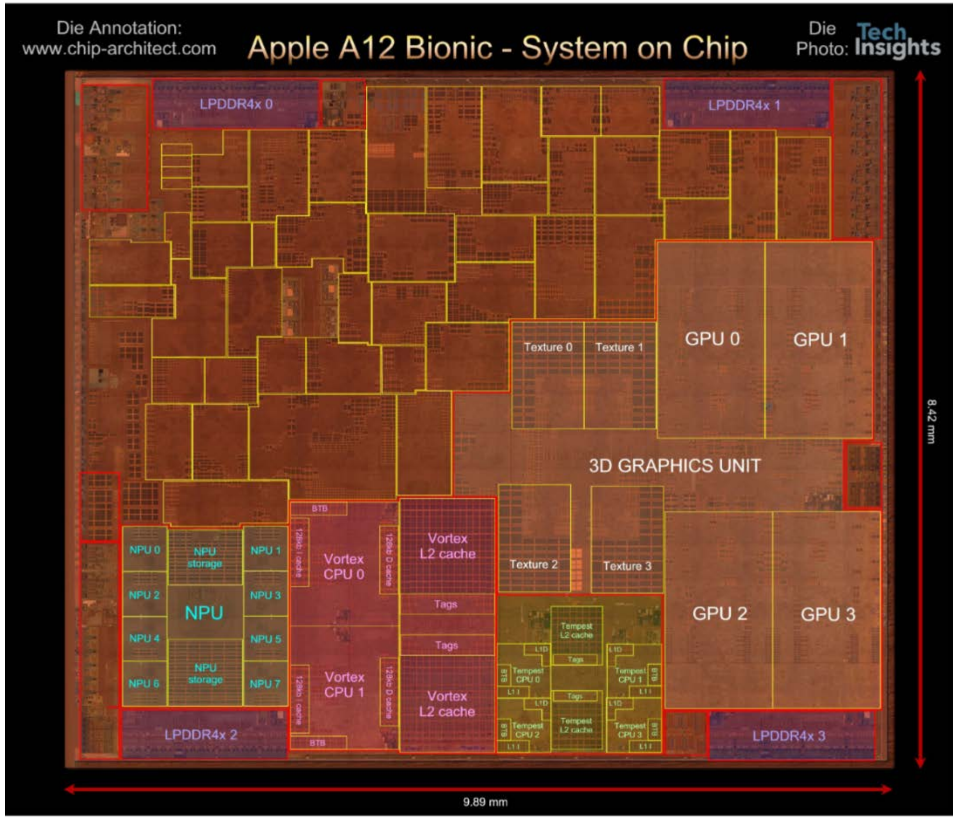
University of Illinois at Urbana-Champaign

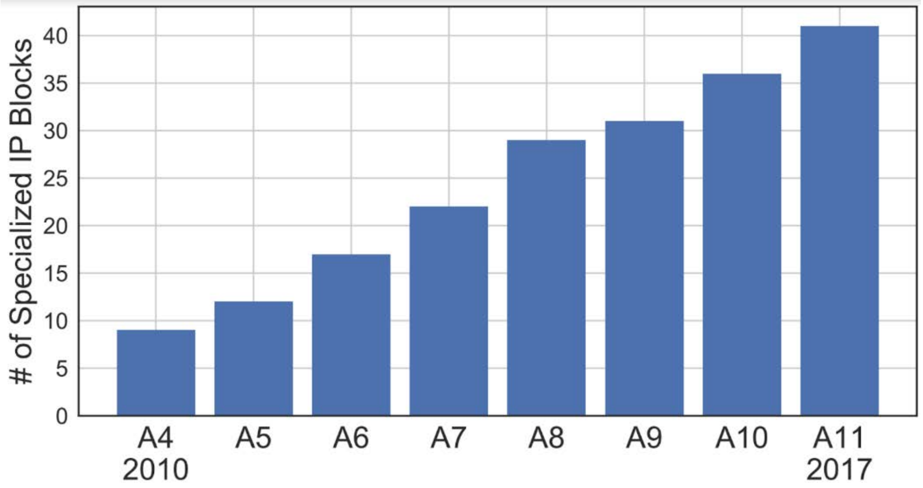
sadve@illinois.edu

w/ many collaborators acknowledged on slides

This work is supported in part by DARPA, NSF, and the Applications Driving Architecture (ADA) Research center (JUMP center co-sponsored by SRC & DARPA)

Era of Specialization





Source: Brooks, Wei group, http://vlsiarch.eecs.harvard.edu/accelerators/die-photo-analysis

Explosion of accelerators in SoCs (System-on-Chip)

~~How to design specialized accelerators?~~

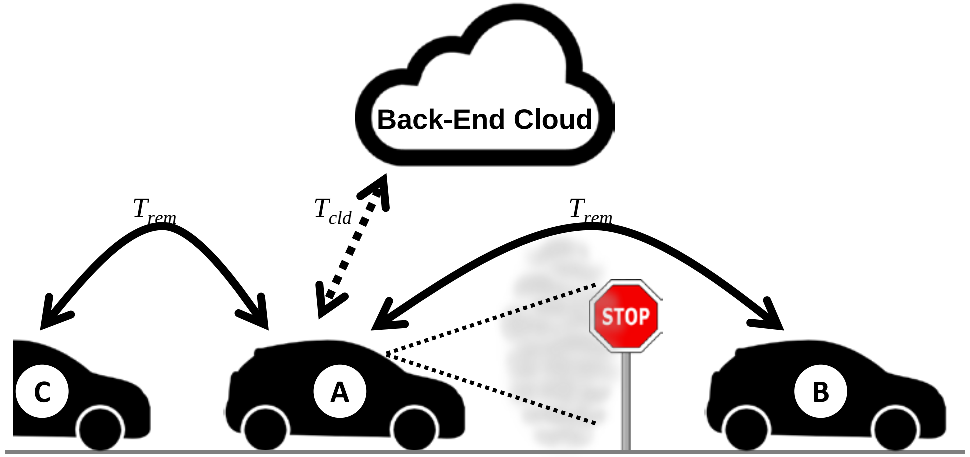
How to design specialized system?

- Communication?
- Software?
- Applications?

Rise of the Edge

VR@Illinois

Chowdhari et al., EarthSense robots

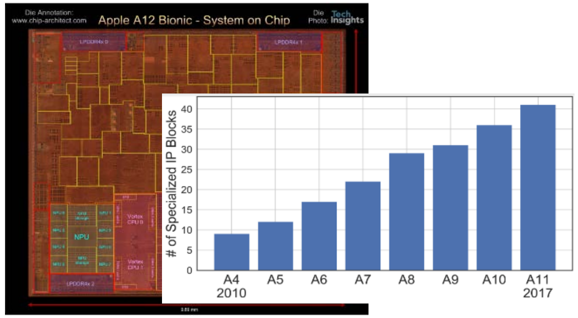


EPOCHS project, IBM, Columbia, Harvard, Illinois

Challenging performance demands

Stringent resource constraints

End-to-end quality metrics



+

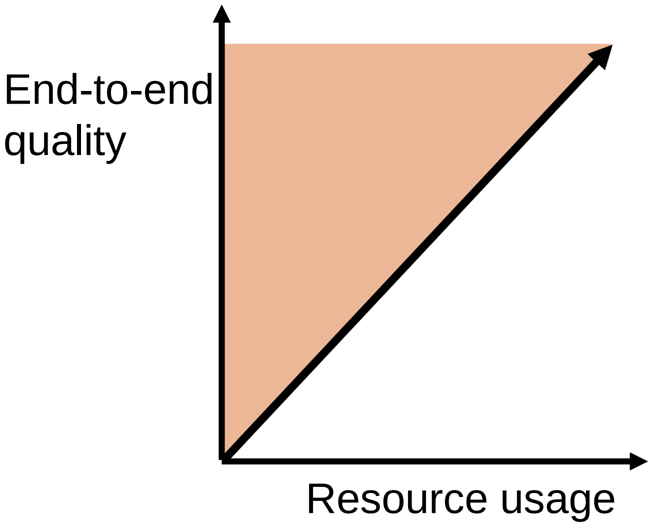
⇒

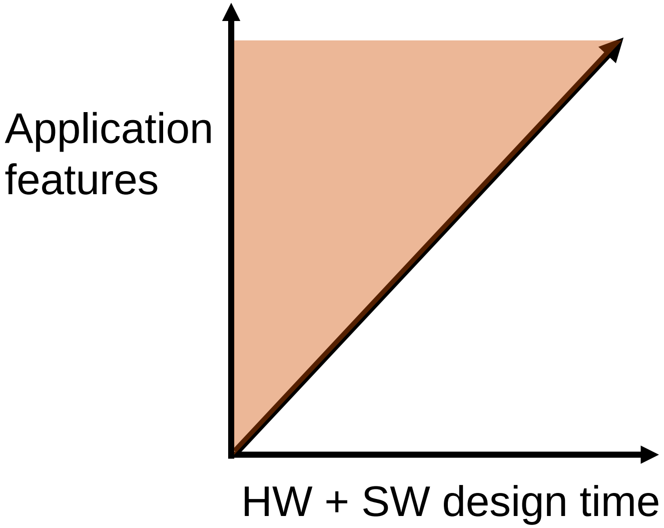
Scalable,

Generalizable

Specialization Techniques

Scalable





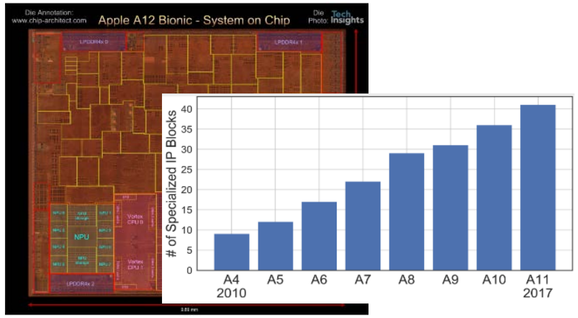
Generalizable

Application-driven,

end-to-end quality driven,

HW-SW-App co-designed system

Systems 2030



+

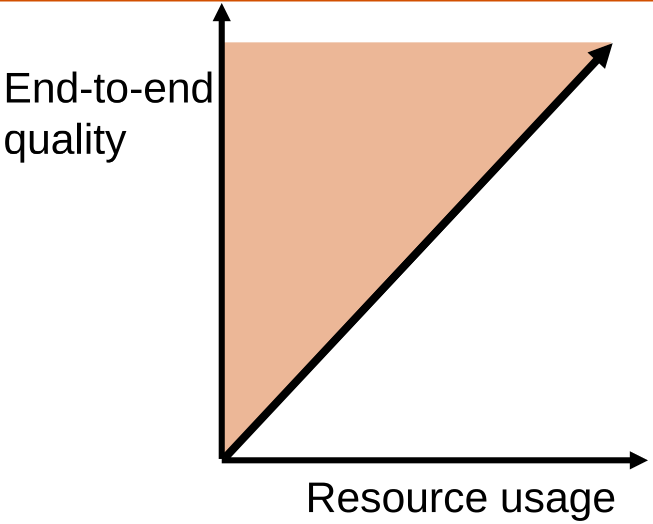
⇒

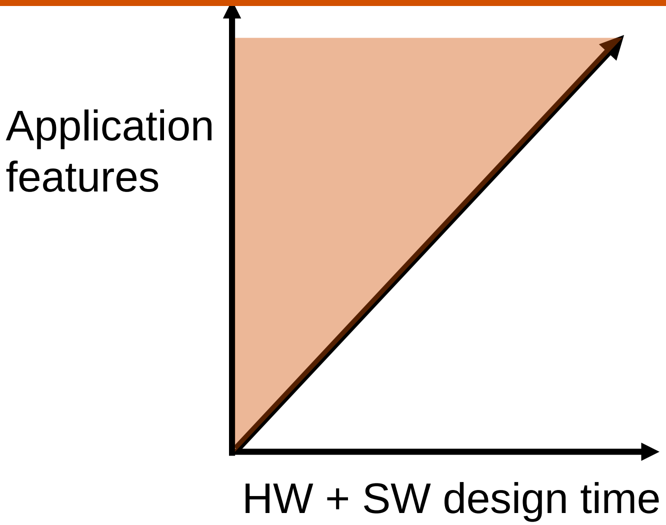
Scalable,

Generalizable

Specialization Techniques

BUT What Application?





