On Network-Aware Visualization

eaviv

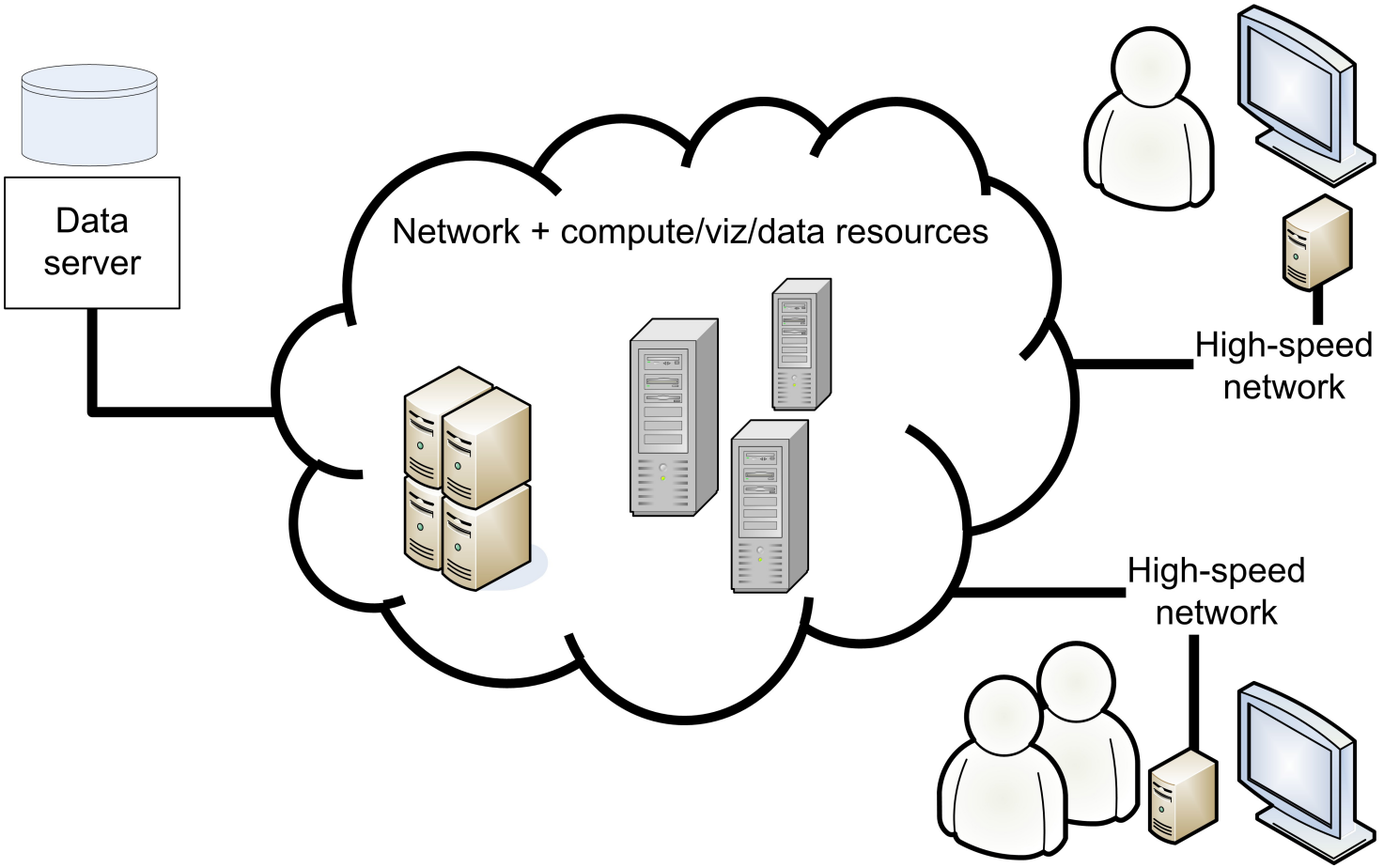
Andrei Hutanu, Jinghua Ge,
Cornelius Toole, Jr., Gabrielle Allen

