Parallel Rendering and Large Data Visualization

PhD Defense Talk

Stefan Eilemann

VISUALIZATION AND MULTIMEDIA LAB

University of Zurich[UZH]

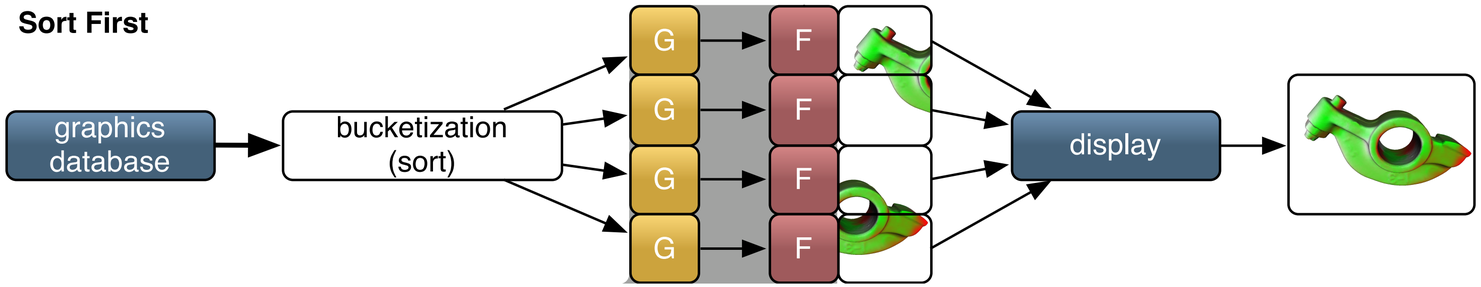
Outline

- Parallel and Scalable Rendering
- Load Balancing
- Equalizer Framework Architecture

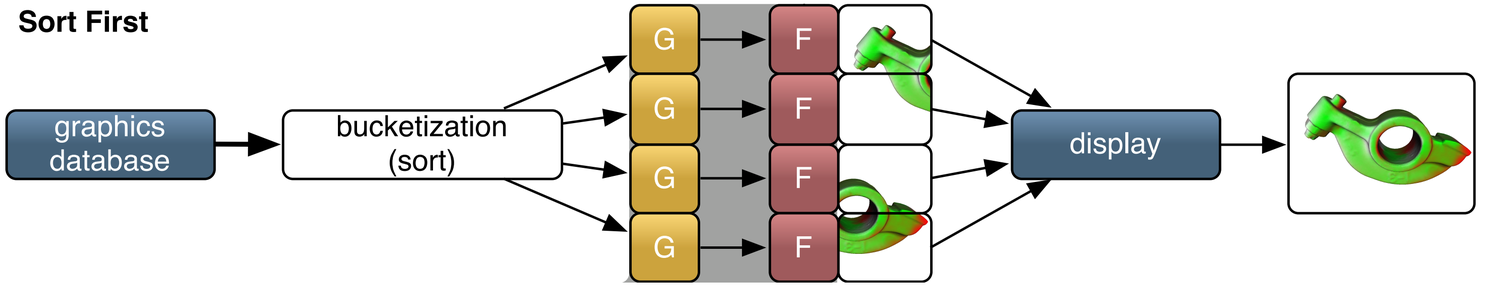
Parallel and Scalable Rendering

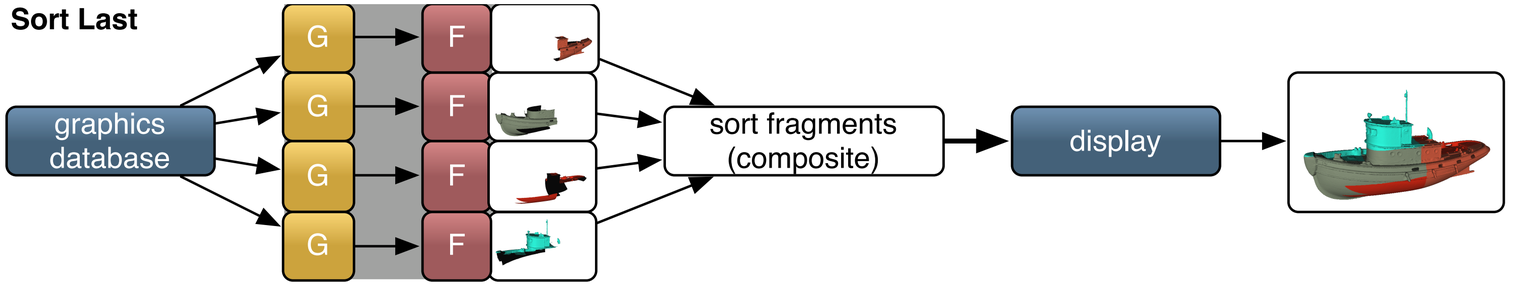
Parallel Rendering

Scalable Rendering

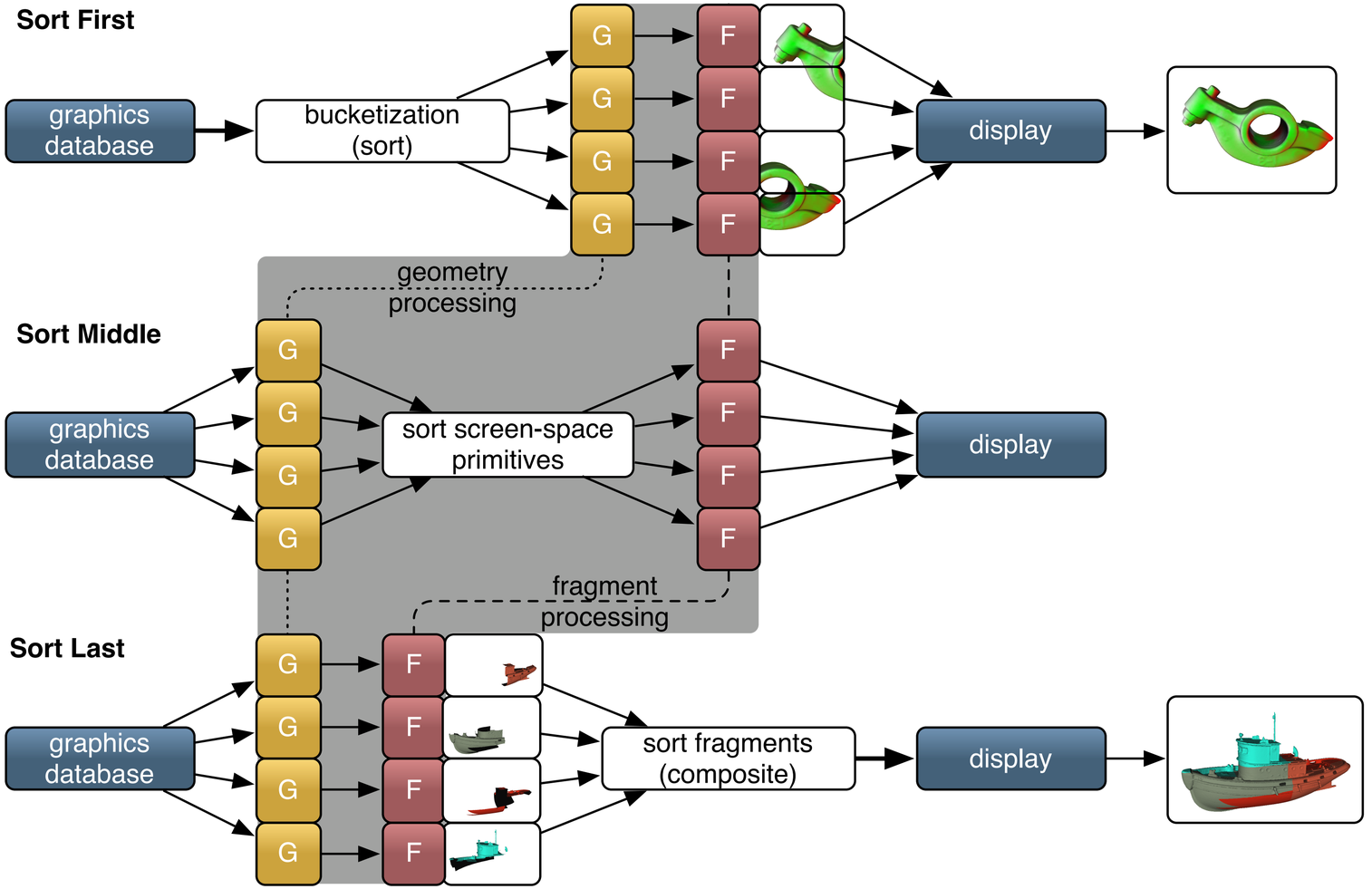


Scalable Rendering





Scalable Rendering



Load Balancing

How we can improve load-balancing for sort-first rendering, in particular for large display systems?

Single Display Load Balancing

- Static, Reactive,
  Predictive, Implicit

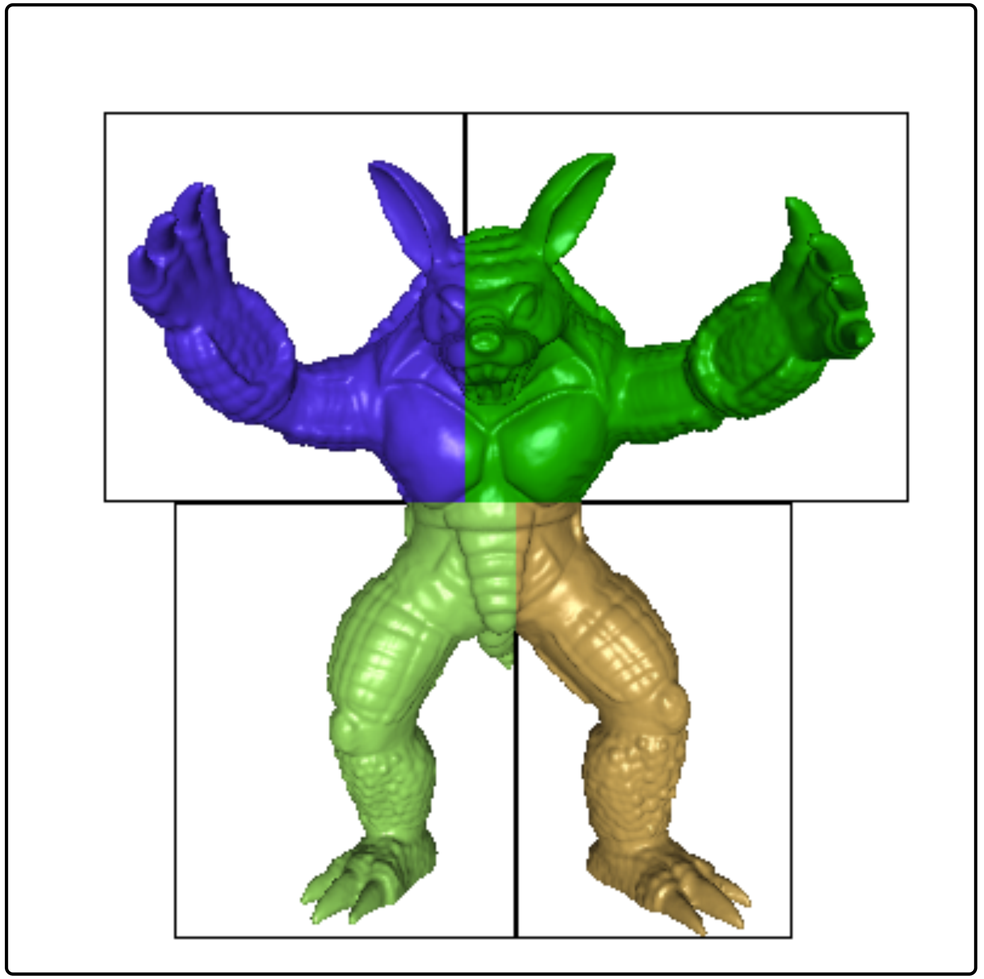
Single Display Load Balancing

- Static, Reactive, Predictive, Implicit

- Equalizers for runtime adaptations

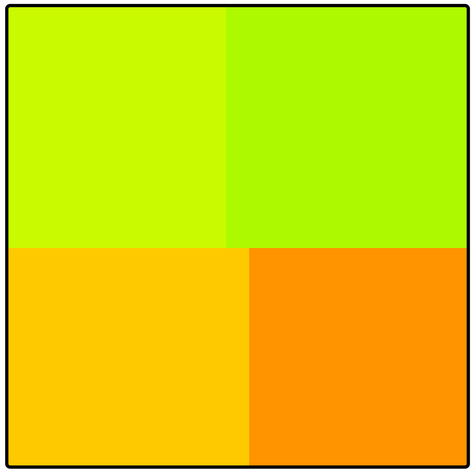
Single Display Load Balancing

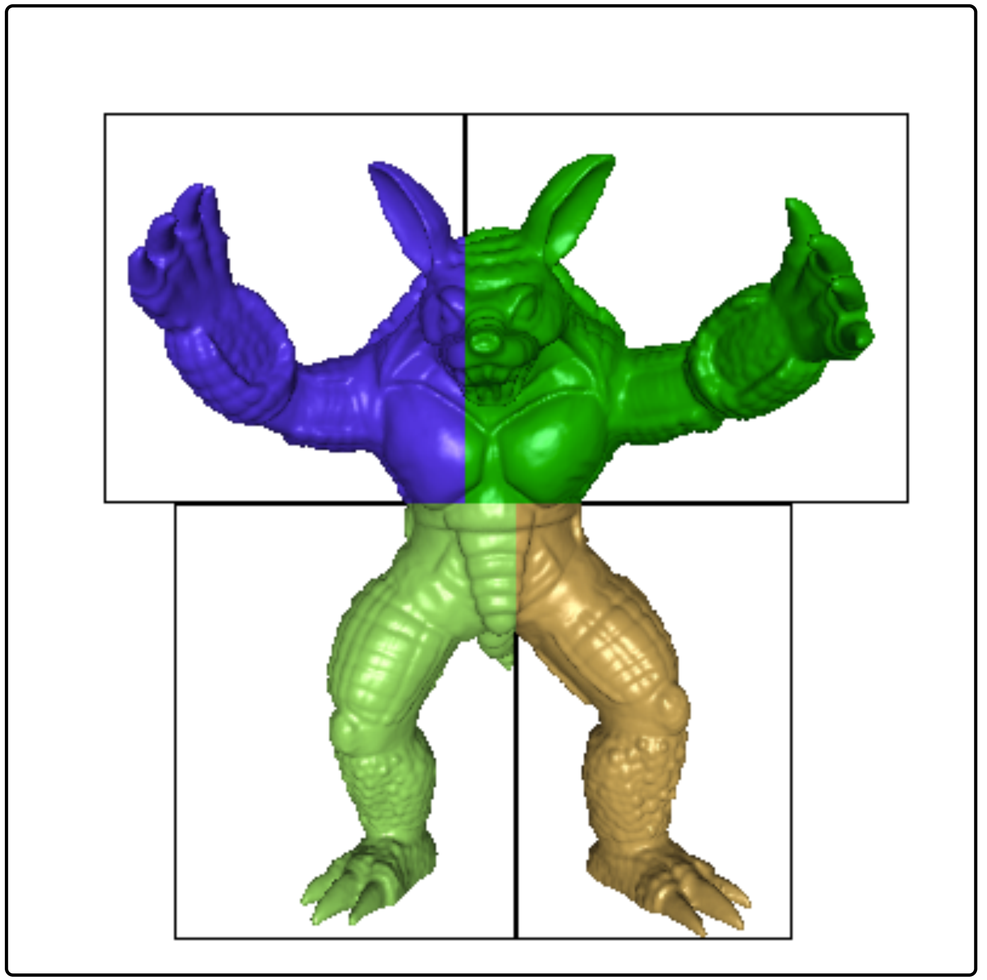
- Static, Reactive, Predictive, Implicit
- Equalizers for runtime adaptations
- Binary split tree



Single Display Load Balancing

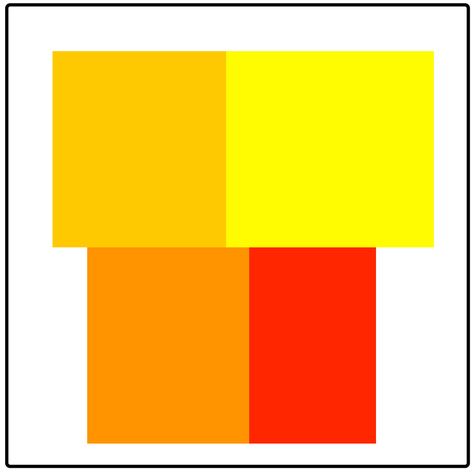
- Static, Reactive, Predictive, Implicit
- Equalizers for runtime adaptations
- Binary split tree
  - Load Equalizer: Integrate over load grid

