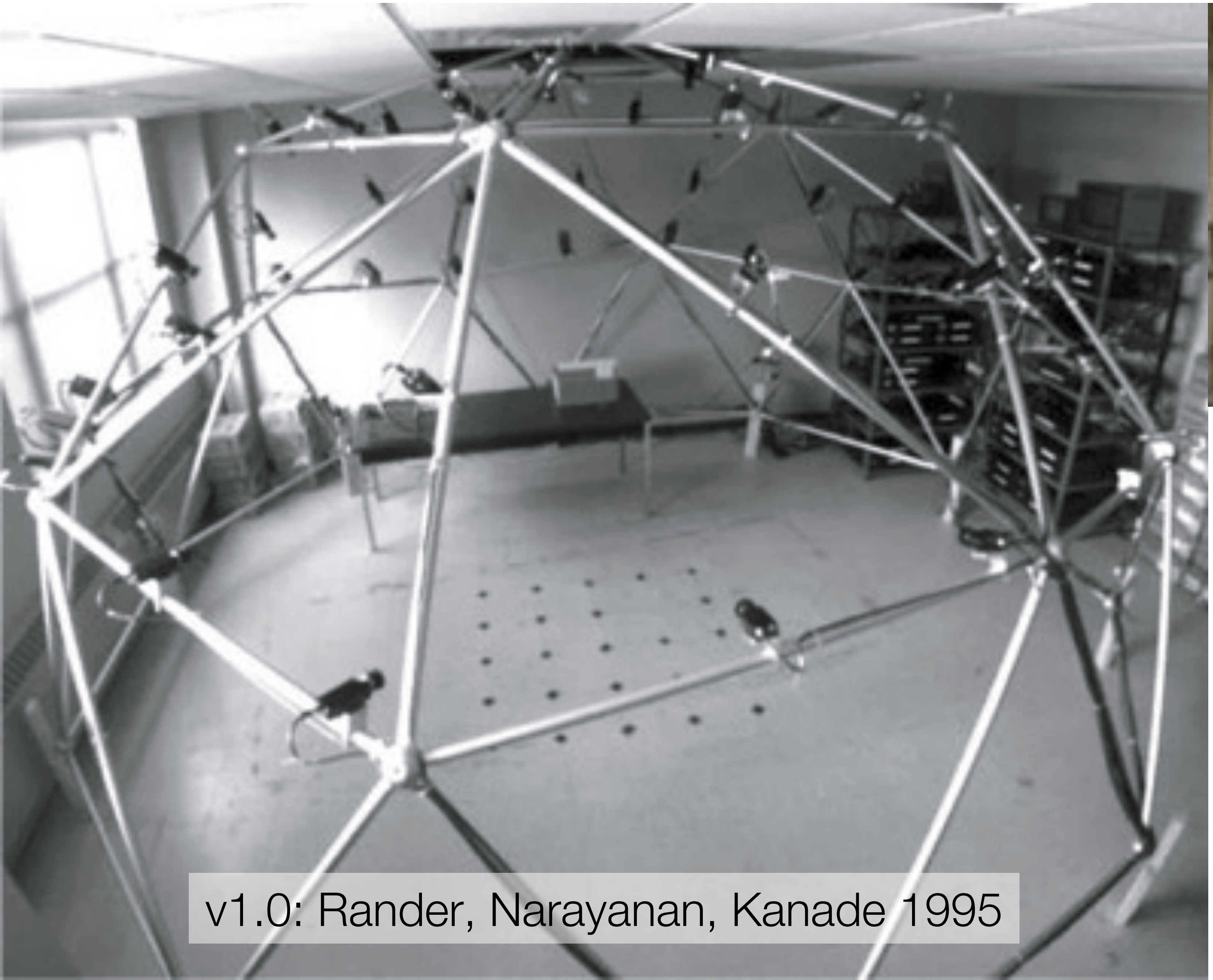
Hardware Design, Structure, and Networking

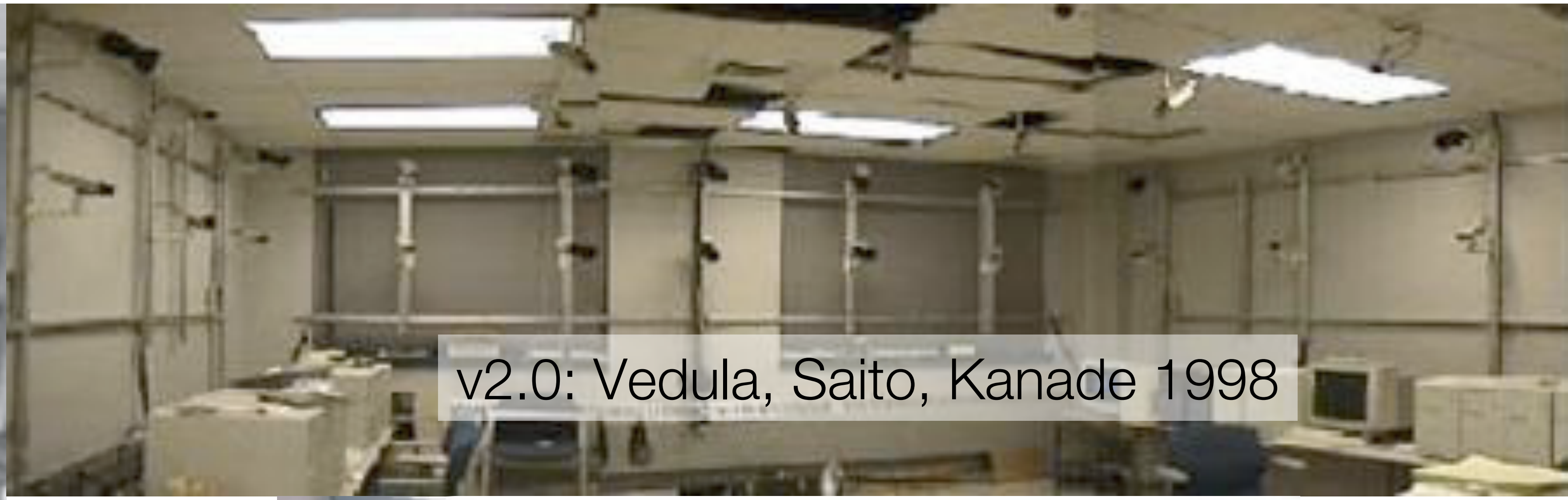
Shohei Nobuhara
Kyoto Univ.