Application-driven,

end-to-end quality driven,

HW-SW-App co-designed system

Systems 2030

Systems 2030: The Extended Reality Case

Extended Reality (XR) = Virtual, augmented, mixed, … reality

Pervasive: Science, medicine, entertainment, education, …

Challenging: Orders of magnitude power, performance, quality gap to reach ideal

Diverse: Involves graphics, vision, audio, video, robotics, optics, haptics, …

Full stack: Challenges span hardware, compiler, OS, algorithm

Flexible: User-driven, end-to-end quality of experience (QoE) metrics

Great driver for research for Systems 2030

XR Requirements

	VR		AR	
	HTC Vive Pro	Ideal	HoloLens 2	Ideal
Resolution (Mpixels)	4.6	200	4.4	200
Field of view (degrees)	110	Full: 165x175 Stereo: 120x135	52 diagonal	Full: 165x175 Stereo: 120x135
Refresh rate (Hz)	90	90-144	120	90-144
Motion to photon lat (ms)	< 20	< 20	< 2	< 5
Power (W)	N/A (server)	1 - 2	> 7	0.1 - 0.2
Silicon area (mm2)	N/A (server)	100-200	> 173	< 100
Weight (g)	470	100-200	566	10s

Orders of magnitude gap in power, performance, area, weight, QoE

Challenges for XR Systems Research

- Needs expertise from many domains

- Closed interfaces
  - Recent: OpenXR opened application-runtime interface

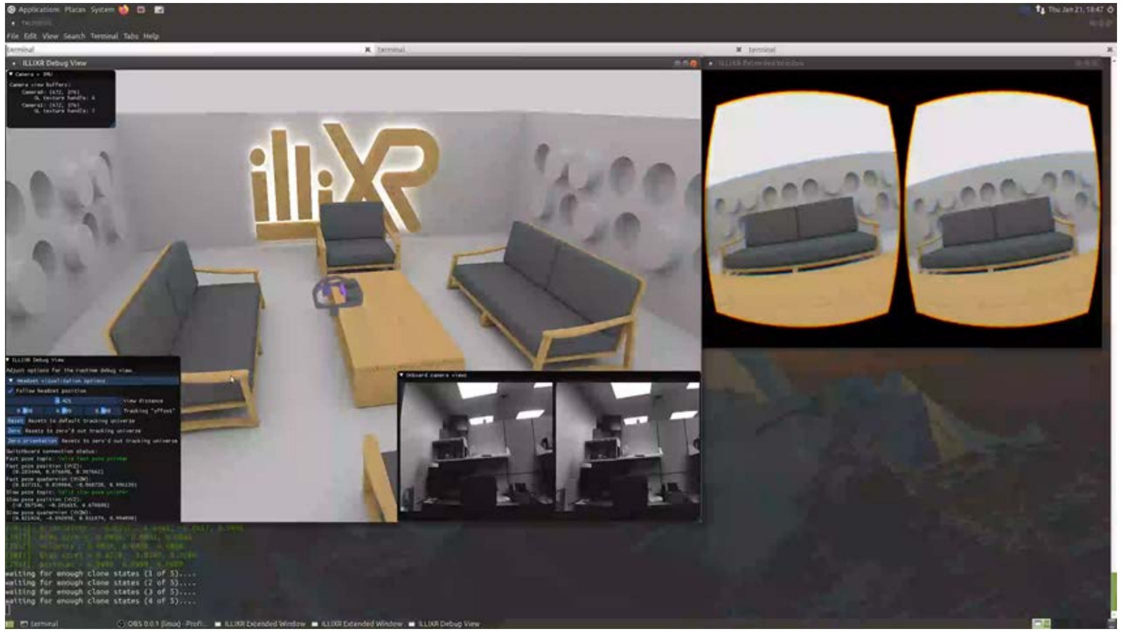
- State-of-the-art closely guarded by industry
  - ~~No open-source benchmarks or systems~~

ILLIXR: Illinois Extended Reality Testbed [Huzaifa et al., ‘20]

- ILLIXR: First open-source full system XR testbed
- State-of-the-art XR components integrated with modular and extensible runtime
- OpenXR compatible
- Several QoE metrics
- Runs on desktops, embedded systems

Soon: Community Consortium

- Industry + academic partners
  - ARM, Facebook, Micron, NVIDIA, …
- Standardize benchmarking, QoE metrics, …



illixr.github.io

A new playground for systems 2030 and XR research

Team ILLIXR

- Rishi Desai
- Samuel Grayson
- Muhammad Huzaifa
- Xutao Jiang
- Ying Jing
- Jae Lee
- Fang Lu
- Yihan Pang
- Joseph Ravichandran
- Giordano Salvador
- Finn Sinclair
- Boyuan Tian
- Lauren Wagner
- Henghzi Yuan
- Jeffrey Zhang

External Consultations

- Wei Cui
- Aleksandra Faust
- Liang Gao
- Matt Horsnell
- Amit Jindal
- Steve LaValle
- Steve Lovegrove
- Andrew Maimone
- Vegard Oye
- Martin Persson
- Archontis Politis
- Eric Shaffer
- Paris Smaragdis
- Chris Widdowson

Team ILLIXR

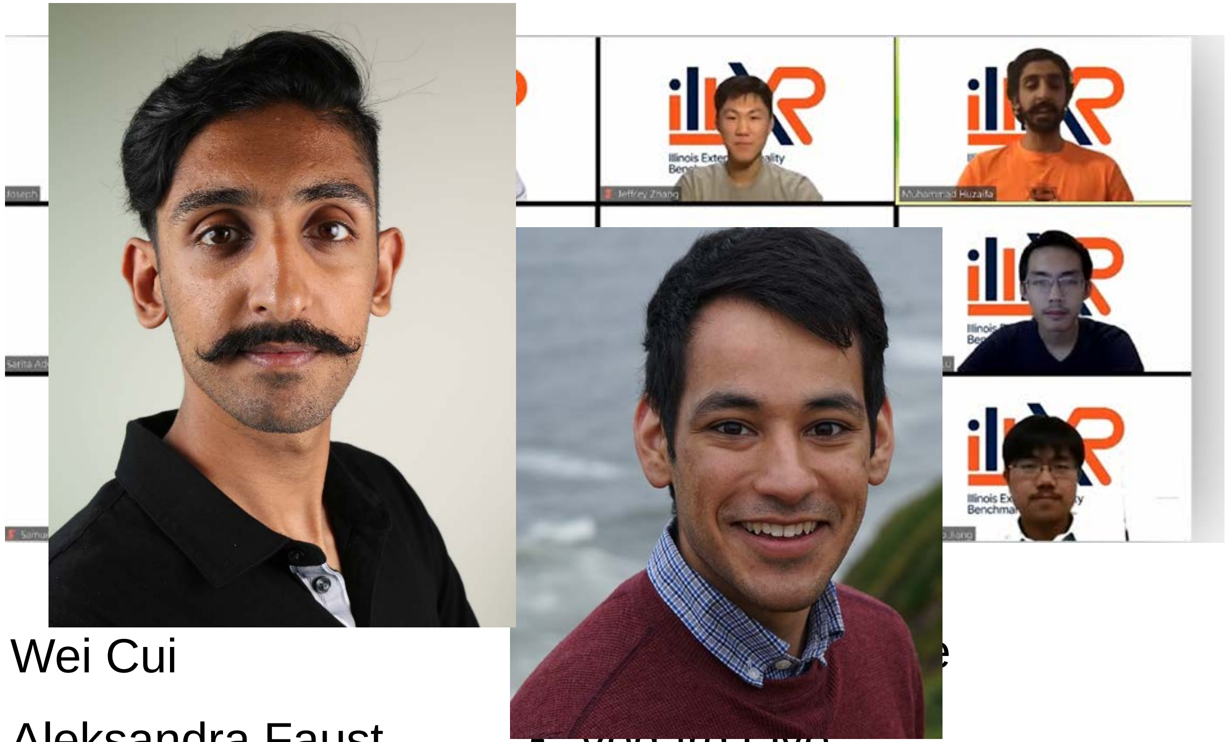
- Rishi Desai
- Samuel Grayson
- Muhammad Huzaifa
- Xutao Jiang
- Ying Jing
- Jae Lee
- Fang Lu
- Yihan Pang
- Joseph Ravichandran
- Giordano Salvador
- Finn Sinclair
- Boyuan Tian
- Lauren Wagner
- Henghzi Yuan
- Jeffrey Zhang

External Consultations

- Wei Cui
- Aleksandra Faust
- Liang Gao
- Matt Horsnell
- Amit Jindal
- Steve LaValle
- Steve Lovegrove
- Andrew Maimone
- Vegard Oye
- Martin Persson
- Archontis Politis
- Eric Shaffer
- Paris Smaragdis
- Chris Widdowson

Team ILLIXR

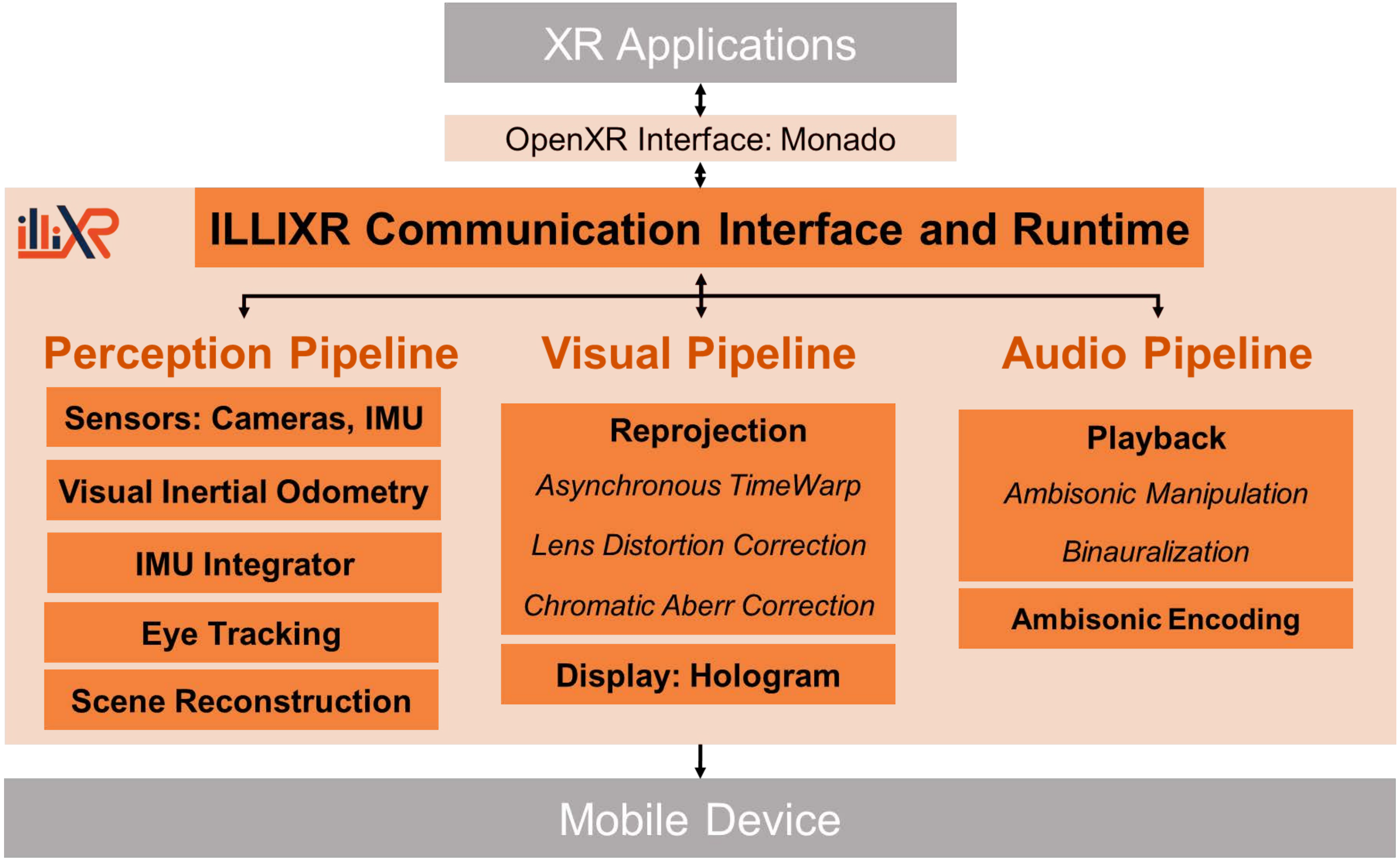
- Rishi Desai
- Samuel Grayson
- Muhammad Huzaifa
- Xutao Jiang
- Ying Jing
- Jae Lee
- Fang Lu
- Yihan Pang
- Joseph Ravichandran
- Giordano Salvador
- Finn Sinclair
- Boyuan Tian
- Lauren Wagner
- Henghzi Yuan
- Jeffrey Zhang



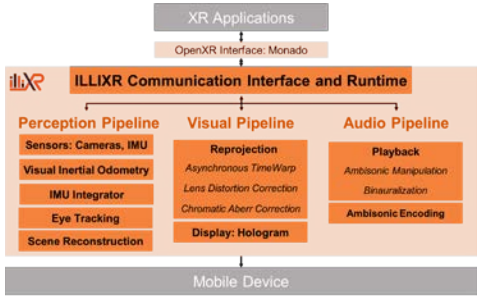
External Consultations

- Wei Cui
- Aleksandra Faust
- Liang Gao
- Matt Horsnell
- Amit Jindal
- Steve LaValle
- Steve Lovegrove
- Vegard Oye
- Martin Persson
- Archontis Politis
- Eric Shaffer
- Paris Smaragdis
- Chris Widdowson

