

Viking: A Multi-Spanning-Tree Ethernet Architecture for Metropolitan Area and Cluster Networks

Srikant, Kartik, Susanta, and Tzi-cker

Srikant@cs.sunysb.edu

Experimental Computer Systems Lab

Stony Brook University

Outline

- Introduction
- Motivation
- Viking Architecture
- Performance
- Summary

Metro Networks

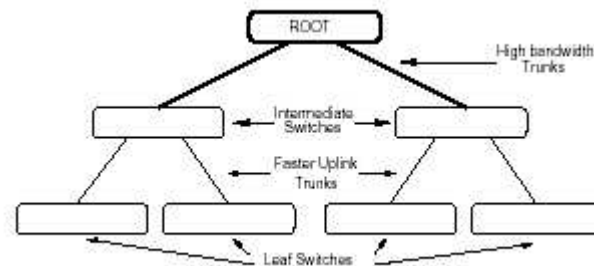
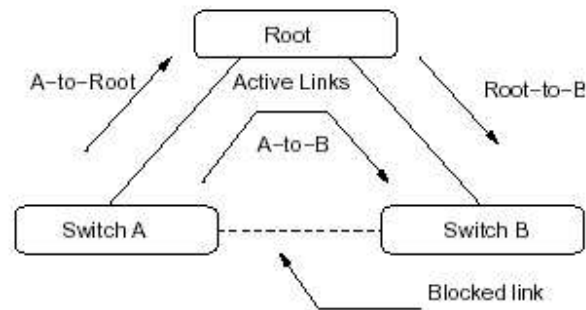
- LAN < MAN < WAN
- Use SONET/SDH, RPR, ATM, DWDM etc.
- Growing shift towards Ethernet
 - Simple, Scalable, Economical, Easy internetworking

Cluster Networks

- Storage Area Networks
- Web Server farms
- Crossbar technology or Fiber channel
 - Myrinet, Bynet
- 1 & 10 Gbps Ethernet a cost effective viable alternative

MAN Issues

- Spanning tree based switching
 - low fault tolerance
 - load imbalance
 - high convergence time (30-60 seconds)
 - Cannot use redundant links
 - Fat-tree hierarchy

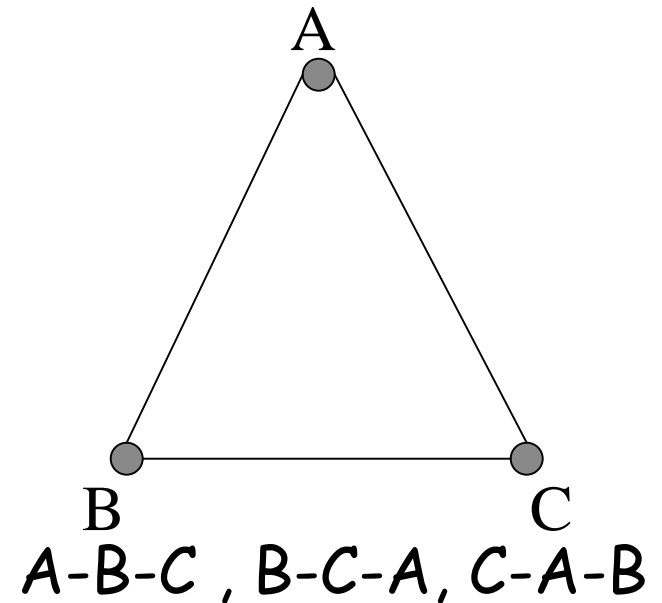


Cluster Network Issues

- Segregation limits bandwidth
- Per-port-backplane bw is limited
- Concentrated points of failure due to aggregation

Viking Solution

- Use multiple spanning trees
- Fault-tolerance
- Load balancing
- Elimination of fat-tree
 - Easy reconfiguration
- Segregation
 - RAID cluster
- Ethernet does not have a mechanism to support multiple spanning trees



Virtual LANs

- Abstraction to define limited broadcast domains
 - Segregation, grouping, administration
- Types
 - Port-based, MAC address, Protocol, IP address
 - Explicit tagging (802.1Q)
- 802.1s: Per VLAN Spanning Tree

