ModelNet: Scalability and Accuracy in a Large-Scale Network Emulator

Ken Yocum, Kevin Walsh, Amin Vahdat, Priya Mahadevan, Dejan Kostic, Jeff Chase, and David Becker

Goal: Develop and explore scalability techniques for distributed service evaluation

- **Simulation**
  - Slow, must modify applications
- **Live Deployment**
  - Limited in size, waiting for failures, not reproducible
- **Emulation**
  - Static, small scale, hard to configure

**ModelNet: Wide-Area Emulation**
- Accurate: error < 1ms @ 95% CPU utilization
- Multiple techniques trade scalability for accuracy
- Run unmodified applications on unmodified OS’s
- Reproducible results (recreate flash crowds, network partitions, etc)
- Support overlays, wireless networking, content distribution networks, peer-to-peer systems

**ModelNet Validation with Chord CFS**

- TCP file transfers across emulated 12-node RON
- RON data from CFS SOSP 01
- Reproduce end-to-end effects
- Uses 12 edge nodes

**ModelNet Performance**

- Single Core Capacity (10vn’s/edgenode)
  - Per-Hop cost = 2.8 µs
  - Per-packet cost = 8 µs
  - Distillation reduces number of hops
- Packet sizes (1500 bytes), full mesh

**ModelNet: Scalability and Accuracy in a Large-Scale Network Emulator**

**Simulation**
- Slow, must modify applications

**Live Deployment**
- Limited in size, waiting for failures, not reproducible

**Emulation**
- Static, small scale, hard to configure

**Core Nodes Emulate a Target Topology, 5 Phases to Running ModelNet.**

1.) **Create**
   - Trade accuracy for scalability
   - Faults cross traffic

2.) **Distill**
   - Able to evaluate large scale performance
   - adaptive mesh refinement

3.) **Assign**
   - Balance load & minimize hardware footprint
   - Topological cuts
   - Balance load
   - Reduce inter-core traffic
   - Replicate pipes
   - Observe/adjust
   - Graph partitioning problem
   - Spectral Analysis
   - Kernighan & Lin

4.) **Bind**
   - Map clients to edge nodes and edge nodes to cores
   - Give clients in target topology IP addresses

5.) **Run**
   - Observe

**Install ModelNet Database Configuration**

**Applications Run on Edge Nodes**

**Bandwidth Latency Loss**

**Faults Cross Traffic**