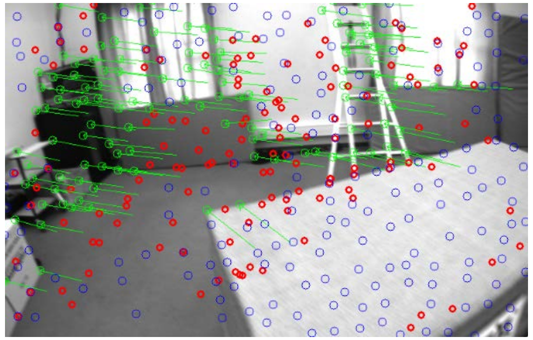
ILLIXR Overview

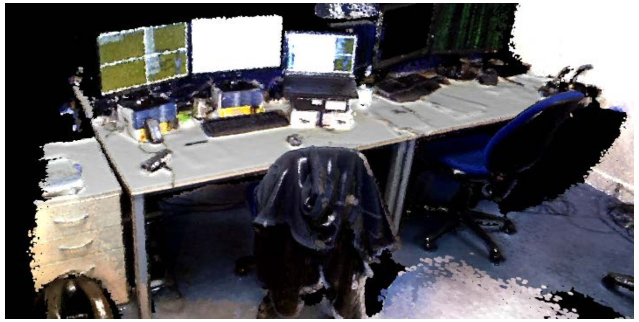


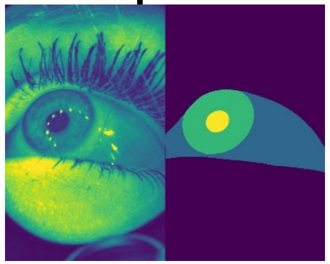
Perception Pipeline



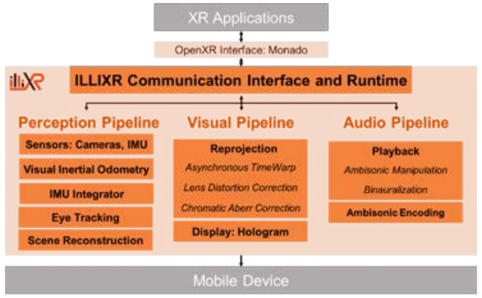
- Sensors: Camera, Inertial Measurement Unit (IMU)
- Visual Intertial Odometry (VIO)
  - Provides position and orientation of user's head (pose)
- IMU Integrator
  - Provides high frequency pose estimates
- Pose Predictor
  - Extrapolates pose to future timestamp
- Scene Reconstruction
  - Uses RGB-Depth camera to build dense 3D map of world
- Eye Tracking



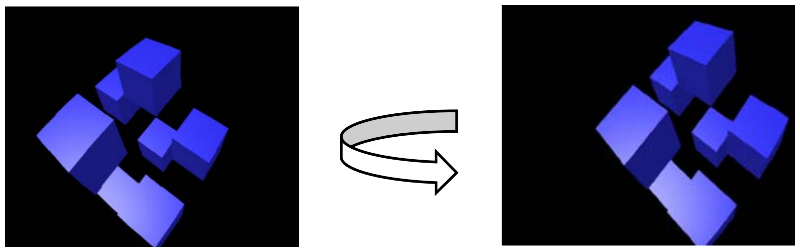




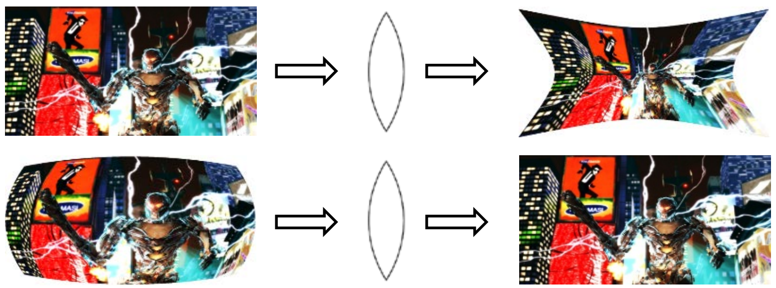
Visual Pipeline



- Asynchronous reprojection (TimeWarp)
  - Warp rendered frame to account for head movement during rendering
  - Uses latest pose estimate and prediction
  - Cuts motion-to-photon latency

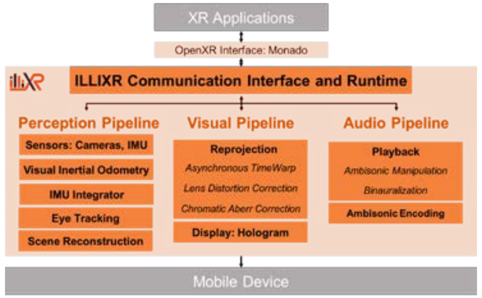


- Lens distortion and chromatic aberration correction
  - Corrects for distortion due to curved lenses



- Adaptive display: hologram
  - Vergence-accommodation conflict (VAC) causes fatigue, headache
    - Eyes focused (accommodated) at fixed point, converge at different points
  - Computational displays w/ multiple focal planes can fix VAC
  - Computational holography: per-pixel phase shift

Audio Pipeline

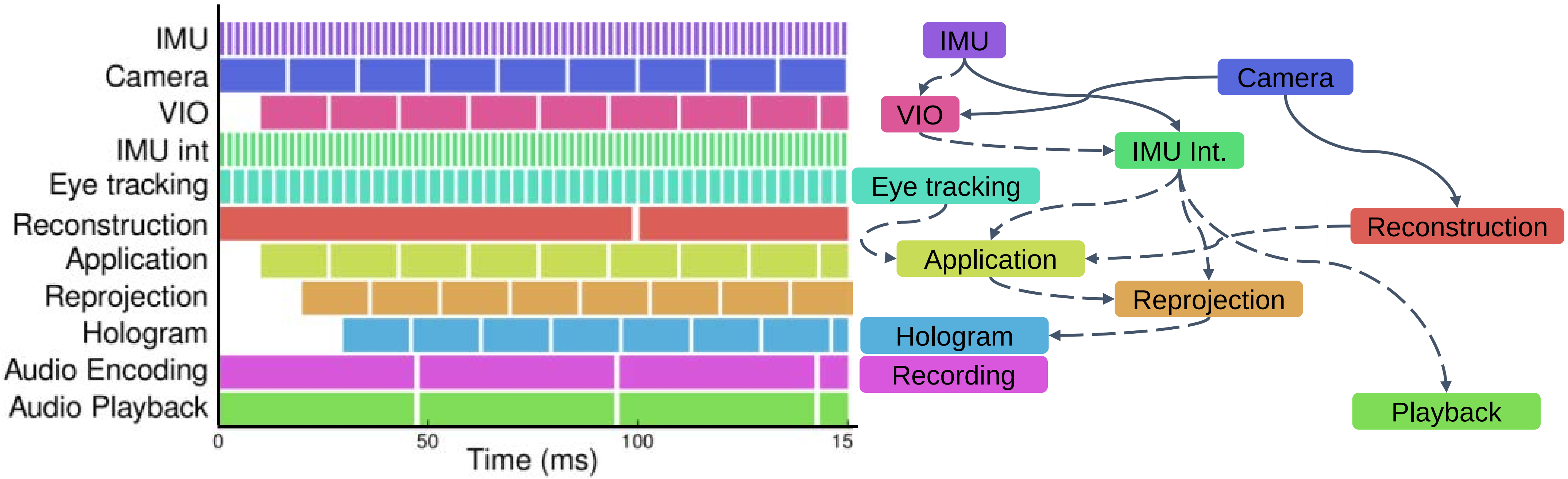


- Audio encoding
  - – Encodes multiple sound sources into Higher Order Ambisonics (HOA) soundfield
- Playback
  - – Rotates and zooms HOA sound field for user’s latest pose
  - – Performs binauralization to account for user’s ear, head, nose

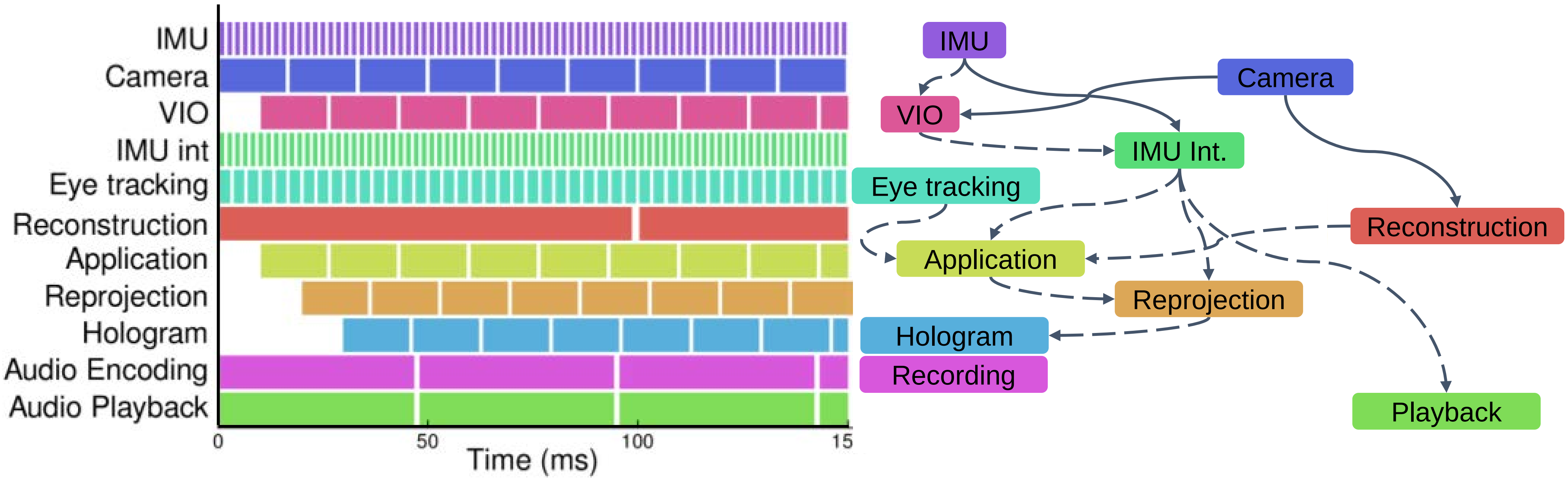
BUT XR is not just a collection of components

It is a SYSTEM

XR System Dataflow

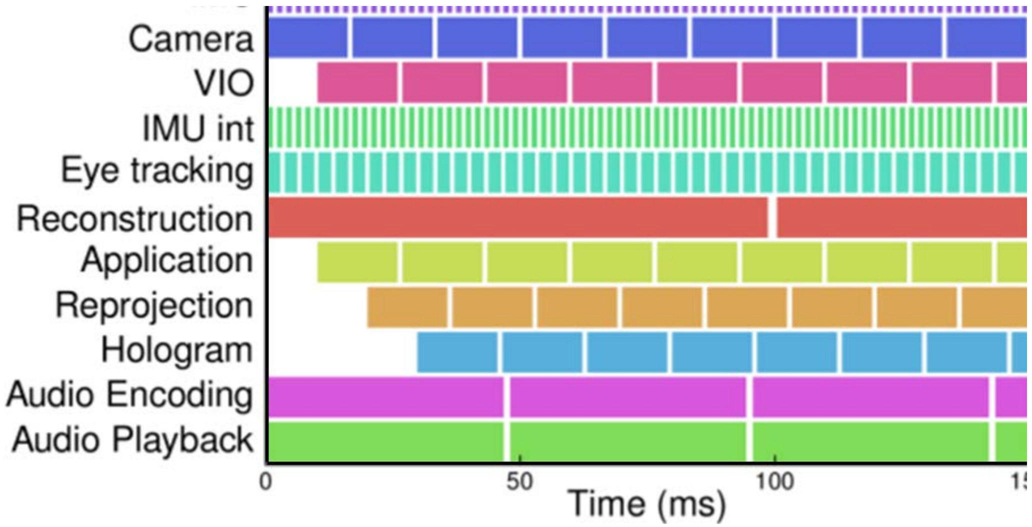


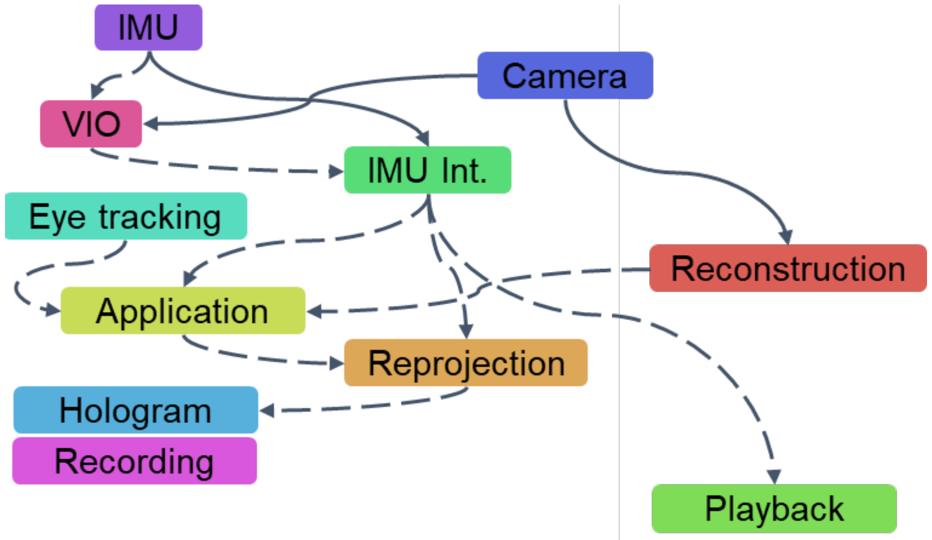
XR System Dataflow



Different components at different frequencies
Multiple interacting pipelines
Synchronous and asynchronous dependences
Multiple quality of experience metrics

