Distributed, Networked Surgical Simulation:

Opportunities for Standardization and Interoperability

Kevin Montgomery PhD

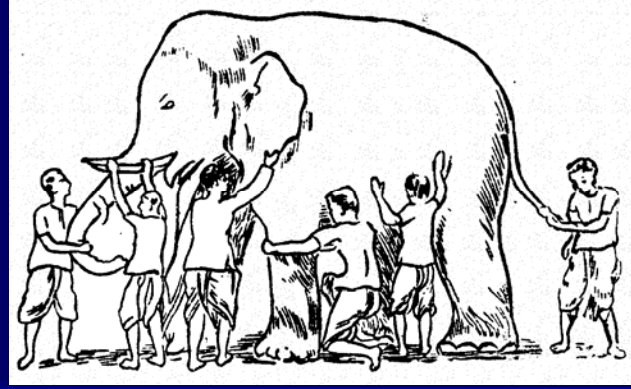
STANFORD UNIVERSITY MEDICAL CENTER

NATIONAL BIOCOMPUTATION CENTER

Surgical Simulation Benefits

- Broader training: Easily provide different scenarios
  - Anatomical variations (gender, age)
  - Pathologies (diseases, trauma)
  - Operating environments (ER, battlefield, space)
- Objective quantification of performance:
  - Simulate results
  - Certification
- Accelerated acquisition of baseline skills
- No risk to real patients

The Idealistic Goal



- Let’s all work together on a common framework of shared code
  - Less time to realization of a working simulator
  - Shared individual expertise/contribution
  - Barrier to entry, deployment, and proliferation lessened
  - Accelerate the production and adoption of simulators
  - Realize the benefits of surgical simulation sooner

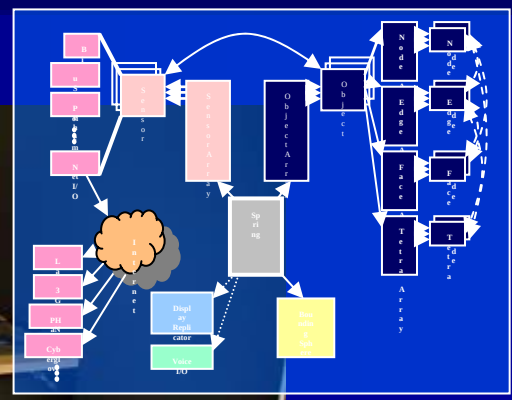
Overview

- Describe mass-spring physical simulation system built over past 6 years: ***Spring***
- Features:
  - Platform: (Sun, SGI, PC, Linux), C++/OpenGL, Parallelized
  - Models: Relatively easy introduction of patient-specific anatomy
  - Simulation: Soft tissue modeling, rigid body dynamics
  - Interfaces: Many devices, multi-user, multi-instrument
  - Haptics: networked, latency dependent or independent
  - Instruments: many surgical/nonsurgical produced
  - Collision Detection/Response: BSP-tree with enhancements
  - Display: stereo CRT, HMDs, projection, anything
  - Misc: Voice I/O, video input, stereo, replicated display (image, geom)
- Applications: Produced during development
- Emphasis on real-time (haptic rate) performance and generality