Structure Design

Different Designs for Different Goals



v1.0: Rander, Narayanan, Kanade 1995



v2.0: Vedula, Saito, Kanade 1998

v3.0: Matthews, Baker, Kanade 2002

Structure Design

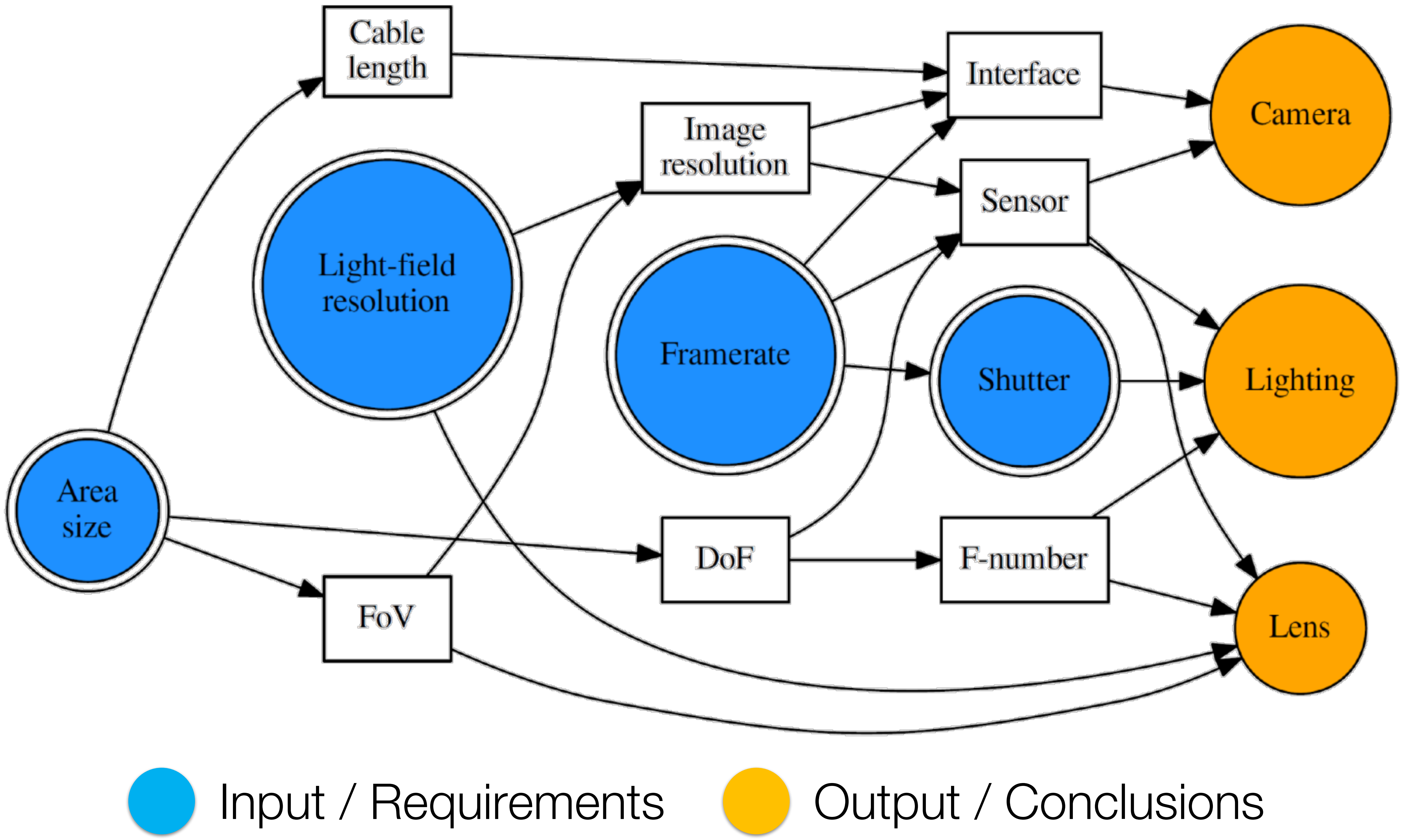
Different Designs for Different Goals

Kyoto U. 2005

Kyoto U. 2008
ICCV2009 demo

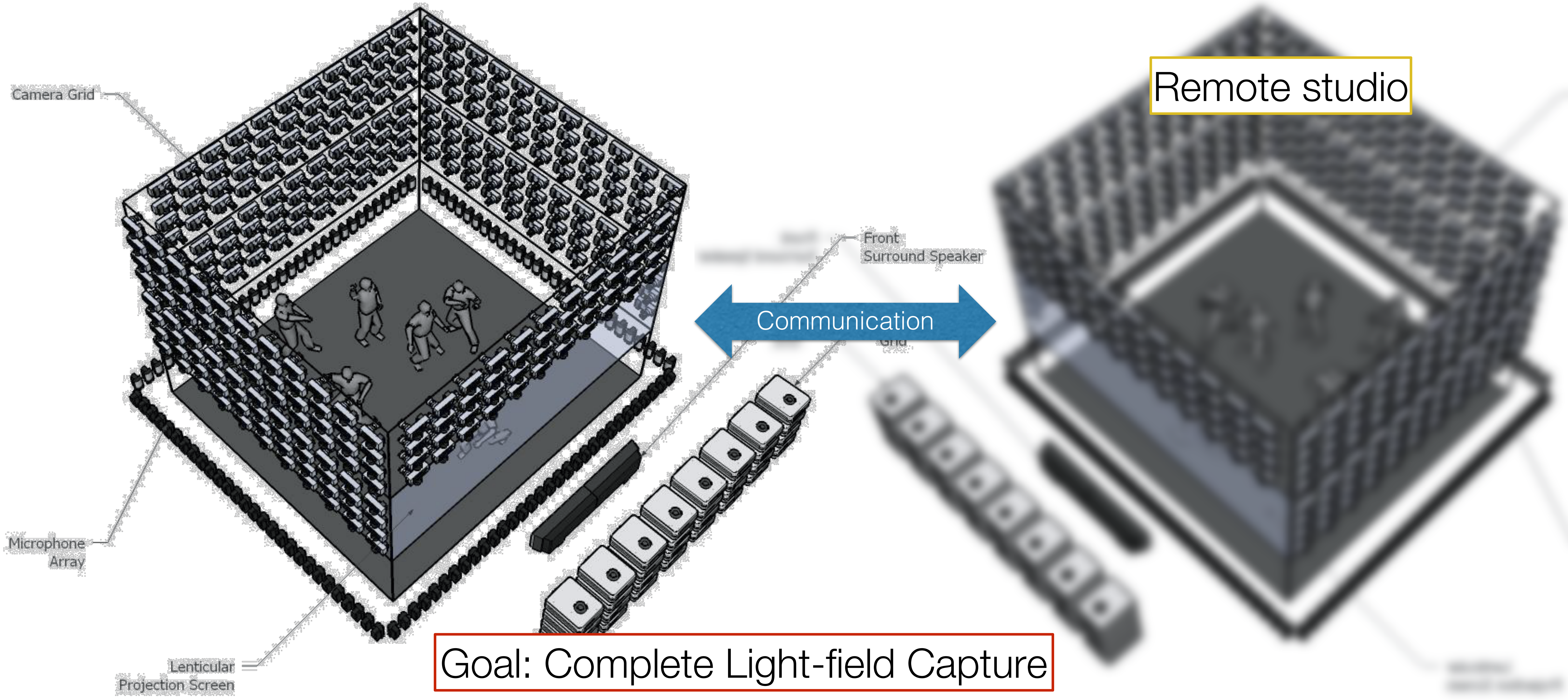
Dependencies

Different Designs for Different Goals



Structure Design

Different Designs for Different Goals

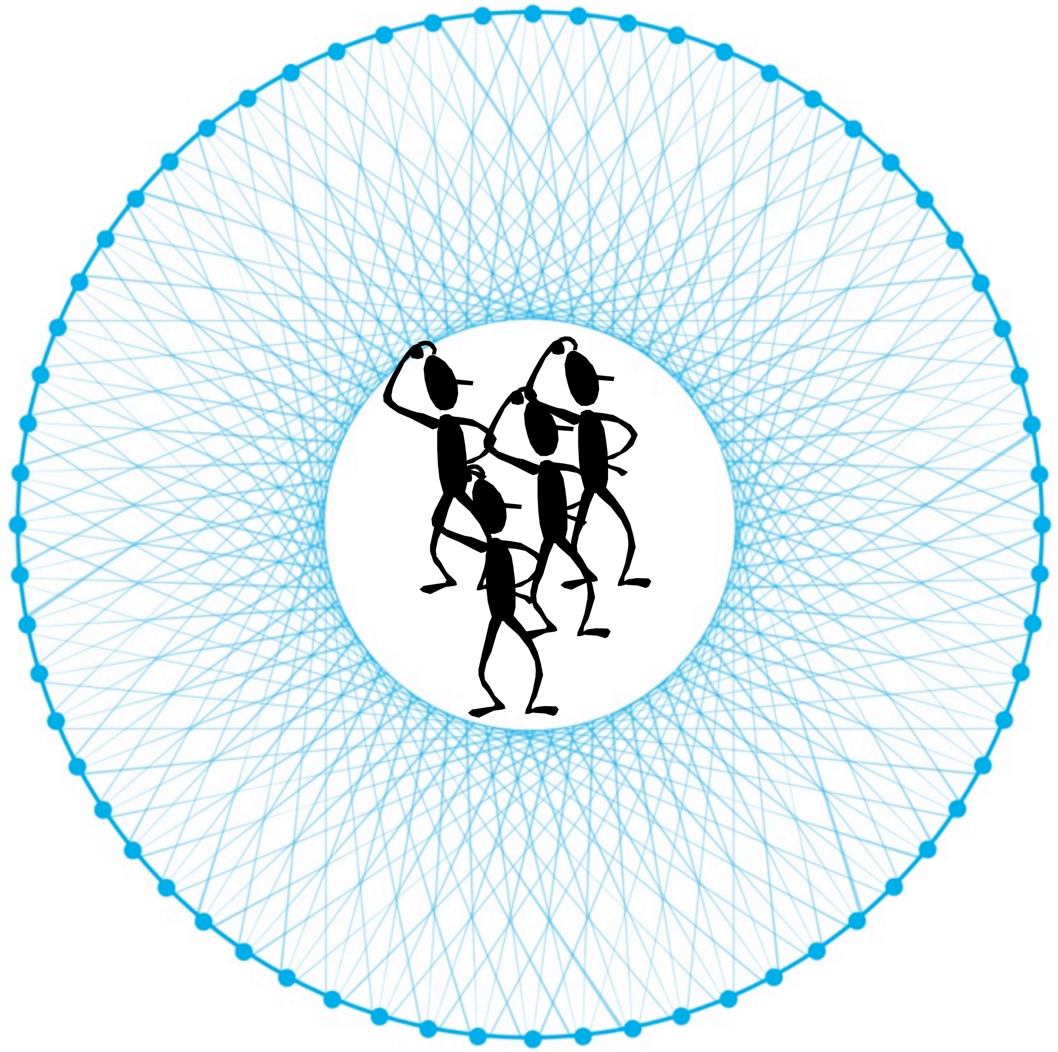


Goal: Complete Light-field Capture

Cameras for Complete LF Capture

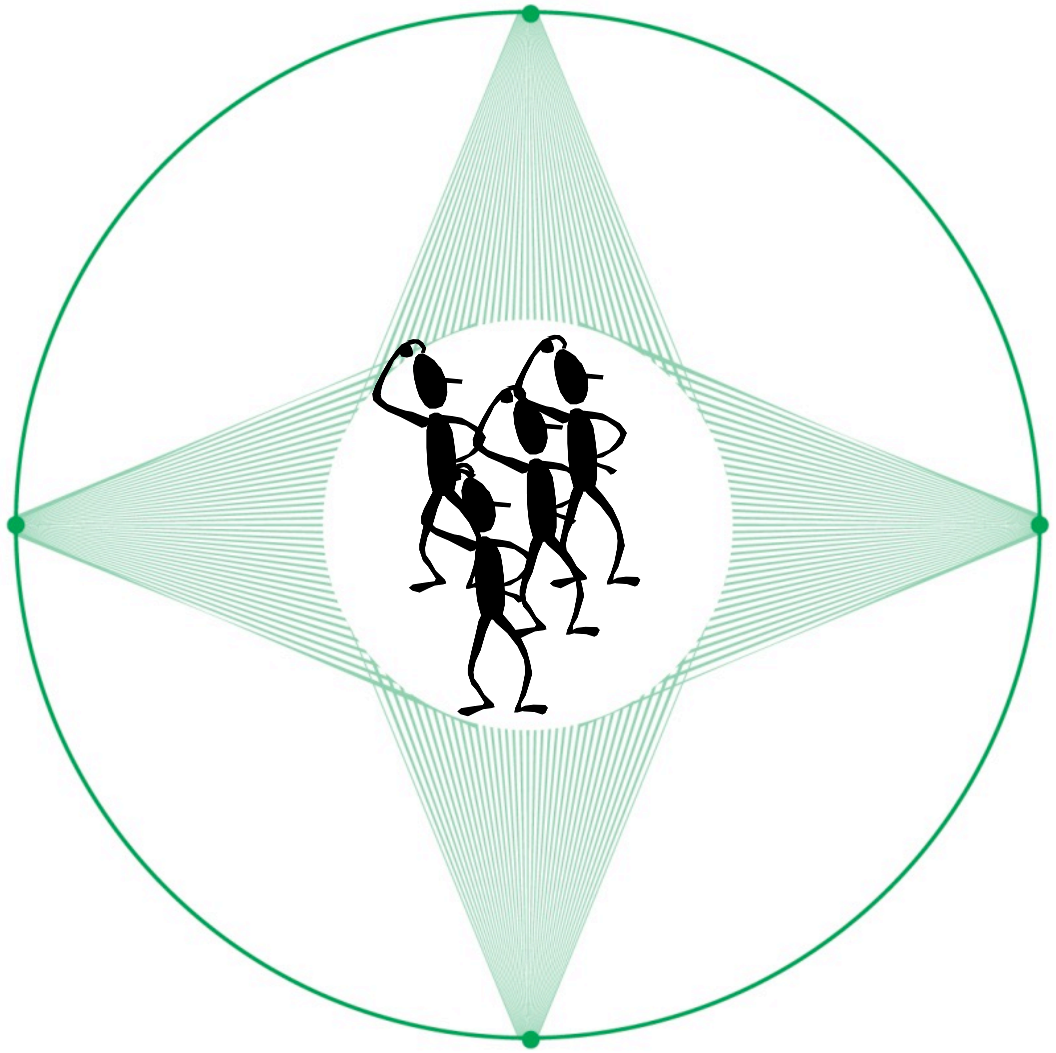
Dense low-res cameras or sparse high-res cameras?

Dense cameras & Low sampling rate



for visibility

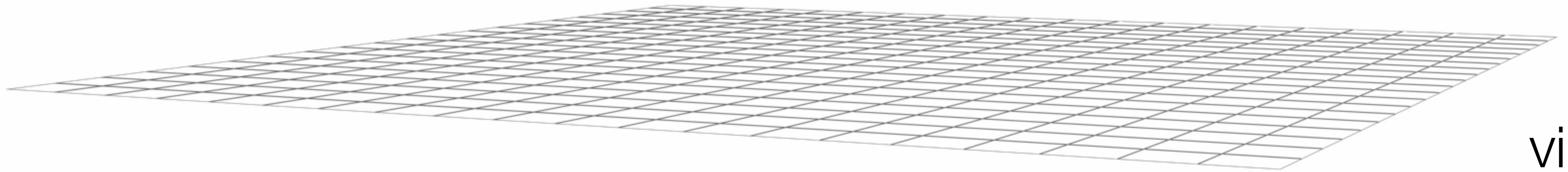
Sparse cameras & High sampling rate



for detailed geometry/texture

Cameras for Complete LF Capture

Dense low-res cameras or sparse high-res cameras?



video at 2010

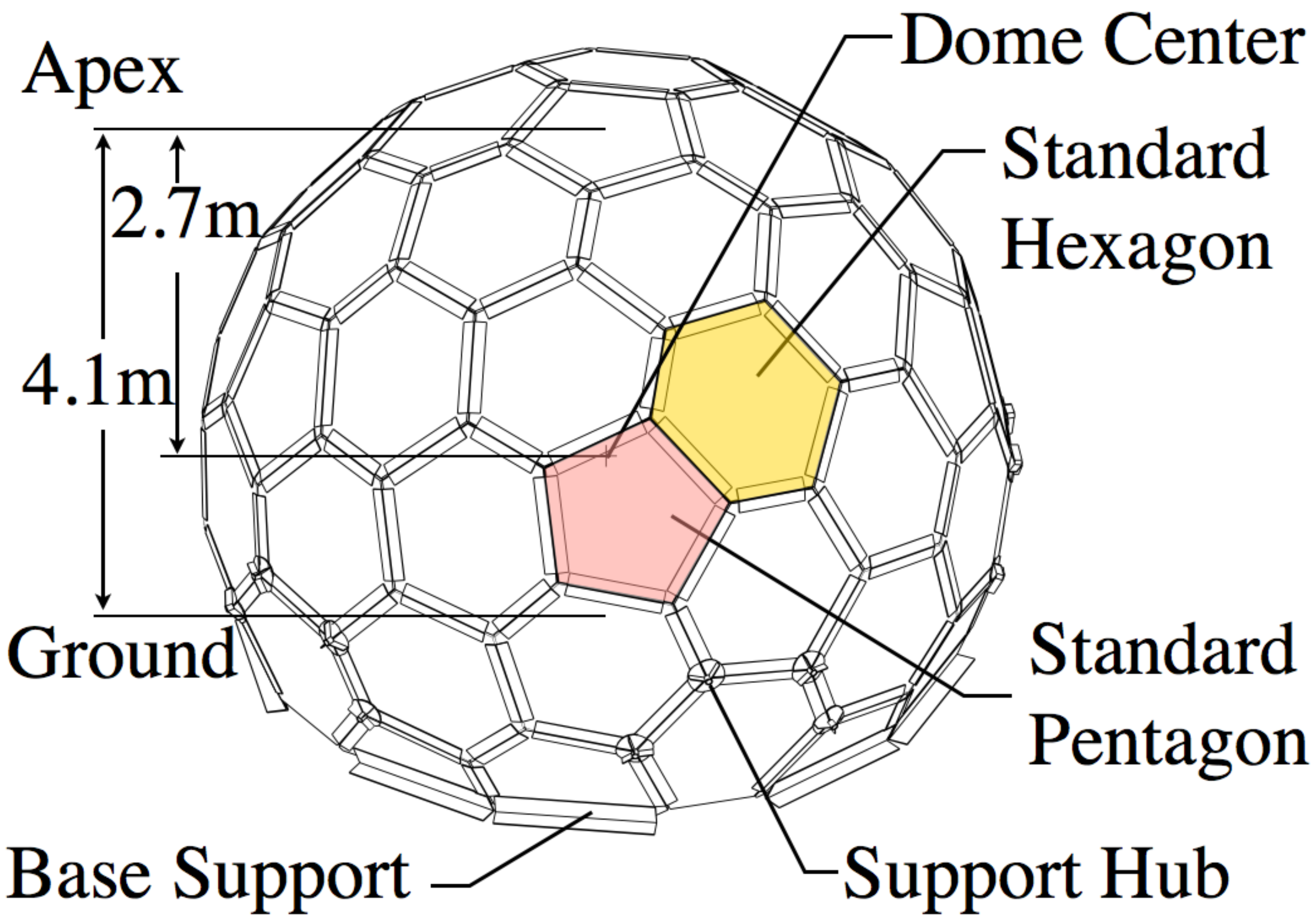
Dense low-res cameras / simple task ⇔ Sparse high-res cameras / complex task

++ Visibility

++ Resolution

Truncated Pentagonal Hexecontahedron

6 Pentagons, 40 Hexagons, 10 Trimmed Panels

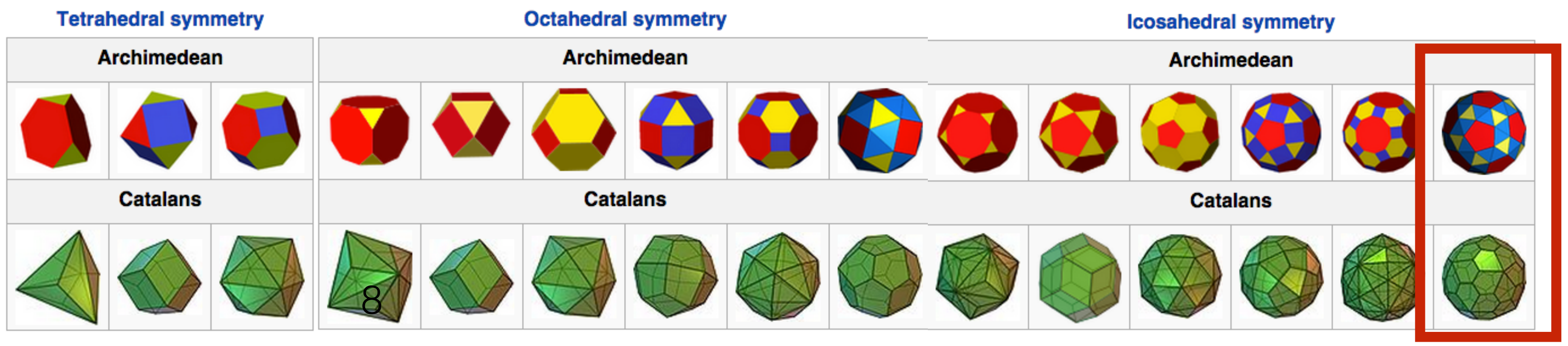


Self-supporting

Modular

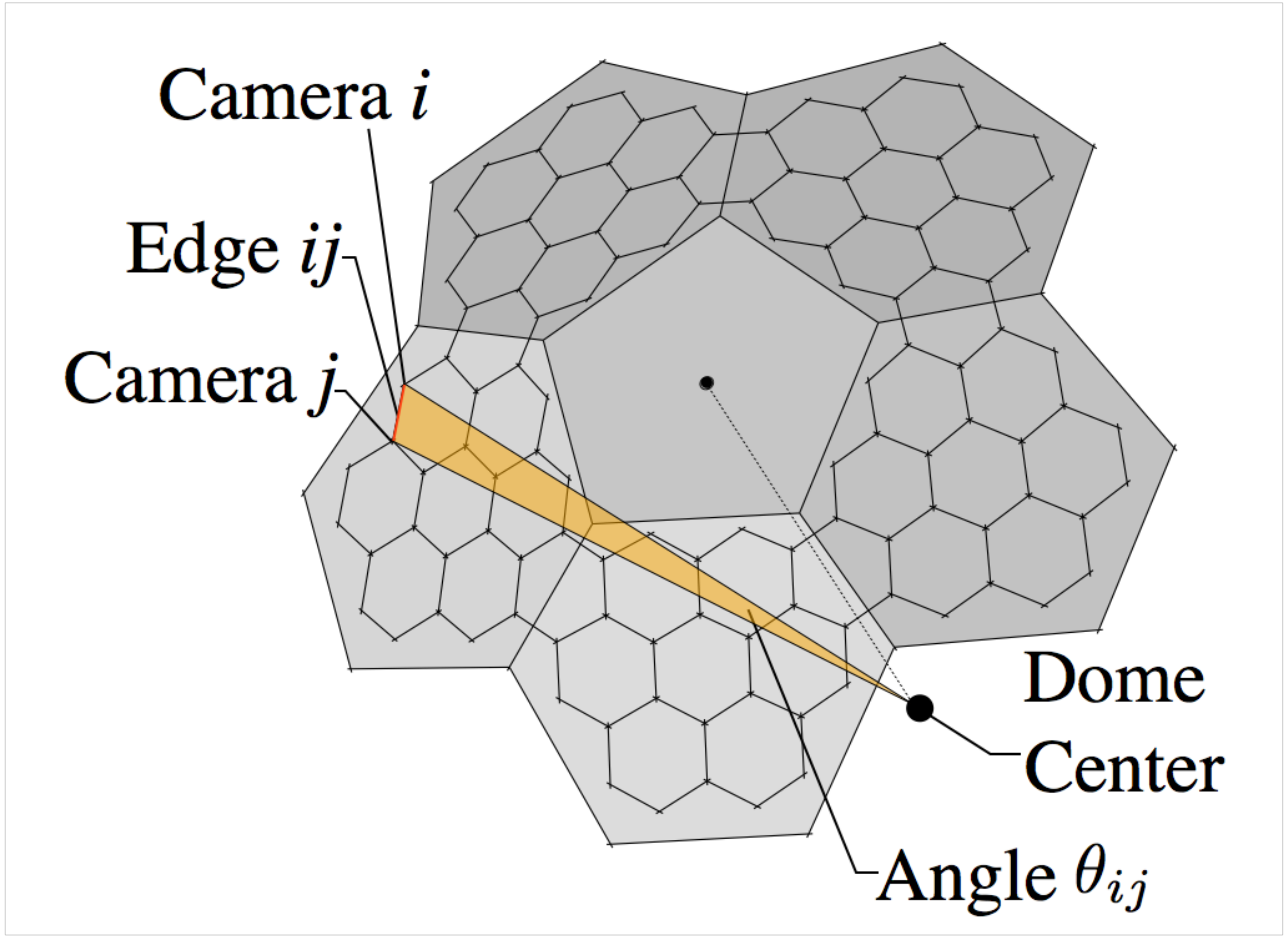
Background

Face-transitive solid
(Catalan solid)



