# Transport Protocols: Bulk File Transfer

Stanislav Shalunov ⟨shalunov@internet2.edu⟩
Steven Senger ⟨senger@cs.uwlax.edu⟩

ESCC/Internet2 Joint Techs Workshop
Salt Lake City, 2005-02-16

# Bulk Transfers

- The killer application for high-performance networks so far
  - What else do we need fat pipes for?
- Several flavors:
  - straightforward huge file transfer
  - interactive high-throughput applications (e.g., ImmSeg)
  - instrument data transfer (e.g., e-VLBI)

## Problem Exists Below Application

- Remains unsolved even in its most simple form (file transfer)
  - best current practice: open $n$ TCP streams, send data
  - typical current practice: $n = 1$ (FTP, HTTP, SCP, etc.)
- Expected performance (links are not congested): ~100 Mb/s
- Typical performance: less than 3Mb/s (Abilene)
- The *wizard gap* gets wider

## Top Reasons of Poor Performance (maybe 80% of cases)

- **Bad transport protocols** (layer 4)
- Ethernet duplex mismatch (layer 2)
- Bad last-hop cables (layer 1)

# Conventional TCP: Bad Transport

- Fundamental problems:
  - Unstable for high-speed networks
  - Too sensitive to non-congestive packet loss (even after minor fixes)
  - Before a loss happens, buffers need to fill: delay is at least doubled
- Implementation problems
  - Buffers are laughably small:
    - Normal default buffer sizes: 8kB, 16kB, 32kB, 64kB
    - Even 64kB over 70ms limits throughput to 7.5Mb/s
  - No provisions for automatic buffer increases

## Remedies for TCP's maladies

(In increasing order of invasiveness.)

- Tuning: buffers, window scaling, timestamps, SACK
- Use multiple streams
- **Something else**
- Replace the kernel and use a different congestion control
- Replace all routers and kernels

## Internet2 Transport Effort

- A group of congestion control researchers and high-end users
- Started in late October 2004
- Goal: do better than conventional TCP
- Most immediate deliverable: a design space survey

## Transport tool

- High performance
- Completely end-to-end: no router modifications
- Suitable for both bulk file transfer and interactive multimedia
- Portable, easy to install and use (no kernel modifications)
- Advanced congestion control using existing research
- Tolerance for minor non-congestive packet loss
- Security

## Design Space for the Tool

- Current version: transport-design-space-07.pdf
- Available from http://www.internet2.edu/~shalunov/transport/
- Copies of -06 on the registration desk
- Specify requirements
- Document independent design questions
- Converge on a design

# Design Space Dimensions

- Explicit signaling
- Implicit signaling
- Kernel- vs user-space
- Protocol features
- Window- vs rate-based
- TCP-compatible vs TCP-friendly
- State at sender vs receiver
- Single vs multiple streams
- UI and API

## Current ideas about the design

- TCP-friendly, not TCP-compatible
- Security nonces
- Implicit congestion signaling
- Delay-based, with fallback to loss-based
- User-space tool with UDP
- State at receiver where possible
- API and a file transfer/distribution application