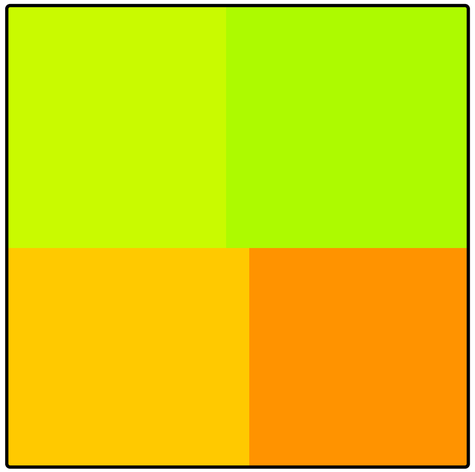


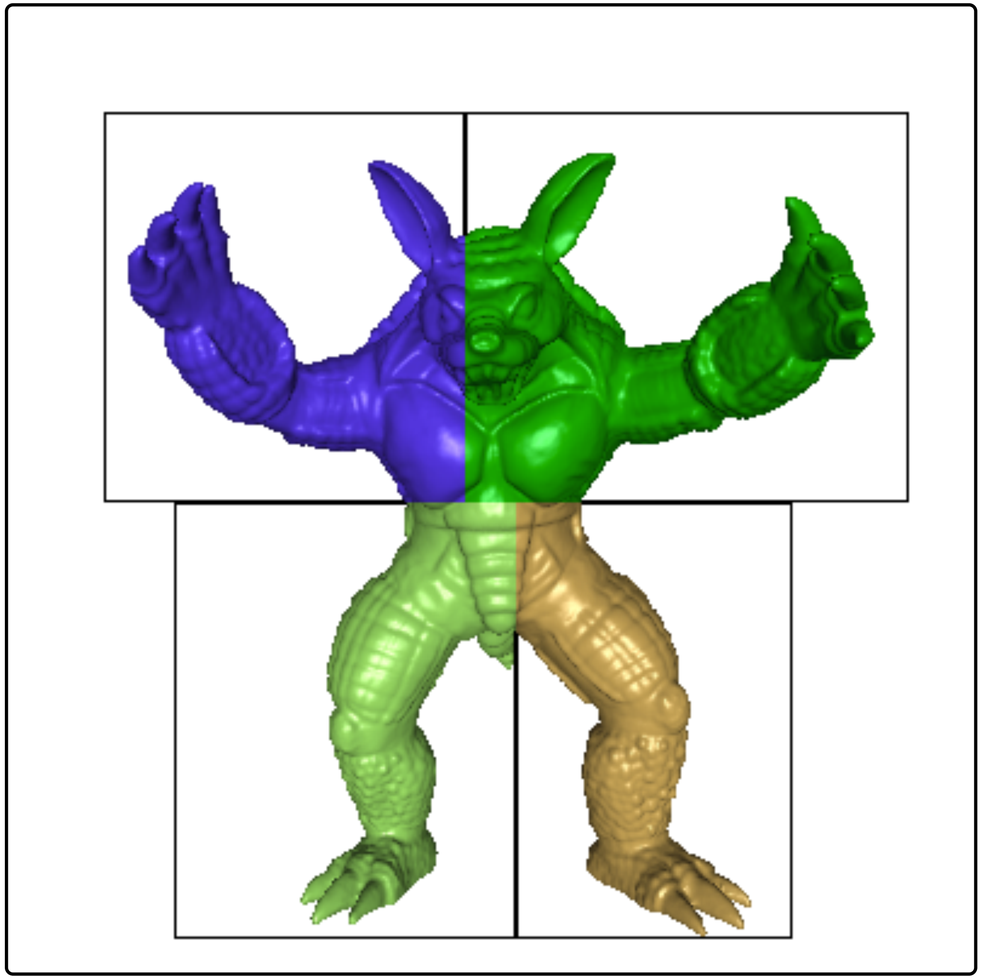


Single Display Load Balancing

- Static, Reactive, Predictive, Implicit
- Equalizers for runtime adaptations
- Binary split tree
  - Load Equalizer: Integrate over load grid

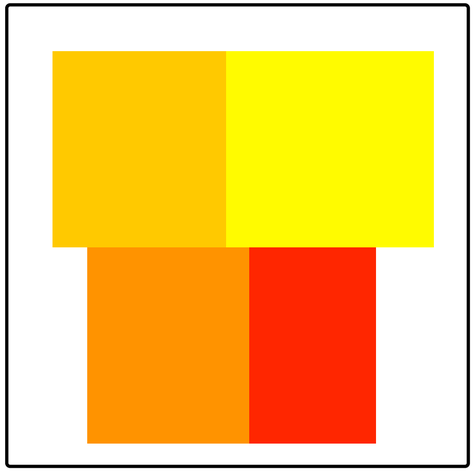


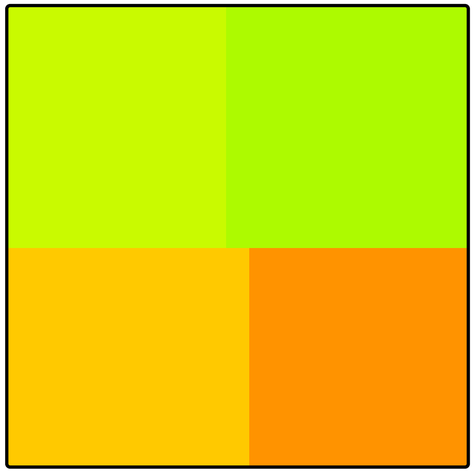


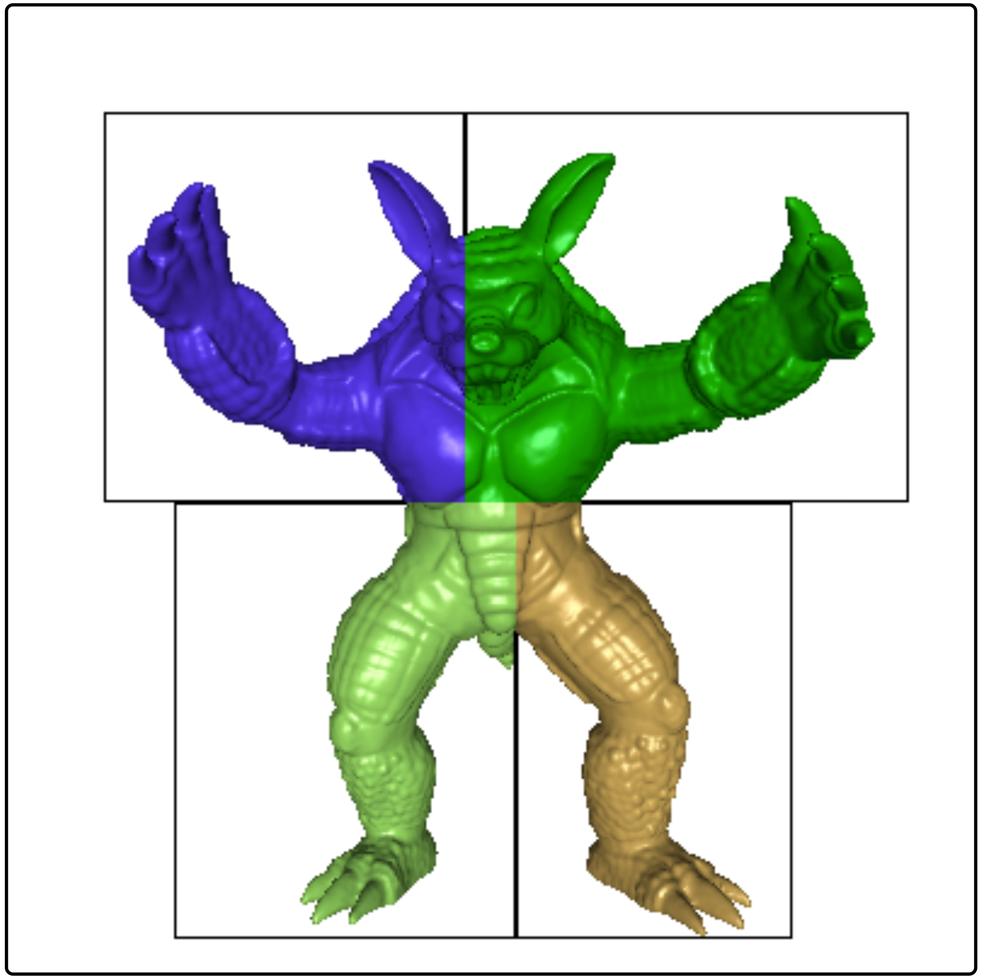


Single Display Load Balancing

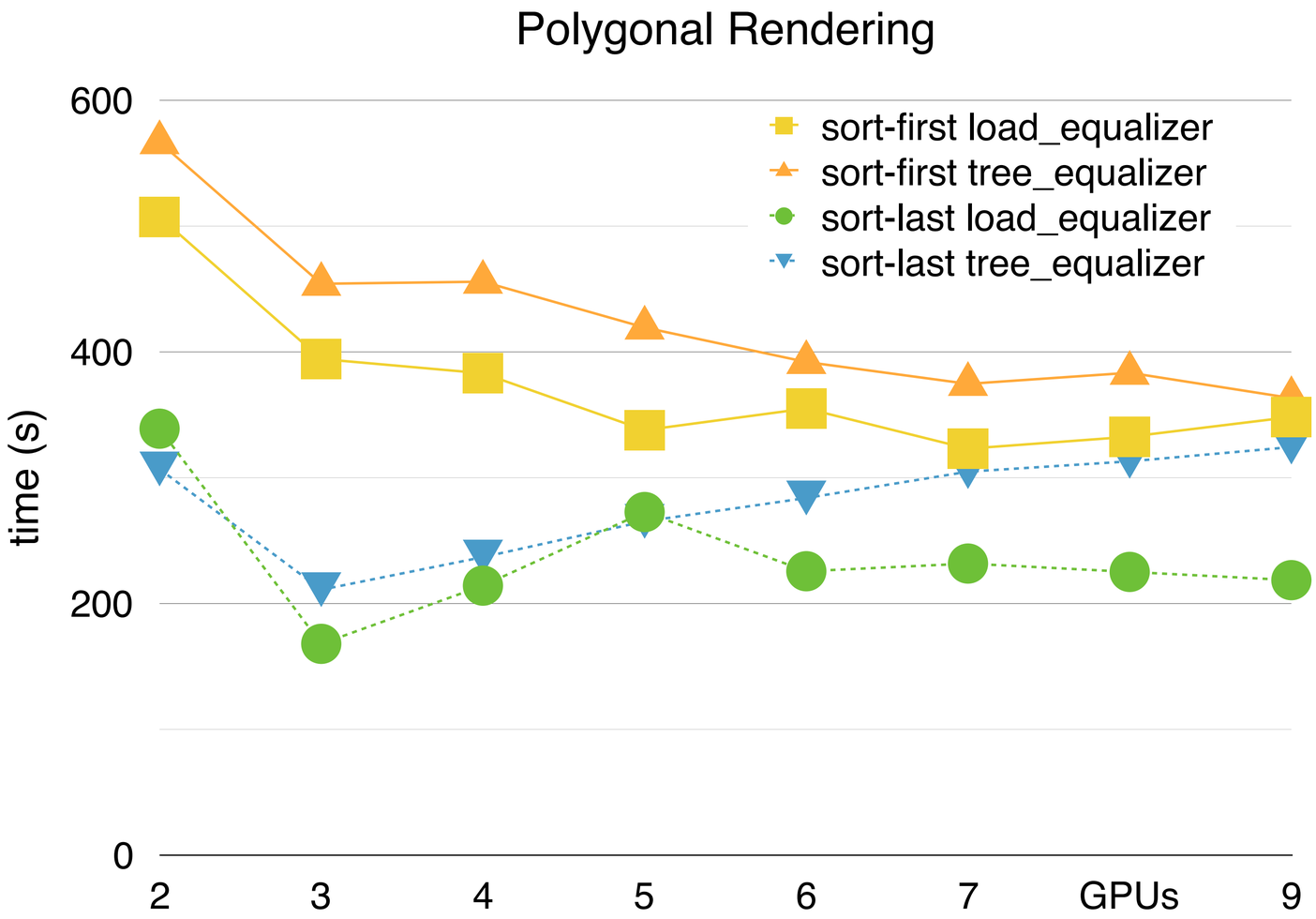
- Static, Reactive, Predictive, Implicit
- Equalizers for runtime adaptations
- Binary split tree
  - Load Equalizer: Integrate over load grid
  - Tree Equalizer: Balance left/right render time

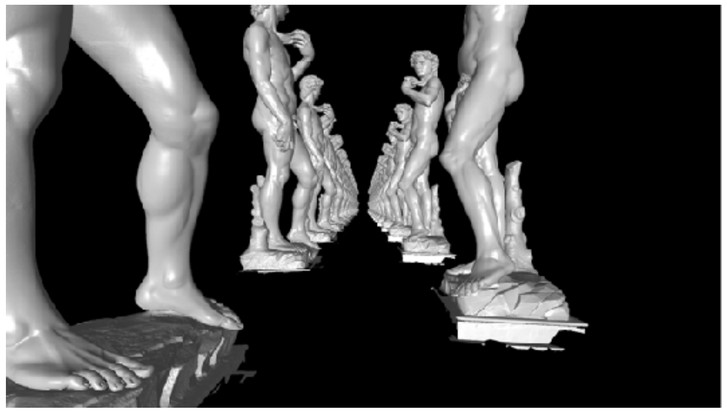






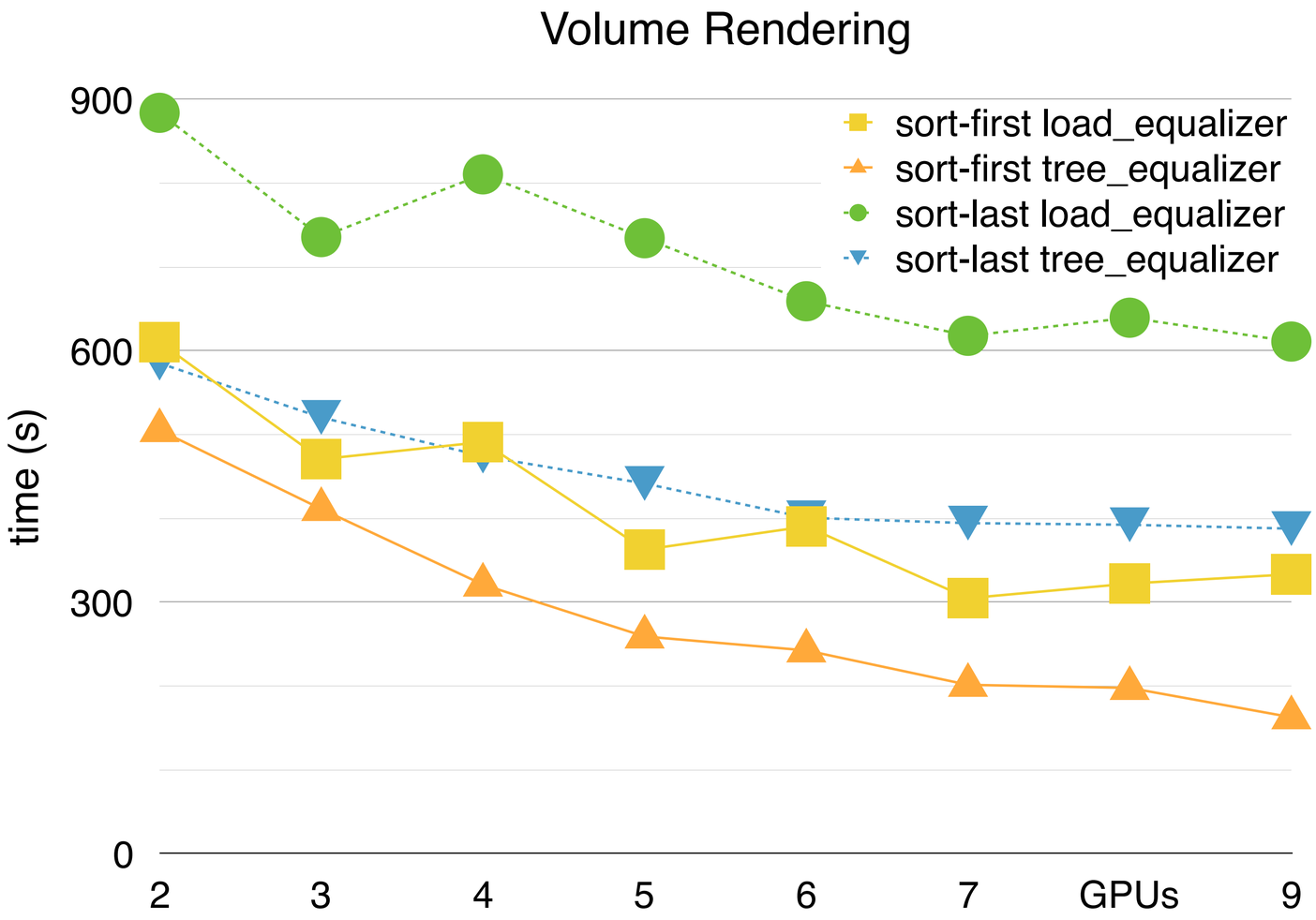
Single Display Load Balancing





Eilemann, S., Steiner, D., and Pajarola, R. (2018). Equalizer 2.0 — Convergence of a Parallel Rendering Framework. IEEE Transactions on Visualization and Computer Graphics