Optimized Placement of Cameras

Minimizing Differences in Camera Angles

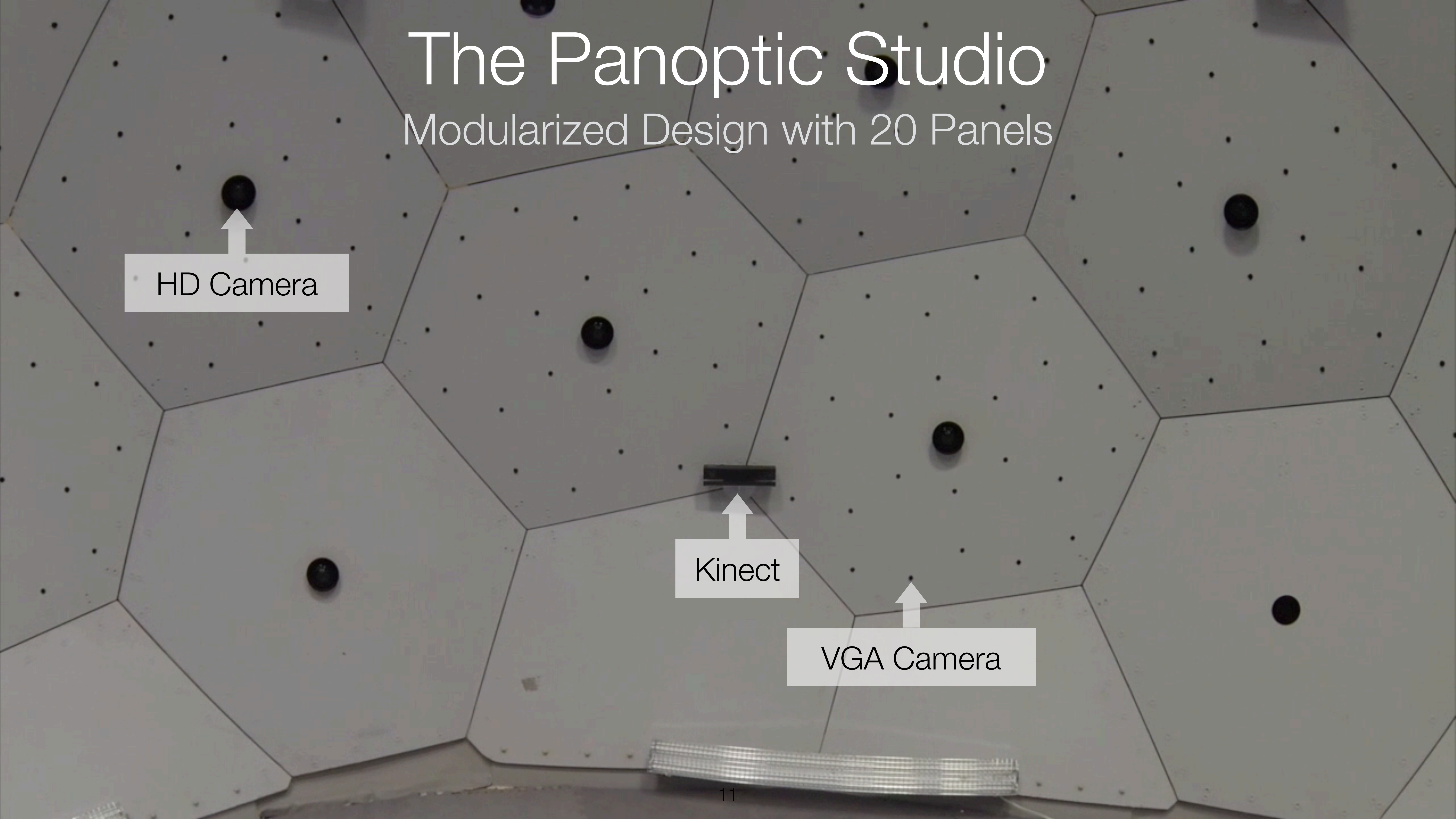


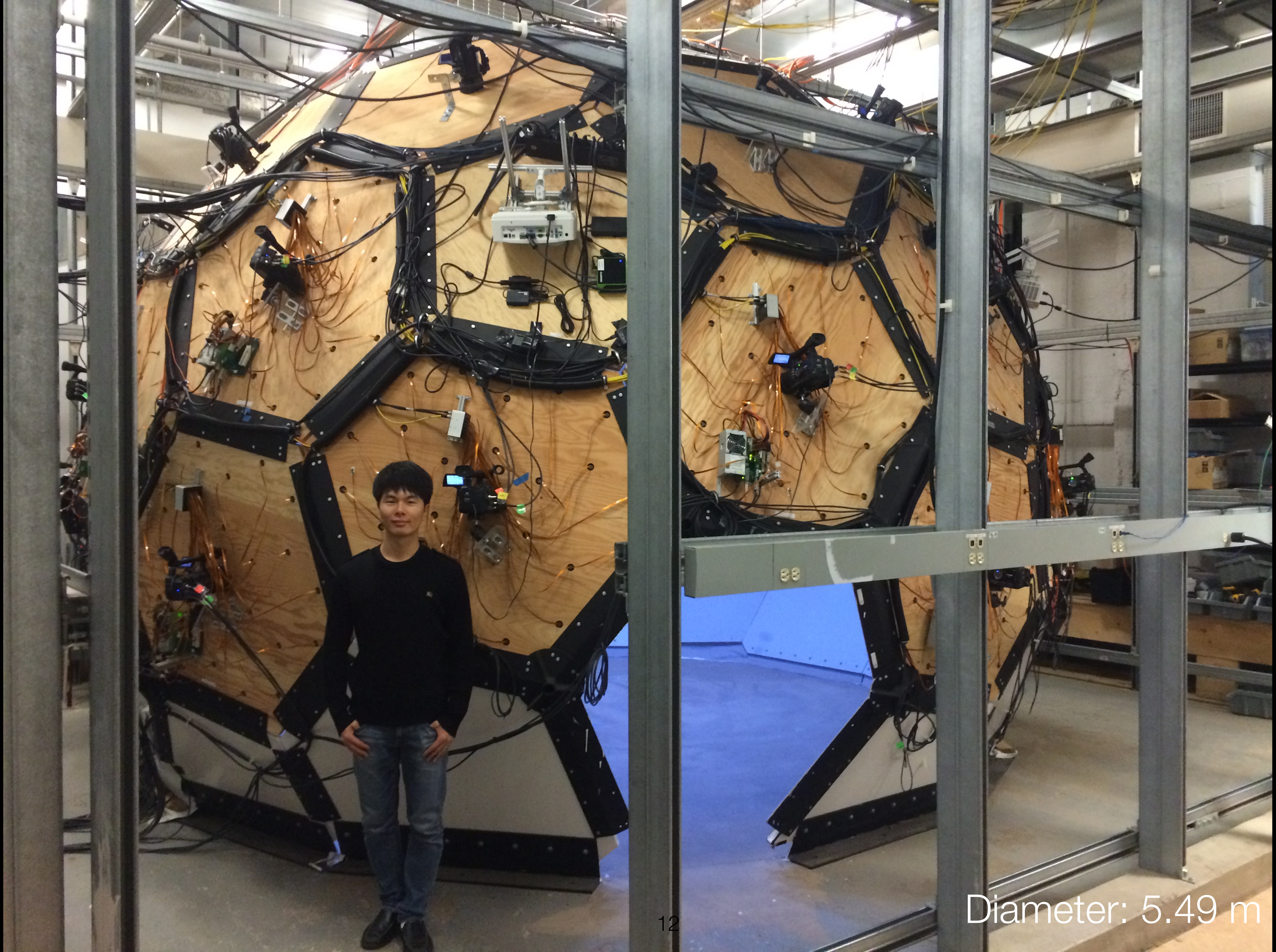
The Panoptic Studio

Modularized Design with 20 Panels

The Panoptic Studio

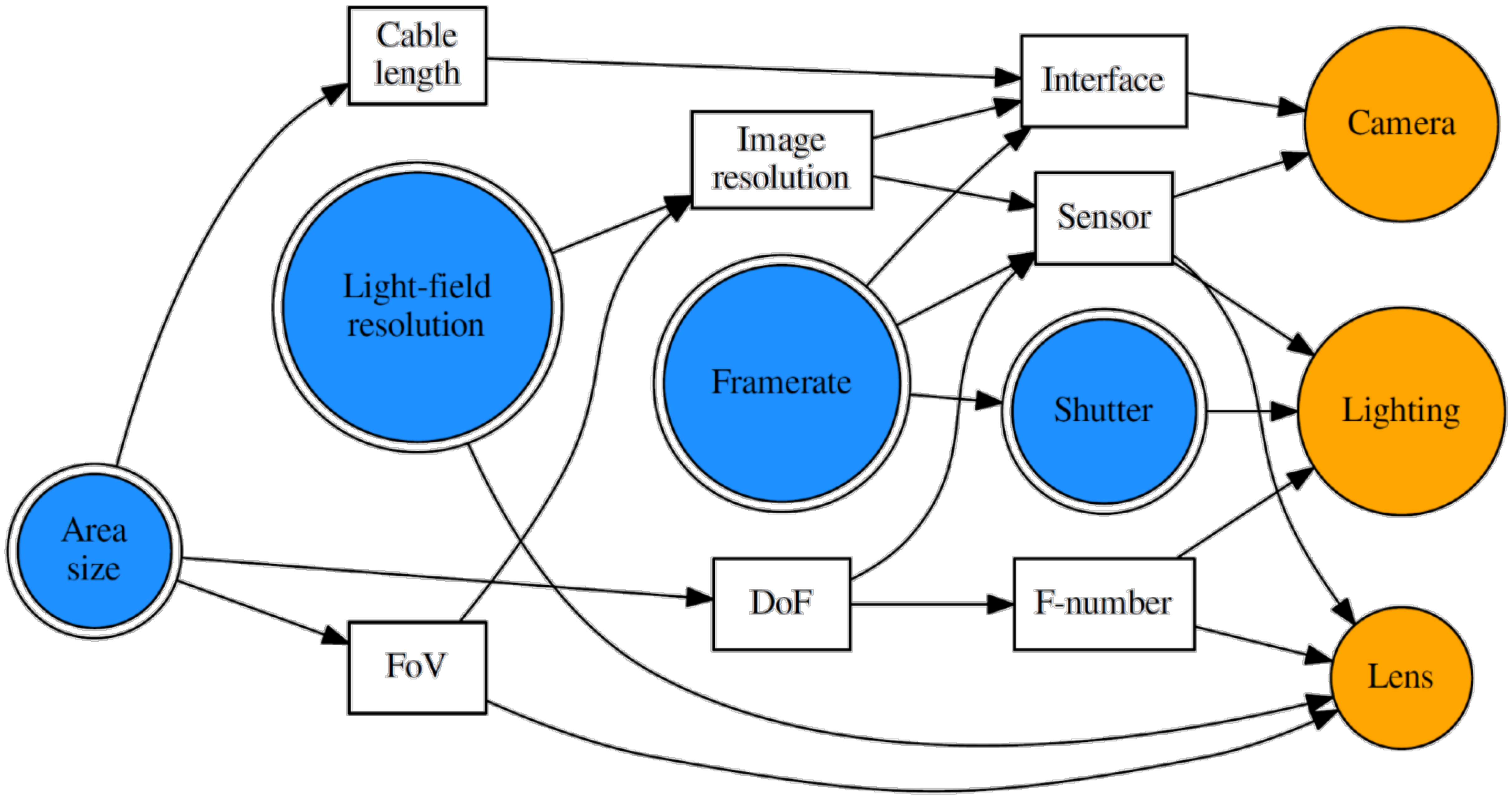
Modularized Design with 20 Panels




Diameter: 5.49 m

Dependencies

Different Designs for Different Goals



Input / Requirements   Output / Conclusions

The Panoptic Studio

Modularized Design with 20 Panels

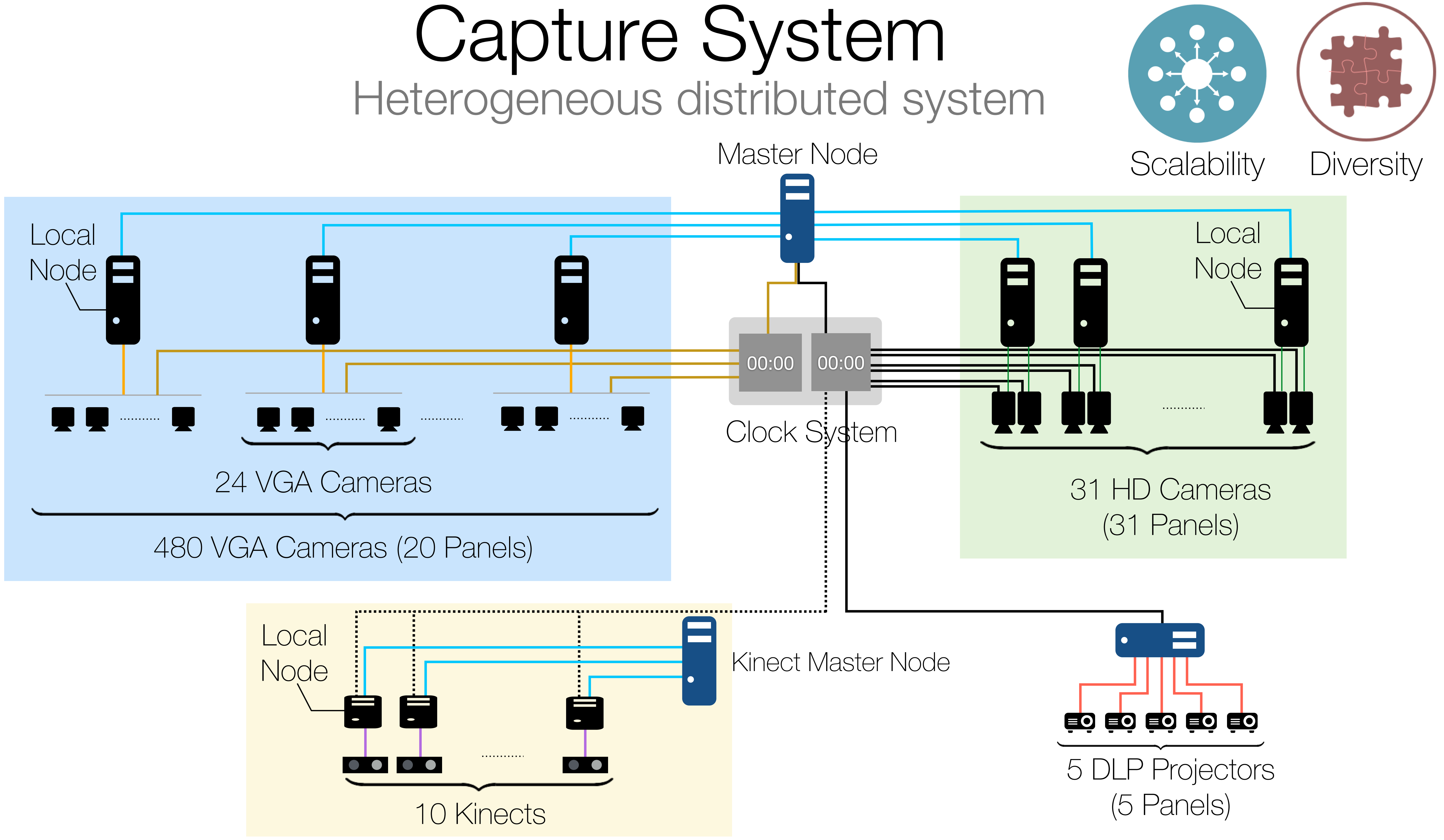
The Panoptic Studio

Modularized Design with 20 Panels

480 VGA Cameras
31 HD Cameras
10 Kinects

Capture System

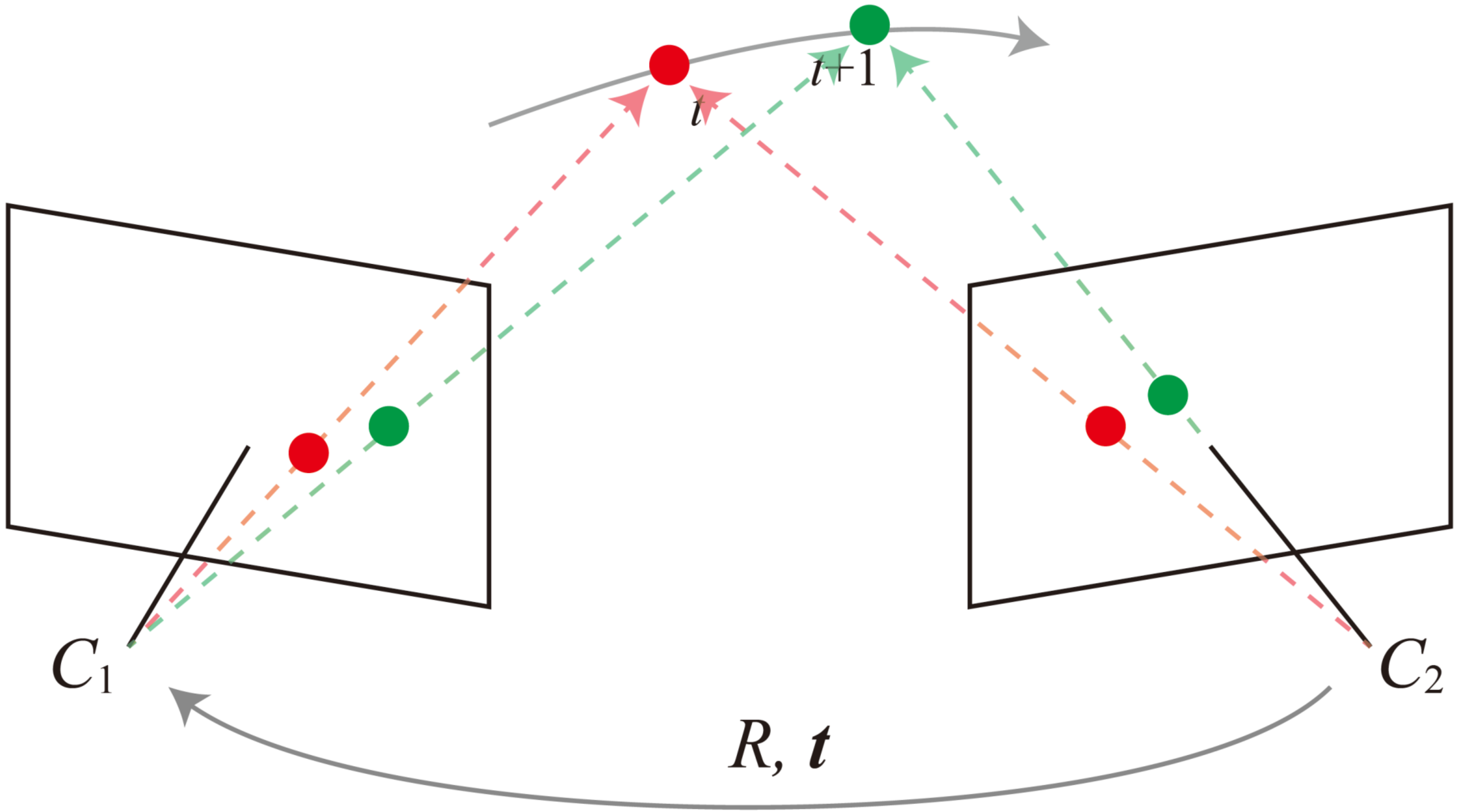
Heterogeneous distributed system



Synchronization

Synchronization

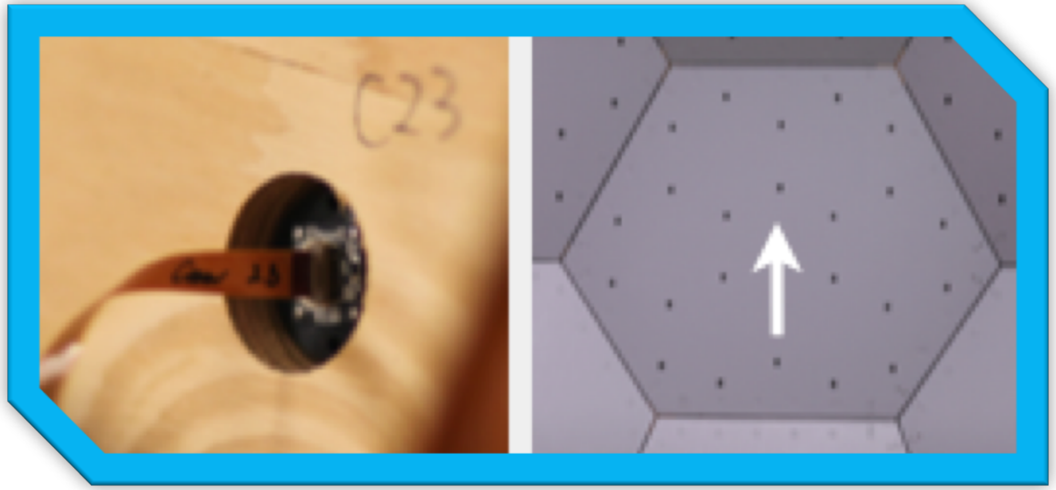
Why do we need this?