## What's your take?

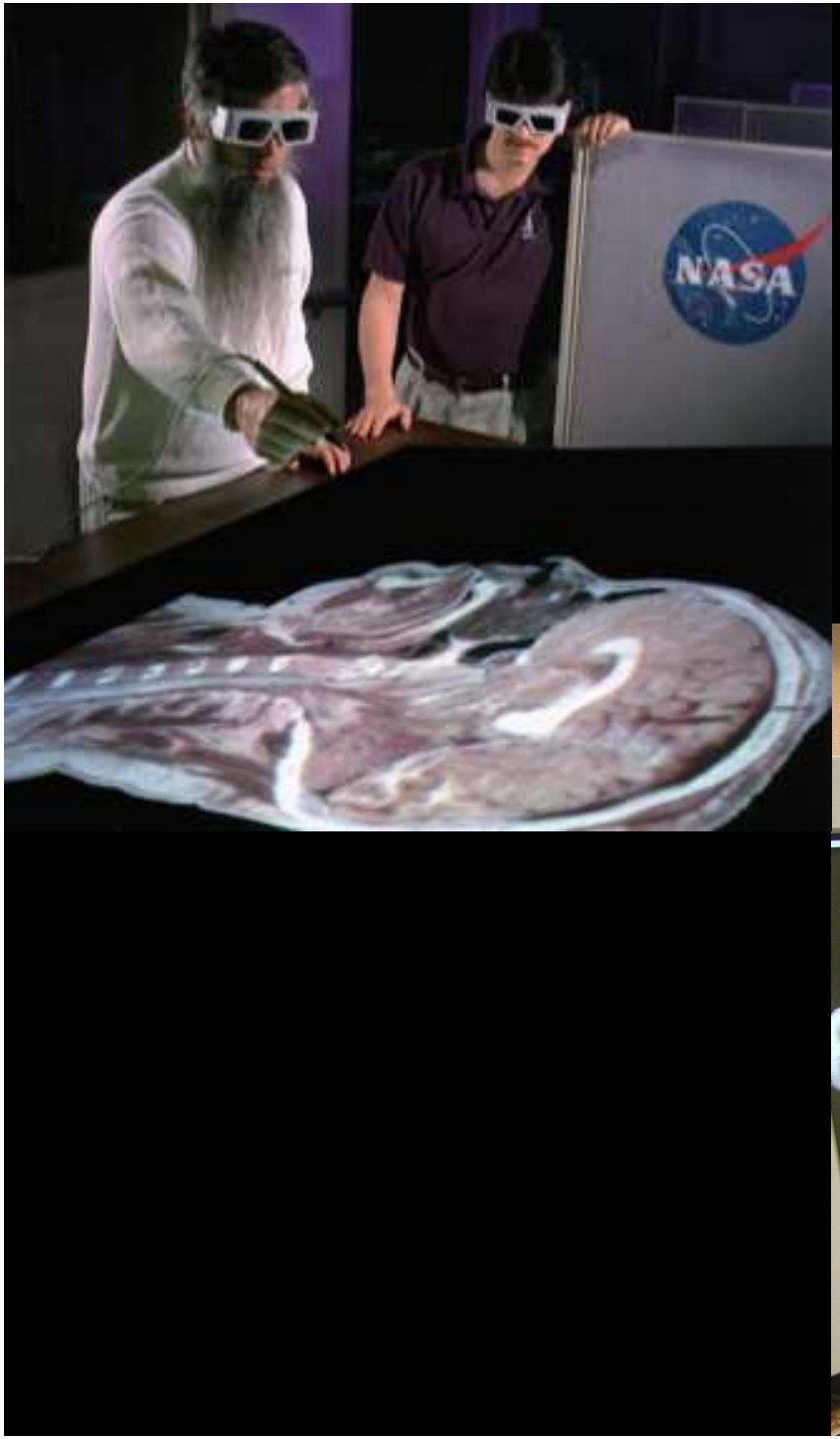
- Your feedback counts
- Mailing list: transport@internet2.edu
- Weekly teleconferences: Friday, noon ET