Viking Goals

- Spanning trees should minimally overlap
- At least two switching paths between any node pair: Fault-tolerance
- Path selection should be load balanced
- Limited number of spanning trees

Viking Architecture

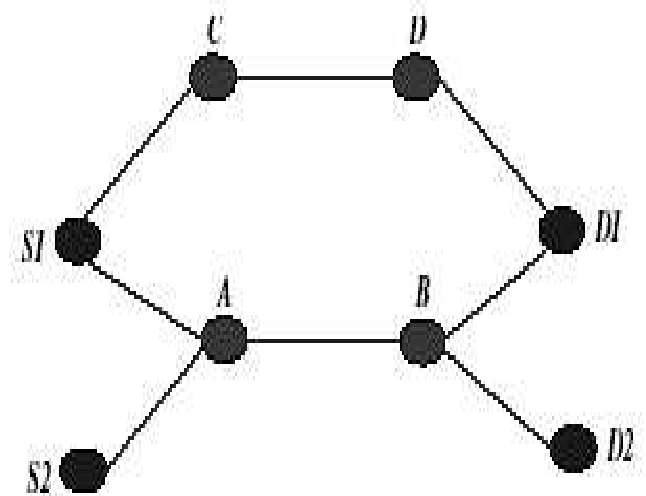
- Topology Knowledge
- Load characterization
- Traffic Engineering
 - Path Selection
 - Spanning tree construction
- Fault-tolerance

Viking Architecture

- Use topology knowledge and load characteristics to come up with appropriate switching paths (LCBR)
- Merge the together to form multiple spanning trees (Path Aggregation)
- Provide fault tolerance by switching the VLANs associated with different spanning trees (Fail Over)

Path Selection

- Link Criticality Based Routing
 - Input: Topology & Load
 - Output: Efficient paths
- Minimize the use of critical links
- Provision backup switching paths



Input: Network topology G .

New route request between nodes s and d

Average bandwidth requirement $B(s, d)$

For all links l : ϕ_l , R_l and C_l

List \mathcal{L} of candidate primary-backup route pairs (X, Y)

Output : Primary route $X(s, d)$ and backup route $Y(s, d)$

$cost_{min} = \infty$; $X(s, d) = Y(s, d) = nil$

For each route pair (X, Y) in the list \mathcal{L} .

If $B(s, d)$ cannot be satisfied along X or Y
then skip to next route.

Recompute the residual capacities R'_l for each link $l \in X \cup Y$.

Recompute the $cost(l) = \frac{\phi_l}{R'_l}$ for each link $l \in X \cup Y$.

Recompute the $cost(G) = \sum_{l \in G} \left(cost(l) - \frac{\phi_l}{C_l} \right)^2$.

If $cost(G) < cost_{min}$ then

$cost_{min} = cost(G)$

$X(s, d) = X$ and $Y(s, d) = Y$.