ILLIXR Runtime





Modular, flexible architecture

ILLIXR components are plugins

Separately compiled, dynamically loaded

Easily swap/add new components, implementations

Efficient, flexible communication interface

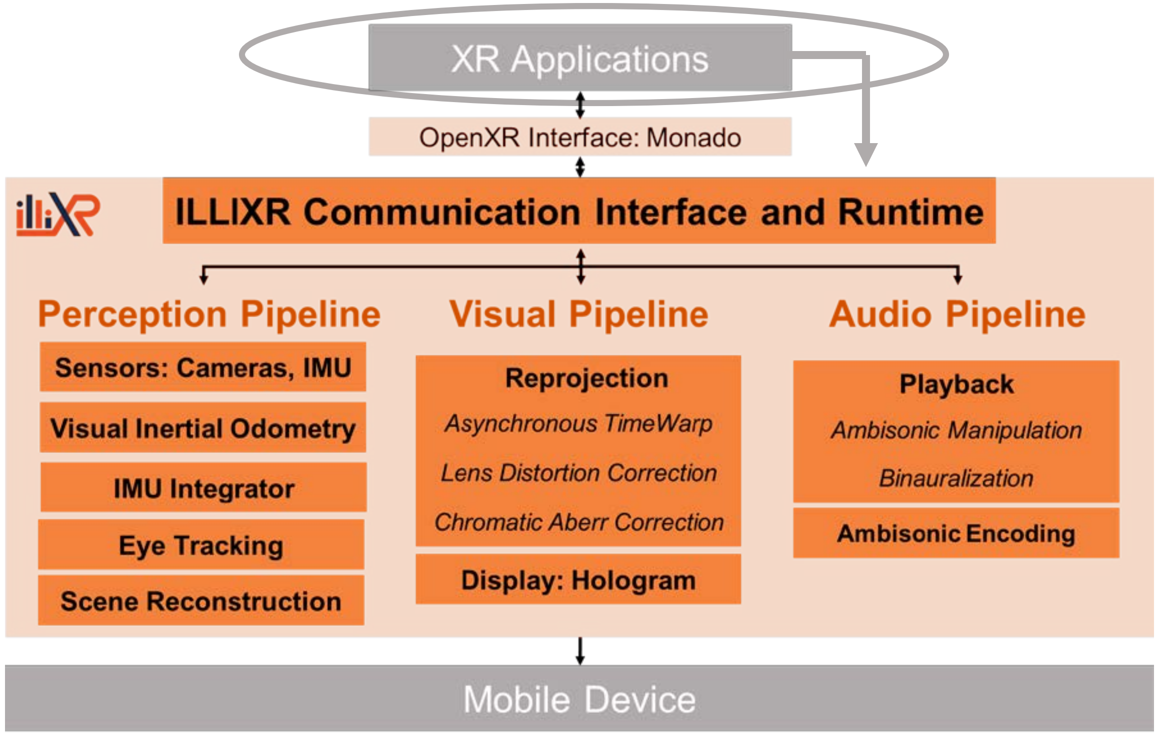
Component specifies event streams to publish, subscribe

Synchronous or asynchronous consumers

Copy-free, shared memory implementation

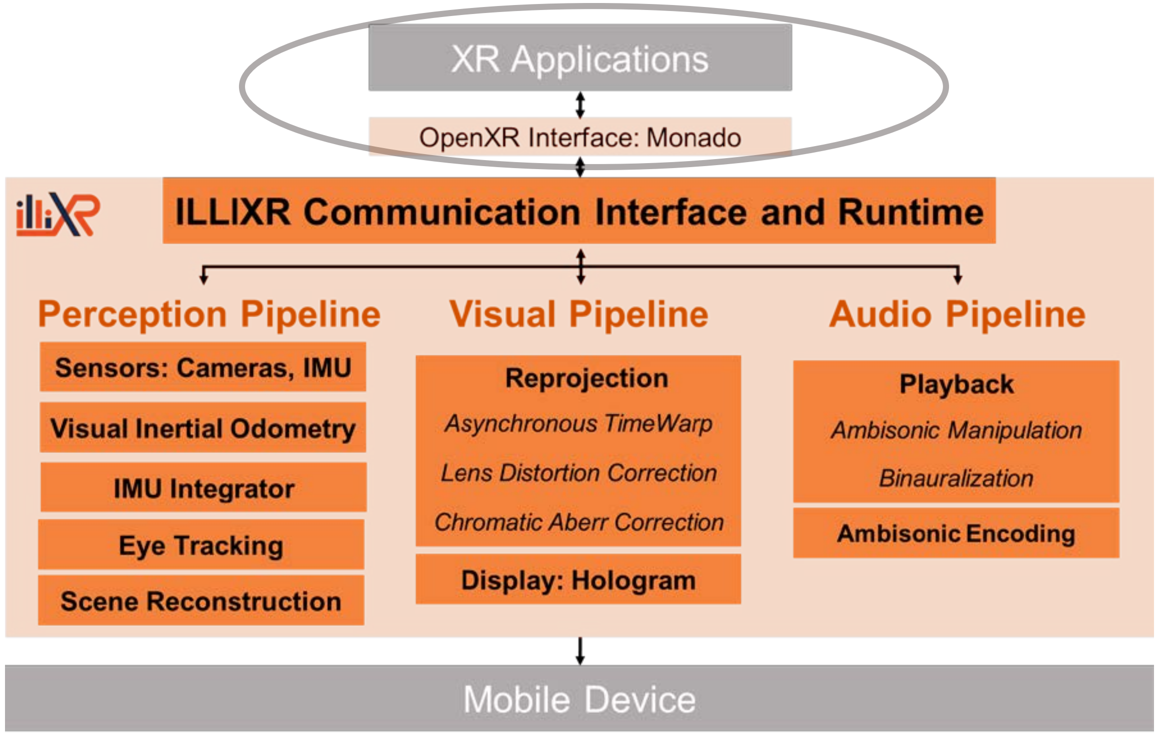
End-to-end system balances flexibility with efficiency

ILLIXR Applications



Can write XR applications directly to ILLIXR

ILLIXR Applications



Can write XR applications directly to ILLIXR

ILLIXR supports OpenXR applications

- Uses Monado implementation of OpenXR
- Today: Godot game engine with many apps
- Soon: Unity, Unreal, …

End-to-End Quality Metrics

- Motion-to-photon latency
  - Time from head motion to display

- Image quality: SSIM and FLIP

- Pose: Average Trajectory Error and Relative Pose Error

+ Extensive telemetry: Frame rates, missed frames, time distributions, power, …

ILLIXR Components Today

	Component	Algorithm	Implementation
Perception Pipeline	Camera Camera	ZED SDK Intel RealSense SDK	C++ C++
	IMU IMU	ZED SDK Intel RealSense SDK	C++ C++
	VIO VIO	OpenVINS Kimera-VIO	C++ C++
	IMU Integrator IMU Integrator	RK4 GTSAM	C++ C++
	Eye Tracking	RITnet	Python, CUDA
	Scene Reconstruction Scene Reconstruction	ElasticFusion KinectFusion	C++, CUDA, GLSL C++, CUDA
Visual Pipeline	Reprojection	VP-matrix reproject w/ pose	C++, GLSL
	Lens Distortion	Mesh-based radial distortion	C++, GLSL
	Chromatic Aberration	Mesh-based radial distortion	C++, GLSL
	Adaptive Display	Weighted Gerchberg-Saxton	CUDA
Audio Pipeline	Audio Encoding	Ambisonic encoding	C++
	Audio Playback	Ambisonic manipulation, binauralization	C++

Evaluation Methodology

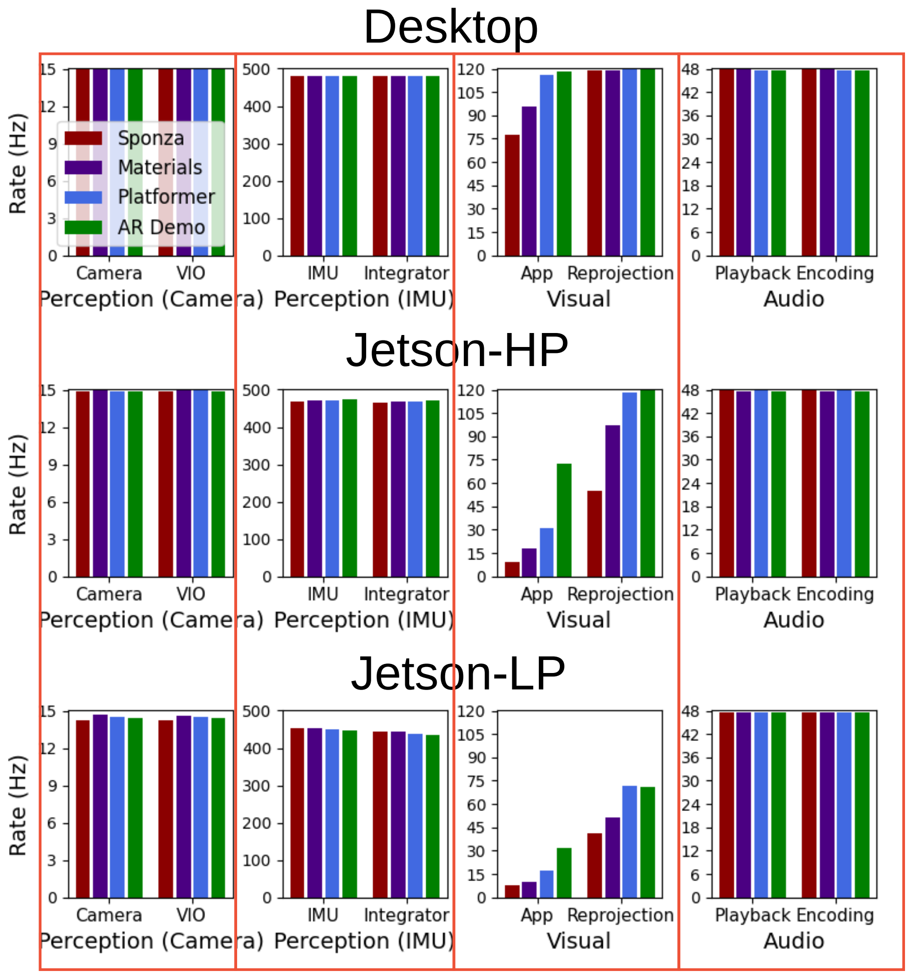
Component	Parameter	Range	Tuned	Deadline
Camera (VIO)	Frame rate Resolution Exposure	15 – 100 Hz VGA – 2K 0.2 – 20 ms	15 Hz VGA 1 ms	66.7 ms – –
IMU (Integrator)	Frame rate	≤ 800 Hz	500 Hz	2 ms
Display (Visual pipeline + Application)	Frame rate Resolution Field-of-view	30 – 144 Hz ≤ 2K ≤ 180°	120 Hz 2K 90°	8.33 ms – –
Audio (Encoding + Playback)	Frame rate Block size	48 – 96 Hz 256 – 1024	48 Hz 1024	20.8 ms –

- Platforms
  - High-end desktop machine
  - Embedded: NVIDIA Jetson-HP (high performance) and Jetson-LP (low power)
- Applications: Sponza, Materials, Platformer, AR Demo on Godot game engine