## ImmSeg

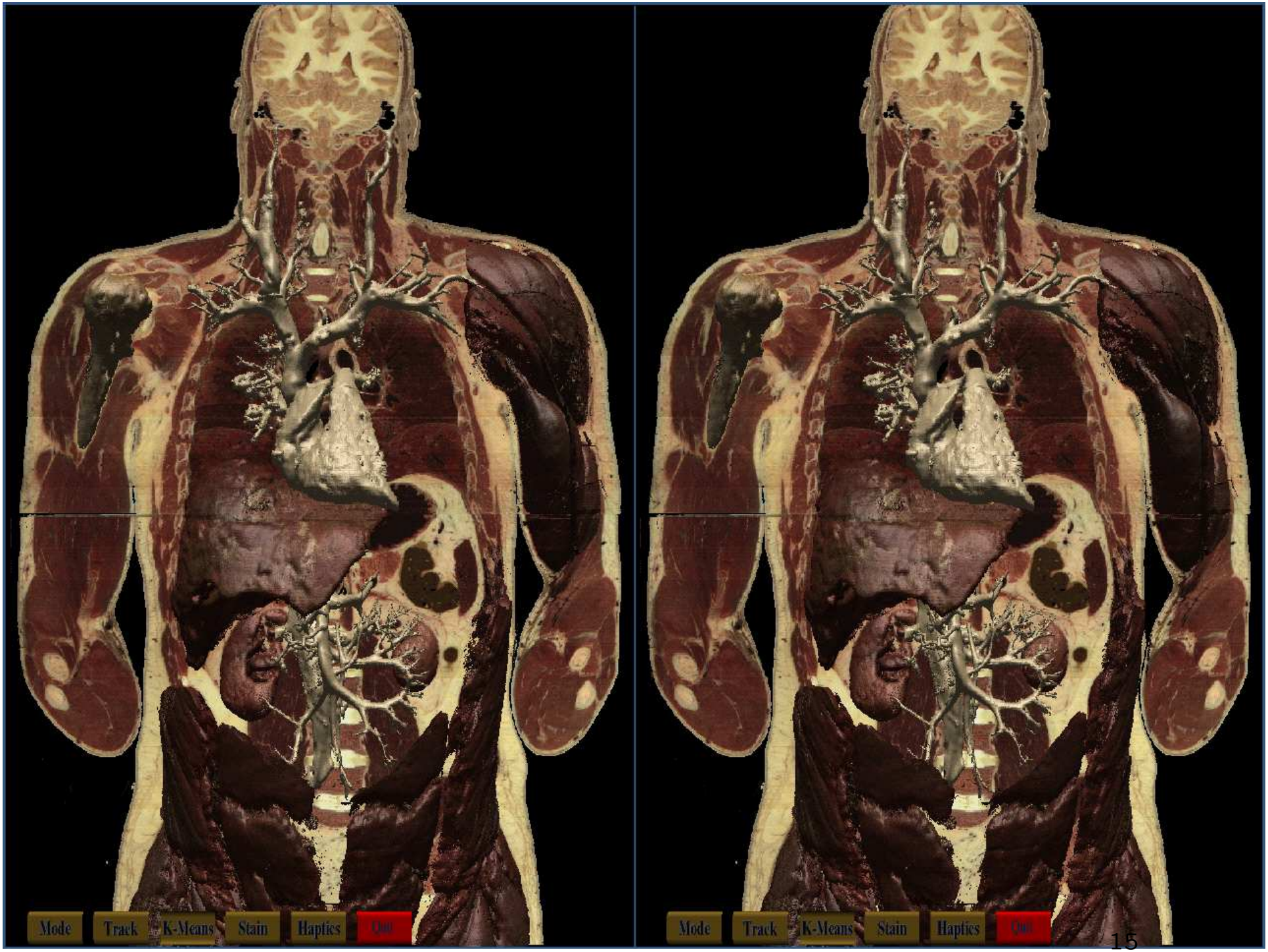
- Interactive bulk multimedia transfer
- TCP not suitable
  - Partially reliable datagrams
  - Unreliable datagrams
  - Very fast ramp-up
  - Performance
- Requirements shaped by applications such as ImmSeg

# Immersive Segmentation

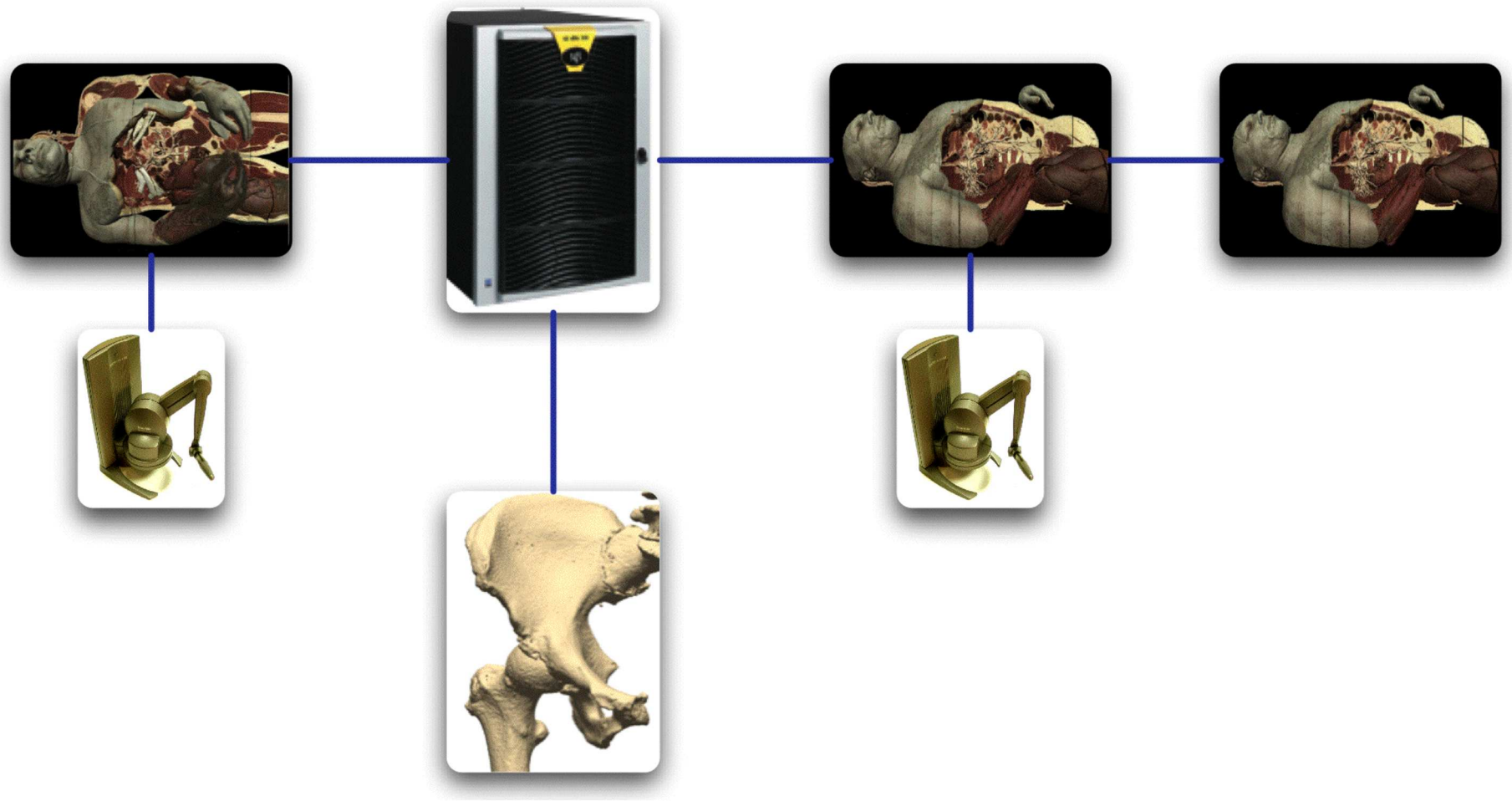
- Segment/Visualize volumetric data sets
- Stereoscopic ray-cast image
- Haptically-enabled segmentation tools
- Continuous control of remote computation
- Exploit user's expert interpretive knowledge
- Discovery-based pedagogy