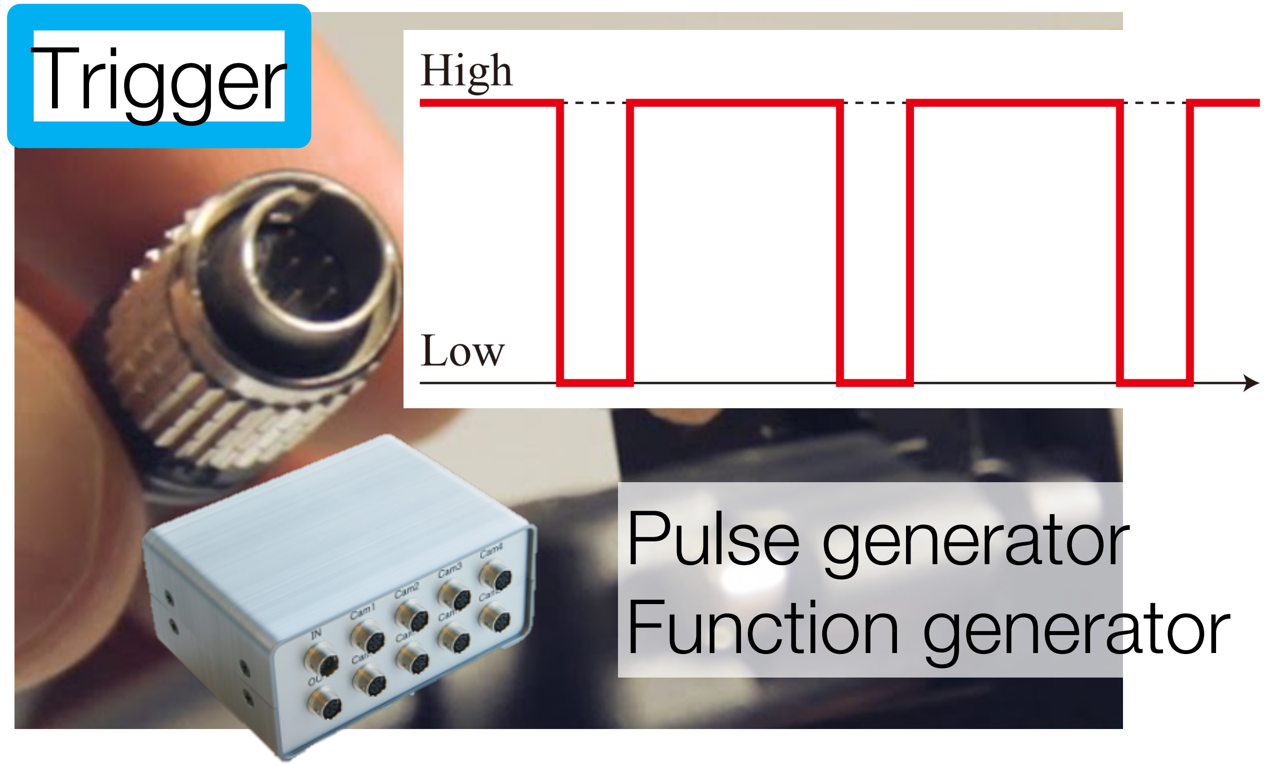


Triangulation assumes same time instant

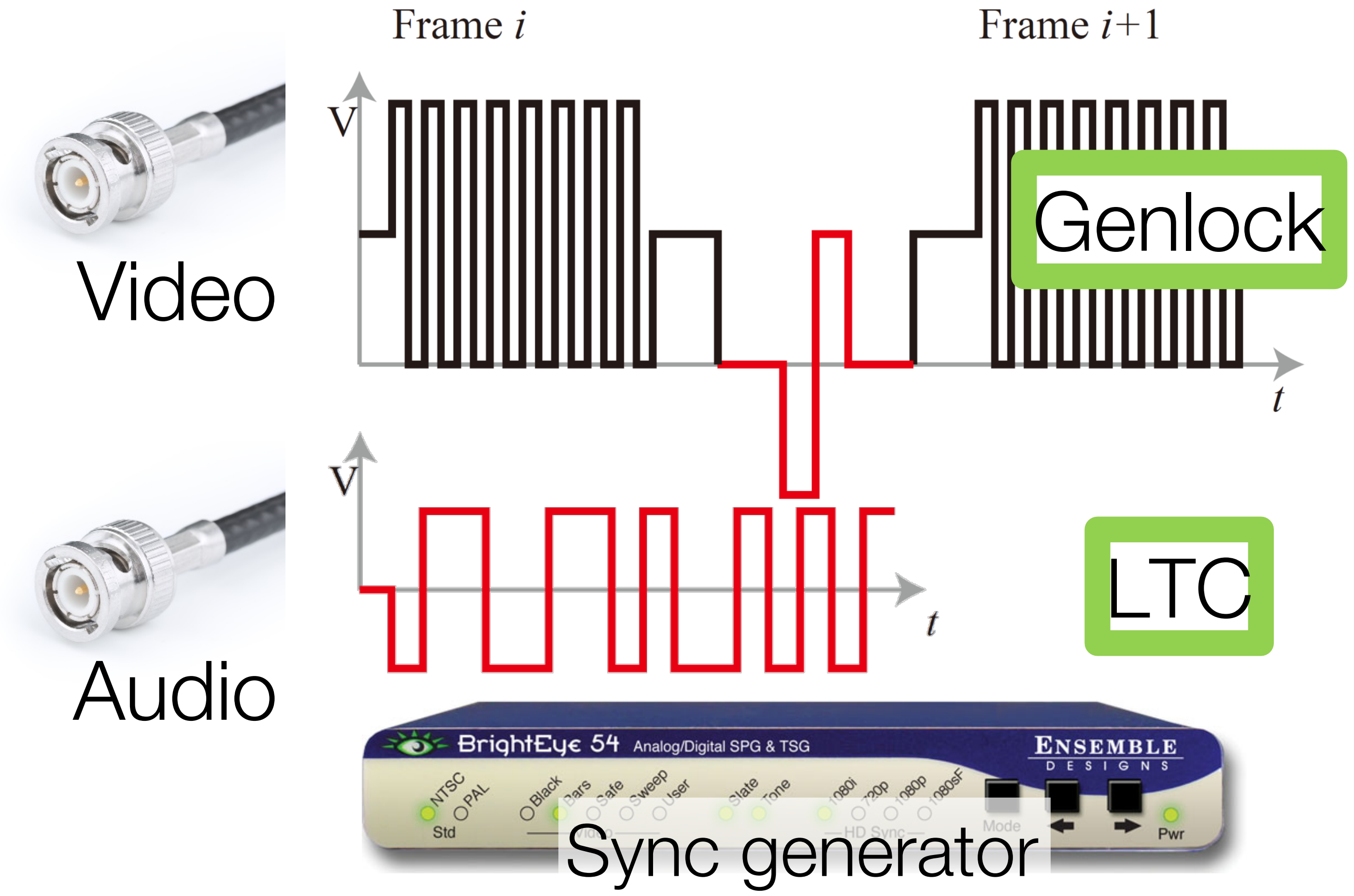
Synchronization

Different standards for machine-vision and production





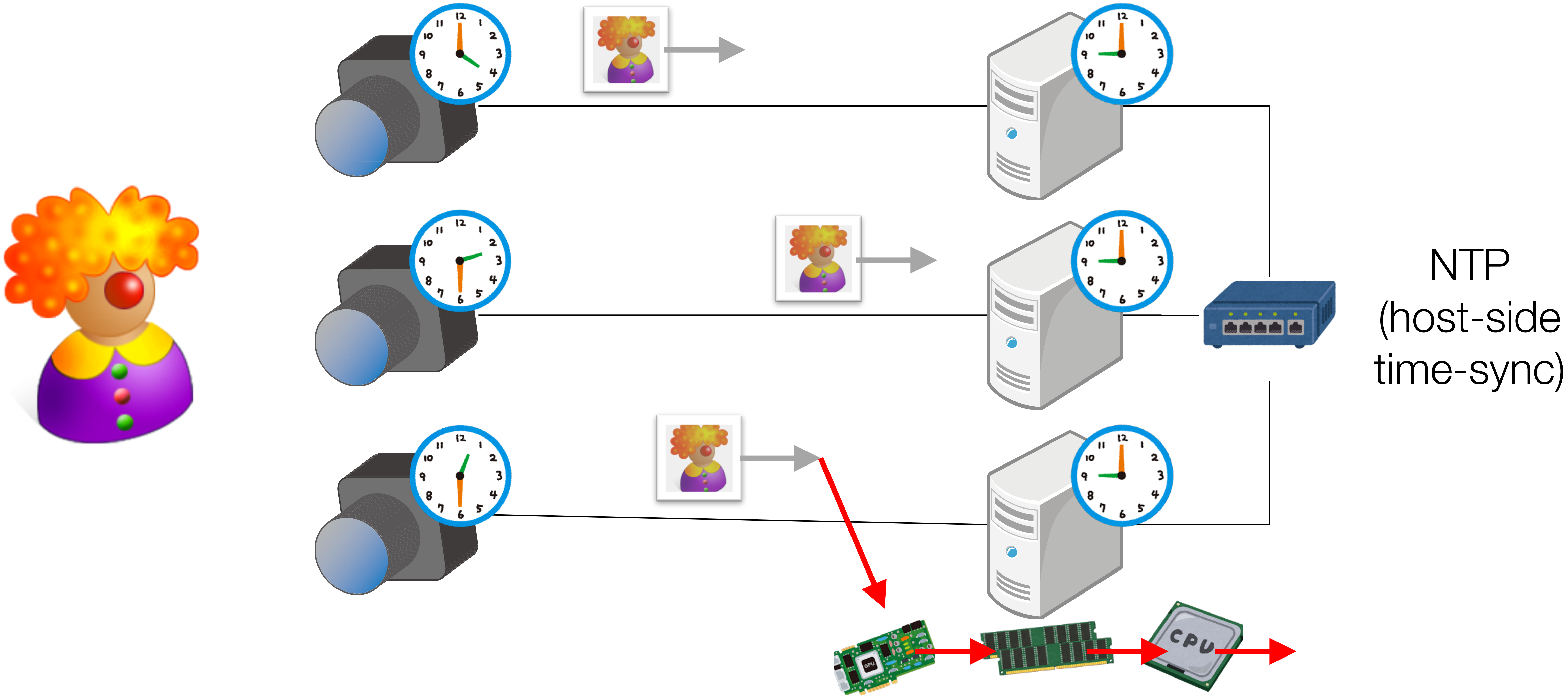
From FLIR/Pointgrey and ViewPLUS



Note: Camera can be a “generator” in both cases

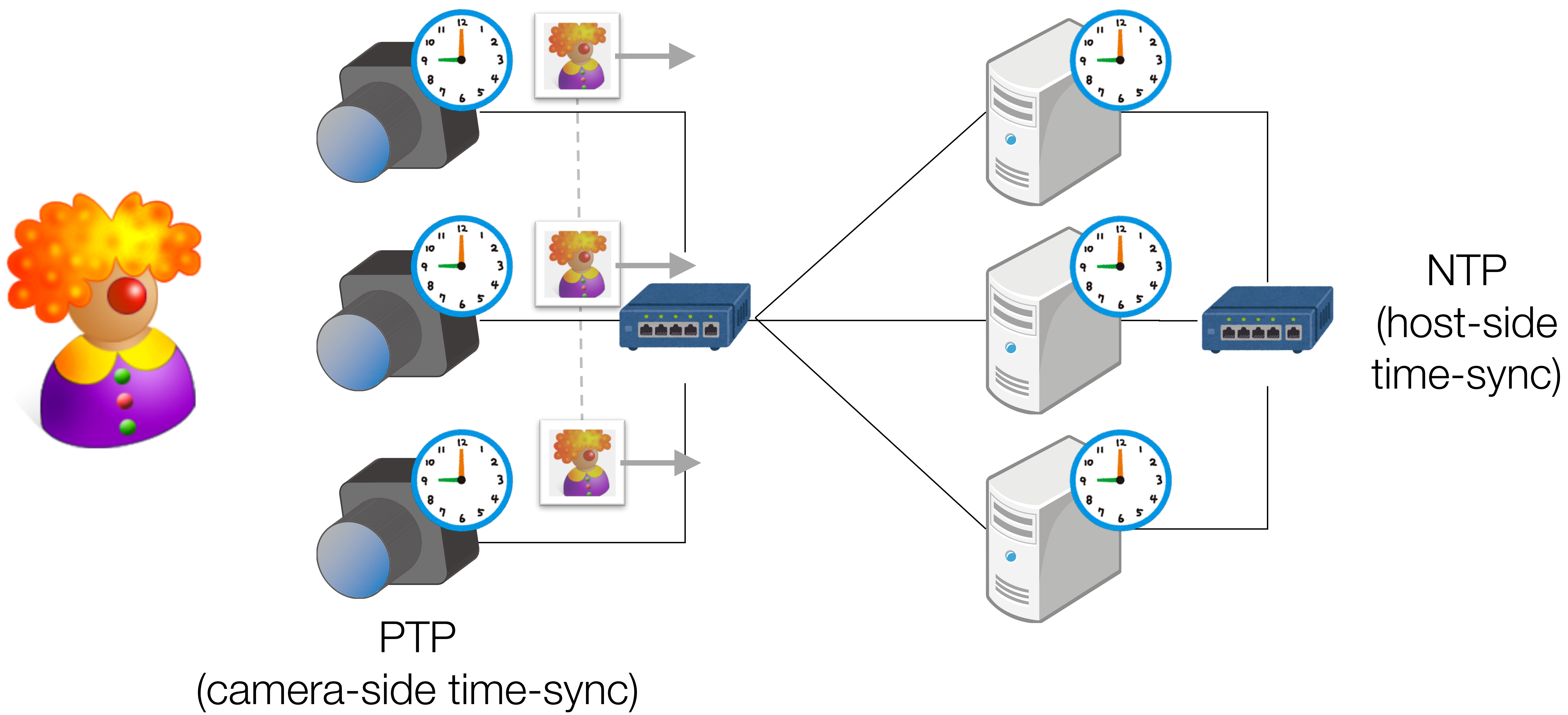
Timestamp for machine-vision cameras

Host-side or camera-side



Timestamp for machine-vision cameras

Host-side or camera-side



Synchronization

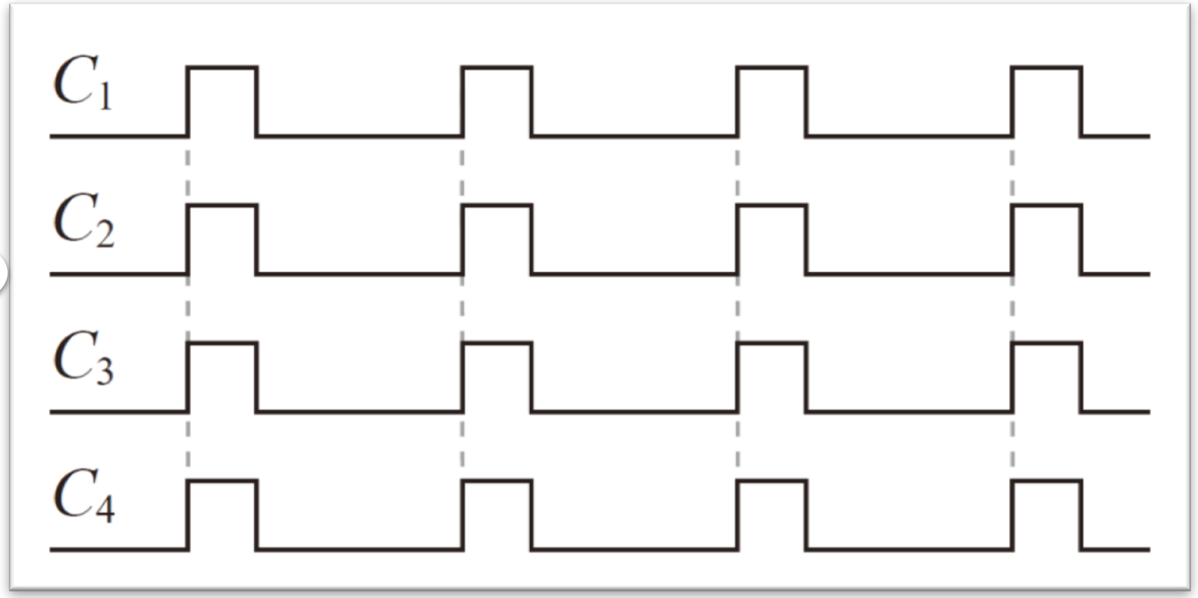