If ($cost_{min} > cost_threshold$) then

Reject the route request

else

Select route-pair $(X(s, d), Y(s, d))$ as

primary-backup route-pair between s and d

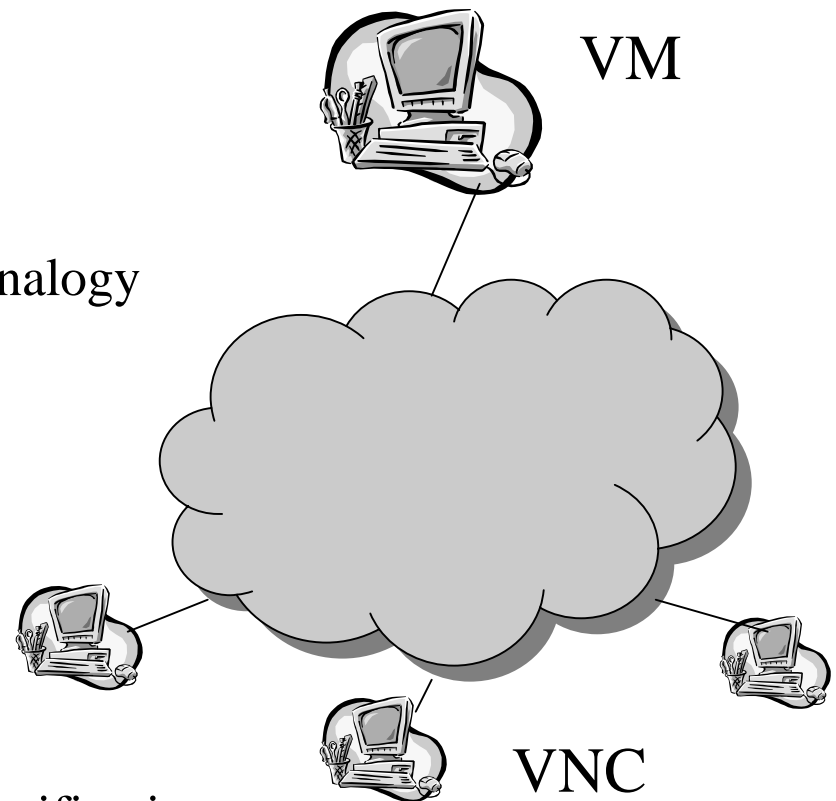
Path Aggregation

- Input: List of Paths; Output: Spanning trees
- Limited number of spanning trees
- Merging based on *common features*
- Common features
 - Common nodes, common edges, common subpaths

- Let the set of all paths be P
- Let the set of all edge pairs be EP
- Let the set of spanning trees be S
- Set $S = \phi$
- Sort the members of P in the descending order of path length
- While ($EP \neq \phi$ and $P \neq \phi$)
- Sort the members of EP in descending order of their frequency of appearance in members of P
 - Set $ep =$ Next element in EP
 - While $\exists p \in P$ such that $ep \subset p$
 - * Remove p from P
 - * Find $s \in S$ such that p and s do not form a loop
 - * Merge p with s
 - * If no such s is found, add p to S