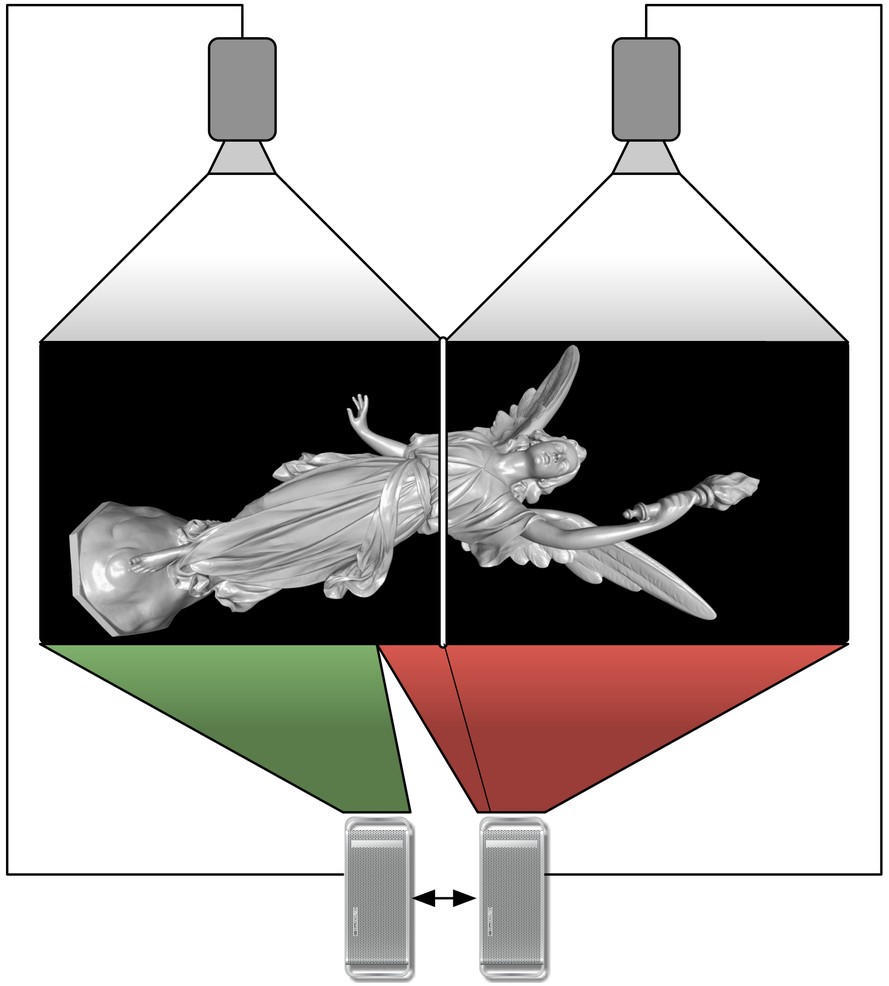
Single Display Load Balancing



Eilemann, S., Steiner, D., and Pajarola, R. (2018). Equalizer 2.0 — Convergence of a Parallel Rendering Framework. IEEE Transactions on Visualization and Computer Graphics

Cross-Display Load Balancing

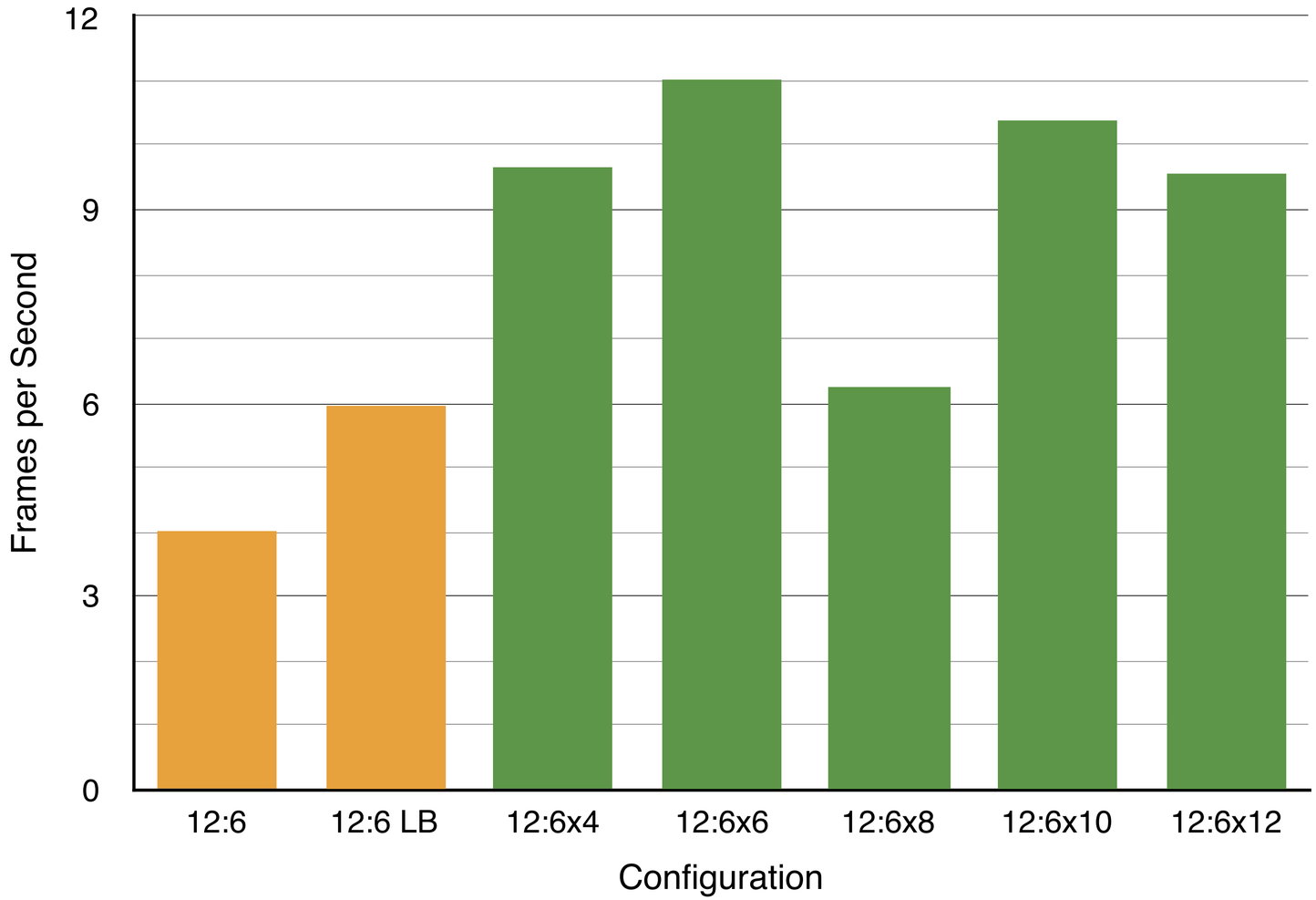
- Assign m GPUs to n Displays (m ≥ n)
- Per-display load balancer
  - Potential sources for each display
- Top-level view equalizer
  - Usage value per source
  - Source channels at most used twice
- Works for non-planar systems



Erol, F., Eilemann, S., and Pajarola, R. (2011). Cross-segment load balancing in parallel rendering. Eurographics Symposium on Parallel Graphics and Visualization

Cross-Display Load Balancing

- 3x2 tiled display wall
- 12 GPU cluster
- Polygonal rendering
- Camera animation



Erol, F., Eilemann, S., and Pajarola, R. (2011). Cross-segment load balancing in parallel rendering. Eurographics Symposium on Parallel Graphics and Visualization

Framework Architecture

How can we reduce end-to-end system latency?

How can we architect the parallel rendering framework to minimise synchronisation between threads?

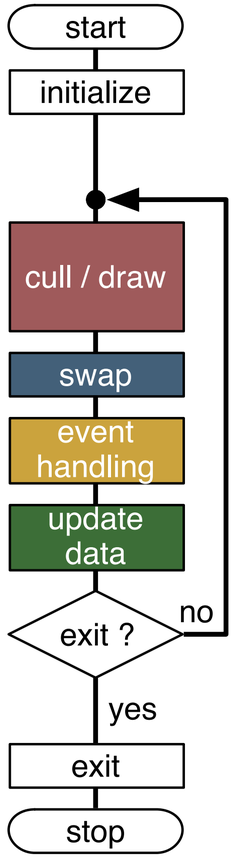
Which new algorithms will decrease the time needed to composite rendering results, in particular for sort-last rendering?

How can we maximise the impact of this research on large data scientists?

Minimally Invasive Architecture

Configuration vs Application

- Main Loop

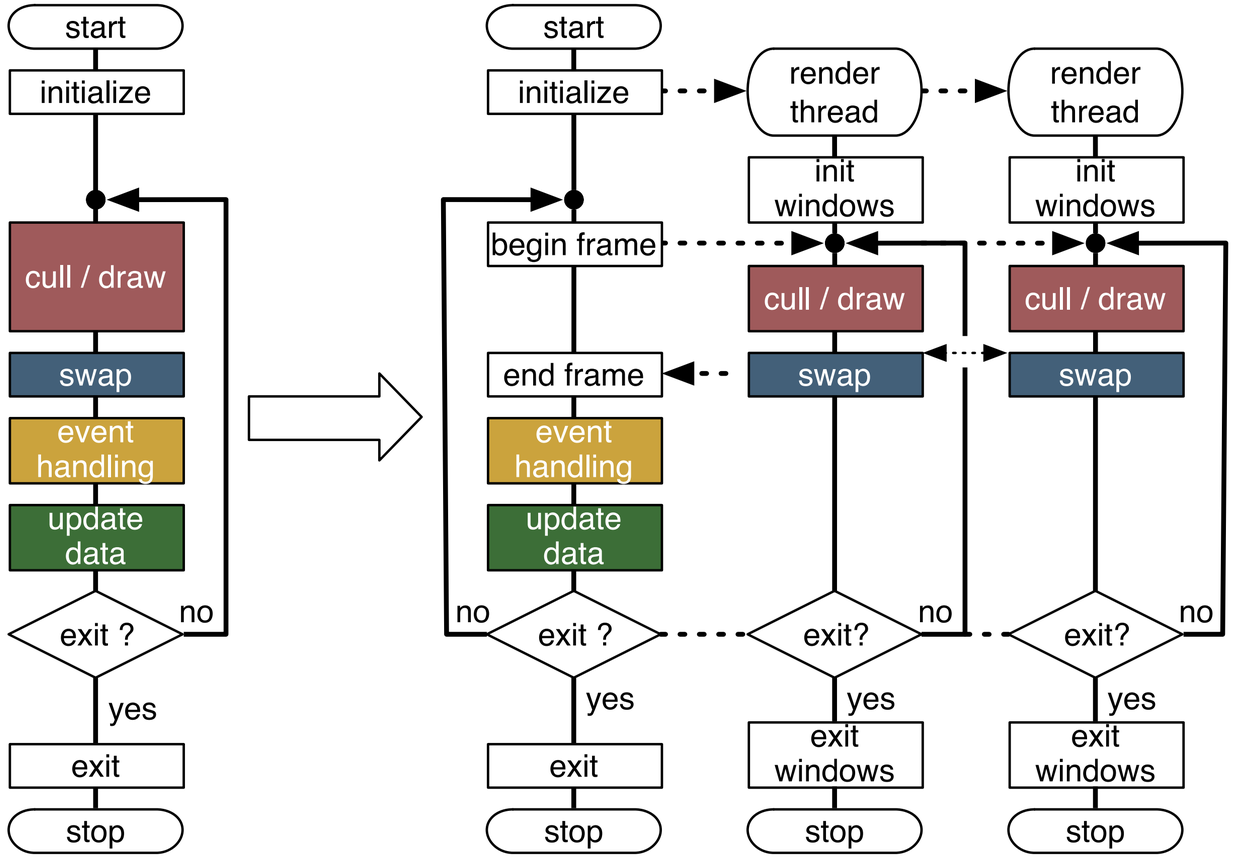


Eilemann, S., Makhinya, M., and Pajarola, R. (2009). Equalizer: A Scalable Parallel Rendering Framework. IEEE Transactions on Visualization and Computer Graphics

Minimally Invasive Architecture

Configuration vs Application

- Main Loop
- Render Client

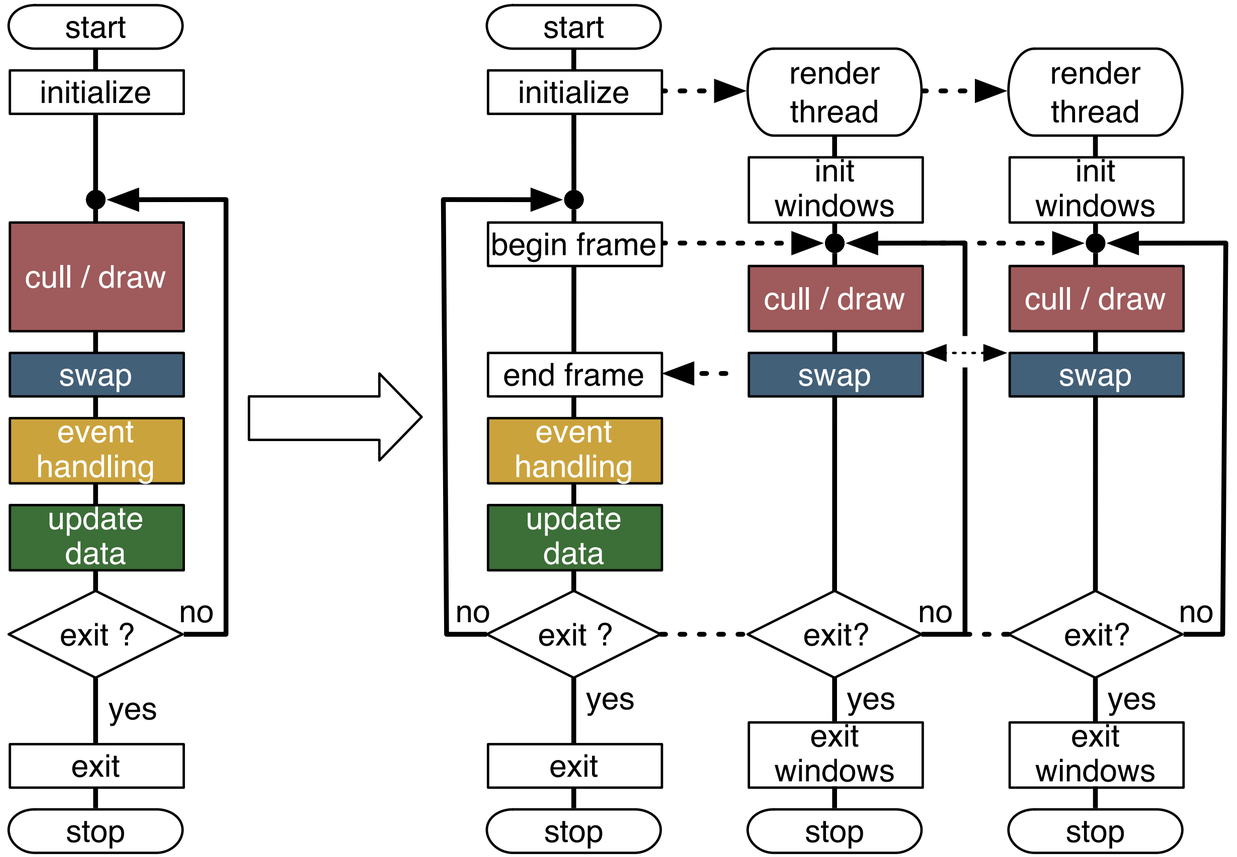


Eilemann, S., Makhinya, M., and Pajarola, R. (2009). Equalizer: A Scalable Parallel Rendering Framework. IEEE Transactions on Visualization and Computer Graphics

Minimally Invasive Architecture

Configuration vs Application

- Main Loop
- Render Client
- RenderContext
  - Viewport
  - Frustum
  - Head Transformation
  - Buffer
  - Range