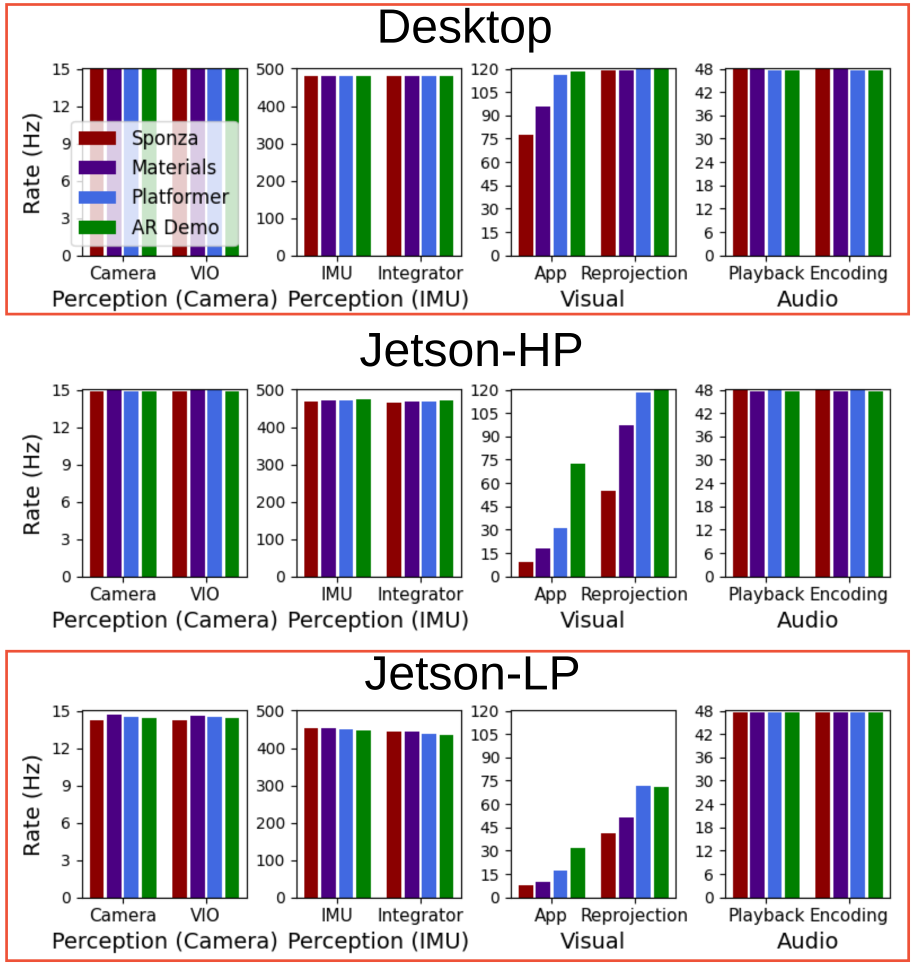
High ⟷ Low

Graphics intensity

Frame Rate

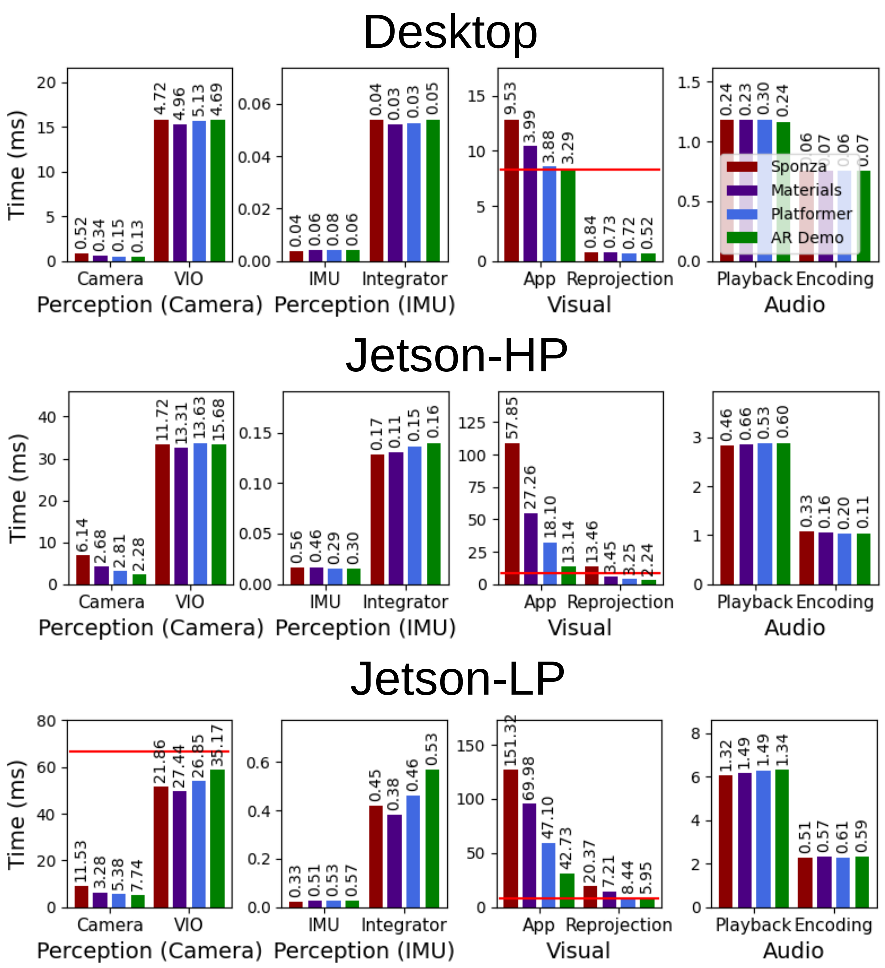

Desktop
Jetson-HP
Jetson-LP
Rate (Hz)
Sponza
Materials
Platformer
AR Demo
Camera
VIO
Perception (Camera)
IMU
Integrator
Perception (IMU)
App
Reprojection
Visual
Playback
Encoding
Audio

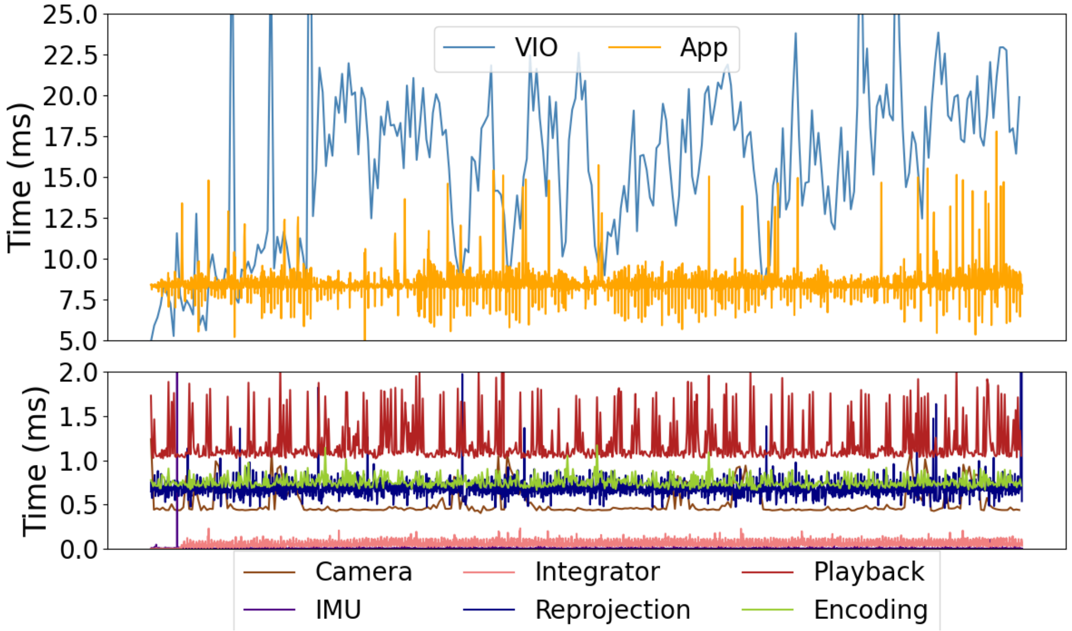
Frame Rate



- Desktop meets performance
  - But at what power cost?
- Jetson-LP can run only audio at target fps
- Gap will increase as displays and components scale

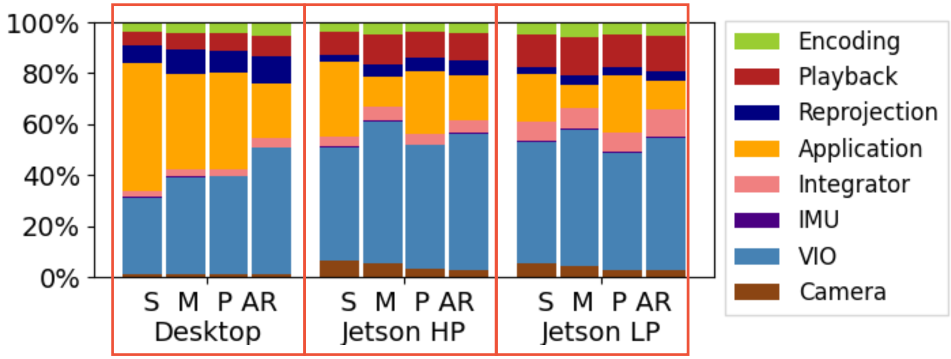
Time Per Frame





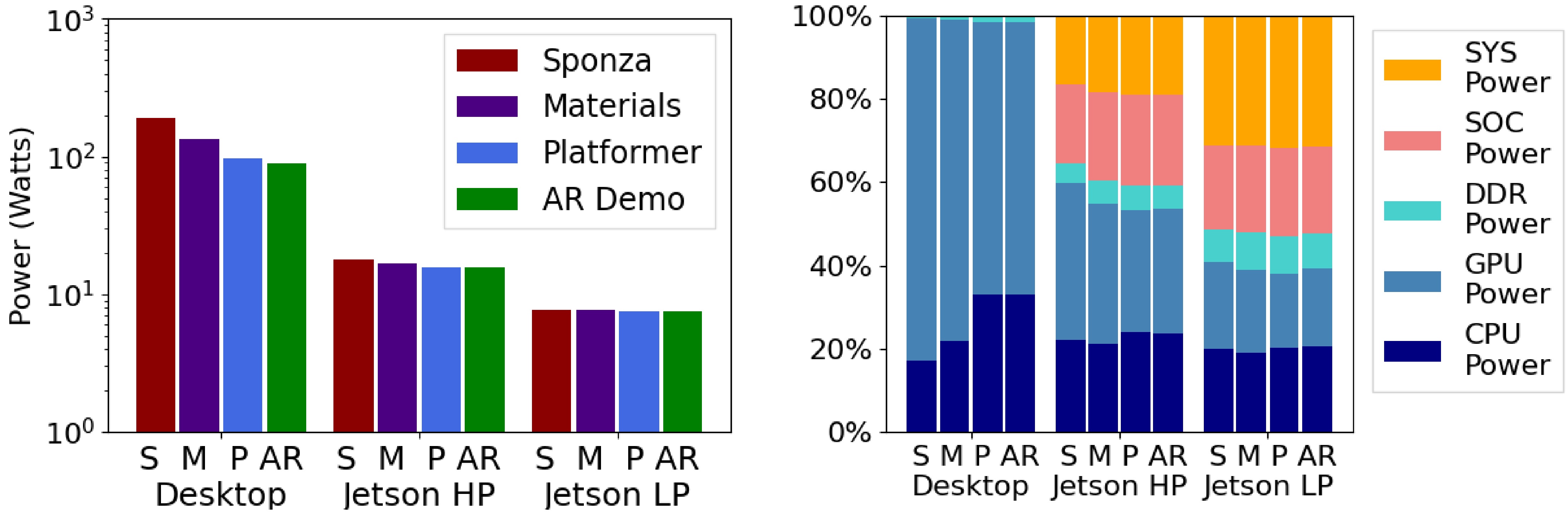
Input-dependence, scheduling, and resource contention lead to significant variability

Distribution of Cycles



- Application and VIO dominate
- Reprojection and integrator take little time, but critical for QoE
- All components and metrics must be considered together