Design space

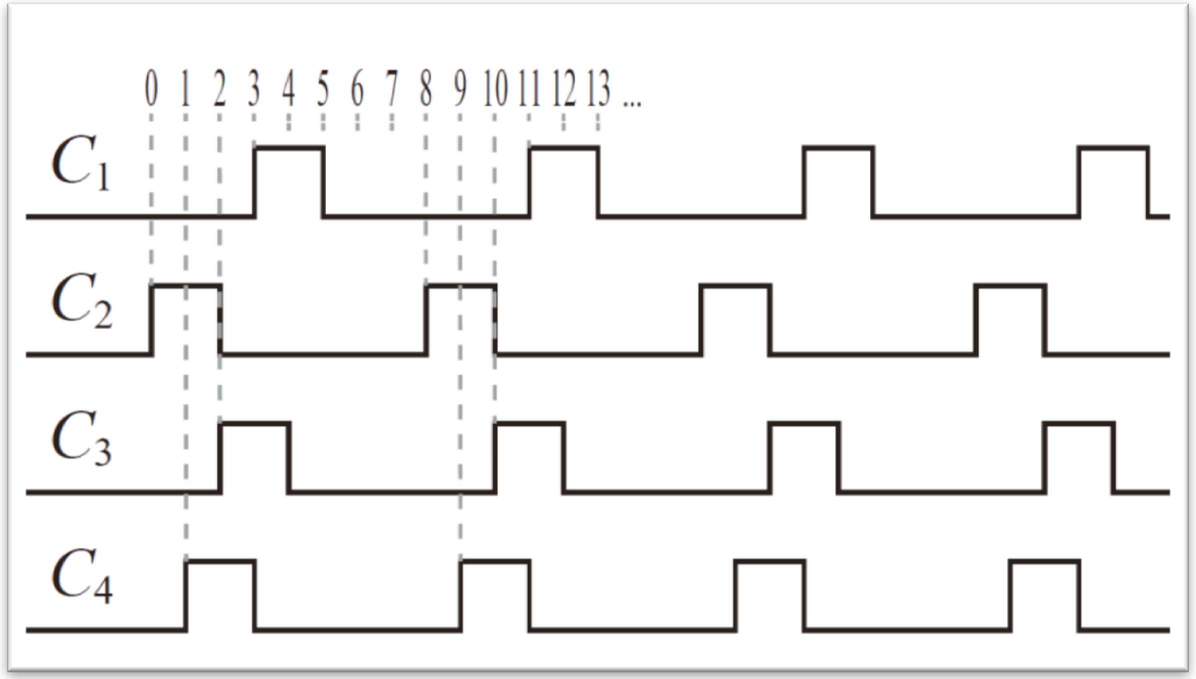
- Synchronized shutter (w/ timestamp)
  - Trigger (+LTC or PTP)
  - Genlock (+LTC)
  - PTP + synchronous free run
- Async with timestamp
  - Clapping / Flashing
  - PTP
  - Multiple LTC recording
- (None)
  - Clapping / Flashing

VGA

HD

Kinect

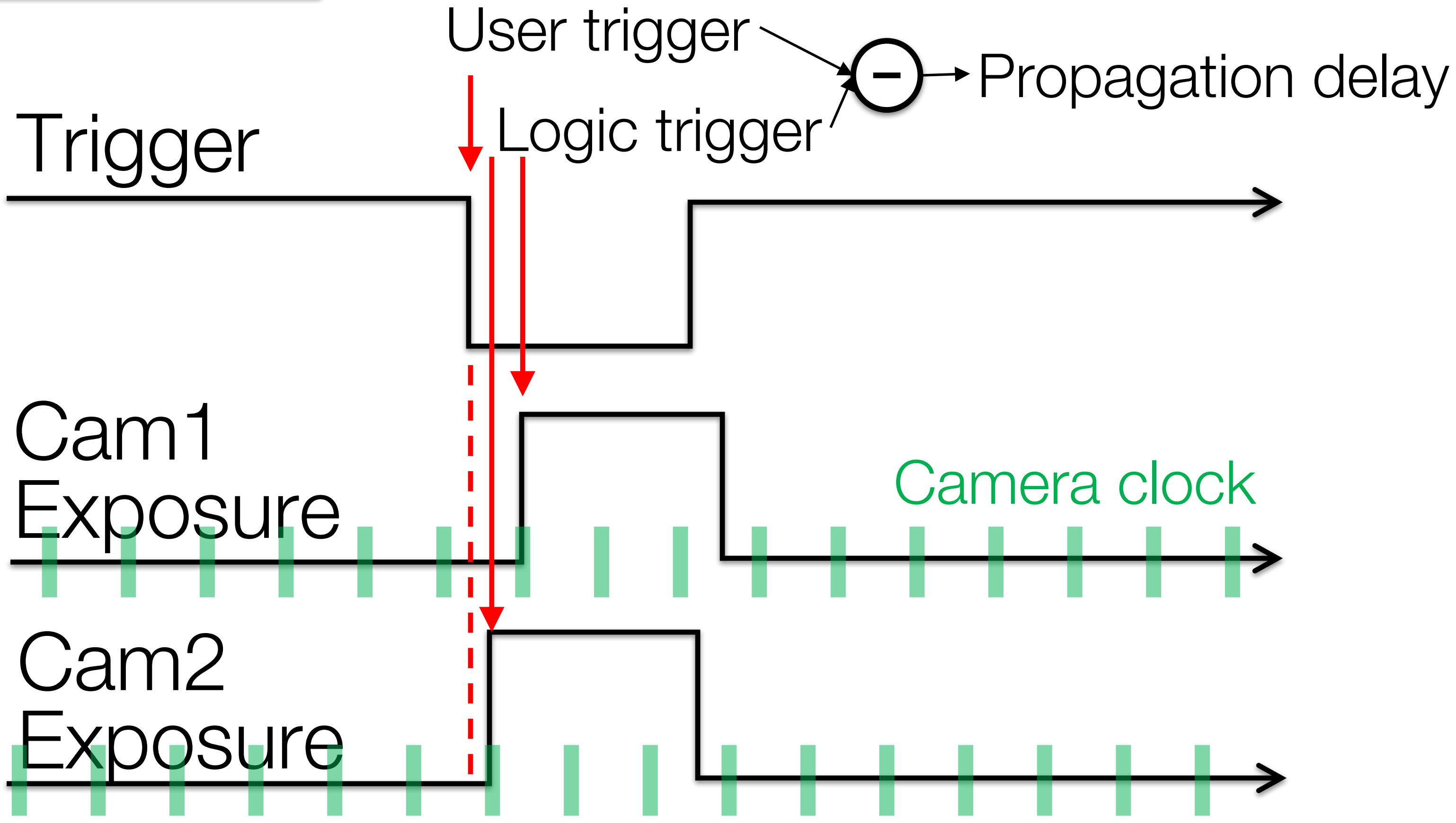




Synchronization

Trigger vs Phase-lock (Genlock)

Machine-vision

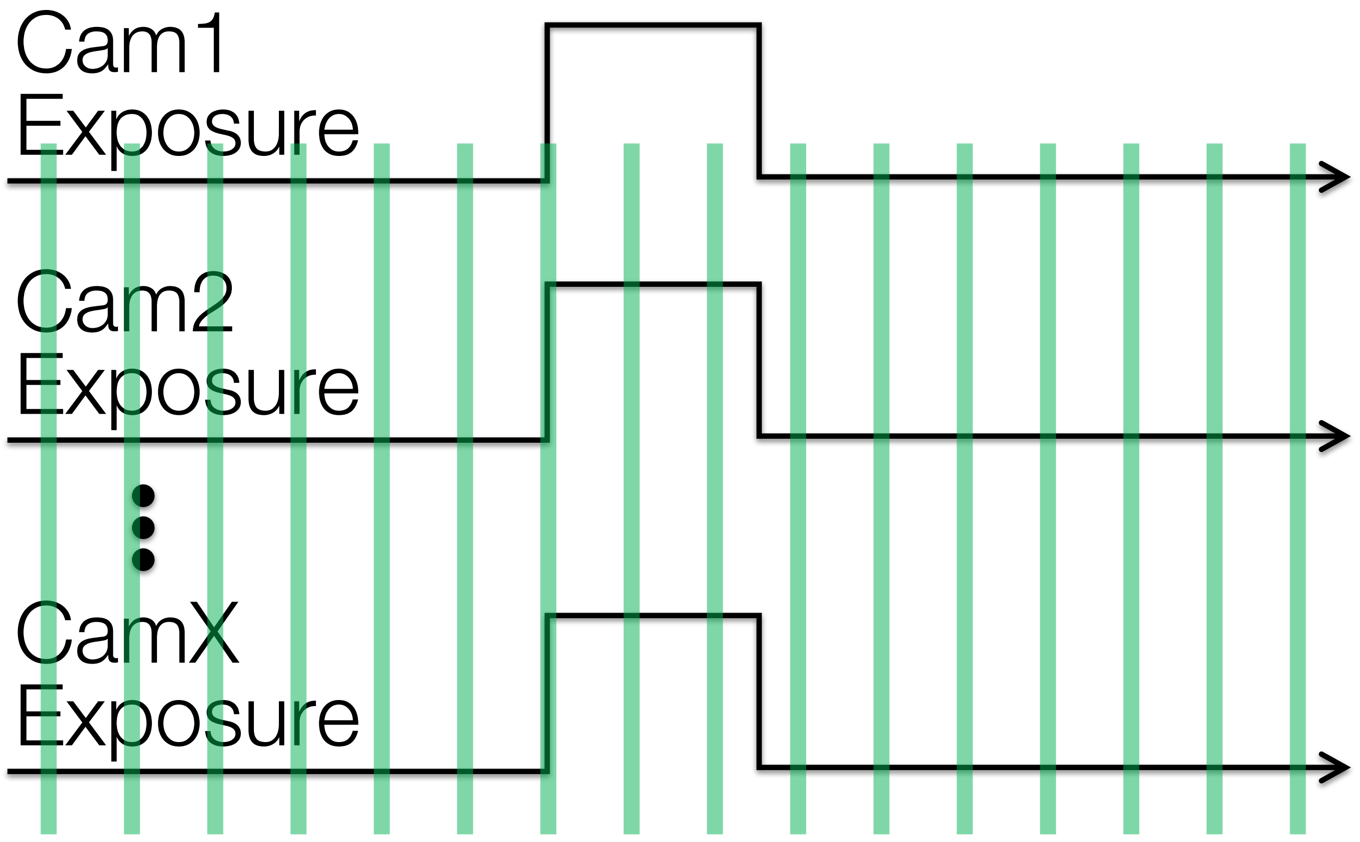


Propagation delay depends on the camera clock and circuit (observable by strobe output)