Center for Computation & Technology
AT LOUISIANA STATE UNIVERSITY

Introduction

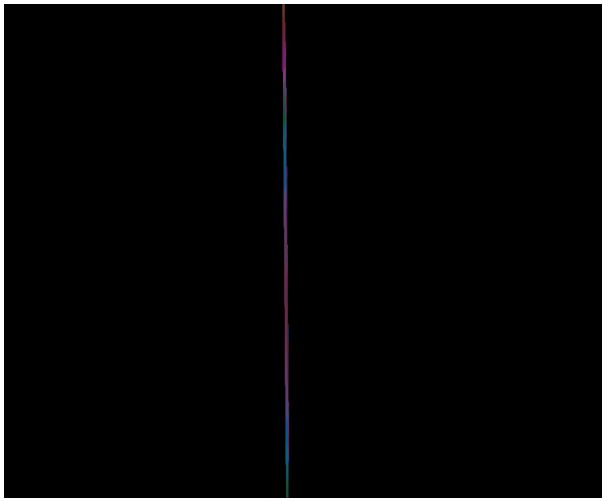
- Data size increase
  - Instruments
  - Simulations
- Emerging high-speed networks
  - Use to improve scalability of applications
- CPU performance limited: use parallel and distributed resources, use GPU, storage resources
- Create virtual meta-computer (OptIPuter)

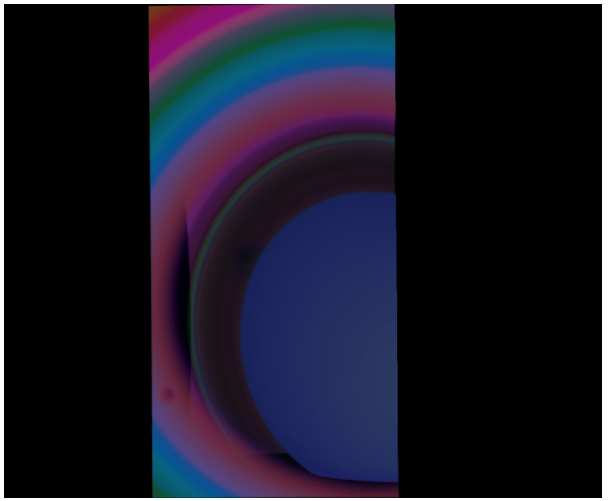
Scenario


Data server
Network + compute/viz/data resources
High-speed network
High-speed network

Improve Application

- Additional motivation
  - Increase I/O rate (see movies)
  - Increase data size (top image: laptop only visualization, bottom image: distributed visualization on laptop using remote cluster)
  - Collaborative visualization capabilities



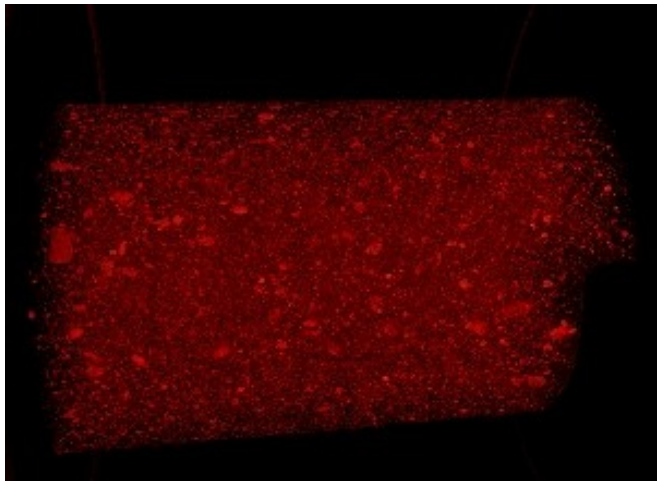


Goal

- Visualization system requirements:
  - Interactive (5fps or more)
  - High data rate for I/O
  - Responsive (1-2 seconds at most between updates)
  - Handles large data (tens of gigabytes/volume, terabytes total data size)
  - High resolution (1 megapixel or more)
  - Good quality (no image artifacts, responsive to interaction)
  - Enables collaborative visualization