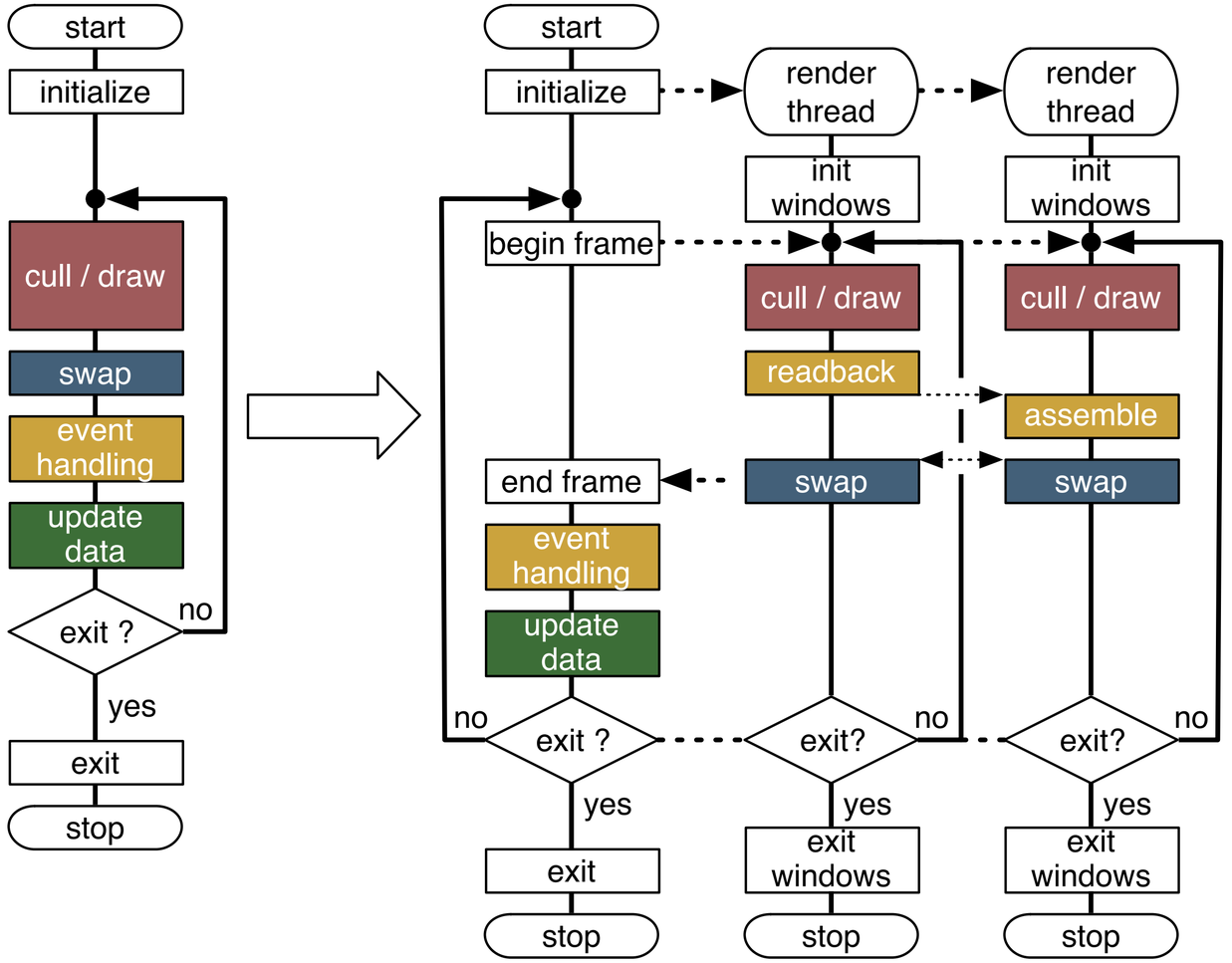


Eilemann, S., Makhinya, M., and Pajarola, R. (2009). Equalizer: A Scalable Parallel Rendering Framework. IEEE Transactions on Visualization and Computer Graphics

Minimally Invasive Architecture

Configuration vs Application

- Main Loop
- Render Client
- RenderContext
  - Viewport
  - Frustum
  - Head Transformation
  - Buffer
  - Range



Eilemann, S., Makhinya, M., and Pajarola, R. (2009). Equalizer: A Scalable Parallel Rendering Framework. IEEE Transactions on Visualization and Computer Graphics

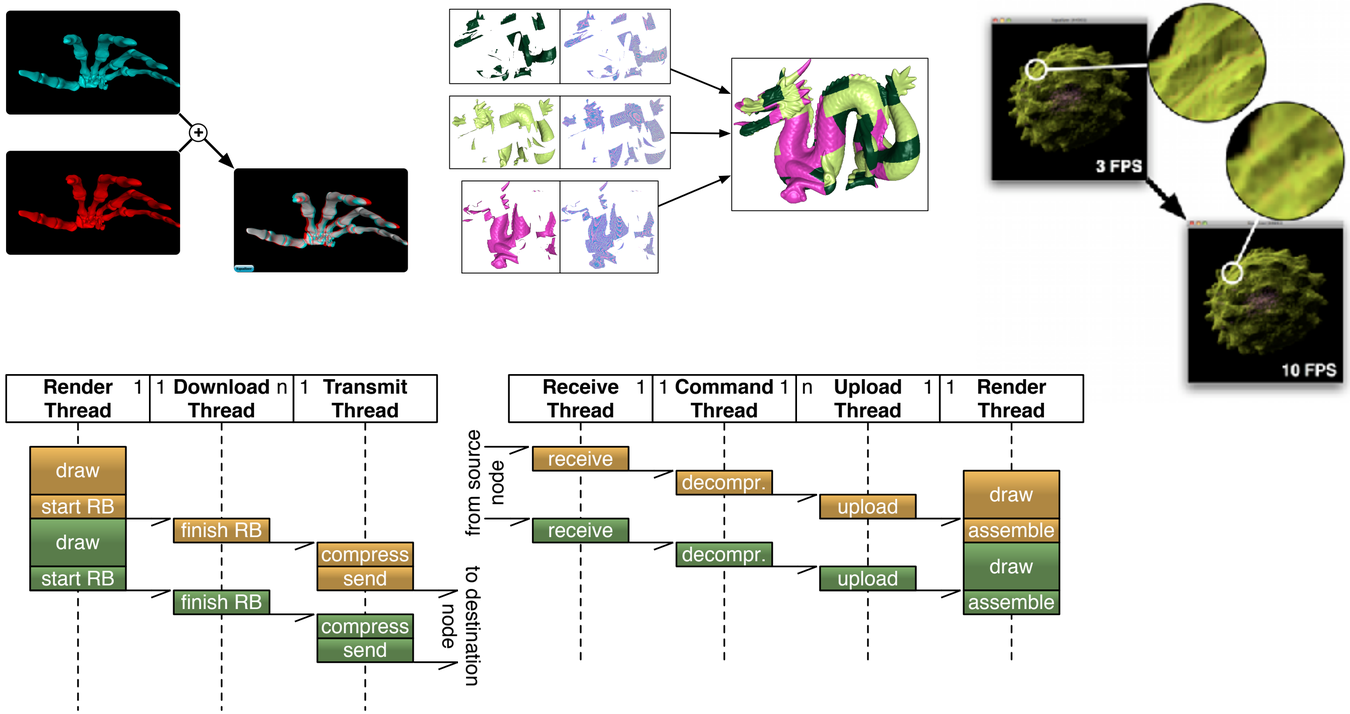
Features

Eight Decomposition Modes

Six Equalizers

Whole System Optimization

Full Application Support



Eilemann, S., Steiner, D., and Pajarola, R. (2018). Equalizer 2.0 — Convergence of a Parallel Rendering Framework. IEEE Transactions on Visualization and Computer Graphics

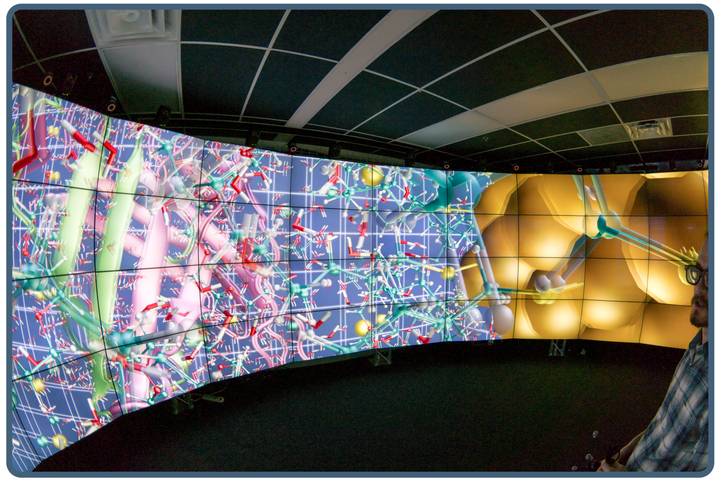
Eilemann, S., Bilgili, A., Abdellah, M., Hernando, J., Makhinya, M., Pajarola, R., and Schürmann, F. (2012). Parallel Rendering on Hybrid Multi-GPU Clusters. Eurographics Symposium on Parallel Graphics and Visualization

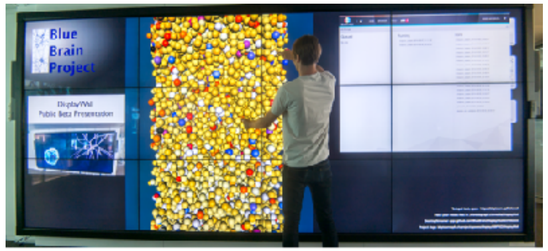
Eilemann, S. and Pajarola, R. (2007). Direct send compositing for parallel sort-last rendering. Eurographics Symposium on Parallel Graphics and Visualization.

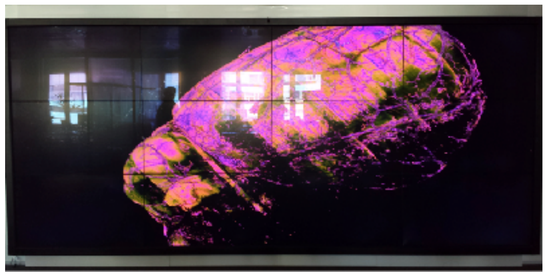
Summary

- Minimally invasive, flexible framework architecture
- Full-fledged application support
- Whole system optimization
- Reliable multicast protocol
- Compression techniques for interactive use
- Novel load balancing algorithms for single and multi display systems

Applications







Hernando, J. B., Biddiscombe, J., Bohara, B., Eilemann, S., and Schürmann, F. (2013). Practical Parallel Rendering of Detailed Neuron Simulations. Eurographics Symposium on Parallel Graphics and Visualization

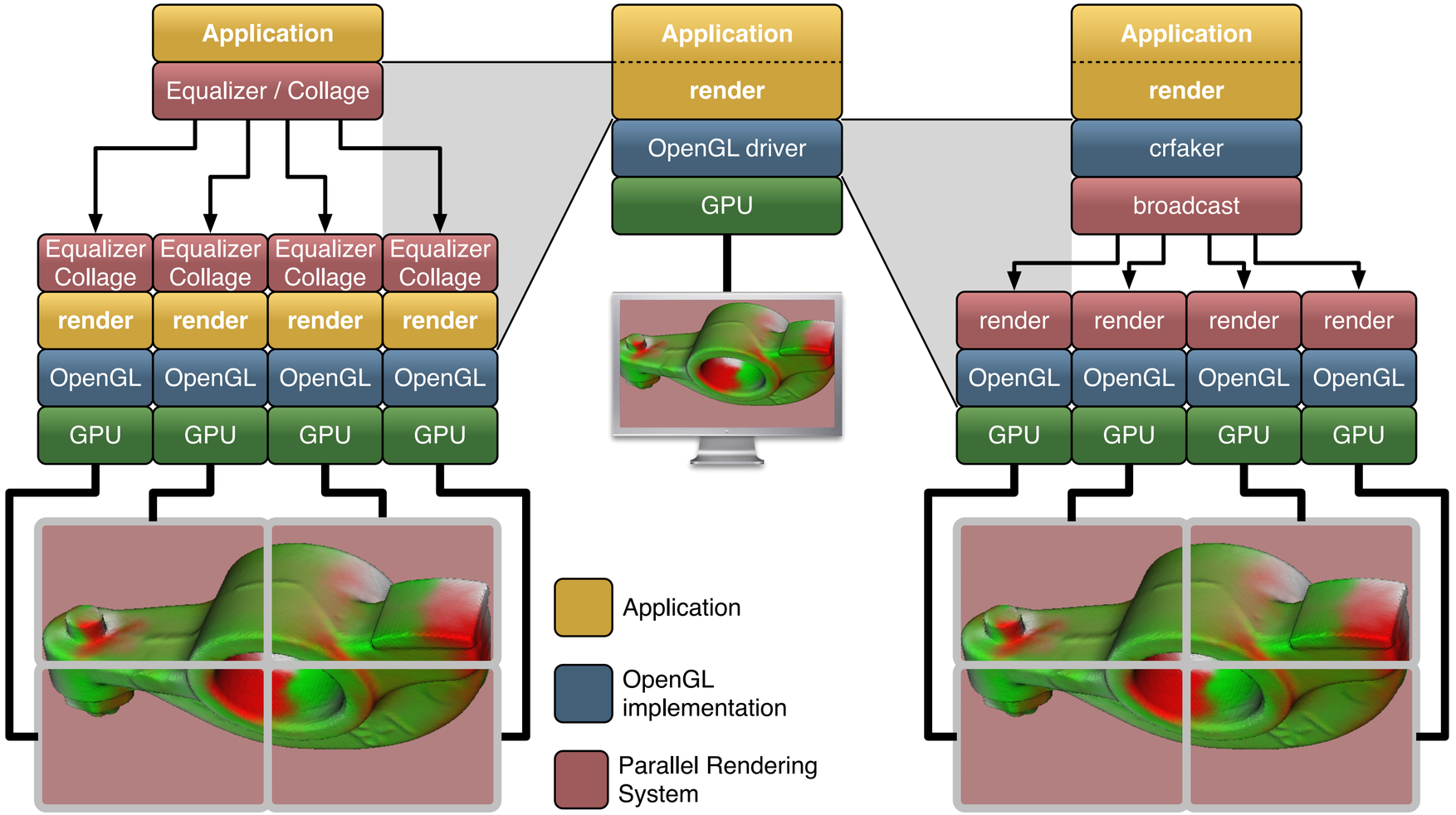
Thank You

Questions?

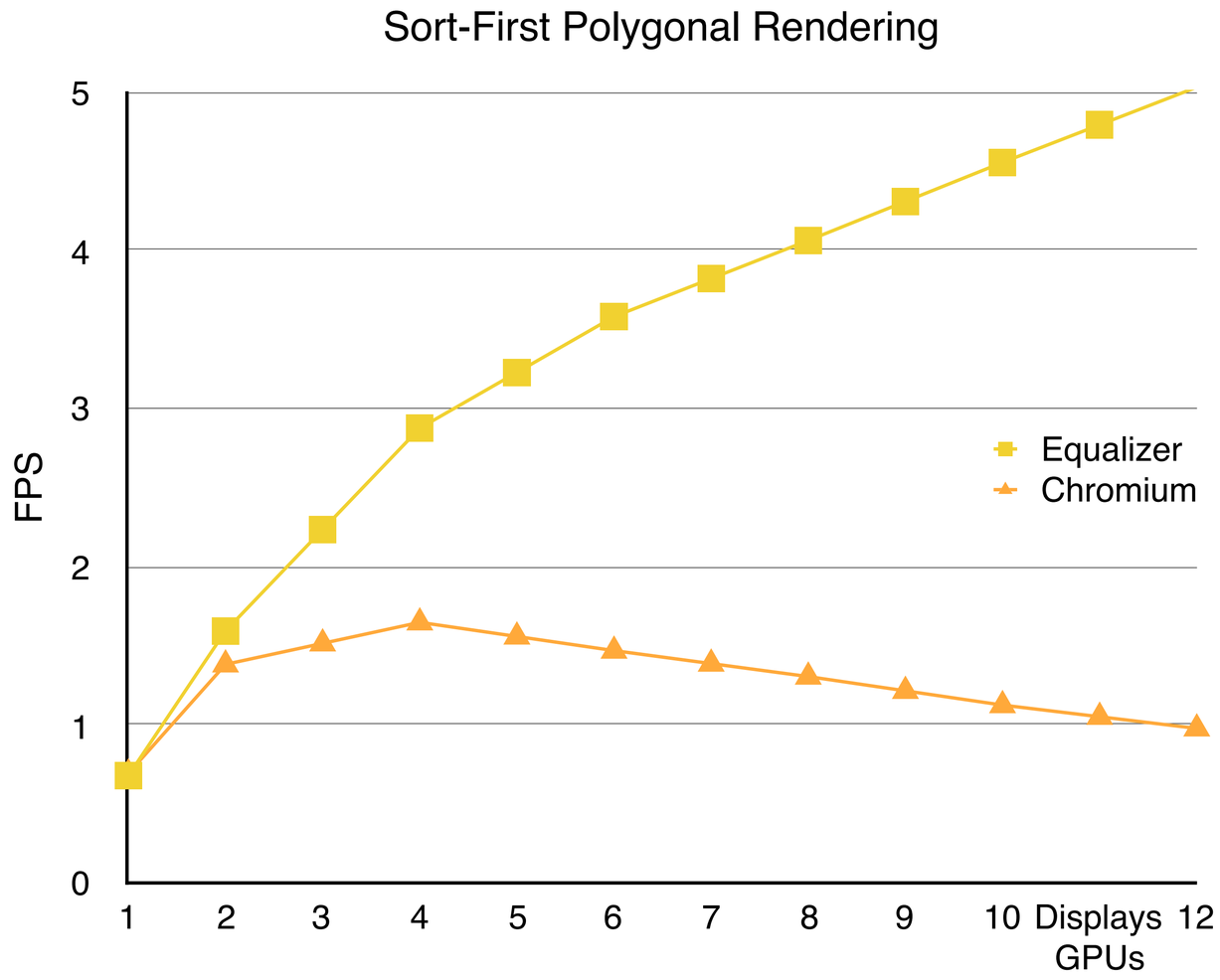
Future Work

- Scene graph and rendering toolkit integration
- Auto-selection of scalable rendering algorithms and resources
- Load balancing sort-last compositing
- Predictive load balancing
- Sort-last scalable raytracing