typ. 10ns for non-isolated IO-pins
typ. 10us for opt-isolated IO-pins

Obj@100km/h travels 0.27mm in 10us

Production

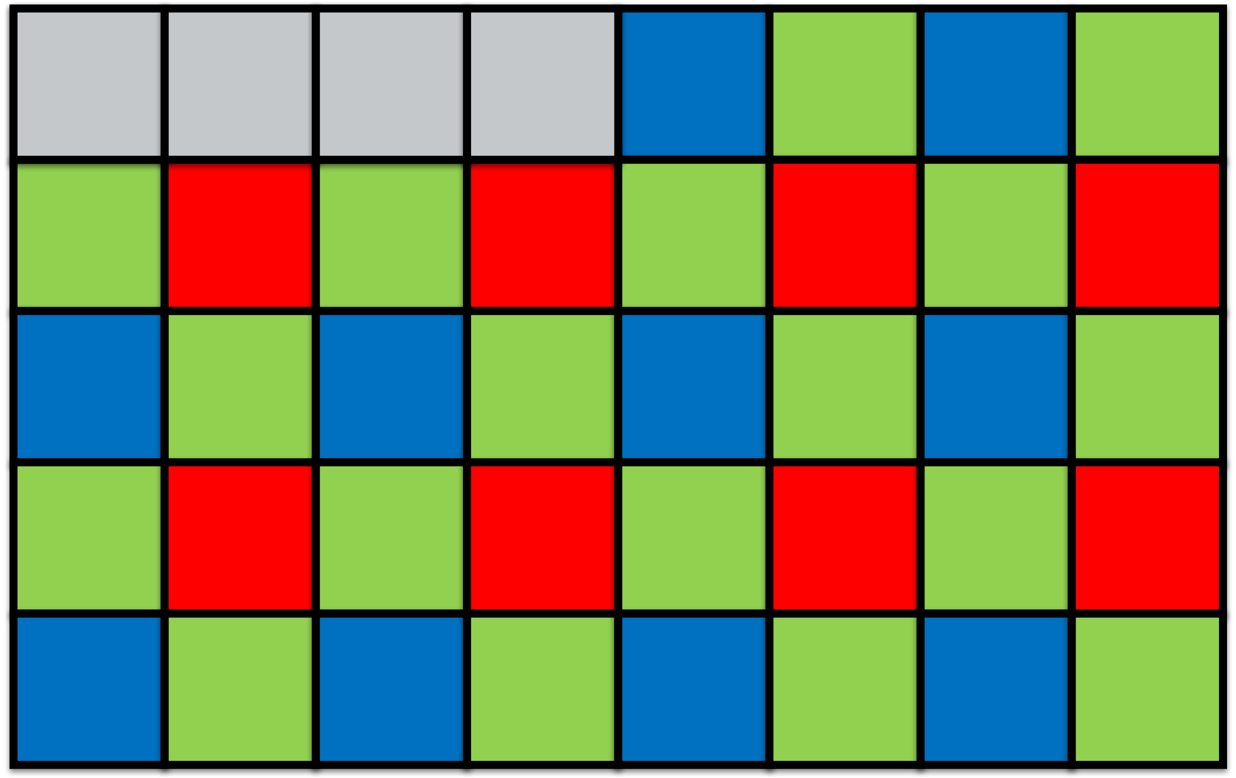


Production camera: Genlock
GigE 2.0 camera: PTP
Old 1394a camera: PGR SyncUnit

Timecode

Frame-embedding vs LTC in audio

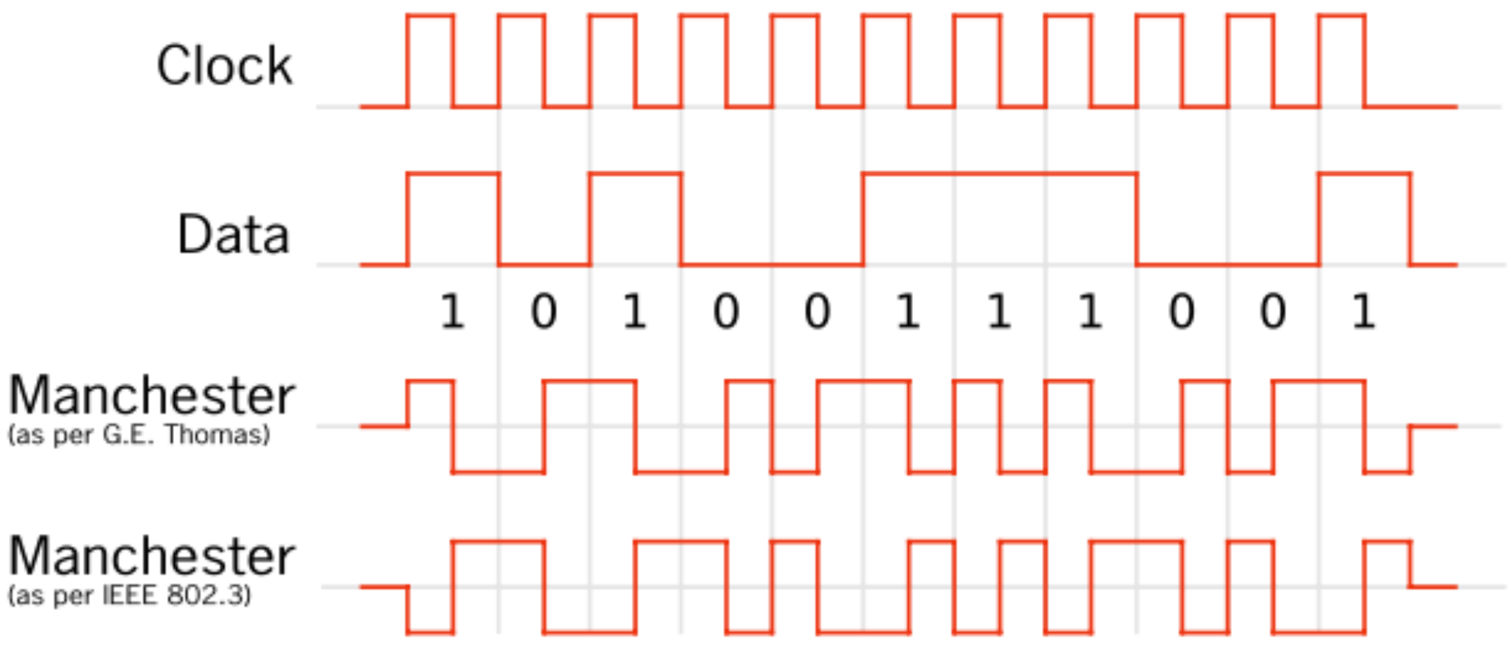
Machine-vision



First n pixels are used for meta-data

Camera clock
→Camera should know the world clock
Image count
→Useful for frame-drop detection

Production



From wikipedia

Manchester-coded binary audio signal

Originally 24/25/30Hz
→Fast-forwarded for our 100Hz system
Not for sync
→Frame start/end in time is not accurate

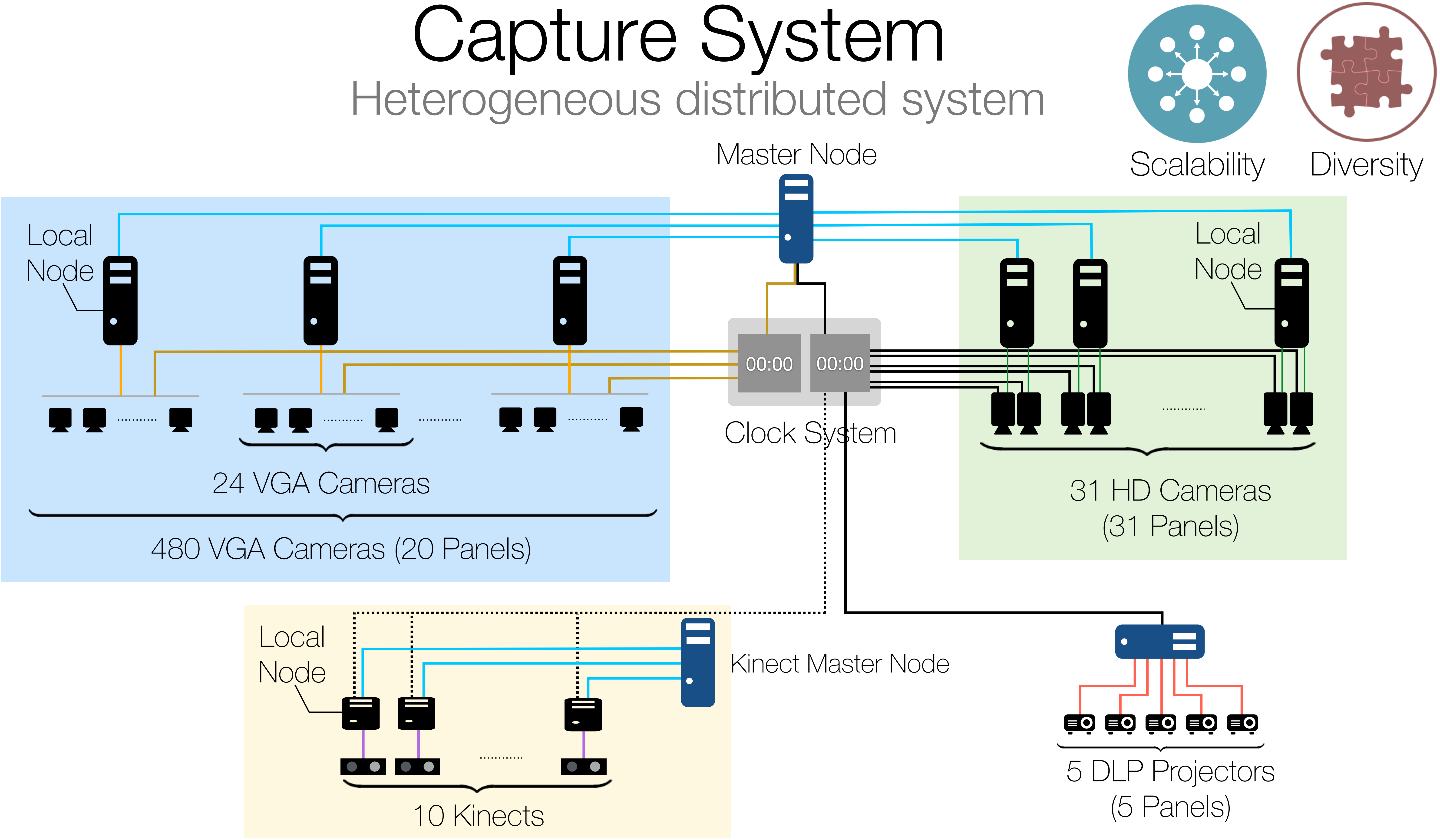
Synchronization

Decision principle

	# of cables	Scalability	Accuracy
• Synchronized shutter			
• Trigger (+LTC or PTP)	1 (+1)	++	++
• Genlock (+LTC)	1 (+1)	++	++
• PTP + synchronous free run	0	+	++
• Async with timestamp			
• Clapping / Flashing	0	+++	?
• PTP	0	+	++
• Multiple LTC recording	1	++	+
• (None)			
• Clapping / Flashing	0	+++	?

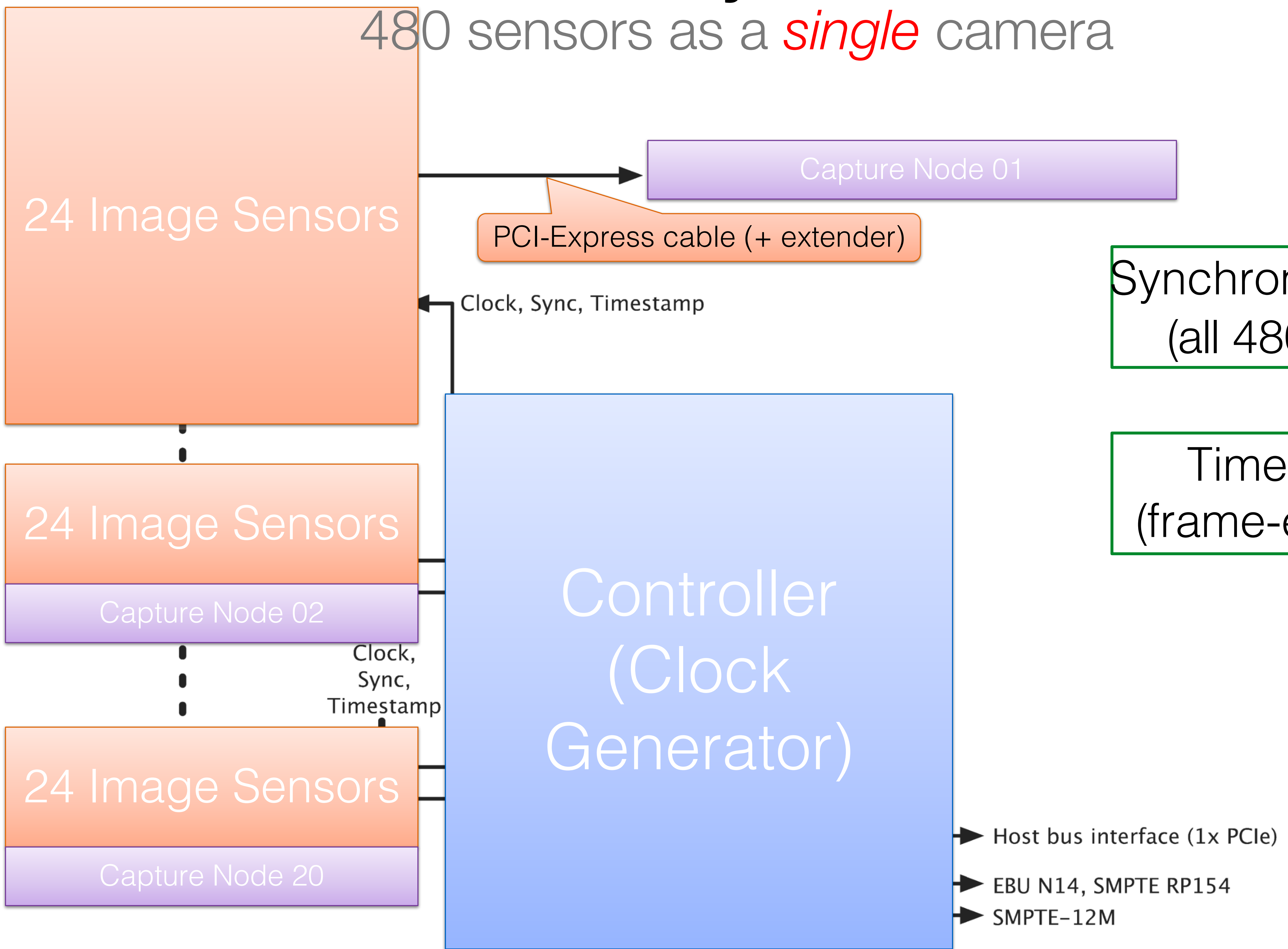
Capture System

Heterogeneous distributed system



VGA system

480 sensors as a *single* camera



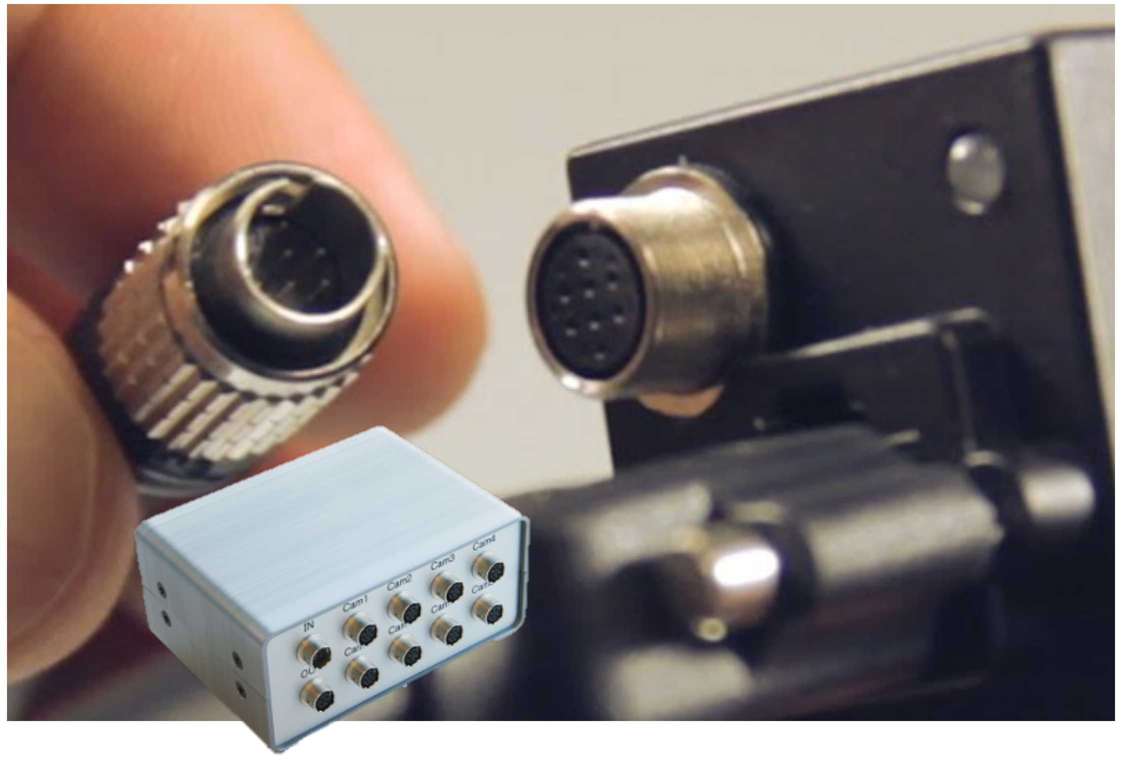
Synchronized shutter
(all 480 sensors)

Timestamped
(frame-embedded)

Synchronization

Take-home messages

- Shutter-level hardware sync is best if you can afford it.
  - Additional cables per camera
  - Signal generator / distributer



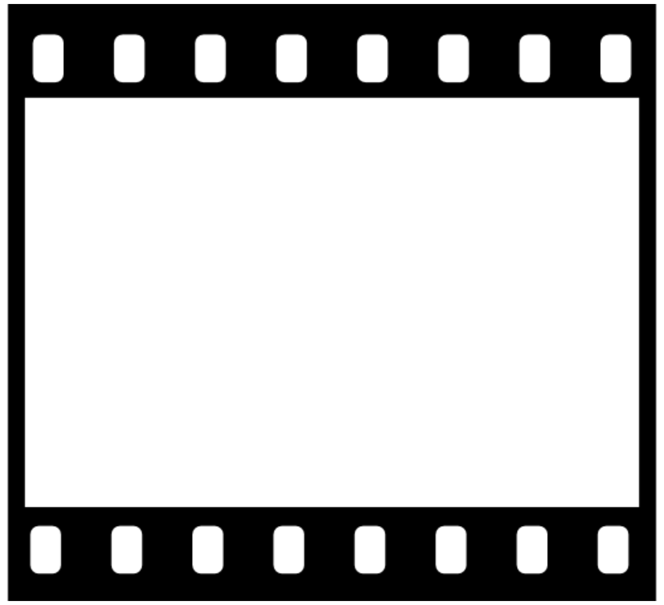
From FLIR/Pointgrey and ViewPLUS

- PTP + synchronous free run can be an option for small camera system
  - All cameras should be in a single LAN

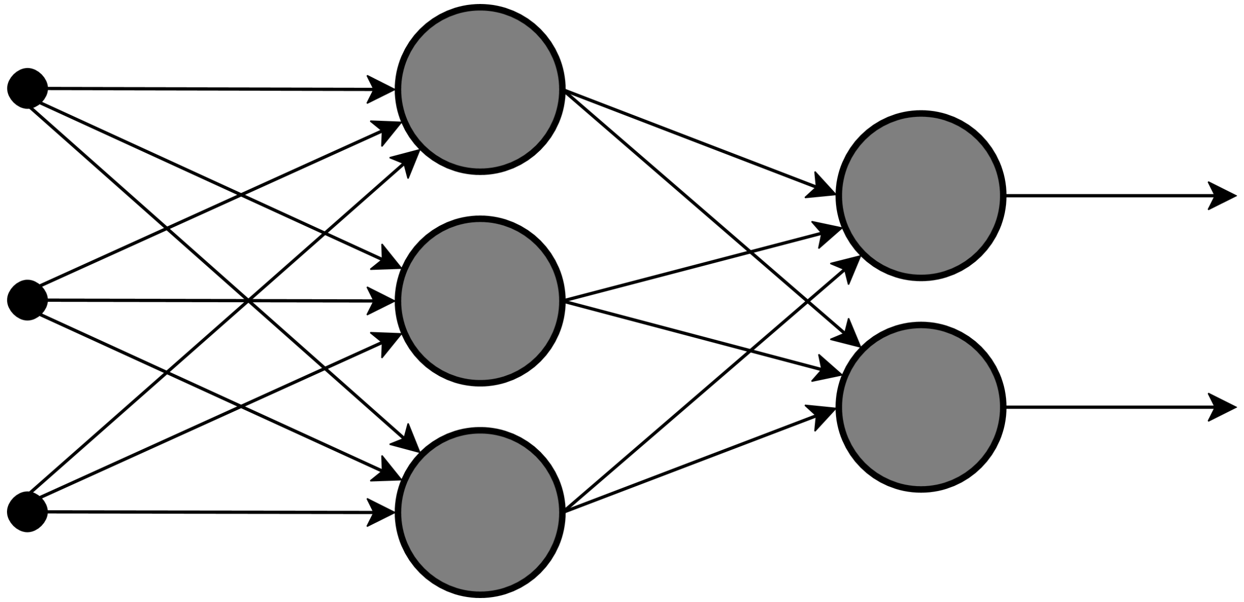
Capture System

Capture System

Why do we need PC system?