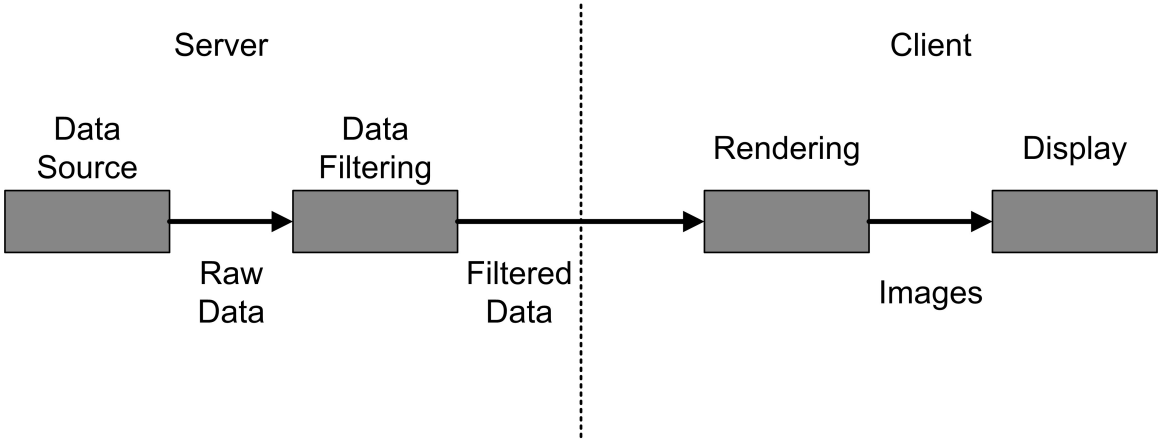
Motivation

- Real datasets generated by scientists
  - Examples: Numerical Relativity (simulation of astrophysical systems) – 4096^3 /variable and timestep (tens of variables, hundreds of timesteps); Chemistry (x-ray tomography scan) – 32 Gigabytes/scan (2048^3); 24 datasets/experiment.
- Want to have usable tools
- Initially focusing on volume rendering