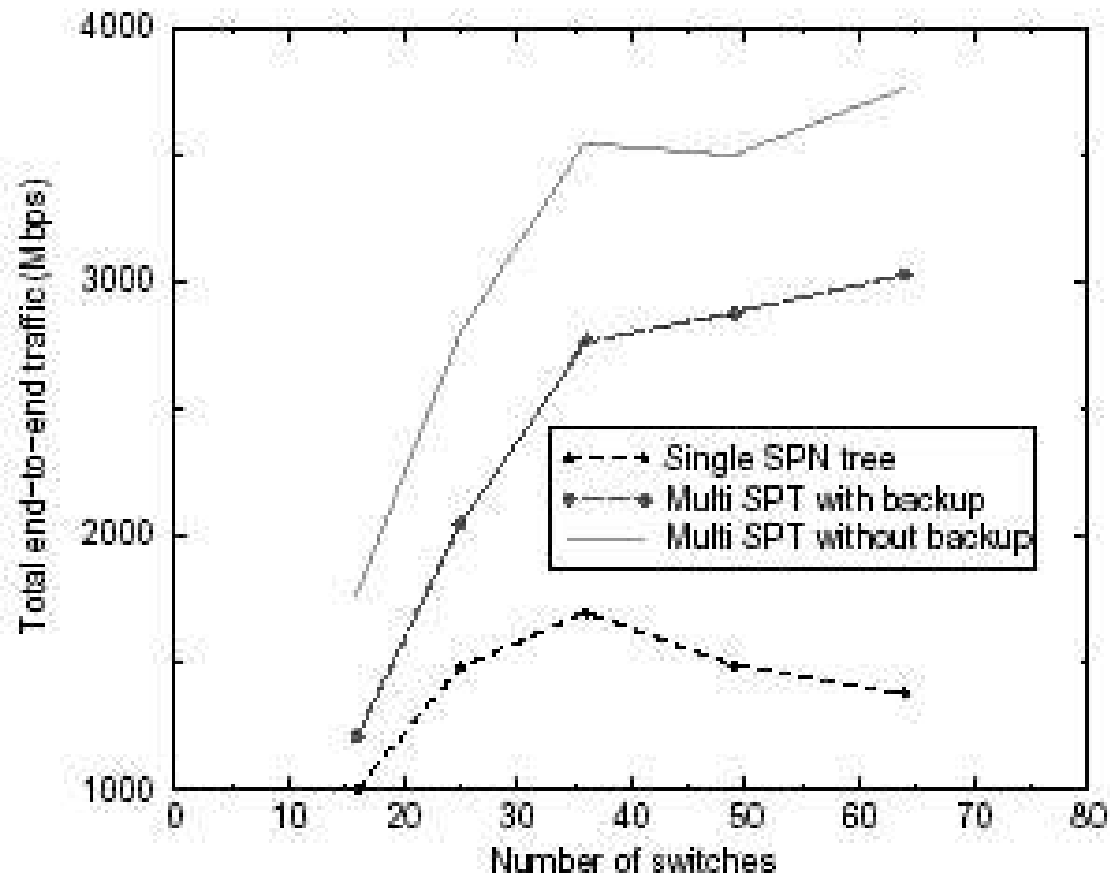
Implementation

- Viking Node Controller
 - Run-time VLAN selection
 - VLAN tag selection, MPLS analogy
 - Load Measurement
- Viking Manager
 - Traffic engineering
 - Fault-tolerance
 - Detection: SNMP
 - Recovery: Alternate VLAN notification