Power

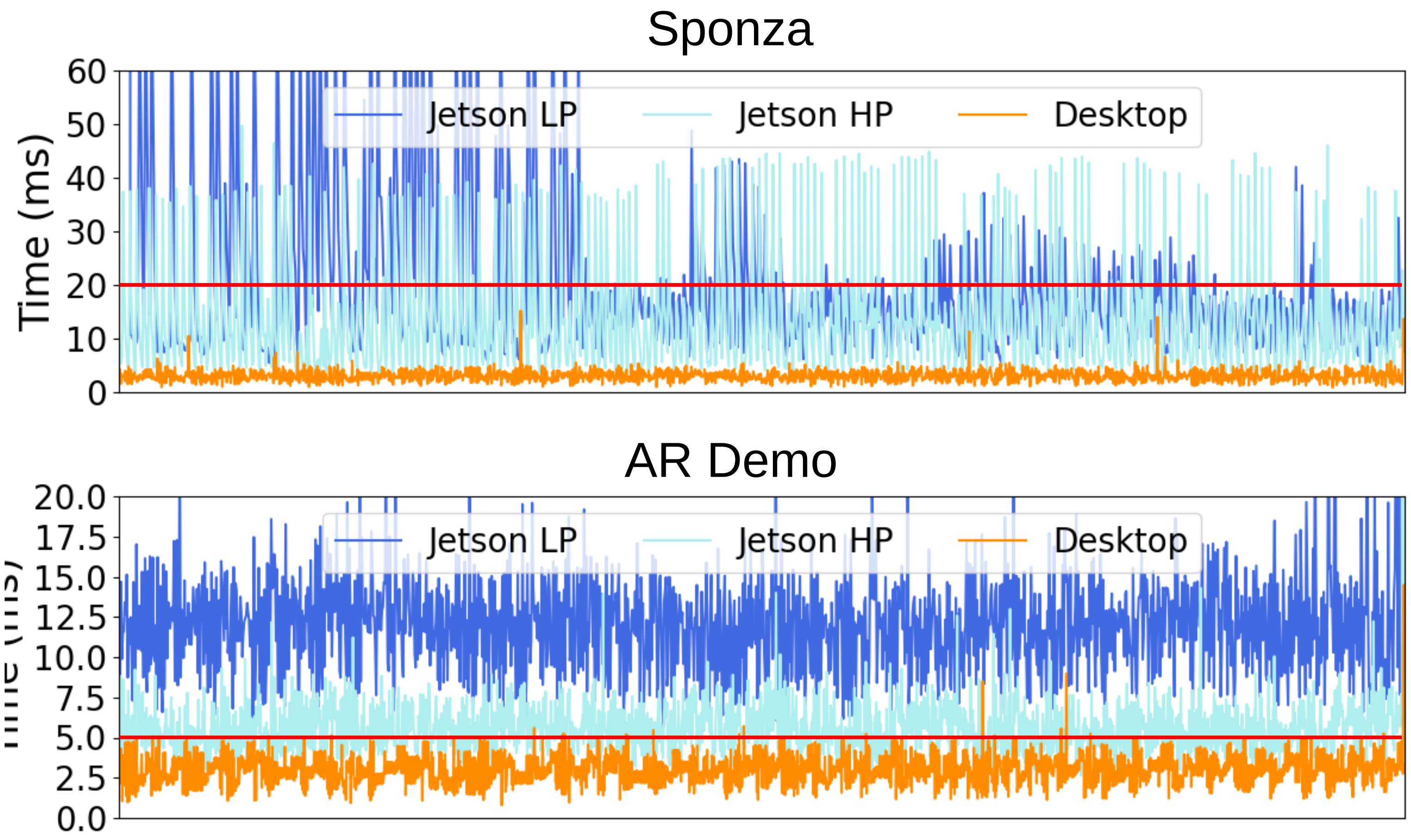


Must consider system-level components such as display and I/O

Motion-to-Photon Latency

Application	Desktop	Jetson-hp	Jetson-lp
Sponza	3.1 ± 1.1	13.5 ± 10.7	19.3 ± 14.5
Materials	3.1 ± 1.0	7.7 ± 2.7	16.4 ± 4.9
Platformer	3.0 ± 0.9	6.0 ± 1.9	11.3 ± 4.7
AR Demo	3.0 ± 0.9	5.6 ± 1.4	12.0 ± 3.4

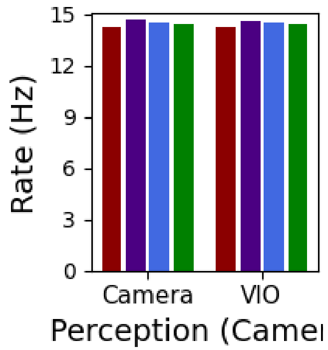
Motion-to-Photon Latency



Extremely unpleasant experience on Jetson

Image Quality

Platform	SSIM	1-FLIP	ATE/degree	ATE/meters
Desktop	0.83 ± 0.04	0.86 ± 0.05	8.6 ± 6.2	0.33 ± 0.15
Jetson-hp	0.80 ± 0.05	0.85 ± 0.05	18 ± 13	0.70 ± 0.33
Jetson-lp	0.68 ± 0.09	0.65 ± 0.17	138 ± 26	13 ± 10

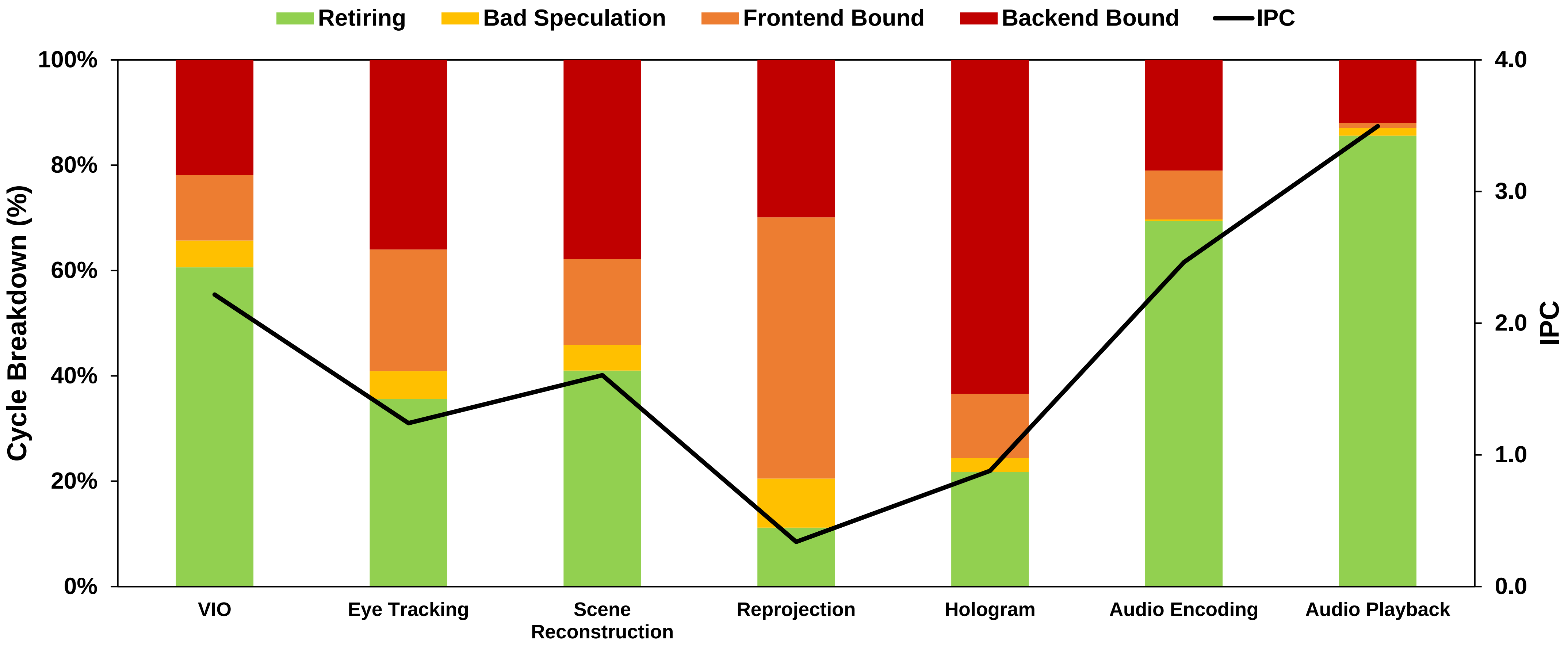


Must consider end-to-end QoE
Need better QoE metrics

Implications for Architects

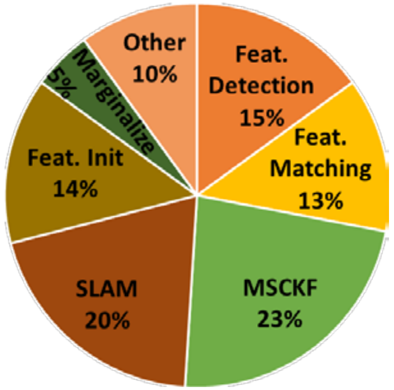
- Substantial performance, power, QoE gap
  - ⇒ Need to specialize hardware, software, *system*
- No application component dominates all metrics
  - ⇒ Must consider all application components in *system* together
- Power consumption goes beyond CPU, GPU, DDR
  - ⇒ Must consider *system*-level hardware components; e.g., display and I/O
- Significant variability
  - ⇒ Need to partition, allocate, and schedule *system* resources
- Per-component metrics do not capture QoE
  - ⇒ Must look at entire *system* to make QoE-driven tradeoffs

Component Microarchitectural Diversity

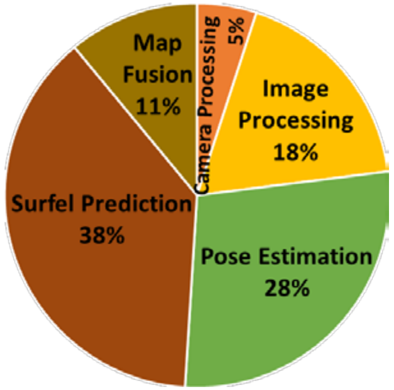


Wide range in IPC and hardware utilization

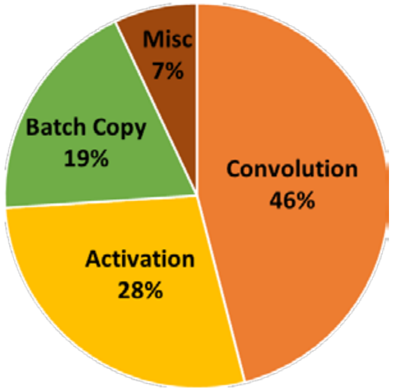
Task Diversity



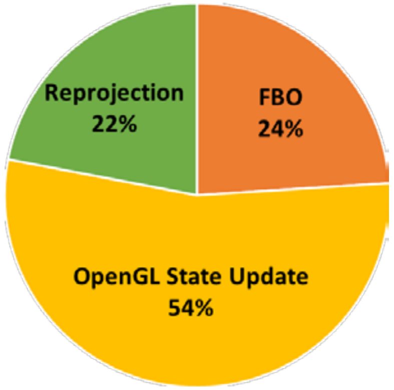
VIO



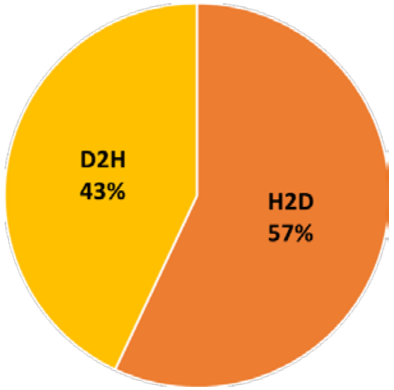
Scene Reconstruction



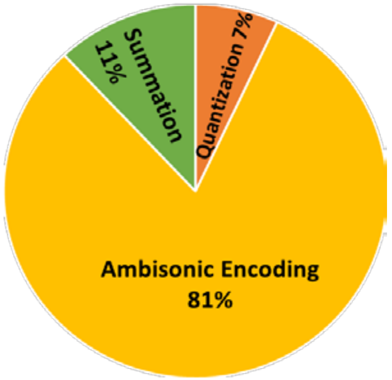
Eye Tracking



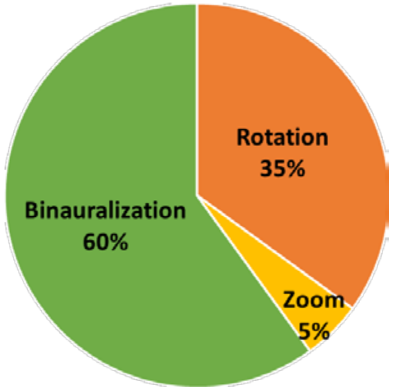
Reprojection



Hologram



Audio Encoding



Audio Playback

Variety (27!) of tasks and no task dominates

Component Deep Dive

Task	Time	Computation	Memory Pattern
Feature detection Detects new features in the new camera images	15%	Integer stencil, once per pyramid level	Subtask 1: Globally dense, local Bresenham stencil Subtask 2: Globally sparse feature accesses, locally dense stencil
Feature matching Matches features across images	13%	Integer stencil; GEMM; RANSAC; linear algebra	Subtask 1: Globally sparse, locally dense pixel accesses Subtask 2: dense feature map accesses Subtask 3: random feature map accesses
Filter Estimates 6DOF pose using camera and IMU measurements	62%	Gauss-Newton refinement; QR decomposition; GEMM; linear algebra	Mixed dense and sparse feature map and filter matrix accesses
Other Miscellaneous tasks	10%	Gaussian filter; histogram	Globally dense stencil

Component Deep Dive

Task	Time	Computation	Memory Pattern
Feature detection Detects new features in the new camera images	15%	Integer stencils per each pyramid level	Locally dense stencil; globally mixed dense and sparse
Feature matching Matches features across images	13%	Integer stencils; GEMM; linear algebra	Locally dense stencil; globally mixed dense and sparse; mixed dense and random feature map accesses
Filter Estimates 6DOF pose using camera and IMU measurements	62%	Gauss-Newton refinement; QR decomposition; GEMM; linear algebra	Mixed dense and sparse feature map and filter matrix accesses
Other Miscellaneous tasks	10%	Gaussian filter; histogram	Globally dense stencil

Task	Time	Computation	Memory Pattern
FBO FBO state management	24%	Framebuffer bind and clear	Driver calls; CPU-GPU communication
OpenGL State Update Sets up OpenGL state	54%	OpenGL state updates; one drawcall per eye	Driver calls; CPU-GPU communication
Reprojection Applies reprojection transformation to image	22%	6 matrix-vector MULs/vertex	Accesses uniform, vertex, and fragment buffers; 3 texture accesses/fragment