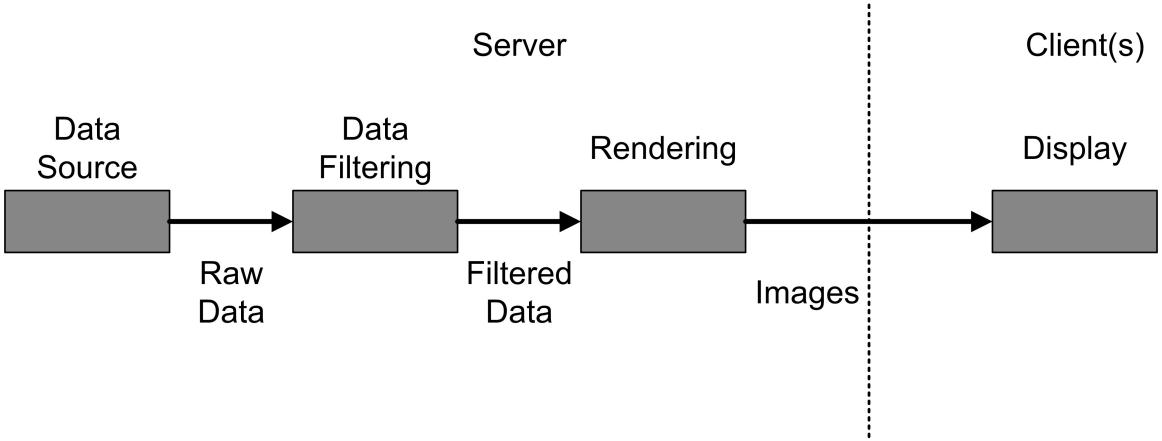


Visualization Pipeline

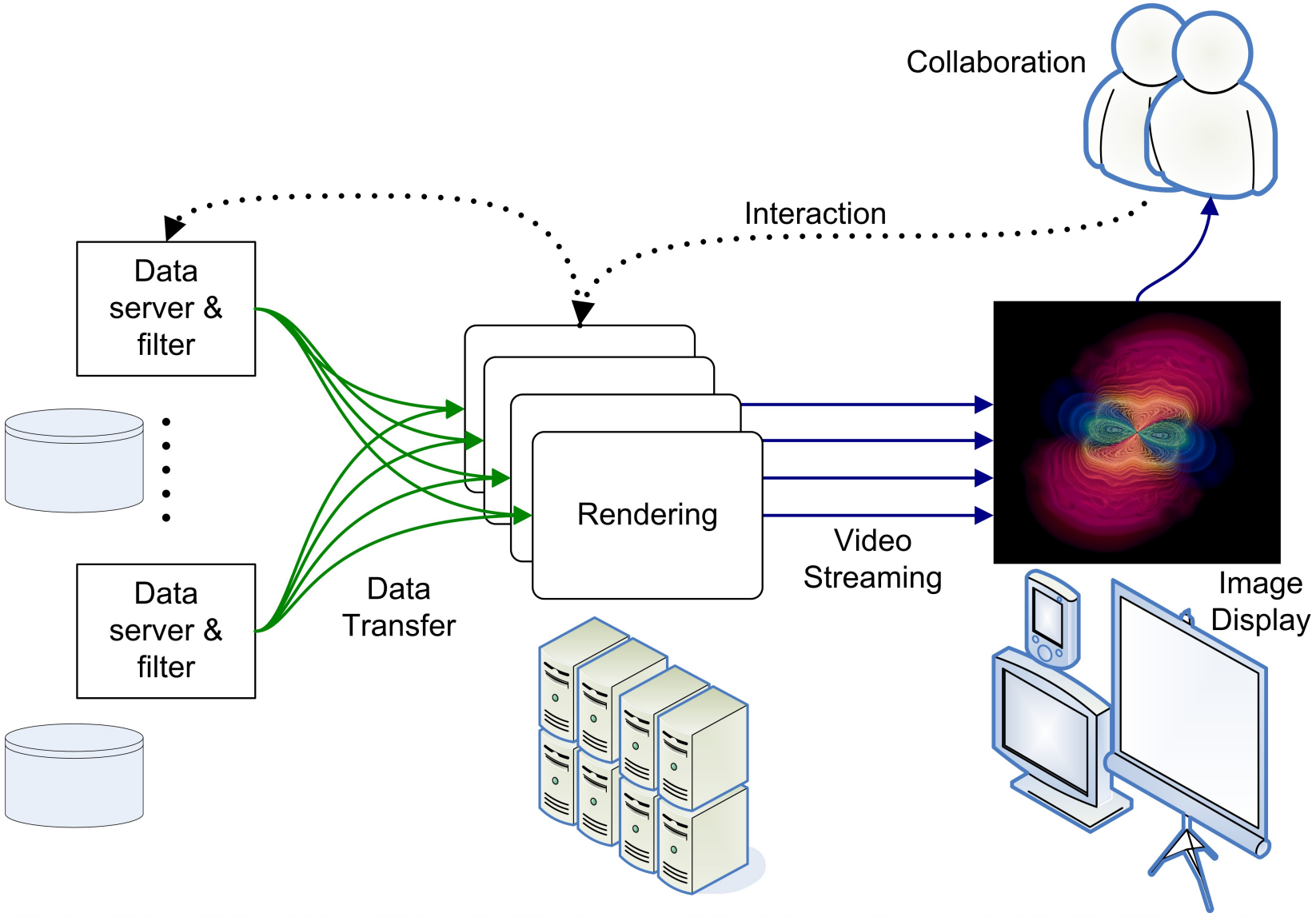
- Visualization of remote data



- Video streaming



eaviv Architecture

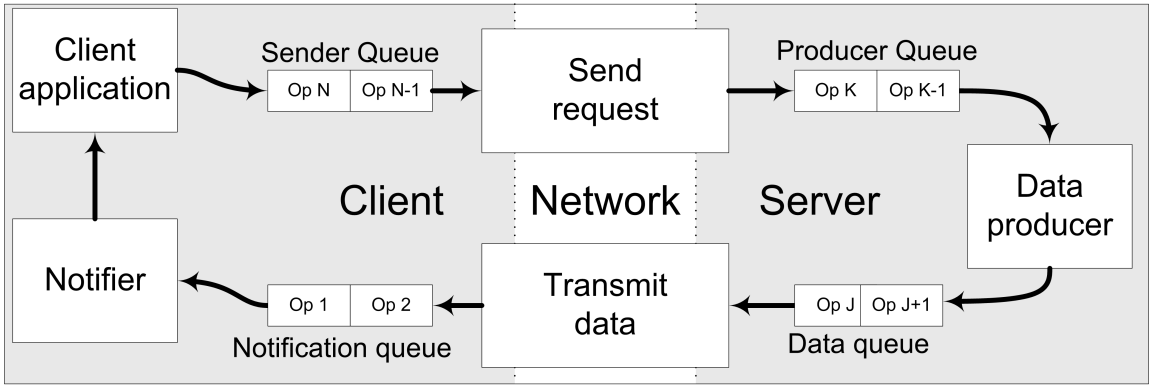


eaviv I/O - bandwidth

- Networks faster than local storage
- Distributed data servers
- Use main memory to cache data
- Fast protocols
  - Short-lived transfers (tens of seconds)
  - Reliable; Use on high-speed, possibly dedicated network links
  - TCP not suitable for high-speed links
  - Few usable alternatives, best is a protocol without congestion control (app sets rate)

