

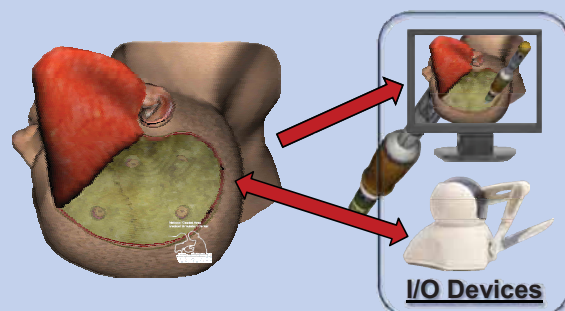
MMVR '11 Workshop – February 10, 2011
Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative

A Pipeline Architecture for Parallel Virtual Environments

Eric Acosta

National Capital Area Medical Simulation Center
Uniformed Services University
<http://simcen.usuhs.edu>

Virtual Environment (VE)



Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative

Head Trauma Simulator VE



Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative

Virtual Environments

VE Modeling

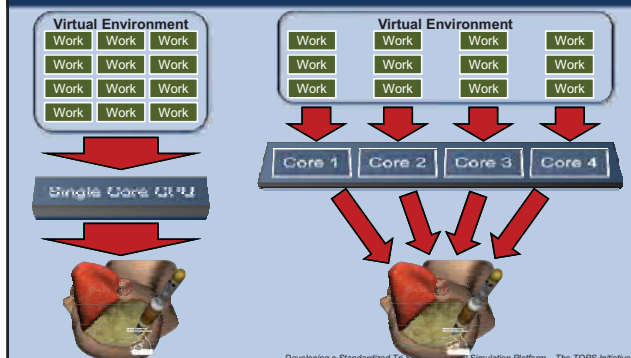
- Geometrical
 - Geometry
 - Visual properties
- Physical
 - Behavior
 - Interactions

VE Execution

- Interactions
- Updates
- Rendering

Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative

Multi-core CPUs



Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative

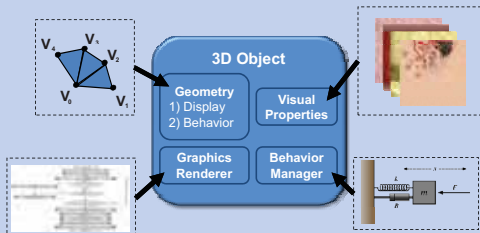
Pipeline Architecture Overview

- Develop VEs for multi-core CPU systems
 - VE Modeling
 - 3D Objects
 - Interactions
 - Multi-threaded VE Pipeline (execution)
 - Parallel simulation & rendering

Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative

VE Modeling

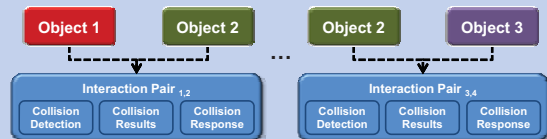
- 3D Objects
 - Modularized data & algorithms



Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative

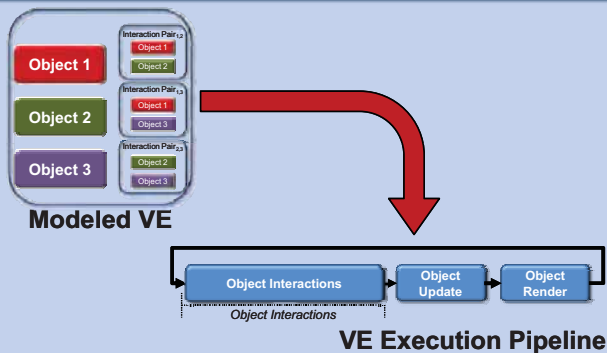
VE Modeling

- Interaction Pairs
 - Pair-wise → parallel processing
 - Interactions of interest