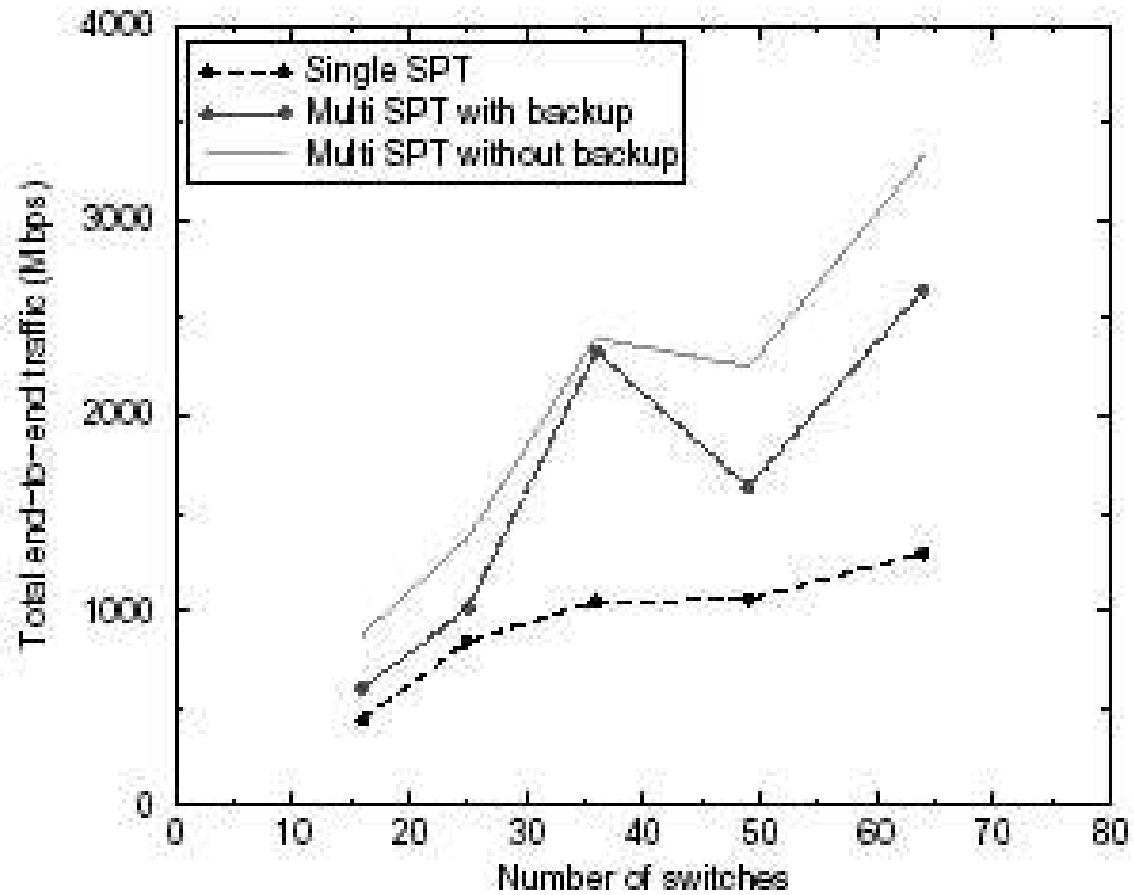


Performance Evaluation

- Simulation based
 - Grid topology, 16, 25, 36, 49, and 64 nodes
 - Uniform and Skewed traffic distribution
 - Uniform : 10, 8, 5, 2, and 1 Mbps
 - 10 % peer-to-peer communication Uniform
 - Skewed : 30, 20, 15, 8, and 5 Mbps
 - 10 % client-server communication
 - Traffic distribution
 - Number of required VLANs