Chromium vs Equalizer



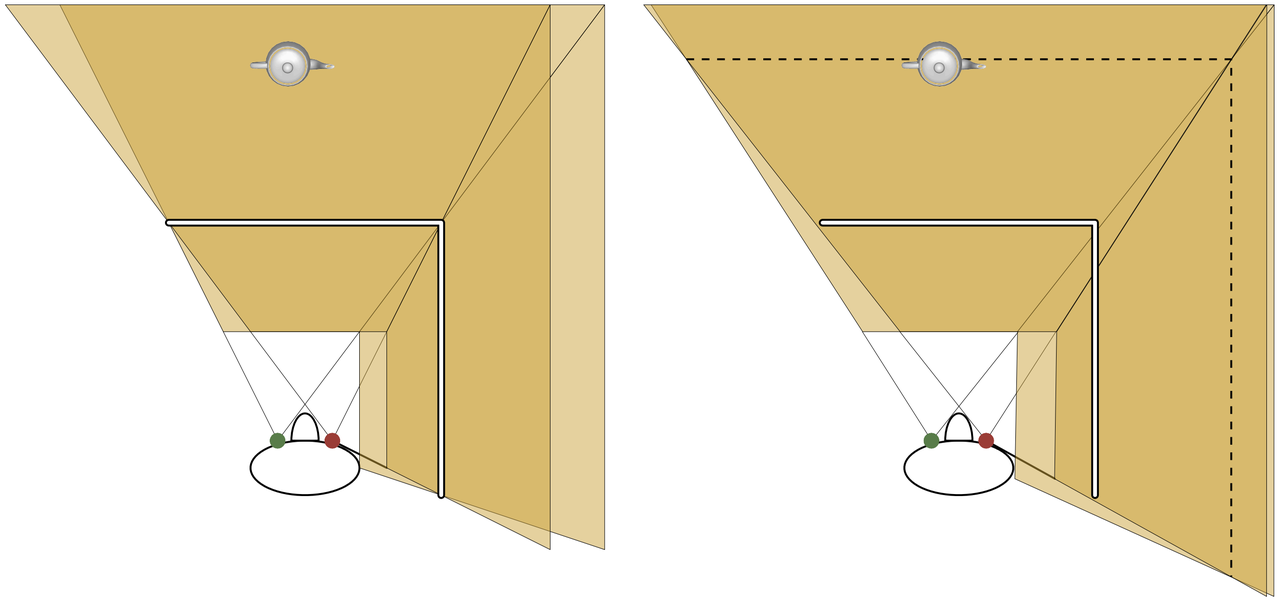
Chromium vs Equalizer



Display Abstraction

Virtual Reality

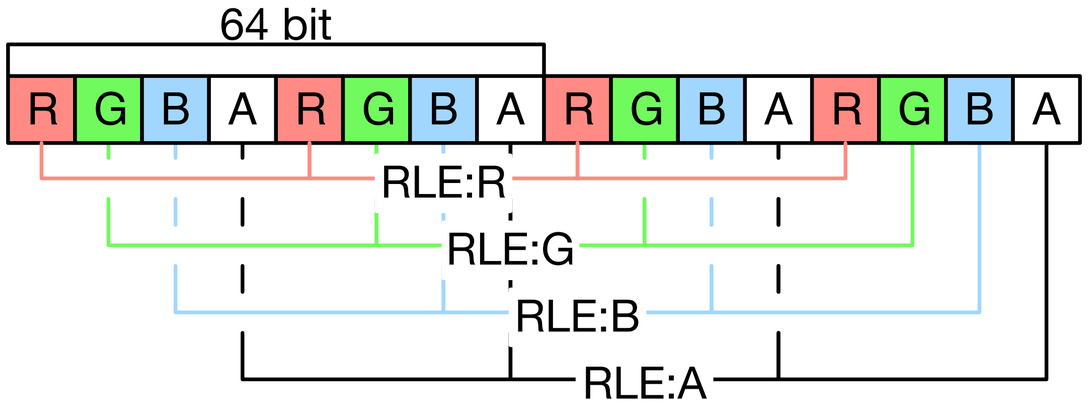
- Head Tracking
  - Cave
  - HMD
- Dynamic Focus Distance
- Asymmetric Eye Position
- Model Unit
- Runtime Stereo Switch

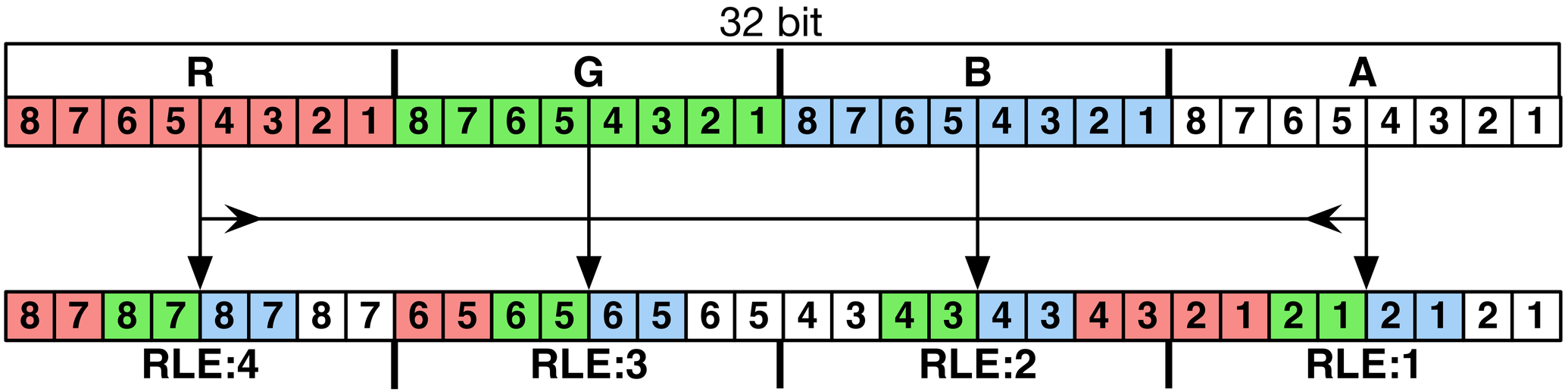


Collage

- Peer-to-peer networking library
  - Connection - DataI/OStream - Node/LocalNode - Objects
- Versioned, distributed objects
  - Compression
  - Chunking, Buffering, Caching
  - Multicast
- Reliable Stream Protocol
  - TCP semantics over UDP Multicast

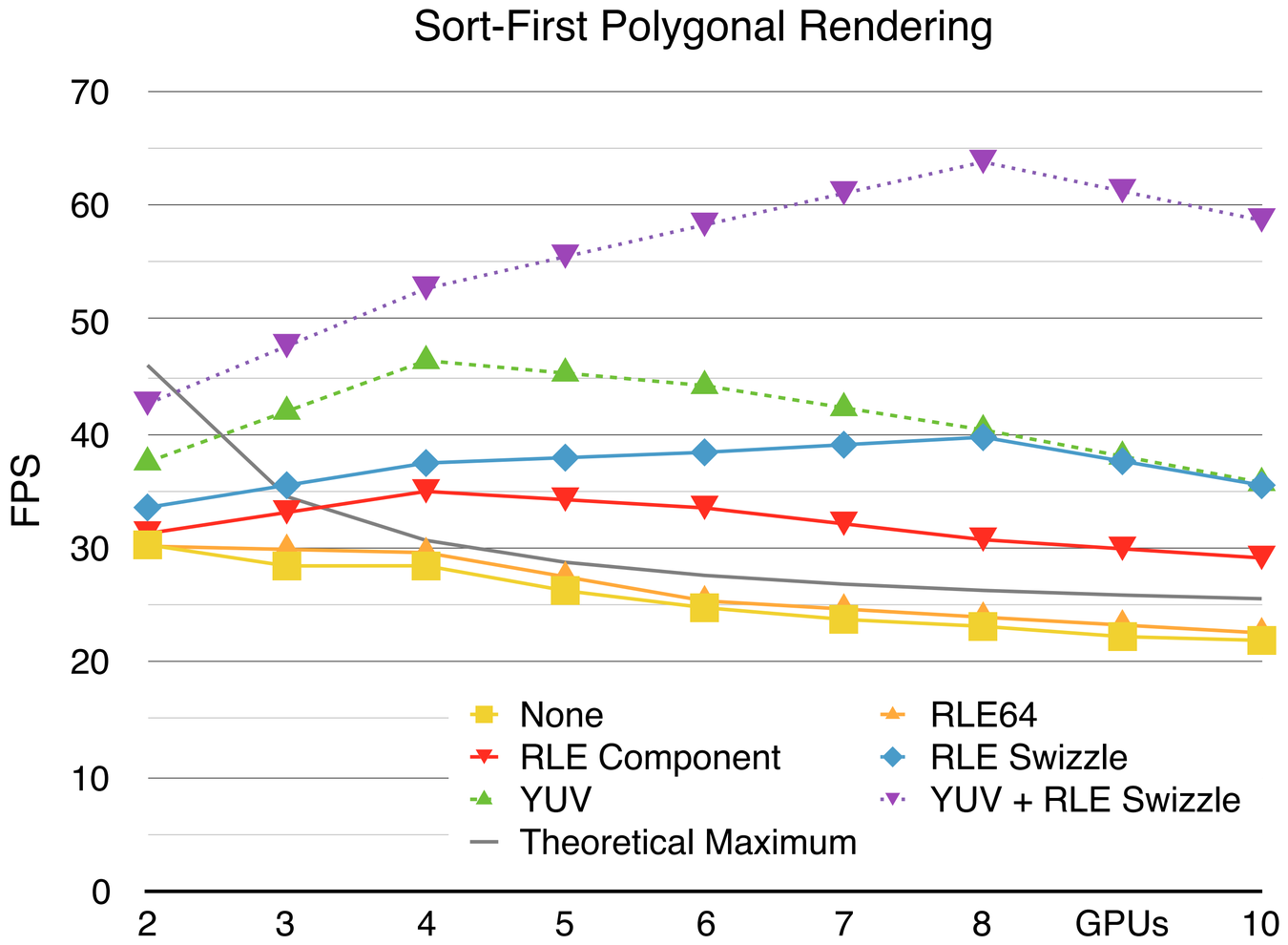
RLE Compression





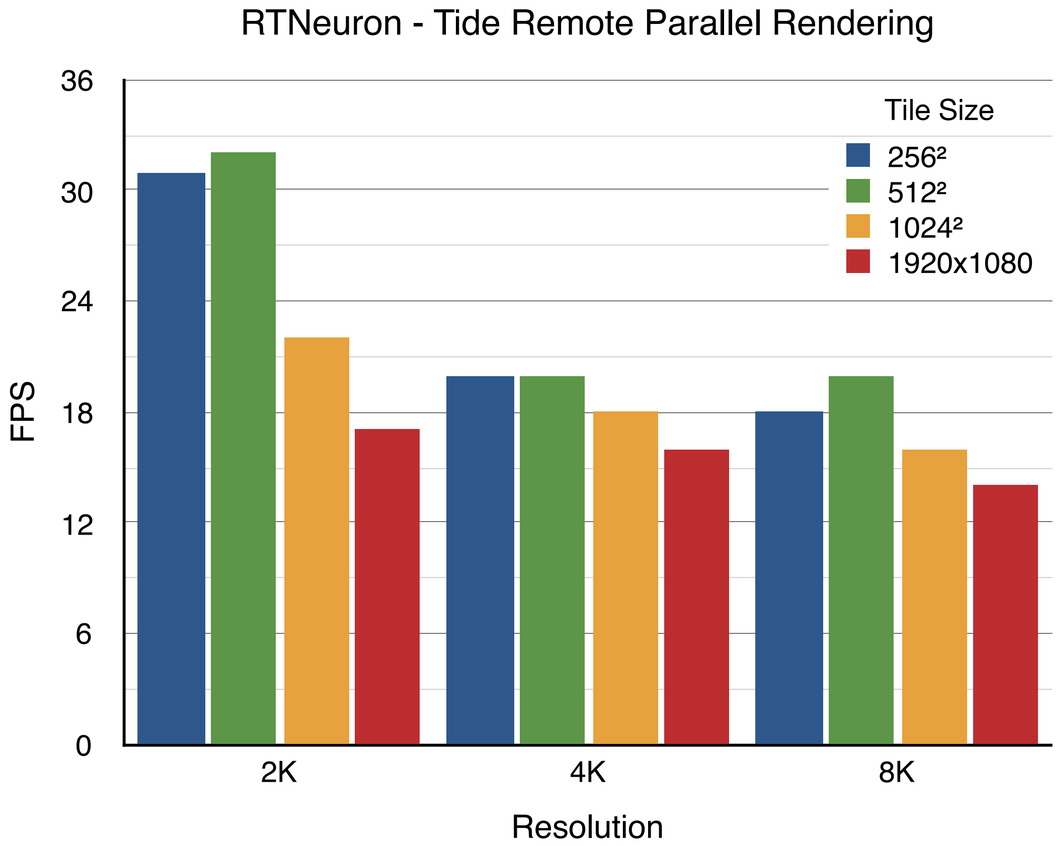
Makhinya, M., Eilemann, S., and Pajarola, R. (2010). Fast Compositing for Cluster-Parallel Rendering. In Proceedings of the 10th Eurographics Conference on Parallel Graphics and Visualization

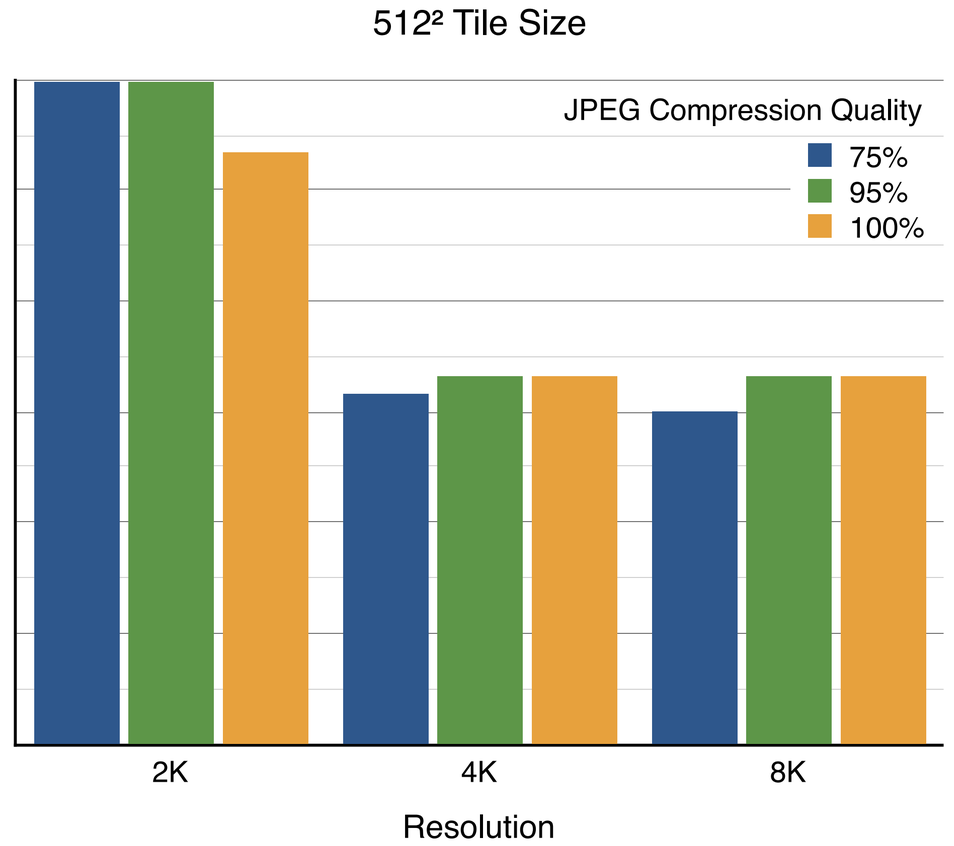
RLE Compression



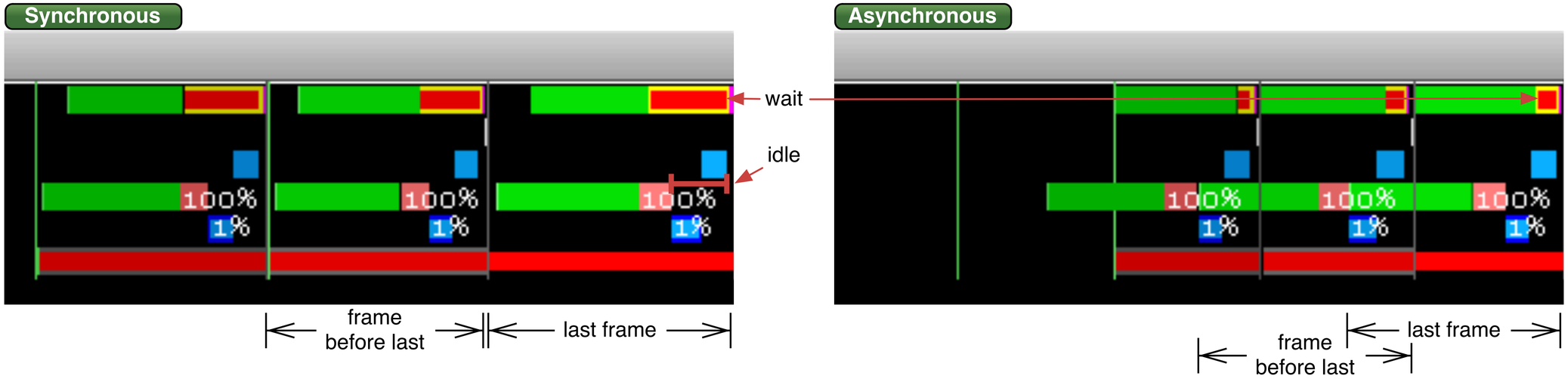
Makhinya, M., Eilemann, S., and Pajarola, R. (2010). Fast Compositing for Cluster-Parallel Rendering. In Proceedings of the 10th Eurographics Conference on Parallel Graphics and Visualization