Task	Time	Computation	Memory Pattern
Hologram-to-depth Propagates pixel phase to depth plane	57%	Transcendentals; FMADDs; TB-wide tree reduction	Dense row-major; spatial locality in pixel data; temporal locality in depth data; reduction in scratchpad
Sum Sums phase differences from hologram-to-depth	$<$ 0.1%	Tree reduction	Dense row-major; reduction in scratchpad
Depth-to-hologram Propagates depth plane phase to pixel	43%	Transcendentals; FMADDs; thread-local reduction	Dense row-major; no pixel reads; pixels written once

Task	Time	Computation	Memory Pattern
Camera Processing Processes incoming camera depth image	5%	Bilateral filter; invalid depth rejection	Dense sequential accesses to depth image
Image Processing Pre-processes RGB-D image for tracking and mapping	18%	Generation of vertex map, normal map, and image intensity; image undistortion; pose transformation of old map	Globally dense; local stencil; layout change from RGB_RGB → RR_GG_BB
Pose Estimation Estimates 6DOF pose	28%	ICP; photometric error; geometric error	Photometric error is globally dense; others are globally sparse, locally dense
Surfel Prediction Calculates active surfels in current frame	38%	Vertex and fragment shaders	Globally sparse; locally dense
Map Fusion Updates map with new surfel information	11%	Vertex and fragment shaders	Globally sparse; locally dense

Task	Time	Computation	Memory Pattern
Normalization INT16 to FP32	7%	Element-wise FP32 division	Dense row-major
Encoding Sample to soundfield mapping	81%	$Y[j][i] = D \times X[j]$	Dense column-major
Summation HOA soundfield summation	11%	$Y[i][j] += X_k[i][j]\ \forall k$	Dense row-major

Task	Time	Sub-task	Computation	Memory Pattern
Rotation Soundfield rotation using pose	35%	**Psychoacoustic filter** Applies frequency-domain filter	FFT; frequency domain convolution; IFFT	Butterfly pattern for FFT/IFFT; dense row-major sequential accesses for convolution
		HOA rotation Rotates virtual channels	Transcendentals; FMADDs	Sparse column-major accesses; some temporal locality
Zoom Soundfield zoom using pose	5%	—	Linear algebra	Dense column-major sequential accesses
Binauralization HRTF application	60%	—	Identical to **psychoacoustic filter**	Identical to **psychoacoustic filter**

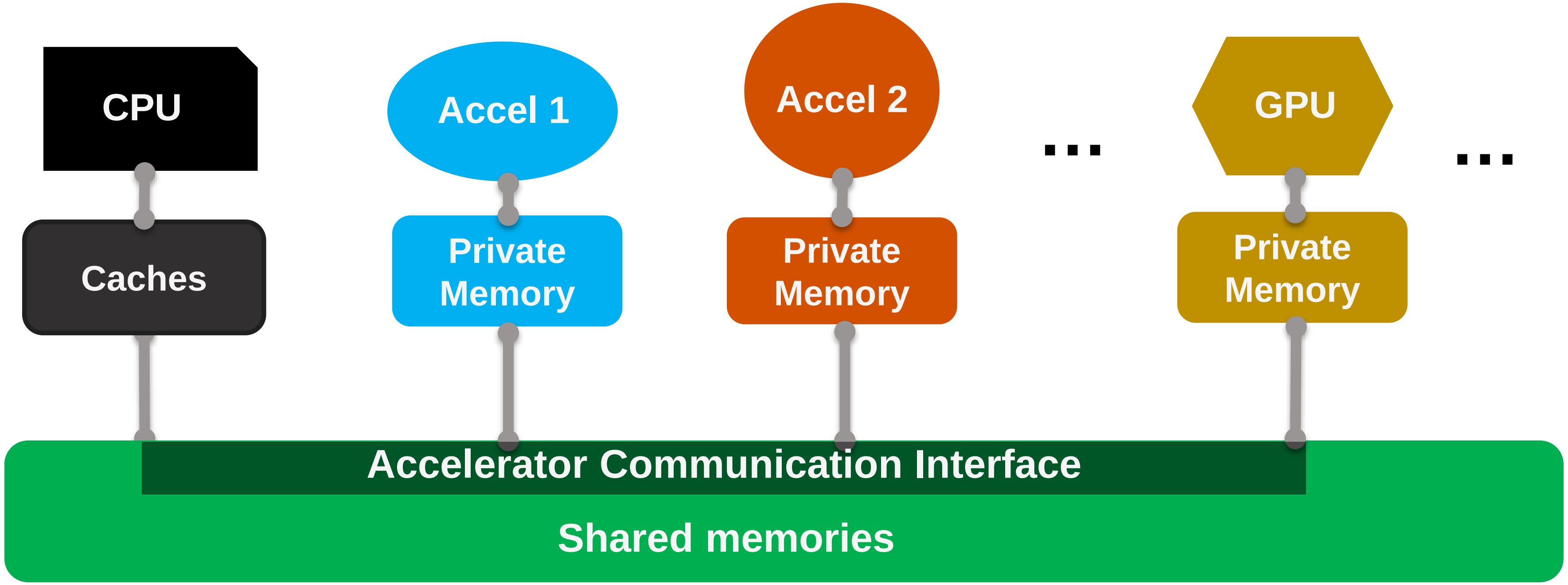
Implications for Architects

- Need to specialize hardware, software, *system*
- Must consider all application components in *system* together
- Must consider *system*-level hardware components; e.g., display and I/O
- Need to partition, allocate, and schedule *system* resources
- Must look at entire *system* to make QoE-driven tradeoffs

- Abundance of tasks and no single task dominates
  - ⇒ Need *automated* techniques to determine what to accelerate
- Impractical to build accelerator for every task
  - ⇒ Must build *shared* hardware
- Diversity of compute and memory primitives
  - ⇒ *Flexible* on-chip memory hierarchy
  - ⇒ *Flexible* accelerator communication interface
- Algorithms in flux
  - ⇒ Must design *programmable* hardware
- Different algorithms have different QoE vs. resource usage profiles
  - ⇒ End-to-end QoE driven *approximate computing*

ILLIXR =
Rich playground for
Systems 2030 research

Accelerator Communication Interface



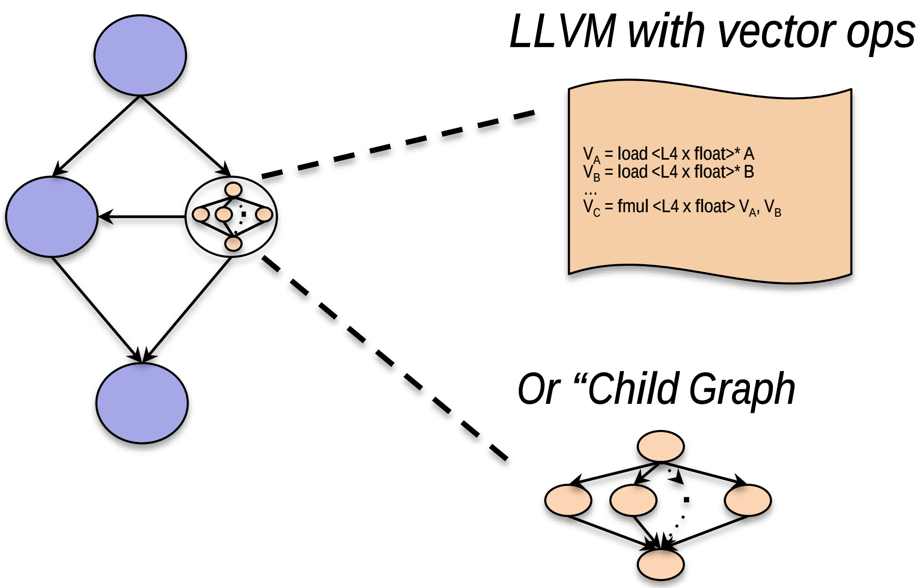
- How should heterogeneous parallel accelerators communicate with each other?
- Programmable, shared hardware ⇒ shared memory
  - Coherence, consistency, communication
  - Build on Spandex heterogeneous coherence interface for coherence specialization [ISCA18, in review]

Representing Heterogeneous Parallelism in Software

w/ V. Adve and S. Misailovic

HPVM: Heterogeneous Parallel Virtual Machine [PPoPP18, OOPSLA19, PPOPP21]

Compiler IR and Hardware Virtual ISA



Model: Hierarchical dataflow graph with side effects

Captures

- coarse grain task parallelism
- streams, pipelined parallelism
- nested parallelism
- SPMD-style data parallelism
- fine grain vector parallelism

Supports high-level optimizations as graph transformations

Targets: CPUs, vector extensions, GPUs, FPGAs, domain specific accelerators

Representing ILLIXR in HPVM

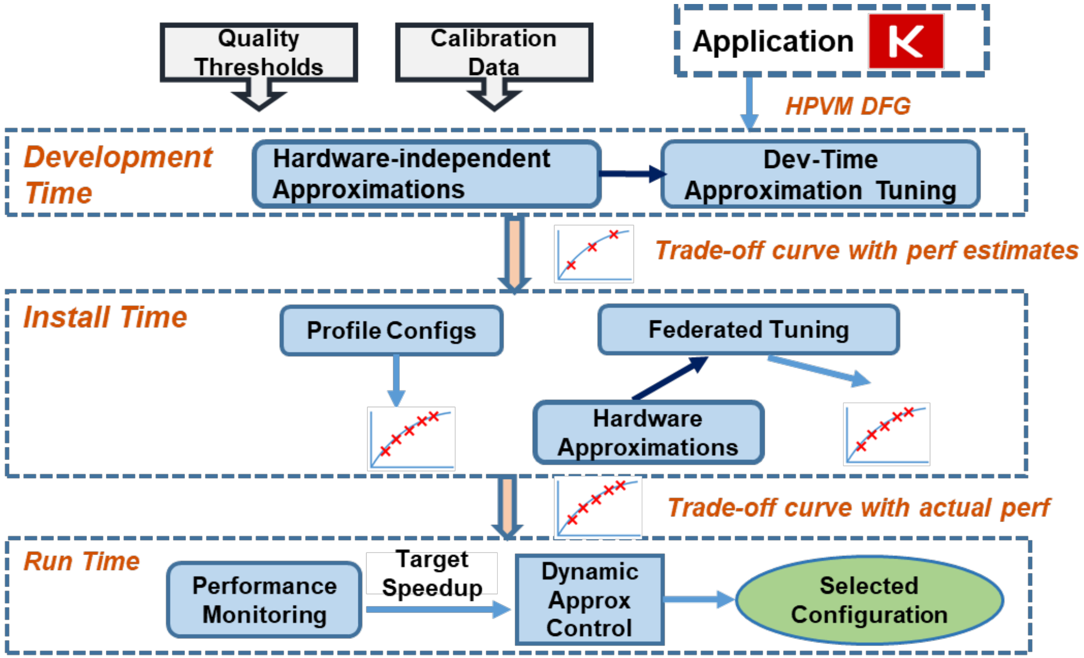
For code generation, automated accelerator selection, approximation, resource mapping, …

Automated Approximation Selection

w/ V. Adve and S. Misailovic

ApproxTuner [PPoPP21]

Combines multiple software and hardware approximations for tensor operations



Uses predictive models to compose accuracy impact of multiple approximations

3-phase approximation tuning

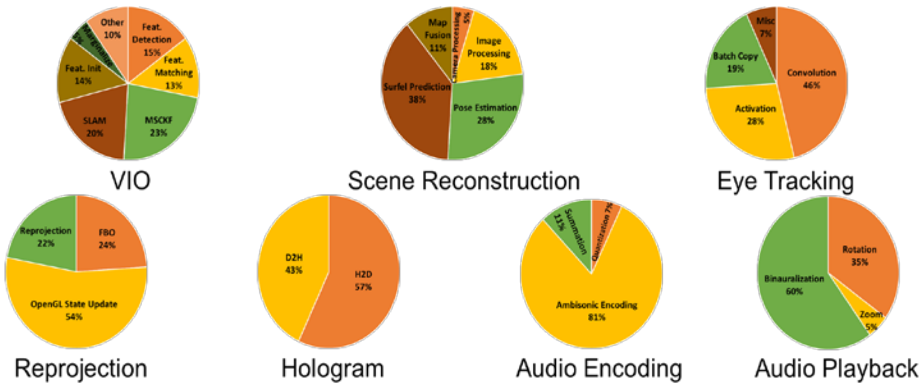
- Development-time preserves hardware portability via ApproxHPVM IR
- Install-time allows hardware-specific approximations
- Run-time allows dynamic approximation tuning