NASA


EIZO


Mode
Track
K-Means
Stain
Haptics
Out
Mode
Track
K-Means
Stain
Haptics
Out

# Collaborative Environment

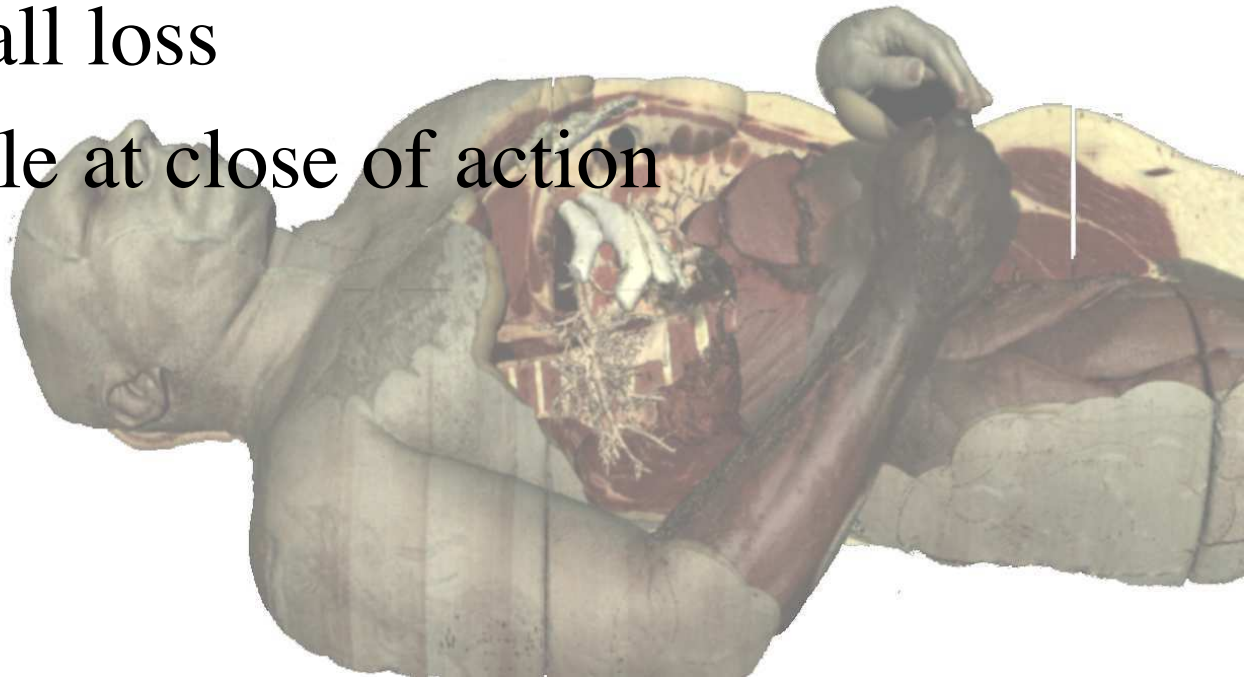


# Modes of Operation

- Region growing
  - Seed action at a point with initial category
  - Interactive steering during propagation
- Neighborhood cluster
  - Identify voxel categories in local neighborhood
  - Heuristically determine opaque category
  - Visual transition between category changes
  - Fix category choice and move through volume