Storage

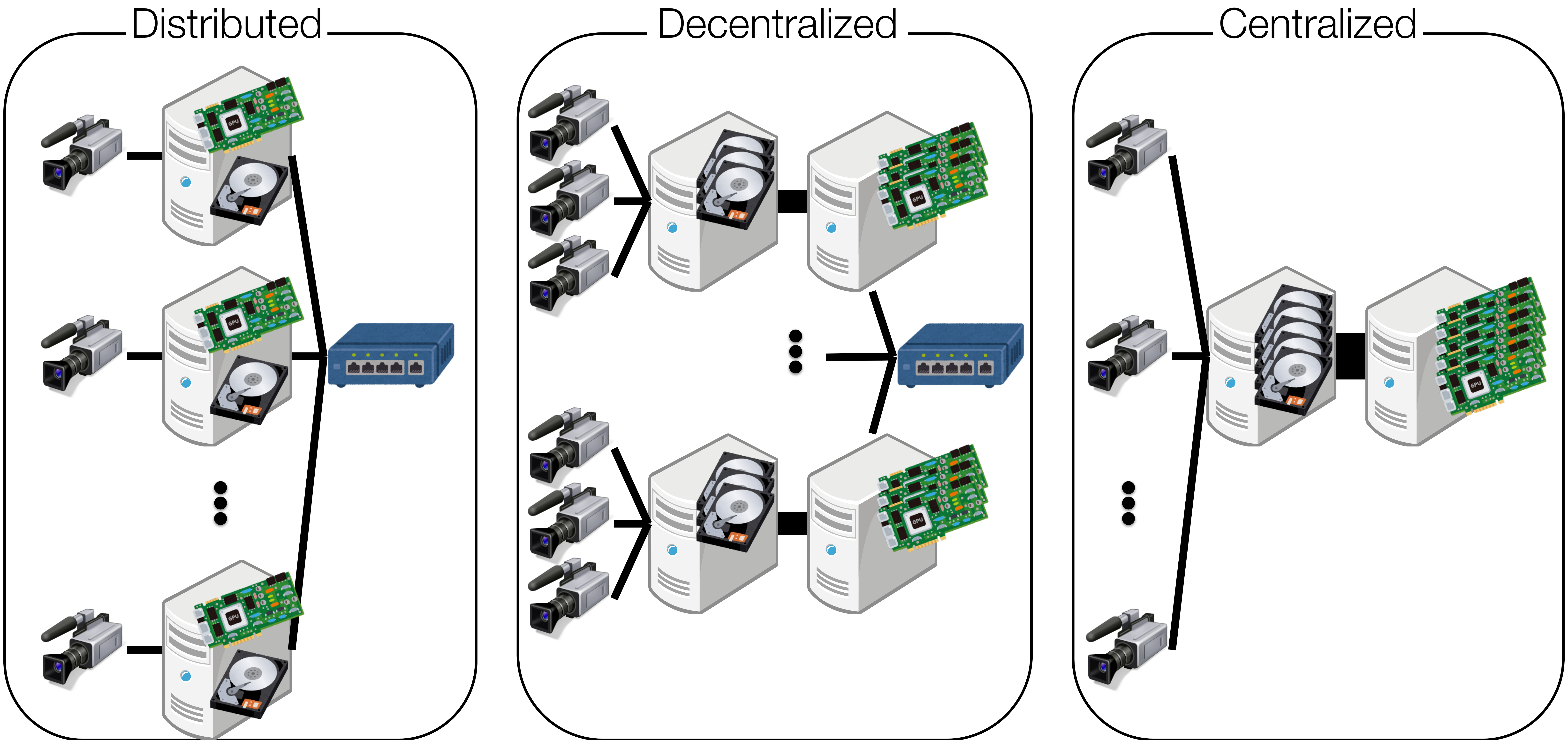


Computation

Visualization / Broadcasting

Capture System

Design space



Capture System

Design principle

Distributed

Scalability

Diversity

++ Scalability
++ Modularity
++ Single-view processing

-- Multi-view processing

Decentralized

Scalability

Diversity

+ Scalability
+ Modularity
++ Stereo processing

- Multi-view processing

Centralized

Scalability

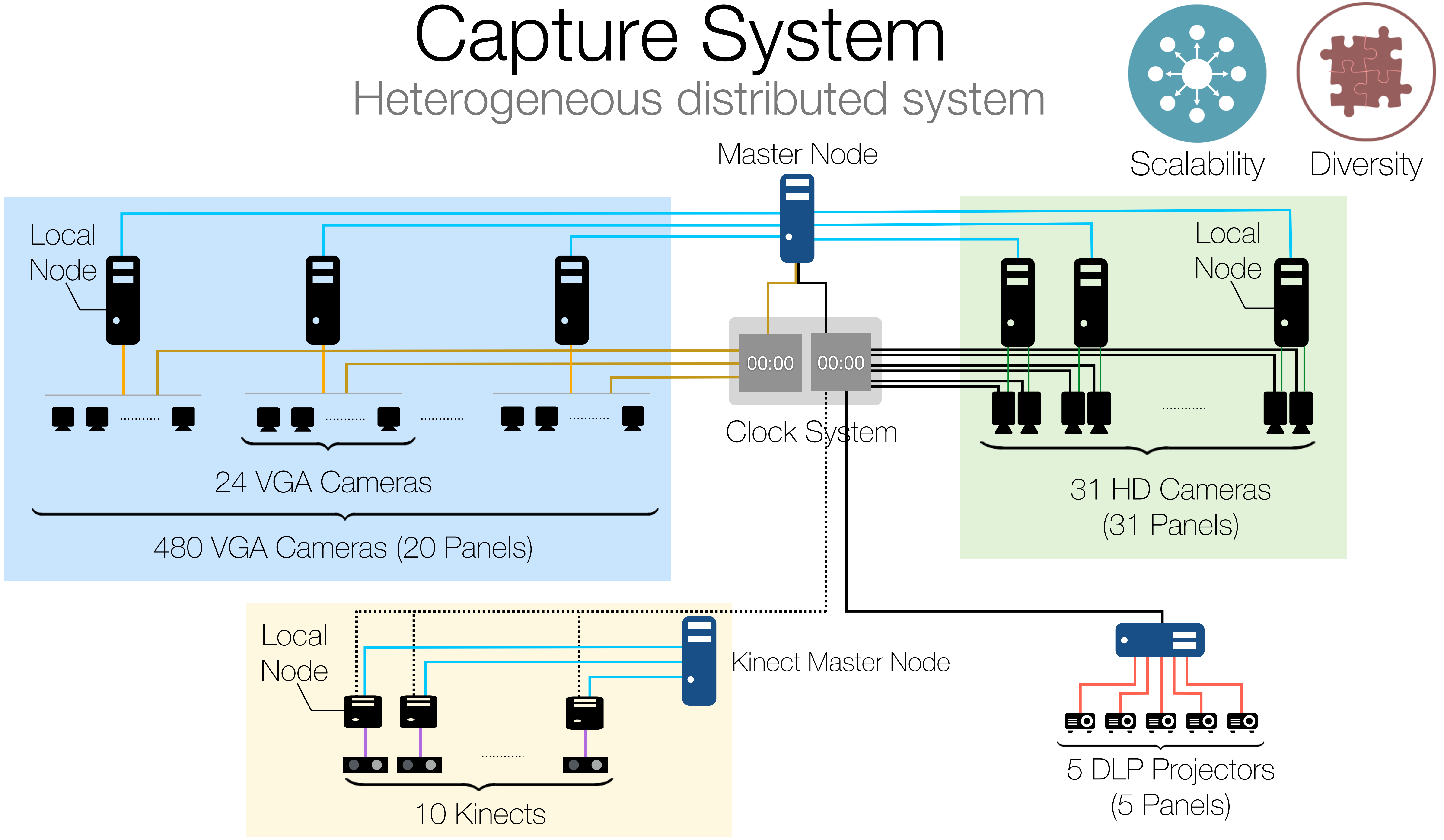
Diversity

-- Scalability
-- Modularity
-- Single-view processing

++ Multi-view processing

Capture System

Heterogeneous distributed system



PC System

Take-home messages

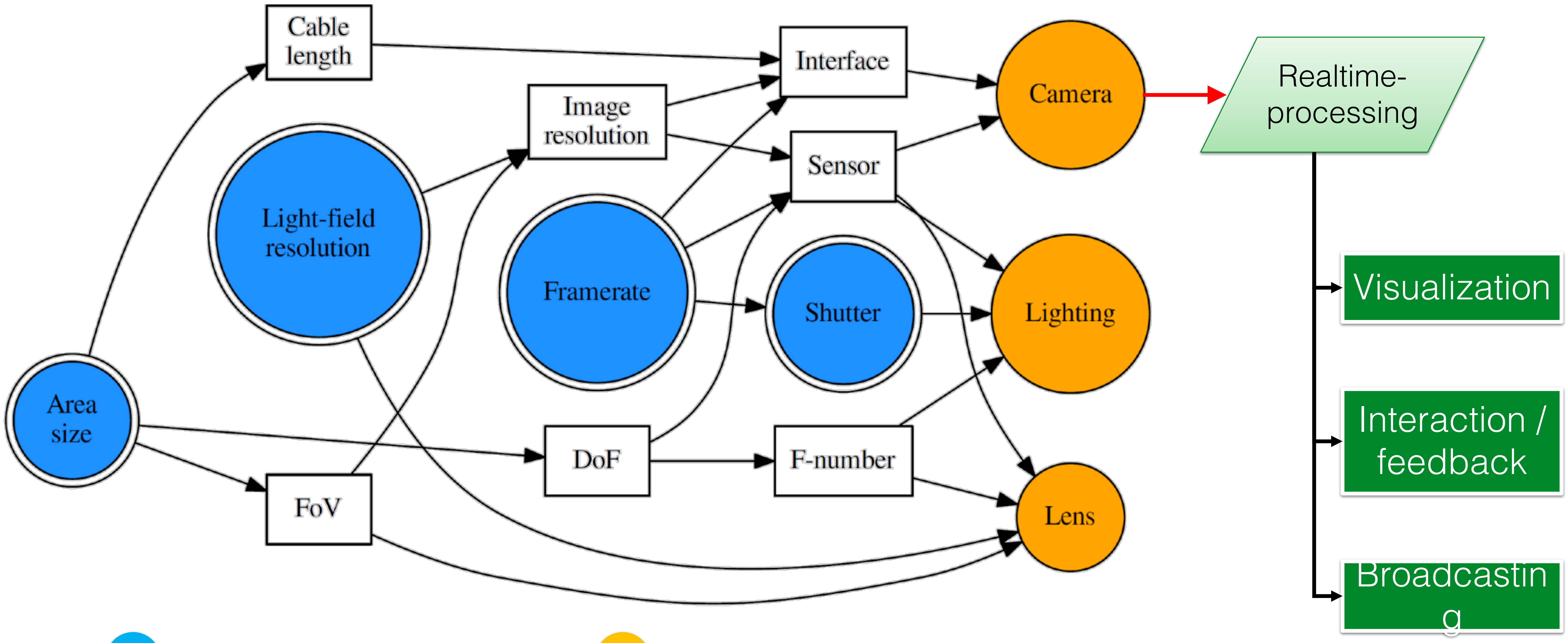
- Distributed architecture is required for large-scale system
    - Modular design = easier maintenance

- Centralized architecture can be better for smaller system
    - Simpler real-time multi-view processing
        - No need to exchange images among nodes

Redesign from scratch?