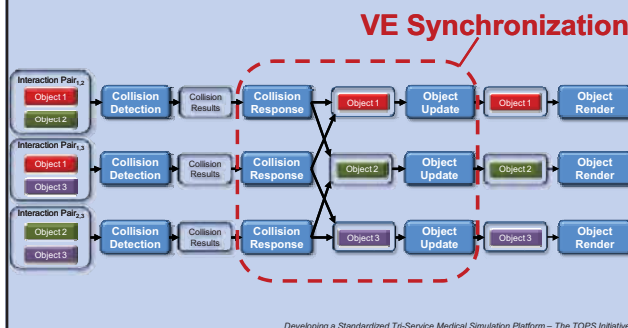


Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative

VE Simulation & Rendering Pipeline



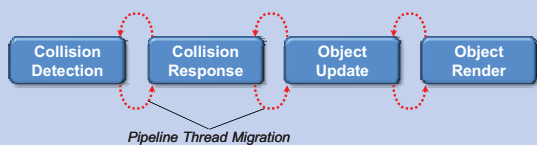
Parallel Pipeline Concept



Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative

Multi-threaded Pipeline Overview

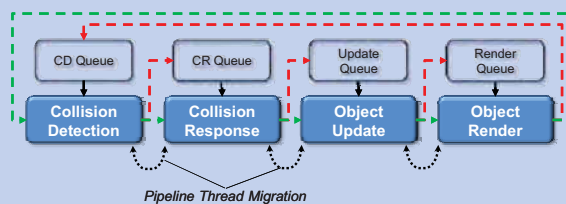
- Thread scheduling
 - Per thread pipeline → Fixed & inefficient
 - Thread migration → Dynamic load distribution & balancing



Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative

Multi-threaded Pipeline Overview

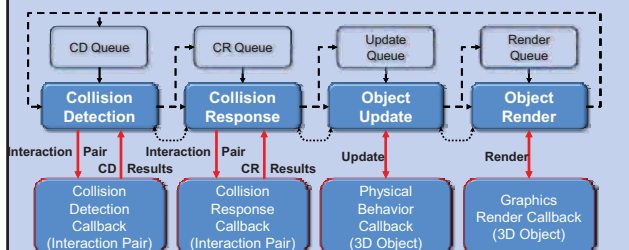
- Work flow overview
 - Thread-safe pipeline queues (synchronization)
 - Carry-over (no synchronization)



Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative

Multi-threaded Pipeline Overview

- Callback mechanism
 - Trigger callbacks (VE modeling)
 - Pipeline remains domain independent

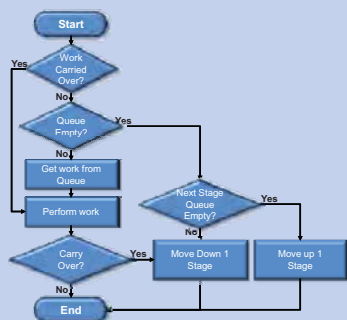


Pipeline Threads

- Thread pooling
- Generalized pipeline threads
 - Callbacks → Execute any pipeline stage
- Autonomous design
 - Loops until terminated
 - Execute work at current stage
 - Embedded scheduling algorithm → current stage
 - No dedicated thread scheduler required

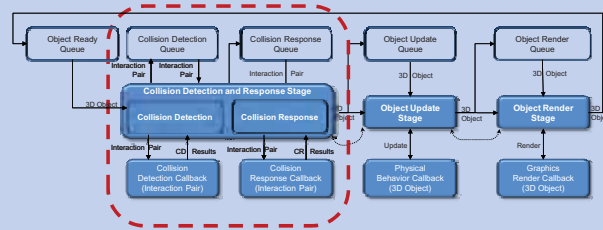
Thread Scheduling Algorithm

- High-level Basic flow



Multi-threaded Pipeline

- CD/CR queues large → Synchronization overhead
 - Eliminate CR queue (carry-over only)
 - Combine interactions stages



Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative

Virtual Environment Interactions

- Collision Detection



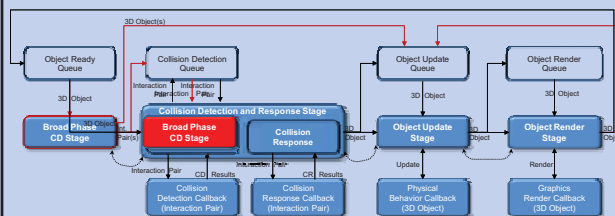
1. Broad Phase

2. Narrow Phase

Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative

Multi-threaded Pipeline

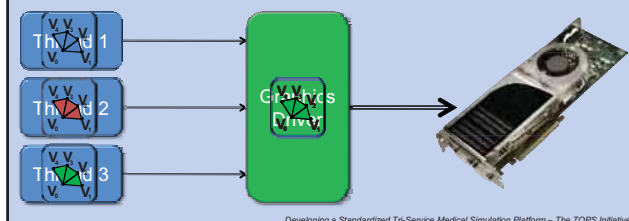
- Separate broad-phase CD stage
- No interactions → Object Update



Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative

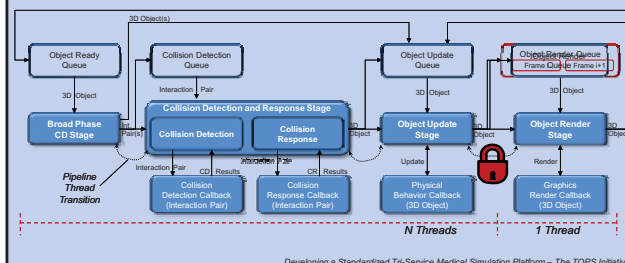
Multi-threaded OpenGL Rendering

- Multiple rendering threads
- Single graphics driver (bottleneck)
 - Context switch overhead

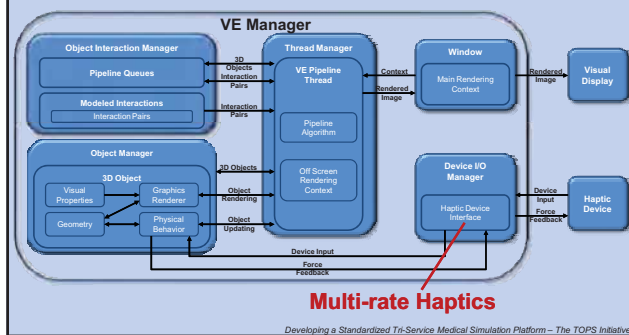


Multi-threaded Pipeline

- Object Render token system
- Dual-frame processing → Reduce idle threads



Modular Architecture



Experimental Results

- Test System
 - Windows 7 64-bit
 - 6-core Intel i7-980x (3.33GHz)
 - 6 GB RAM
 - Nvidia Quadro 5000 (2560 MB)

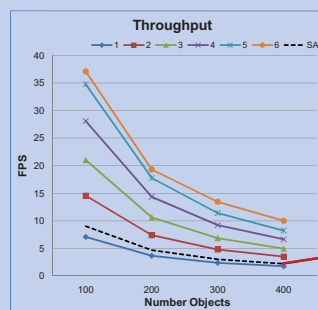
Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative

Experimental Results

- Deformable mass-spring particles
 - Vary # particles and # threads
 - Particles
 - 42 nodes & 162 springs
 - Random step size (1-100x per frame)

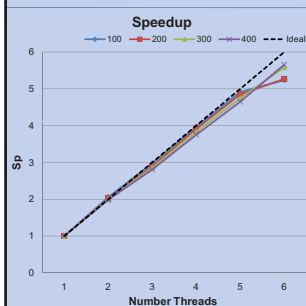


Deformable Particles Results



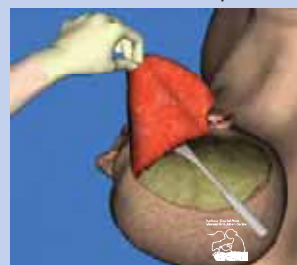
Single Threaded Optimized App.

Deformable Particles Results



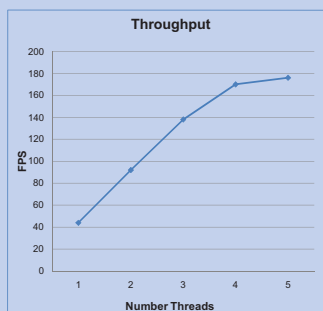
Experimental Results

- Head trauma simulator
 - Automated skin retraction step

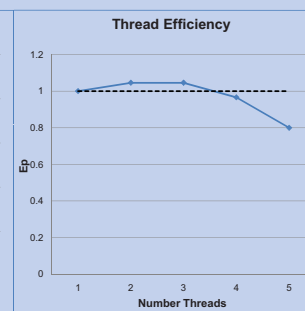
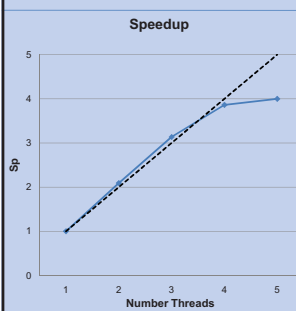


Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative

Head Trauma Simulator Results



Head Trauma Simulator Results



Conclusions

- Object-centric development
 - Objects + interactions
 - Support parallel and sequential algorithms
 - Parallelization transparent to developer
- Near-linear performance
- Scales efficiently vs. number cores used

Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative

Conclusions

- Thread scheduling algorithm
 - Simple and effective
 - 1 thread → Sequential pipeline behavior
 - Work load distribution (no scheduler overhead)
- Automatic load balancing
 - Write application once → scale performance

Developing a Standardized Tri-Service Medical Simulation Platform – The TOPS Initiative