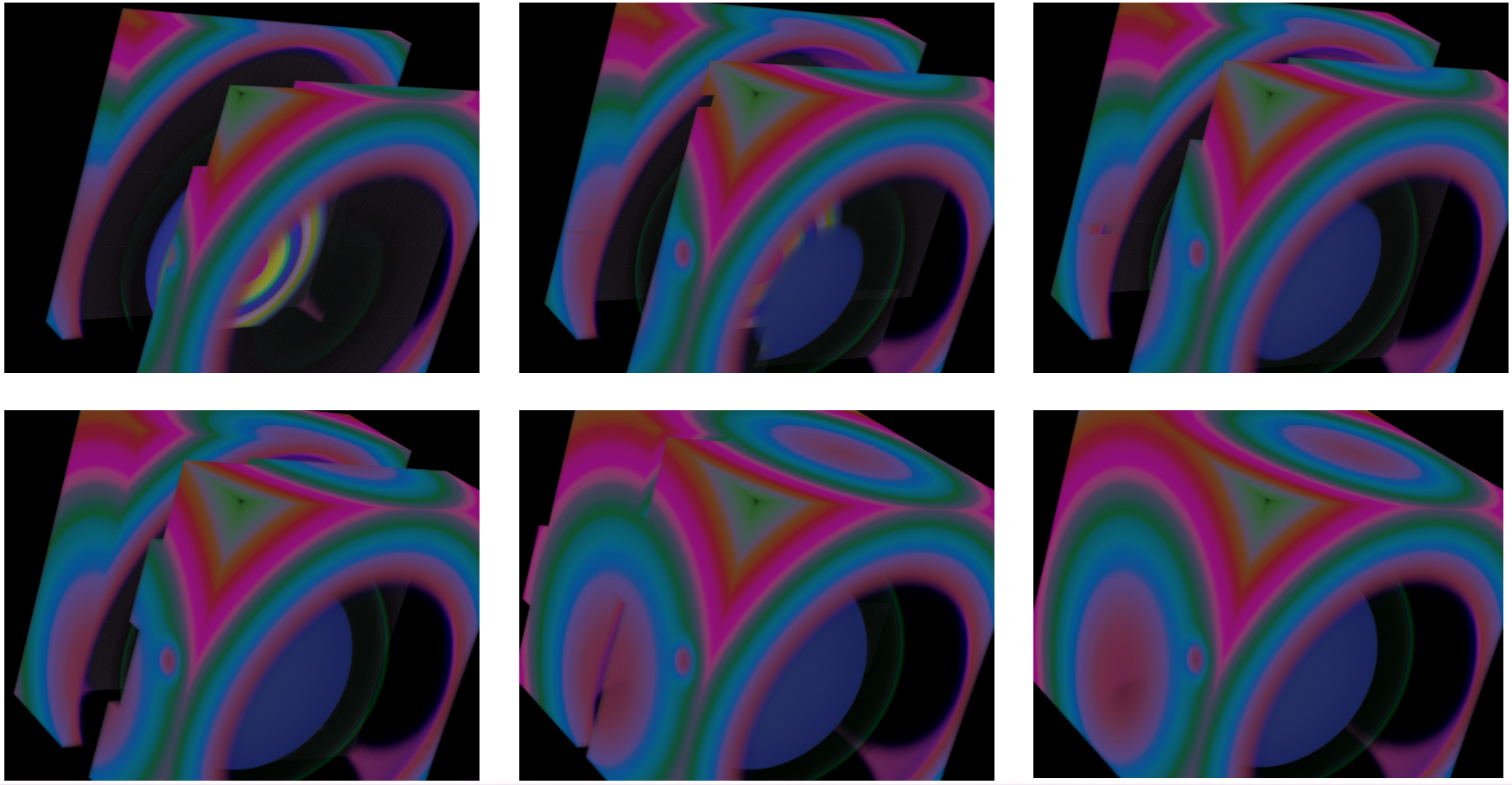
eaviv I/O - latency

- Blocking on I/O, serialized operations
  - Very expensive when doing remote I/O over high RTT links
- Pipelined, non-blocking system
  - High operation throughput
  - Configurable operations (bulk, data formats)



Rendering

- Parallel, GPU volume rendering, ray-casting
- Only data sections. Progressive visualization



Interaction

- Modify parameters (zoom, viewing direction)
- Tangible devices: interfaces that enable direct manipulation of digital objects and actions through physical means
  - Support collaboration

Streaming

- Images from remote renderer; collaboration
- Avoid quality degradation using high-speed networks
  - High resolution
  - High frame rate
  - No compression (low latency)
- Using SAGE
  - Parallel streaming from each node, UDP (though some issues when combined)

Results

- Rendering performance; I/O speed; 8 node quad core Xeon, 4 Tesla S1070-16GB each

# processes	Data size	Frame rate (fps)	Local speed	Network speed
1 (workstation)	0.8 GB	30	0.68 Gbps	0.8 Gbps
8 (cluster)	15 GB	15-21 (18 avg)	0.11 Gbps	6.6 Gbps
16 (cluster)	30 GB	11-13 (12 avg)	0.12 Gbps	5.3 Gbps
32 (cluster)	60 GB	4-5 (4.5 avg)	0.2 Gbps	4.3 Gbps

- Video streaming requirements

Resolution	Frame rate	Streaming requirements
1024x800	28 fps	525 Mbps
1920x1080	11.5 fps	546 Mbps
2048x2048	6 fps	576 Mbps

Results

Feature	*eaviv*	ParaView	VisIt
Data loading	Progressive, Asynchronous	Single operation, Blocking	Single operation, Asynchronous
Data protocols	UDT, TCP, fully configurable	TCP only	TCP only
Data servers	Distributed and parallel	Parallel only (MPI)	Parallel only, must be on same cluster
Data throughput	*5.3 Gbps Network (30 GB data)*	*0.12 Gbps Local (32 GB data)*	*0.12 Gbps Local (32 GB data)*
High-speed data limit	*Yes: Main memory*	*No: Disk size*	*No: Disk size*
Parallel volume rendering	GPU	CPU	CPU
Frame rate	*11-12 fps (30 GB)*	*0.5-1 fps (32 GB)*	*0.28-0.35 fps (32 GB)*
Render size limit	*60 GB (GPU memory)*	*120 GB (CPU memory)*	*120 GB (CPU memory)*
Time to first image	*5 s*	*35 minutes (load time)*	*35 minutes (load time)*
Visualization updates	*0.1 s (frame rate)*	*N.A.*	*N.A.*

Results

Feature	*eaviv*	ParaView	VisIt
Video streaming	Parallel (SAGE)	Serial	Serial
Video transmission	TCP, UDP (SAGE)	TCP only	TCP only
Interaction	Tangible devices (mouse & keyboard with VNC)	Mouse & keyboard	Mouse & keyboard
Collaborative support	Yes: SAGE video distribution, tangible devices	No	No
Direct simulation connectivity	No	No	Yes
Fully-featured visualization application	No (Prototype)	Yes	Yes
Programming effort	High	Lower	Lower
Execution complexity	High	Low	Low

Conclusions

- Using networks to improve I/O speed
- Remote rendering cluster to increase data size
- Support for high quality collaboration
- Future: increase data size, multiple clusters, multiple views

eaviv Project (300K, NSF EAGER)

- Distributed visualization using dynamically configurable optical networks