Tide Streaming

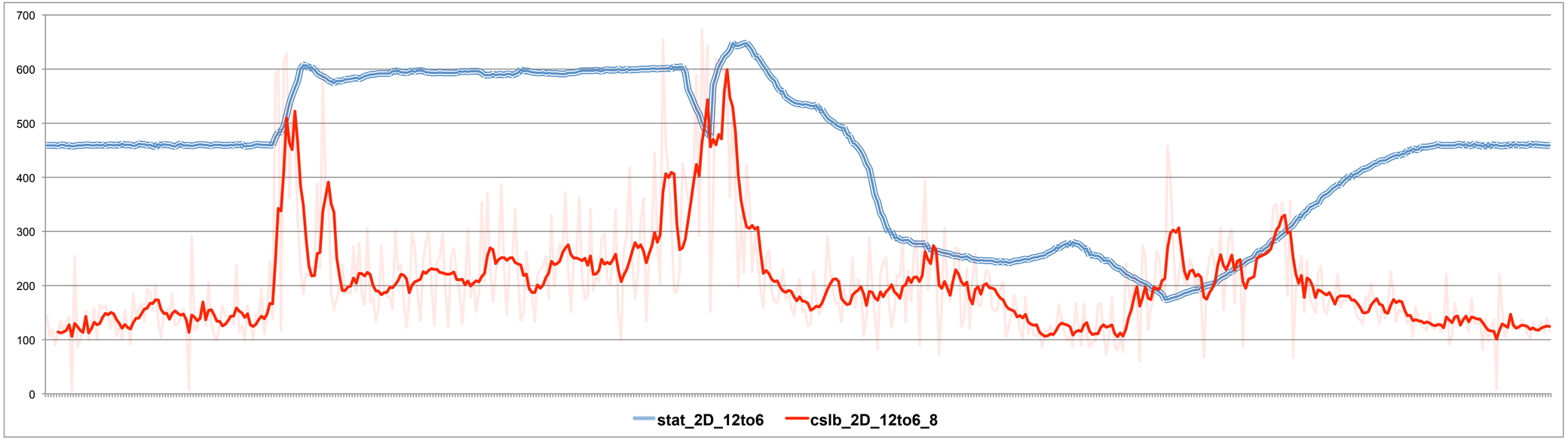




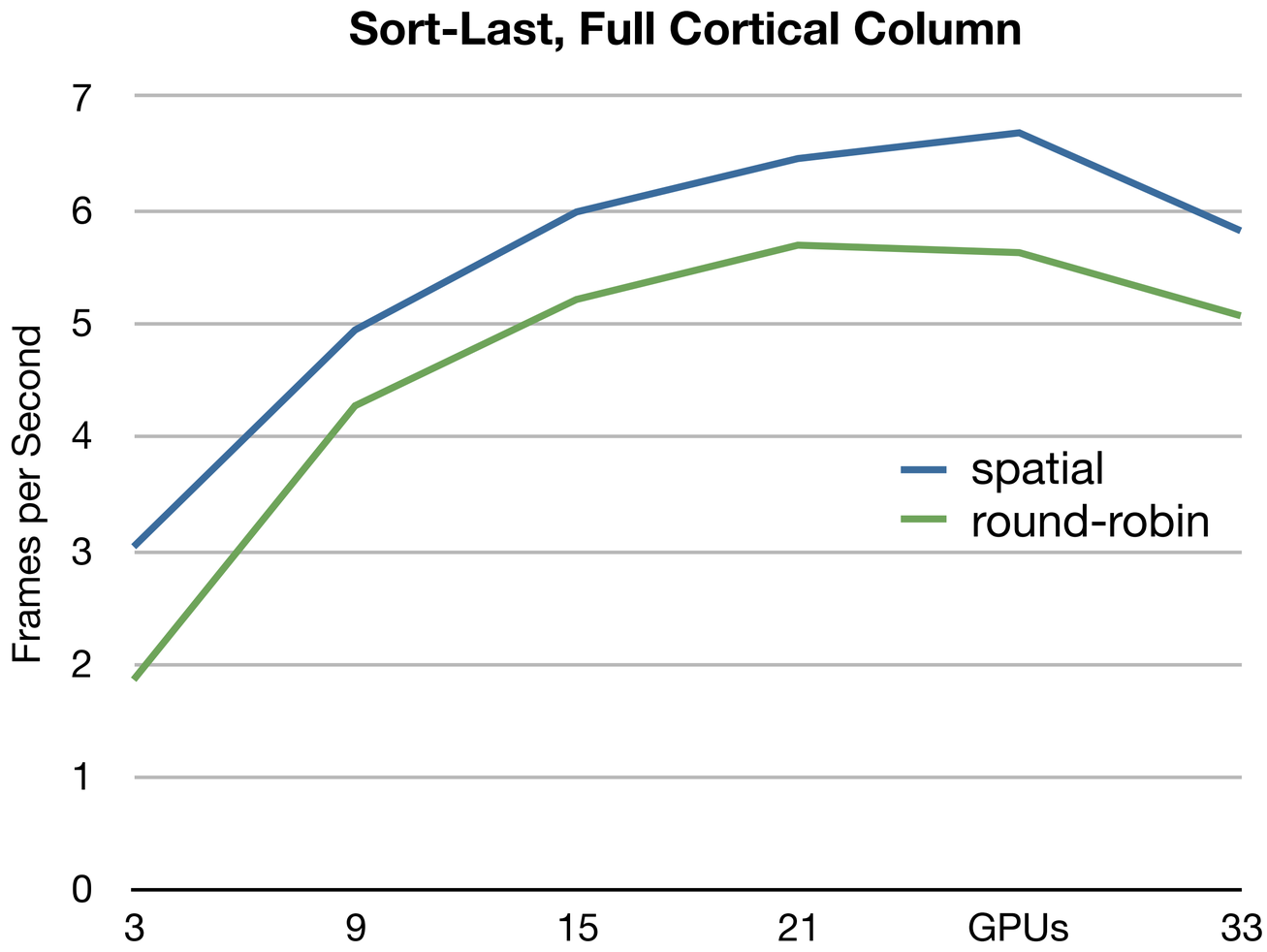
Asynchronous Execution



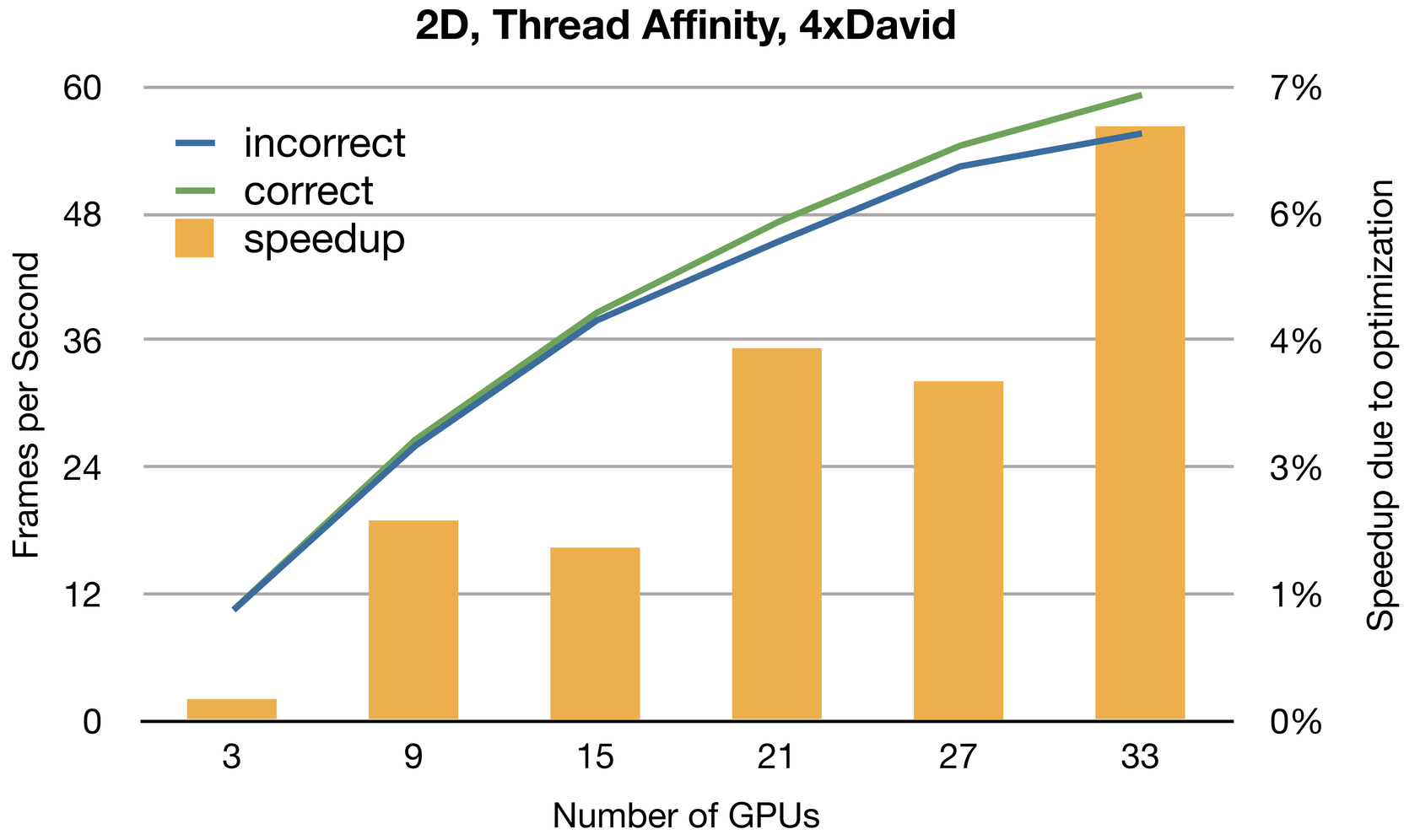
Cross-Display Load Balancing



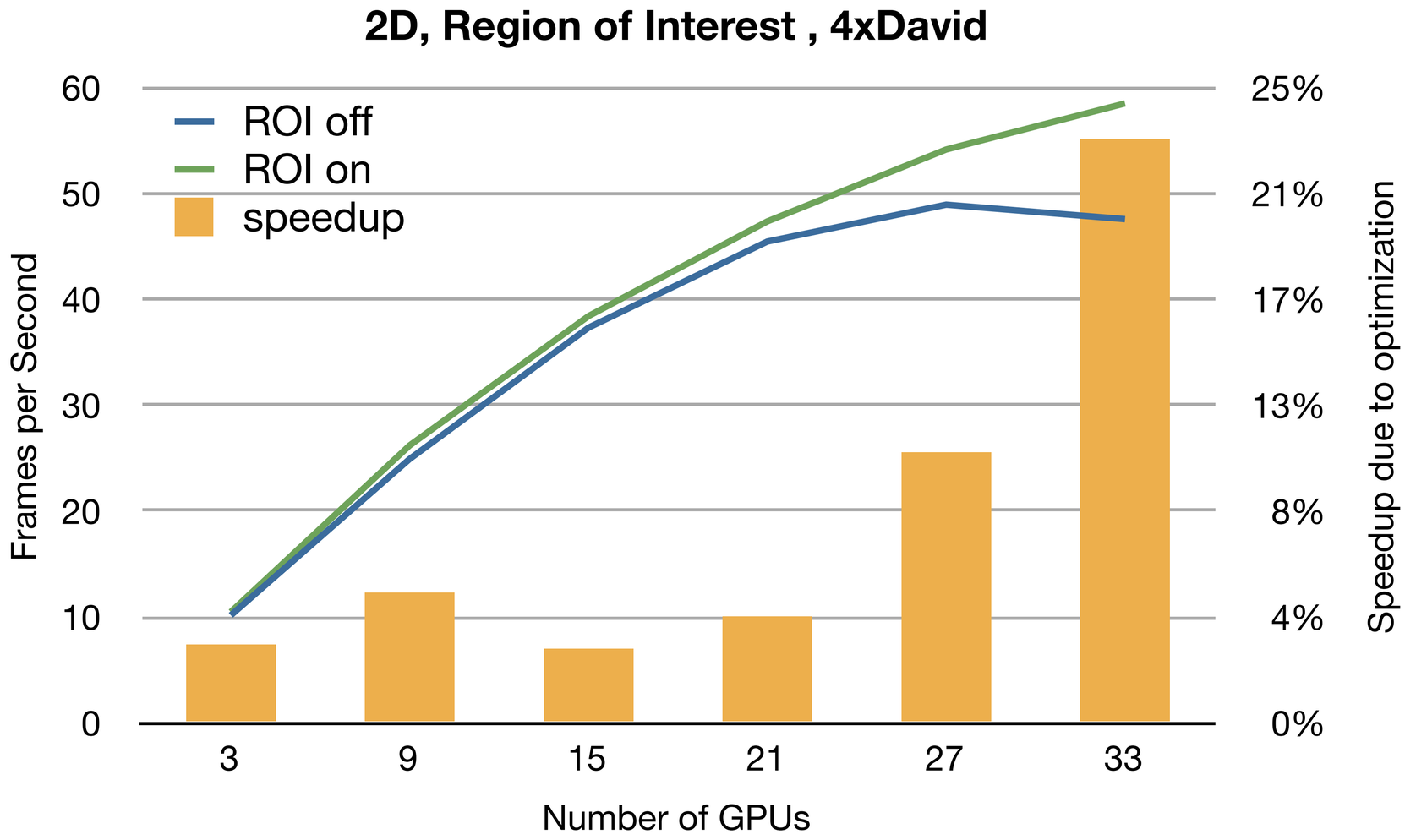
Spatial vs Random Sort-Last



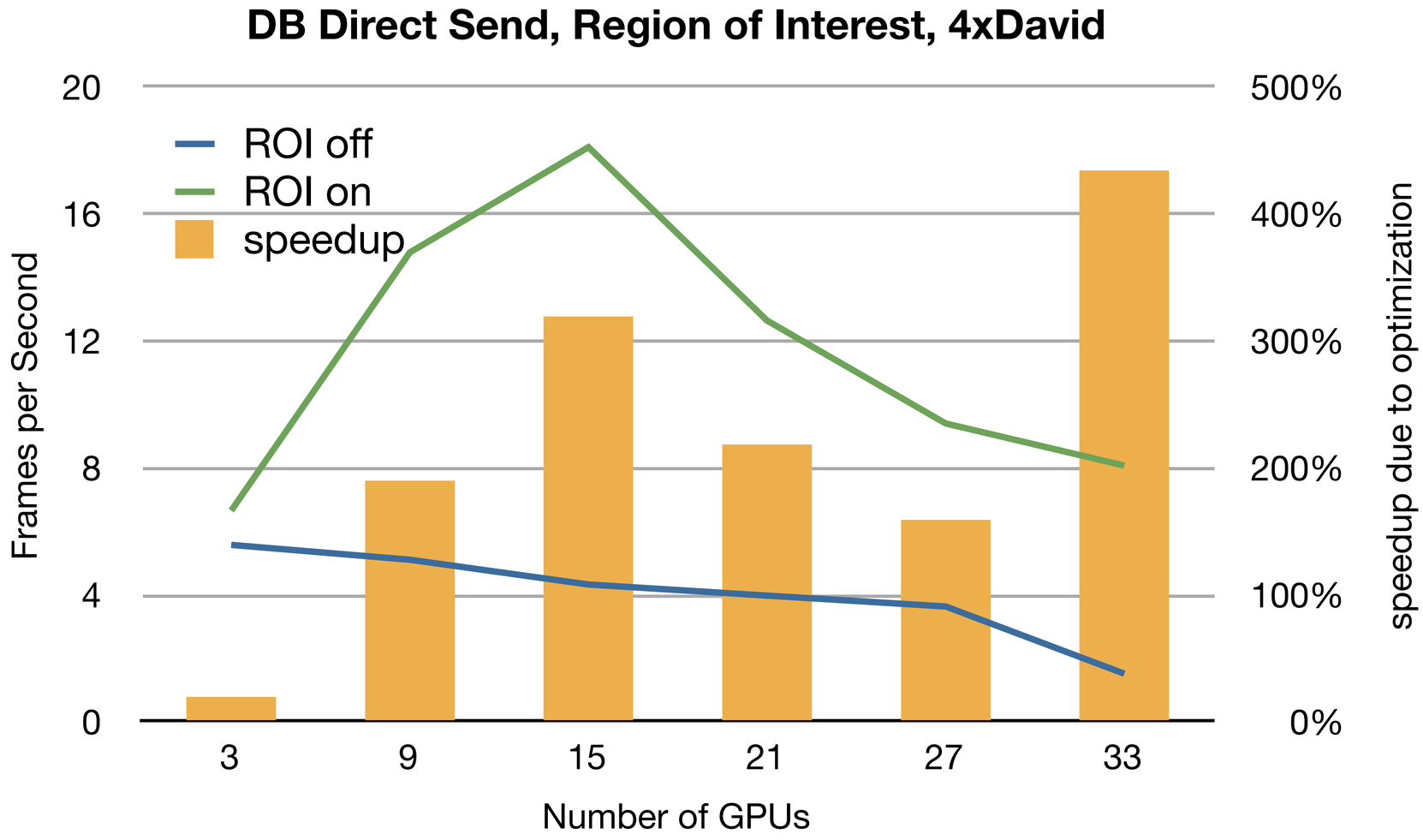
NUMA Thread Affinity



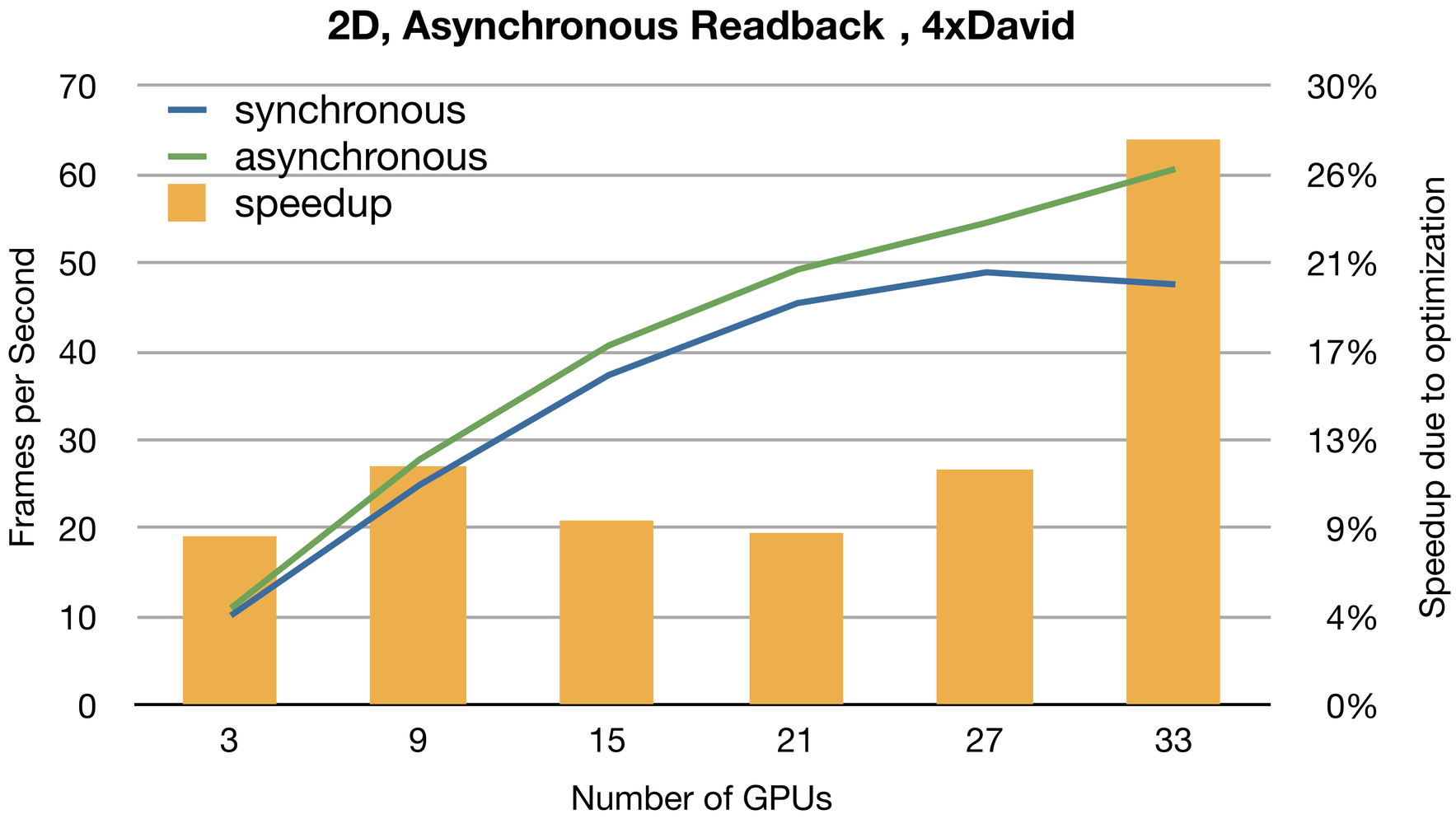
Sort-First Region of Interest



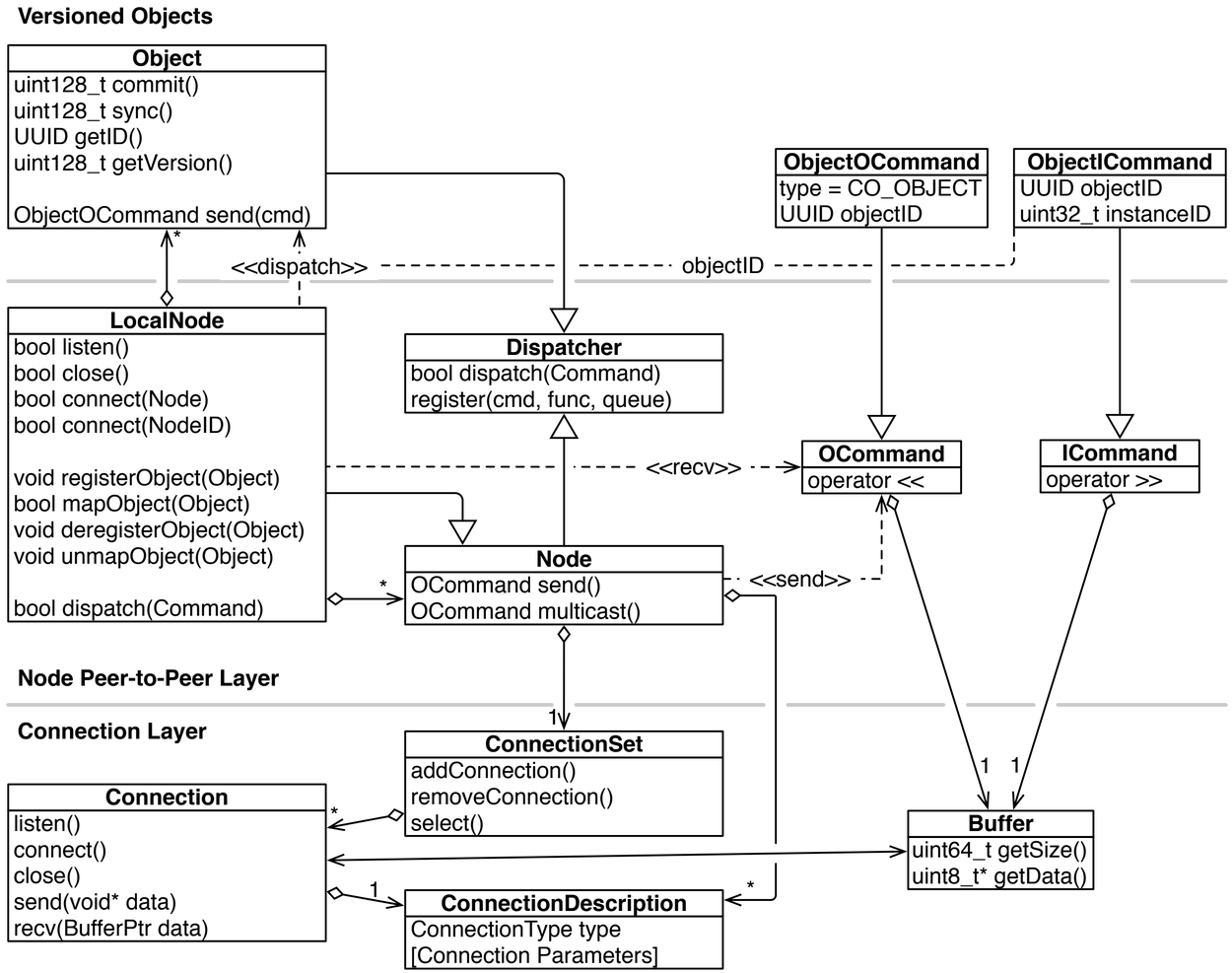