Dependencies

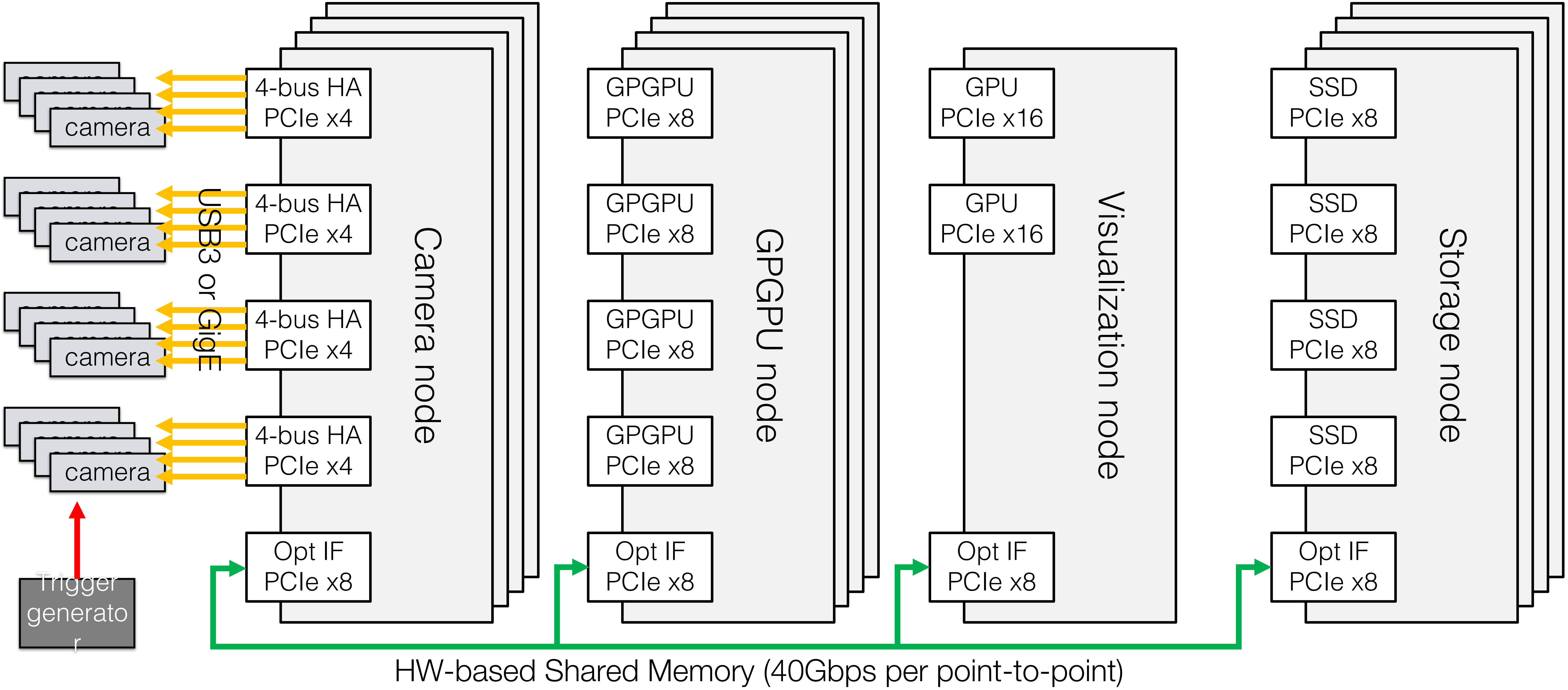
Different Designs for Different Goals



Input / Requirements   Output / Conclusions

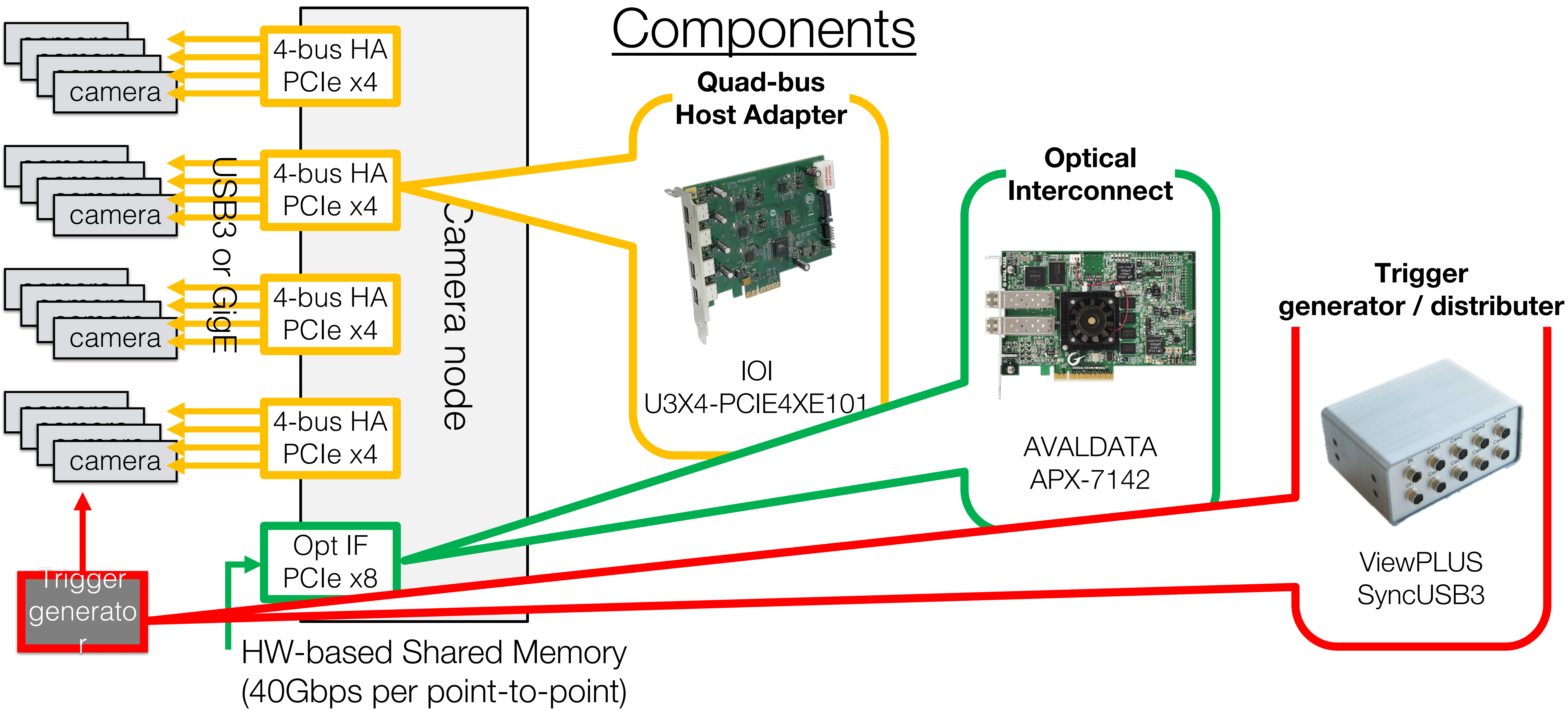
Redesign from scratch?

Diversity / scalability / realtime-ness



Redesign from scratch?

Diversity / scalability / realtime-ness

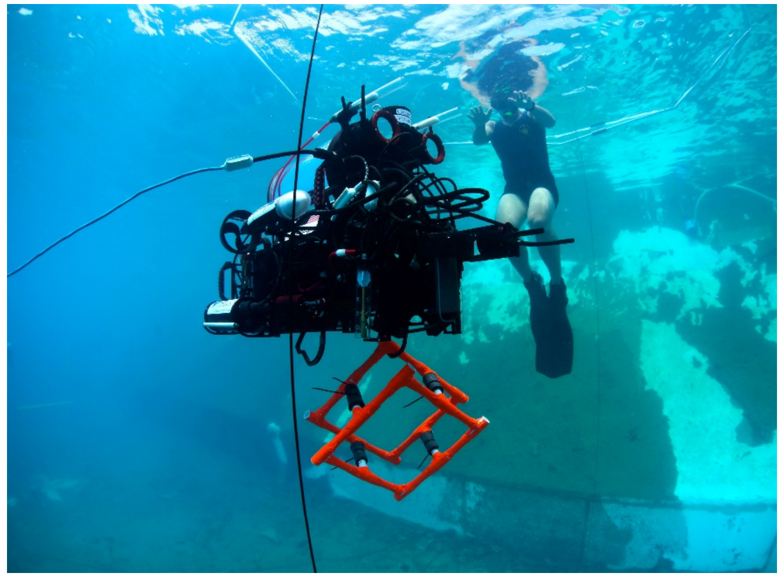


Discussions

Limitations, future work

Large-scale, Outdoor

- Cabling?
- Pan/tilt robot?
- Lens control?



Underwater

- Cabling?
- Illumination?

Microscale

- DoF?
- Multi-view?

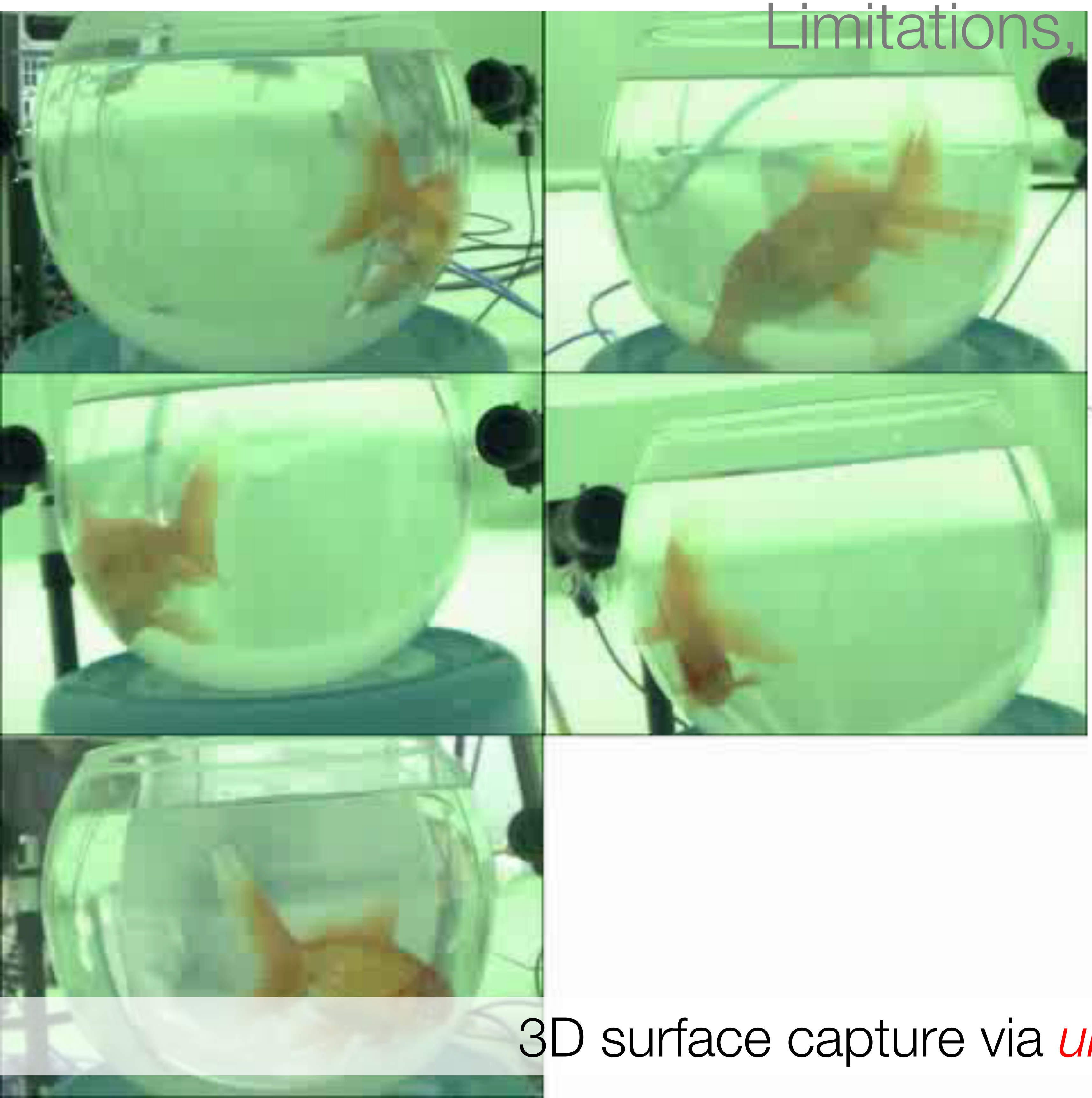
Discussions

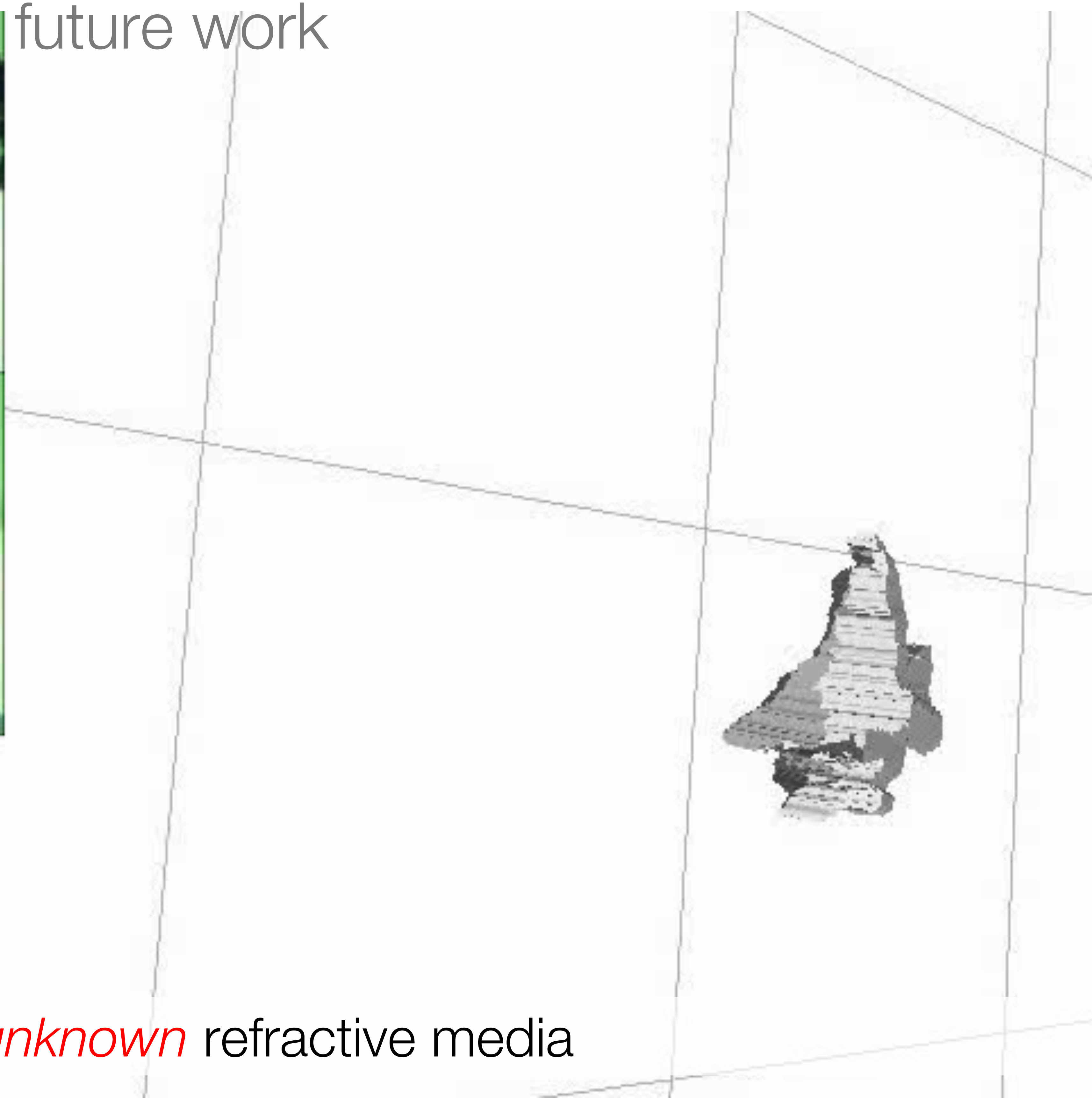
Limitations, future work

3D surface capture via *unknown* refractive media

Discussions

Limitations, future work

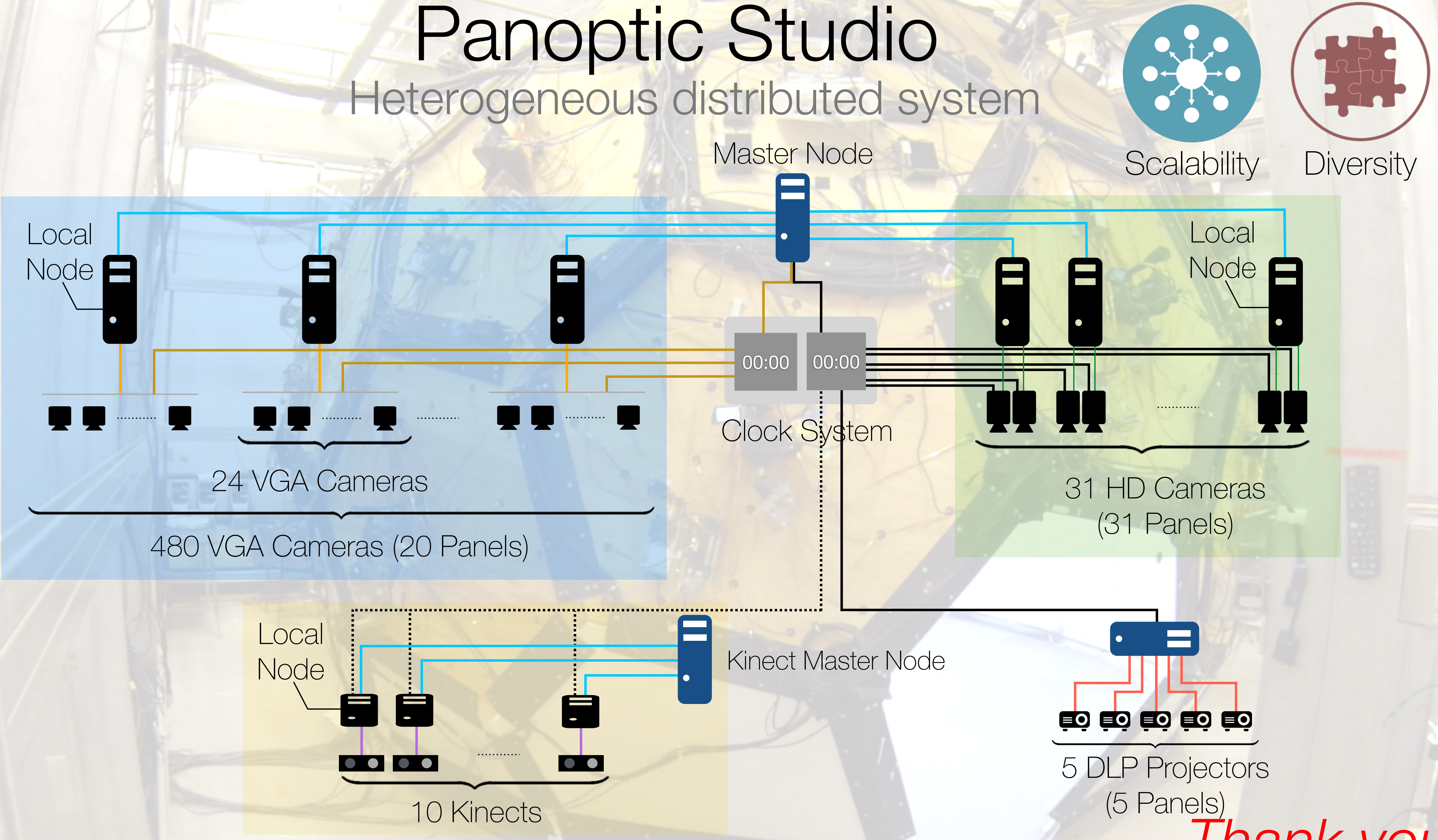




3D surface capture via *unknown* refractive media

Panoptic Studio

Heterogeneous distributed system



Thank you!