System Overview

Haptic Device/User

TCP/IP

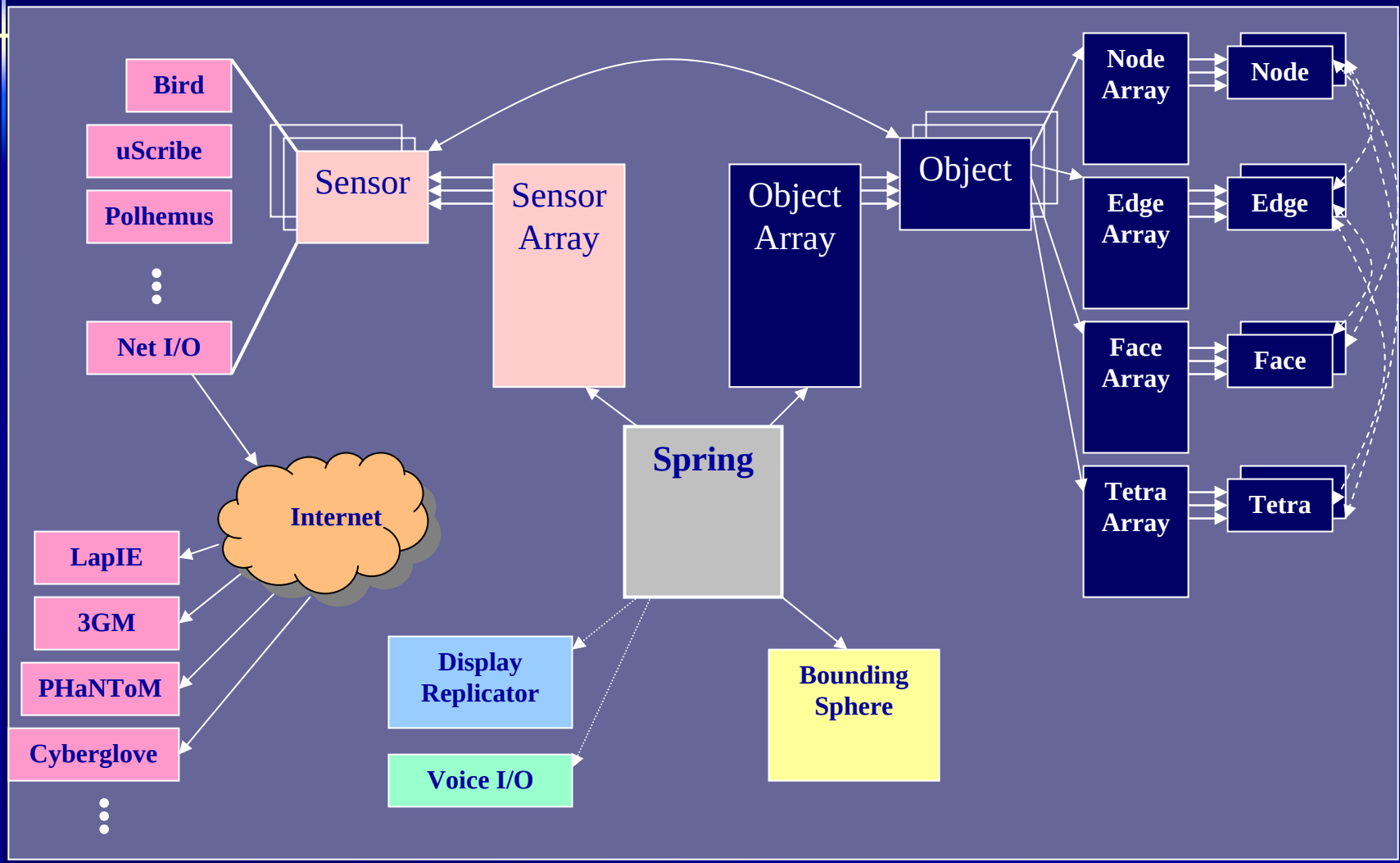


Simulation Engine

Proprietary

Haptic Device Controller

Simulation Architecture



Areas for Standardization

- Data:
  - “Your liver in my abdomen”
- Simulator:
  - Open source simulation engines with standard APIs & Code sharing
  - Standard methods for tool-tissue interaction- predictable
- Devices:
  - Standard tracker/haptic device interfaces
- Network protocols:
  - Networked Haptics protocol
  - Distributed rendering protocol
  - Streaming video protocol (MJPEG)
  - Voice control protocol

Soapbox

- Best standards are those borne from real-world implementation- Otherwise:
  - Unimplementable standards
  - Standards glut (so many to choose from!)
  - Standards churning/turnover
  - *De facto* wins over *de juris*
- Standards must be produced at the right time (not too early, not too late)

Summary

- Open our kimonos (and code) now
- Work together
- Establish interfaces
- Learn a lot about what works and doesn’t

- Then- useful, independent/open, 3D standards are coming