Approximations for ILLIXR

Build on ApproxTuner for QoE-driven automated selection

Automated Selection, Generation of Accelerator HW & SW

w/ V. Adve, D. Brooks, V. Reddi, G.-Y. Wei



Manual identification of common compute, memory patterns

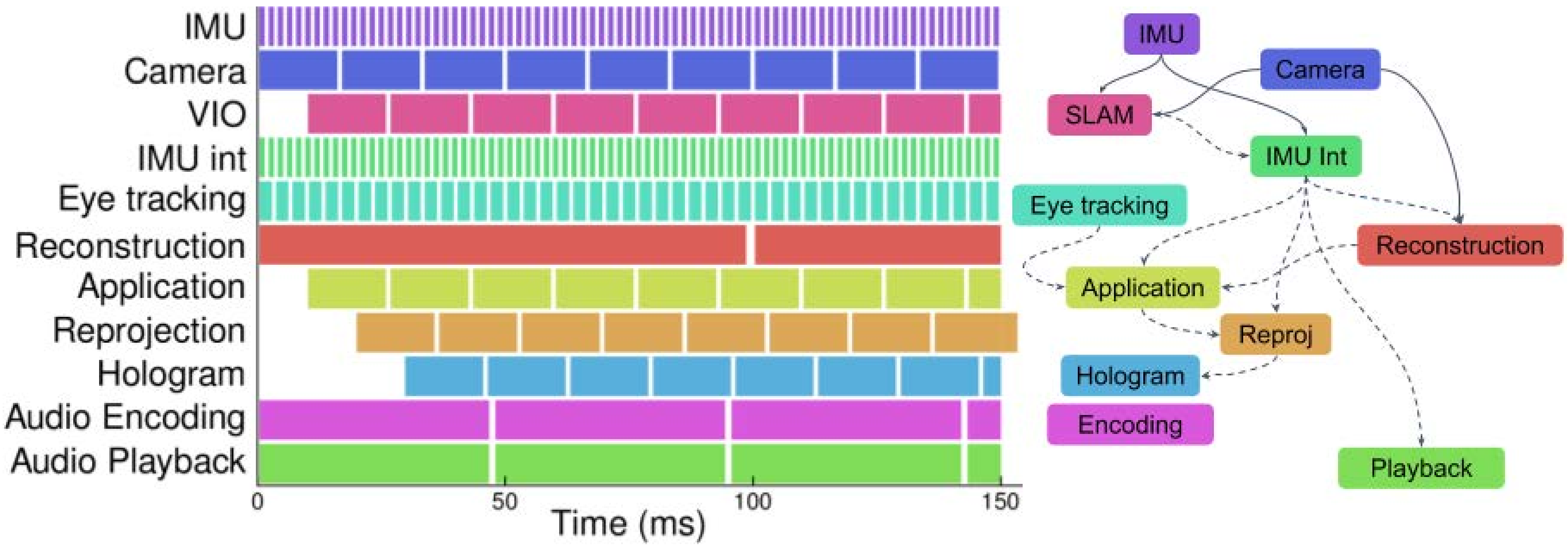
$\Rightarrow$ Cross-component co-design allows hardware, computation, and data reuse w/ large benefits

Goal: Automated design space exploration to identify profitable acceleration, generate HW+SW

- Use HPVM's parallelism and communication representation
- Compiler analysis and transformations for common patterns and optimizations

QoE-Driven Scheduling

w/ P. B. Godfrey, R. Mittal



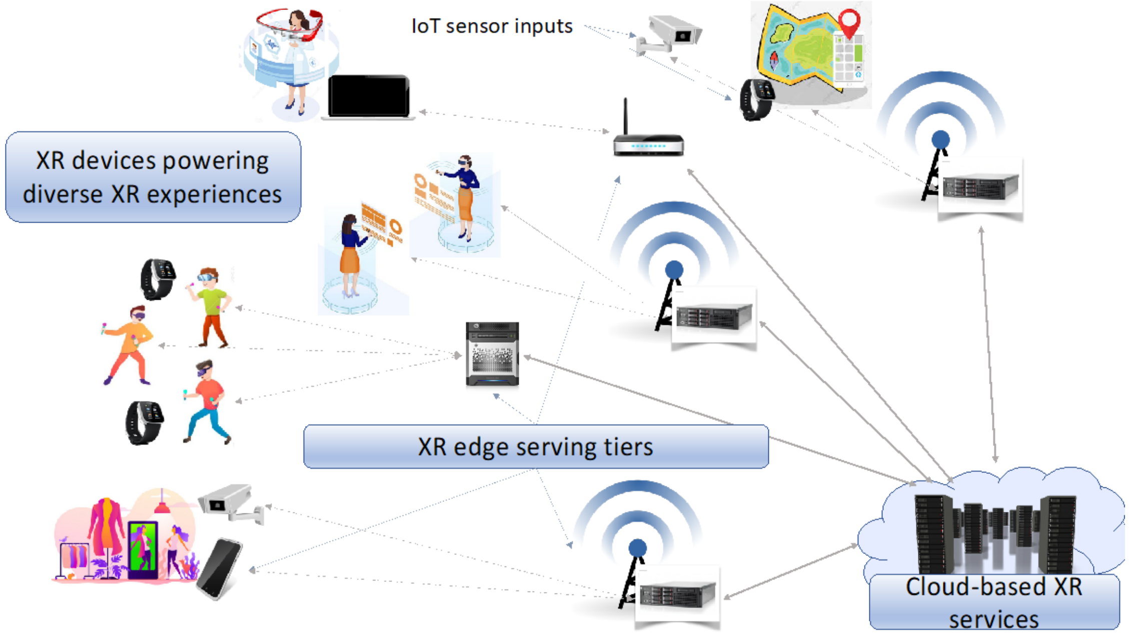
ILLIXR task graph is a DAG with multiple critical paths and QoE constraints

Scheduler goal: Determine frame rates and schedule to meet QoE for given hardware mapping

Future: Multiple hardware targets for given task, hardware and software approximations

From Single-Device to Distributed Systems

w/ A. Gavrilovska, K. Nahrstedt



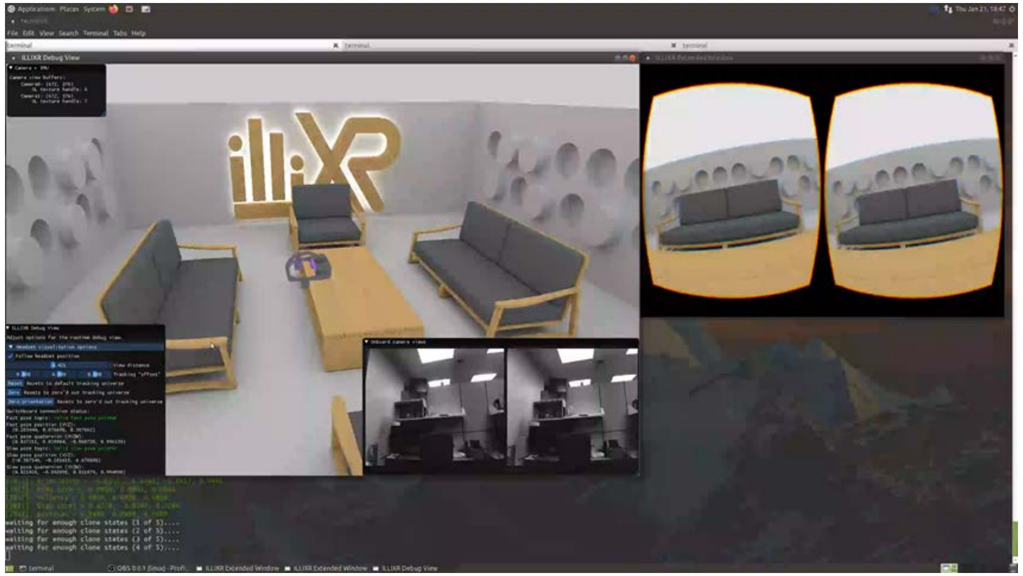
- Offload computation to edge, cloud servers
- Content streaming
- Multiparty AR/VR experiences

More Use Cases

- Security and Privacy
- 360 Video streaming
- Multiparty AR programming stack
- Displays
- On-sensor computing
- QoE metrics
- XR algorithms
- …

ILLIXR Testbed

- New components: translational reprojection (spacewarp), hand tracking, ...
- Add North Star head set
- Broaden hardware/software platforms supported
- Create and curate data sets and applications
- Incorporate research results
- …



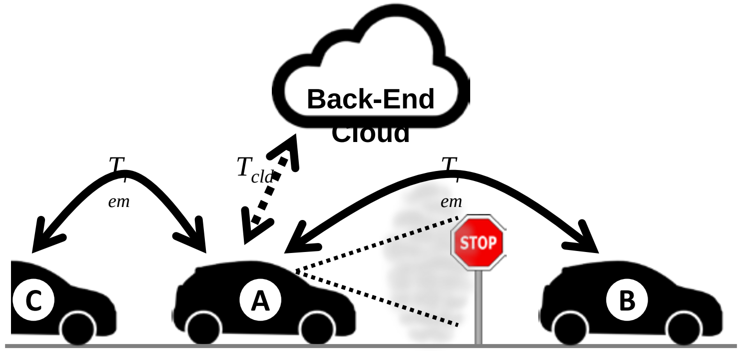
illixr.github.io

Soon: Community Consortium

- Industry + academic partners
  - ARM, Facebook, Micron, NVIDIA, …
- Standardize benchmarking, QoE metrics, …

VR@Illinois

Chowdhari et al., EarthSense robots

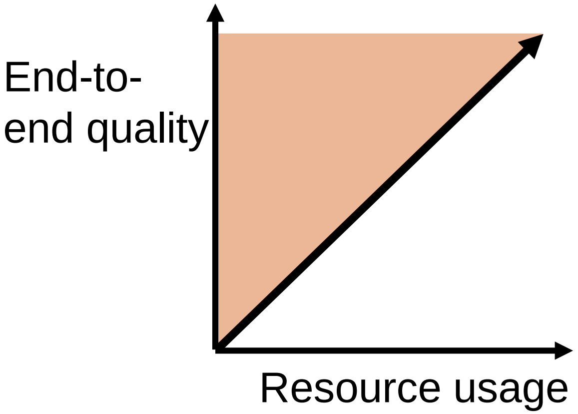


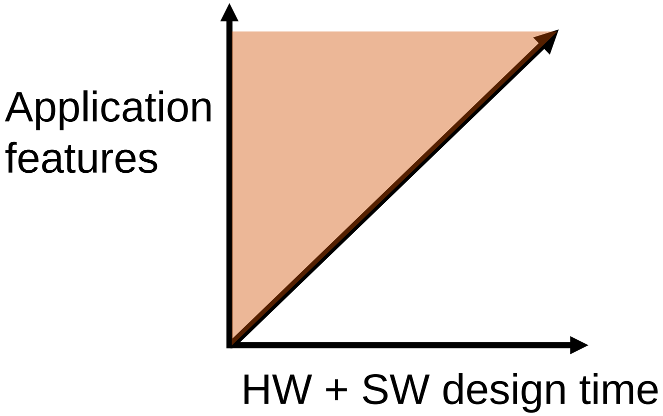
EPOCHS project, IBM, Columbia, Harvard, Illinois

Challenging perf demands. Stringent resource constraints. End-to-end quality metrics.

ILLIXR is a rich playground for research for Systems 2030

Scalable & Generalizable Specialization





Application-driven,

end-to-end quality driven,

HW-SW-App co-designed

system specialization techniques

illixr.github.io

Team ILLIXR

- Rishi Desai
- Samuel Grayson
- Muhammad Huzaifa
- Xutao Jiang
- Ying Jing
- Jae Lee
- Fang Lu
- Yihan Pang
- Joseph Ravichandran
- Giordano Salvador
- Finn Sinclair
- Boyuan Tian
- Lauren Wagner
- Henghzi Yuan
- Jeffrey Zhang

- **Spandex team:** Johnathan Alsop, Robert Jin, Weon Tak Na, Matthew Sinclair, Zeran Zhu

- **HPVM team:** Vikram Adve, Adel Ejjeh, Muhammad Huzaifa, Keyur Joshi, Rakesh Komuravelli, Maria Kotsifakou, Akash Kothari, Sasa Misailovic, Yasmin Sarita, Ben Schreiber, Hashim Sharif, Matthew Sinclair, Prakalp Srivastava, Elizabeth Wang, Yifan Zhao, Nathan Zhao

Some Life Stories and Lessons

- ILLIXR story
  - Born out of frustration and desire to impact something real
  - Had no clue about XR except that my colleague Steve Lavalle had recently returned from a successful stint as founding chief scientist at Oculus ☺
  - CFAR seed proposal (1 page) + Encouraging colleague + Excited student ⇒

    Detour became main research thrust w/ many students, faculty, and industry collaborators
  - 3+ years of work, real impact still to come, but already satisfying
- Memory models story
  - Frustration with HW memory models, called on SW community to fix
  - Joined Java memory model effort – didn't know Java or PL-ese, no students, no funding
  - 5 years of work, 1 paper, but real impact
- SIGARCH chair story
  - Colleagues gathered frustrating data on diversity in architecture community
  - Joined hands with colleagues for intense activism, concrete actions
  - ~3 years of work, no papers, but real impact; e.g., CARES movement

Some Life Stories and Lessons

- Follow your passion
- Take risks. Believe in yourself.
- Impact = Change minds. Takes time and hard work. (!= # Papers)
- It takes a village. Pay it forward.