Sort-Last Region of Interest



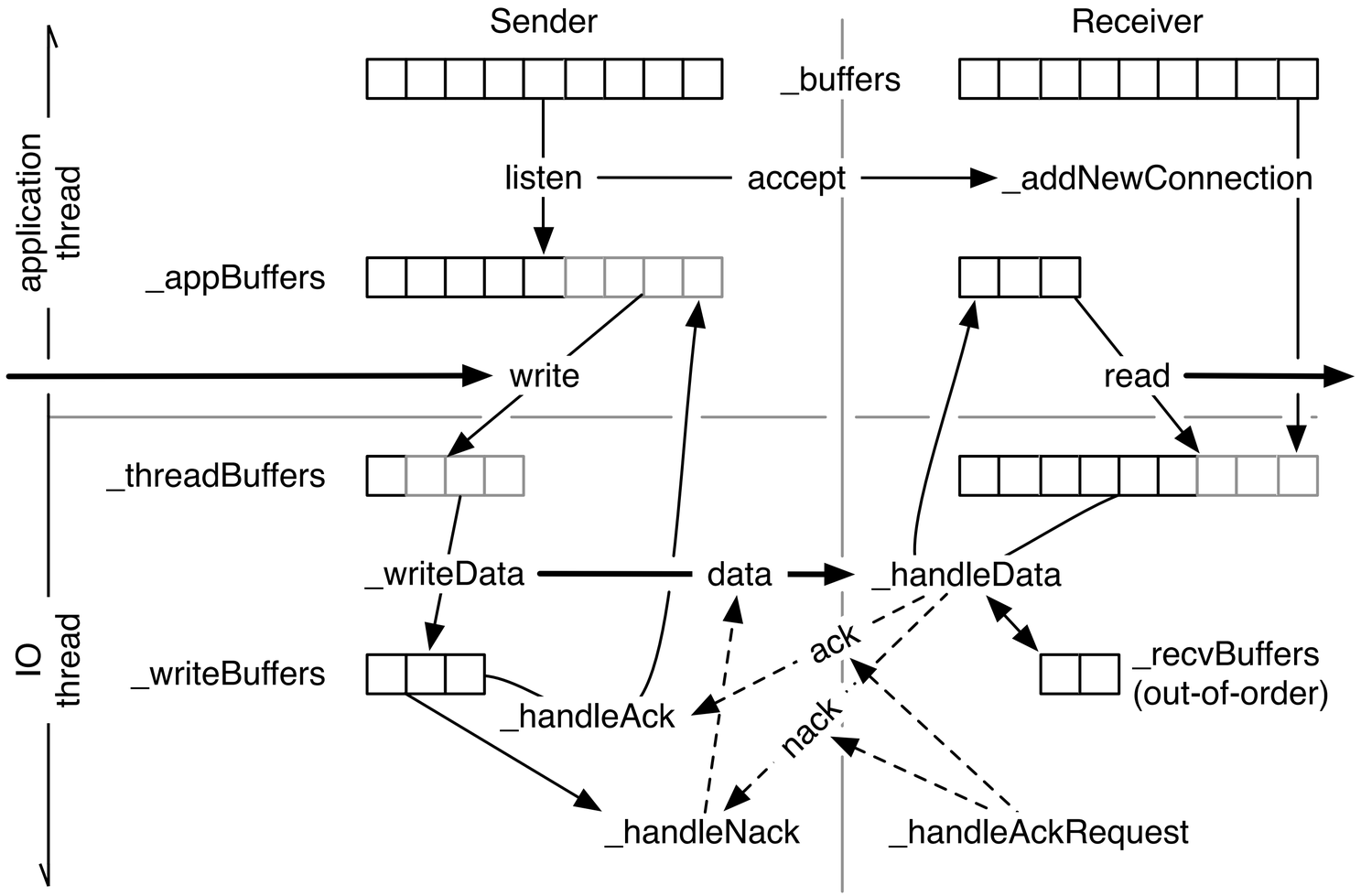
Asynchronous Readback



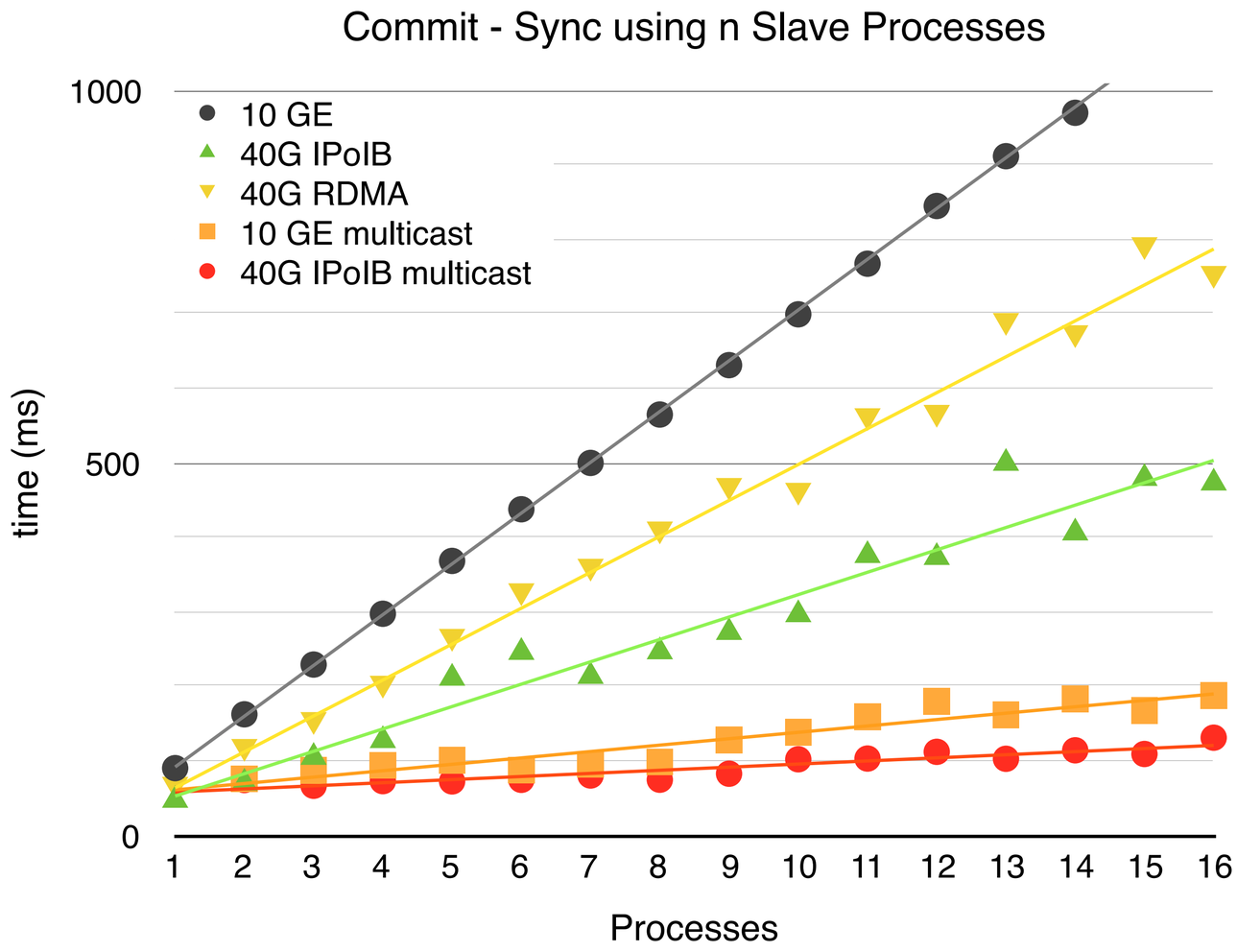
Collage Architecture



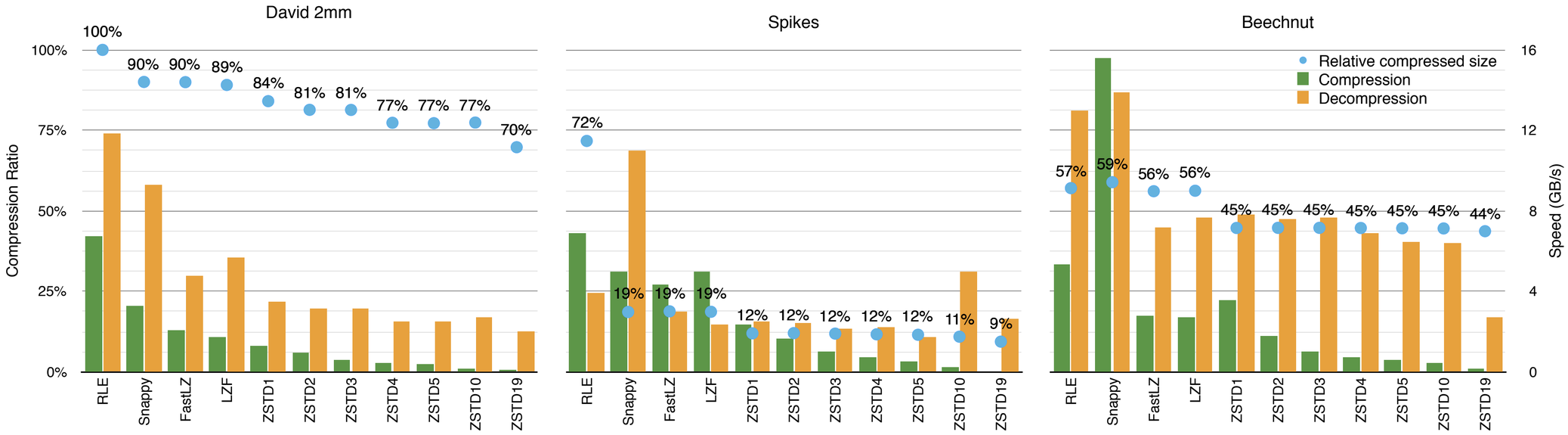
Reliable Stream Protocol

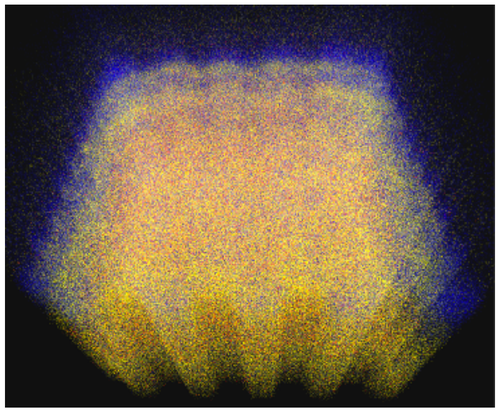


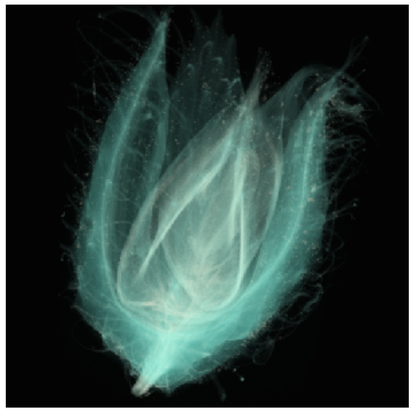
Reliable Stream Protocol



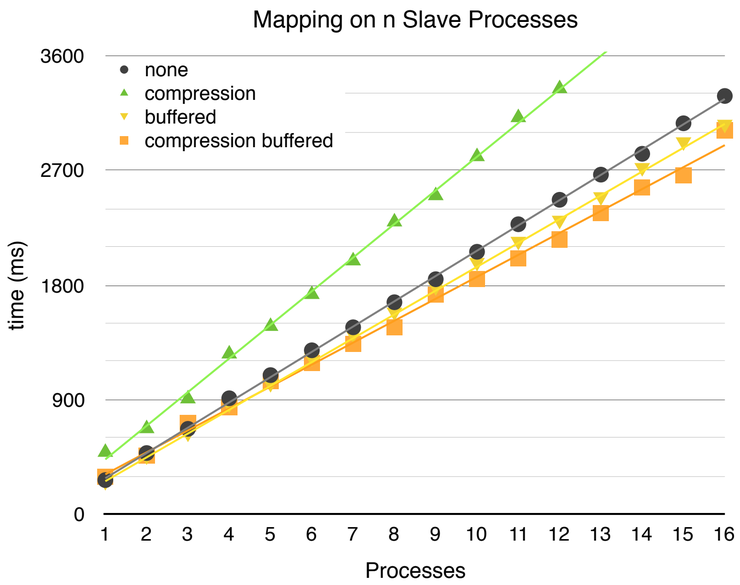
Compression

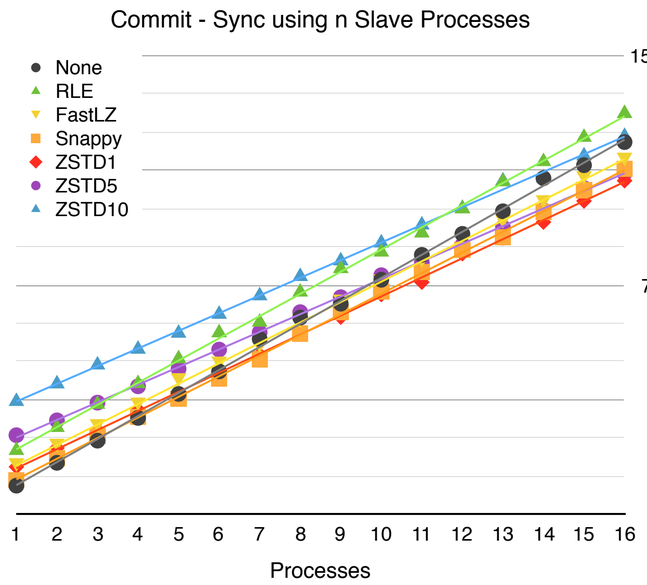