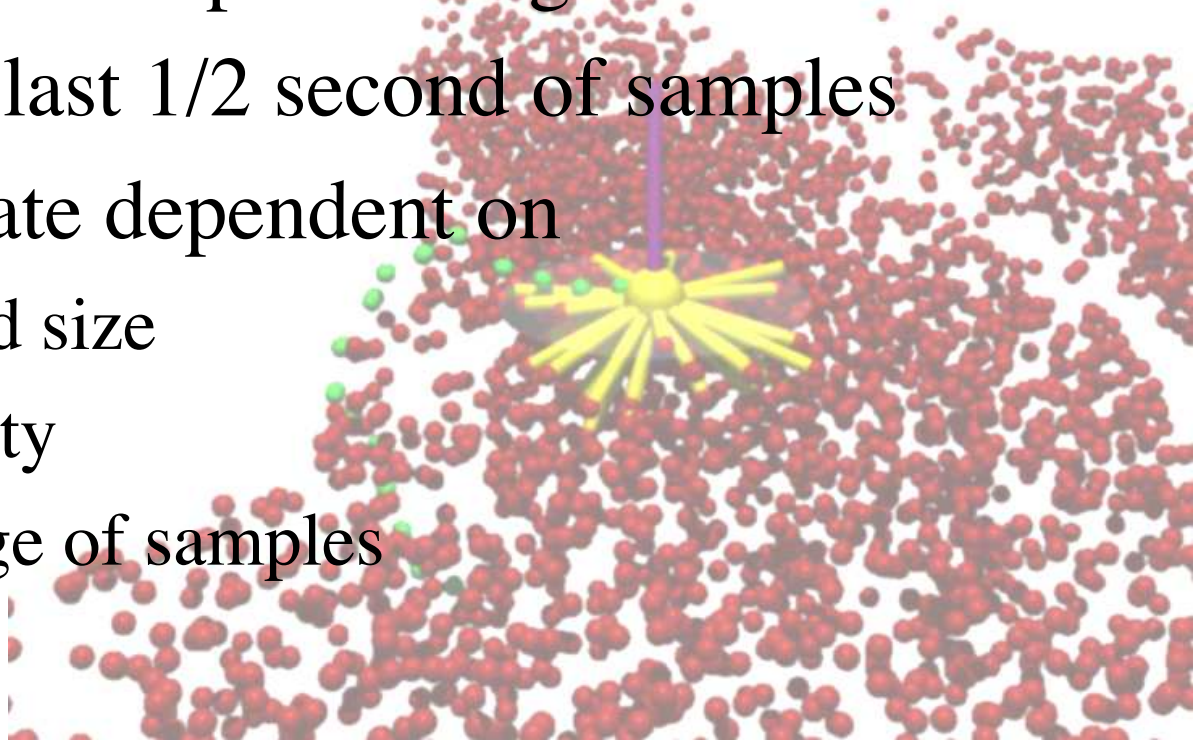
# Visualization Stream

- Pixel updates during user action
- Varies by mode, up to 30Mb/s per client
- Tolerate small loss
- Prefer reliable at close of action



# Haptics Stream

- Reflect changing segmentation on server
- Stream random sample of neighborhood
- Client retains last 1/2 second of samples
- Unreliable - rate dependent on
  - neighborhood size
  - desired density
  - acceptable age of samples



# Surface Data

- Interactively mesh segmented structures
- Press against and move over structure
- Mesh data streamed to auxiliary client
- Reliable, ~40Mb/s bursts

## More information

- http://www.internet2.edu/~shalunov/transport/
- Join the mailing list: transport@internet2.edu
- Send comments on the design space survey

## Contributors

Lawrence D. Dunn (Cisco), Yunhong Gu (UIC),
Steven Low (Caltech), Injong Rhee (NCSU),
Bartek Wydrowski (Caltech), Lisong Xu (UNL)

# Questions?