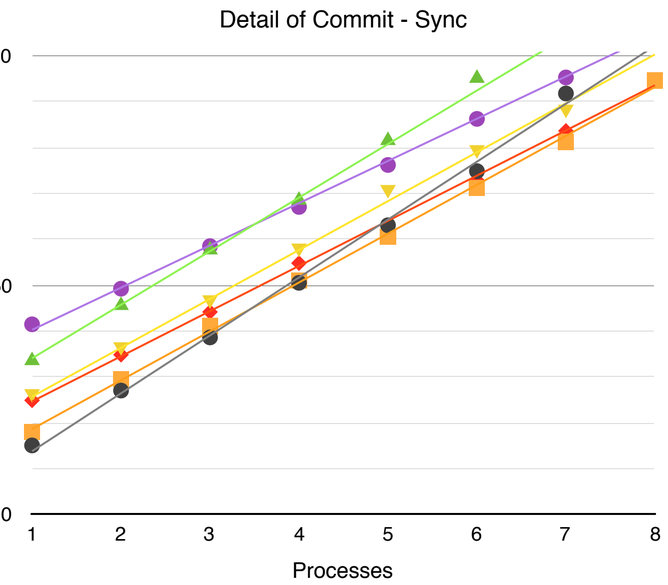




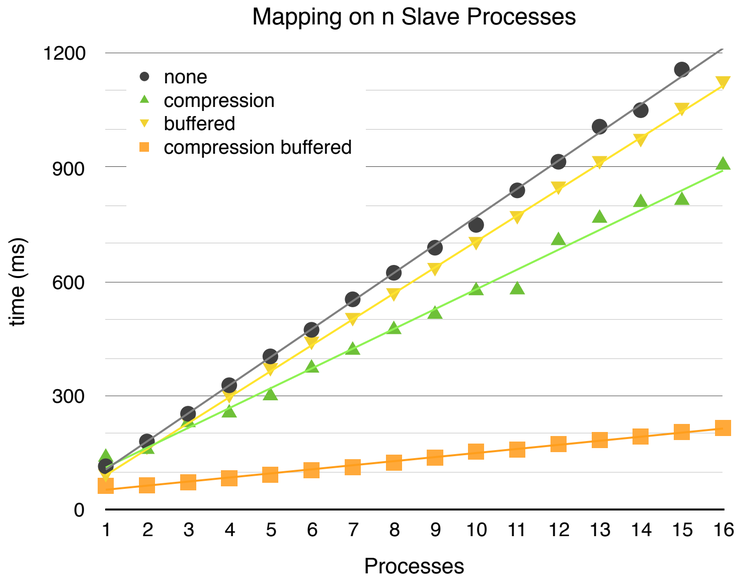
Distribution

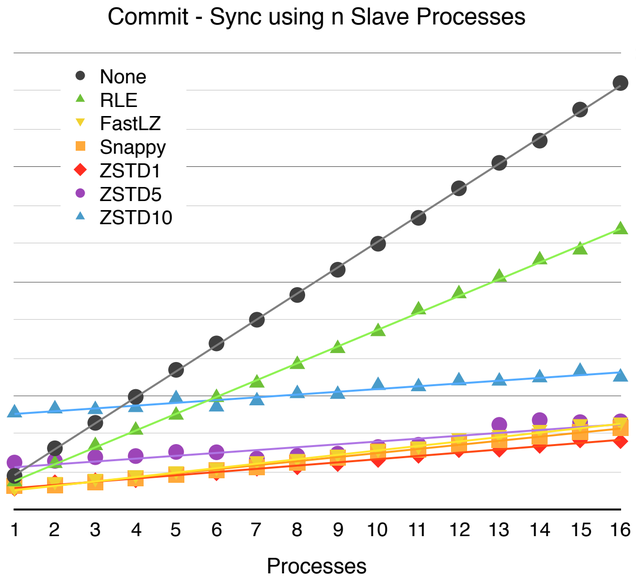


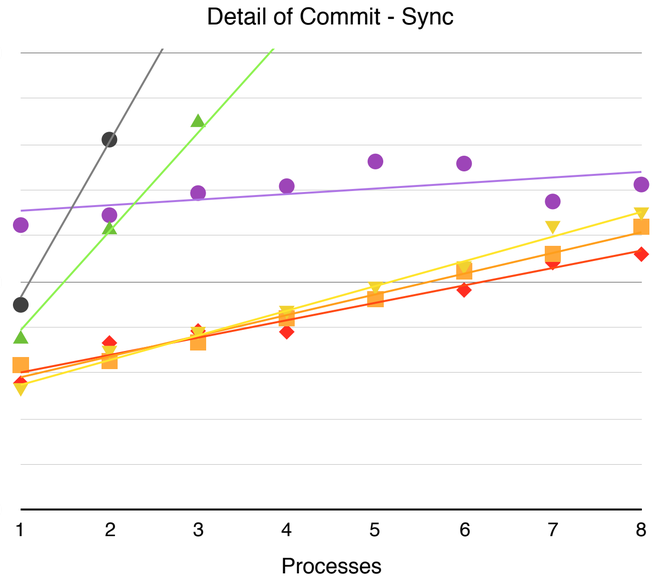


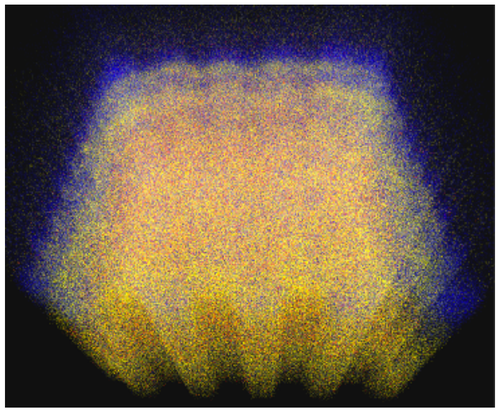


Distribution

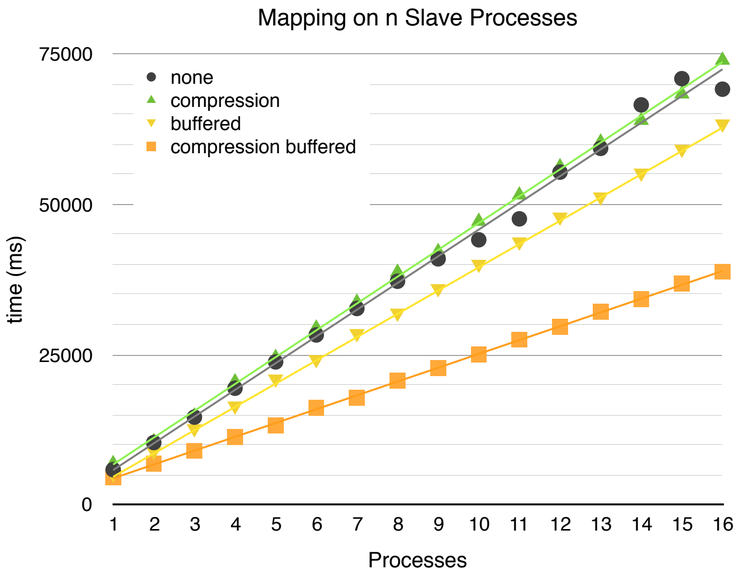


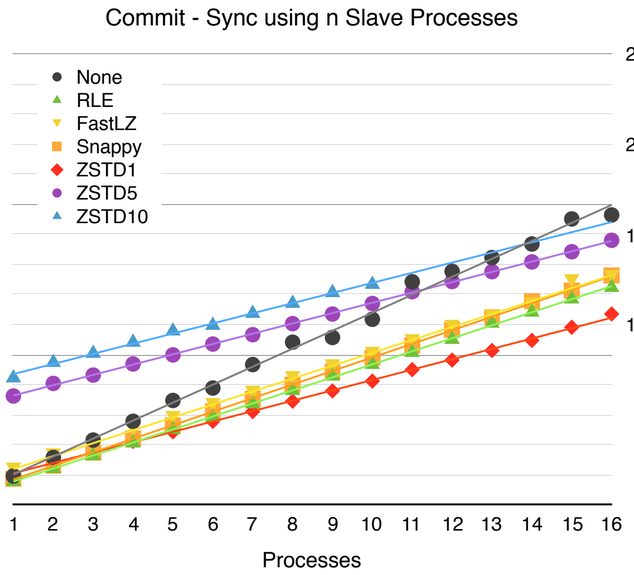


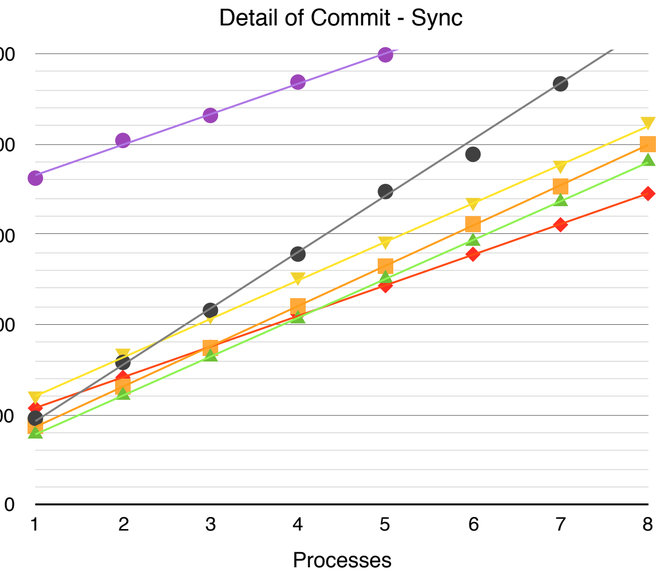


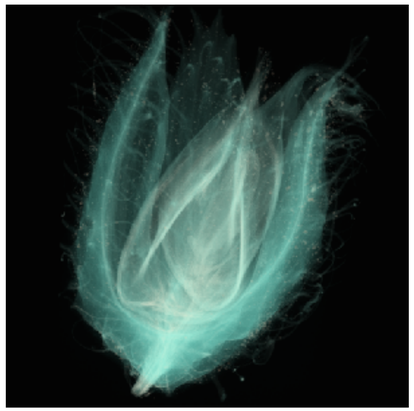


Distribution









Distribution