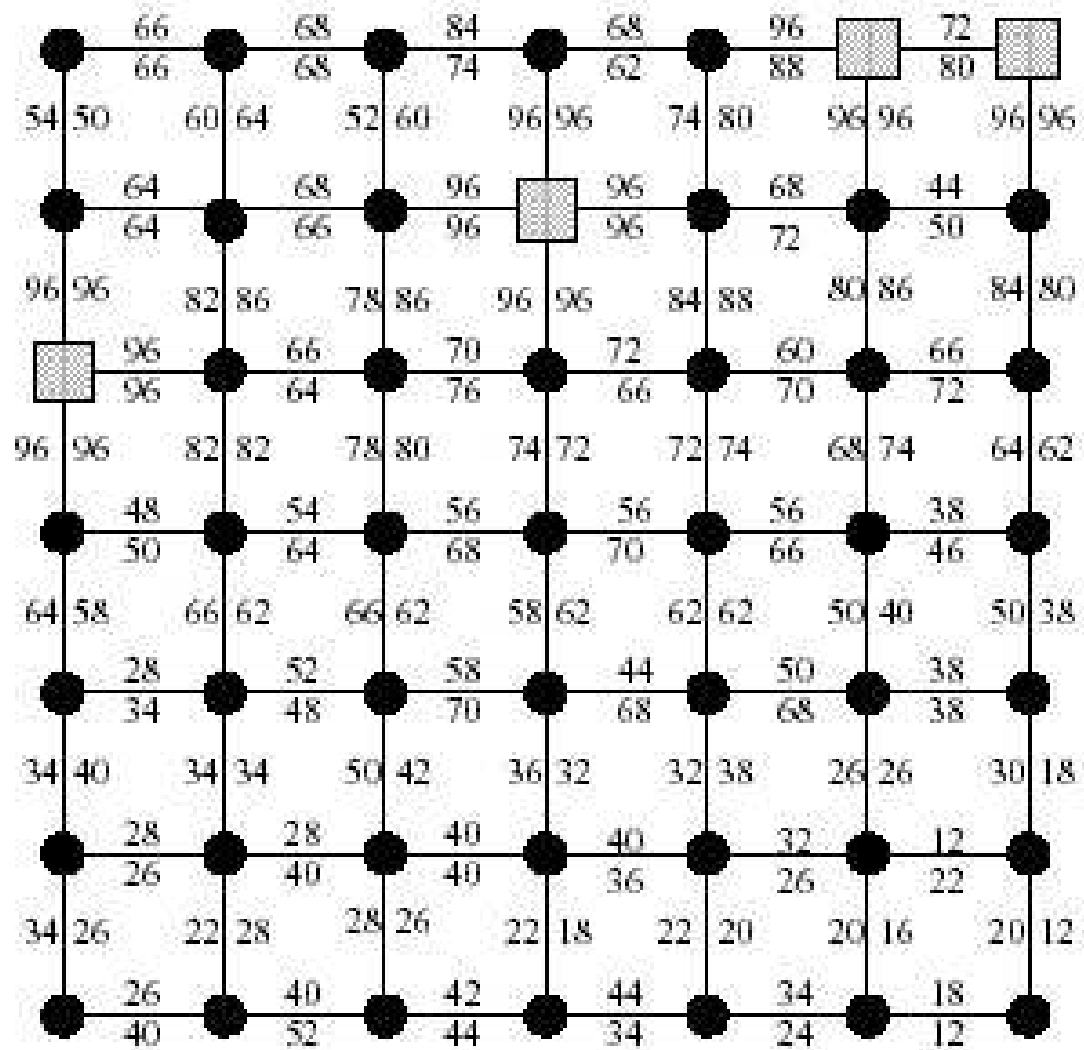
Uniform Traffic Distribution



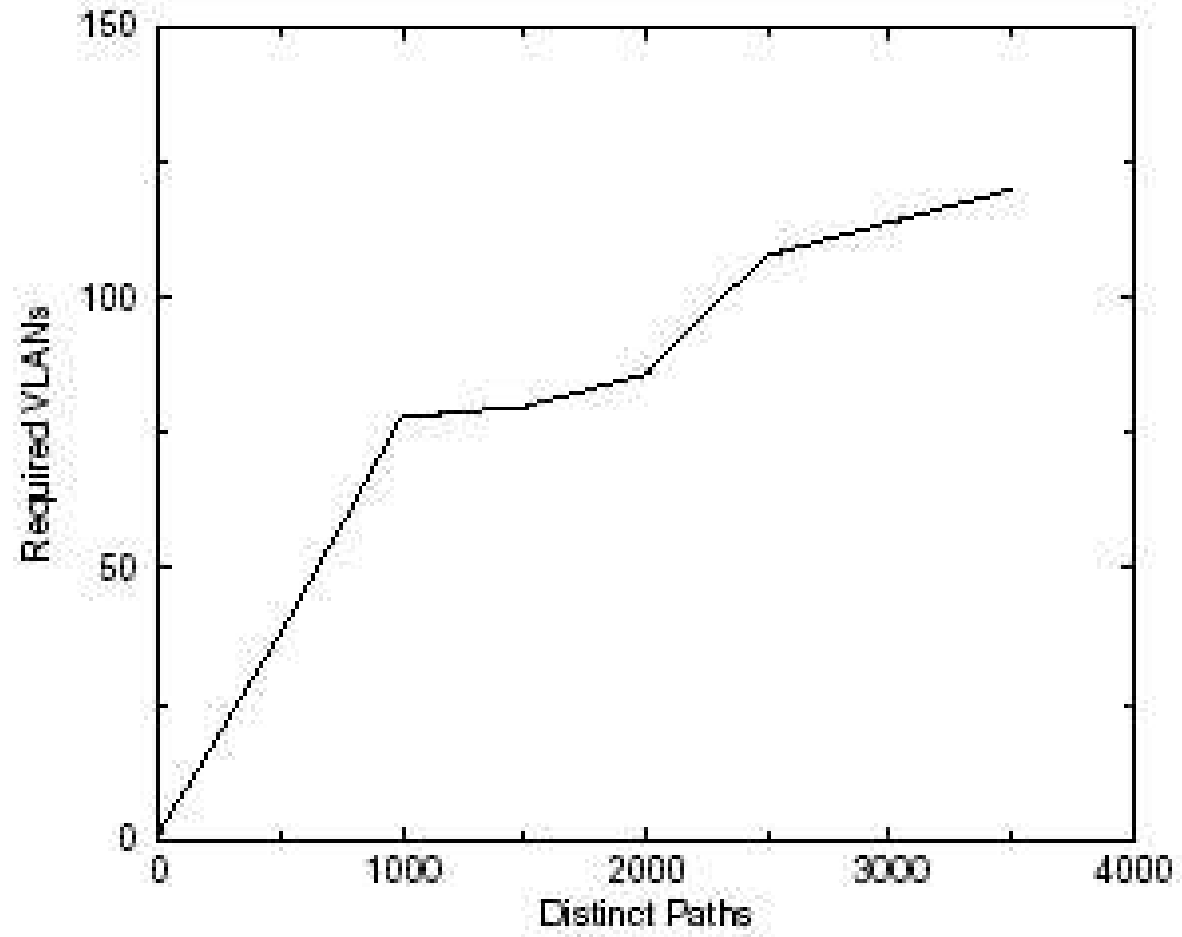
Skewed Traffic Distribution

9/28/2004

ECSL, Stony Brook University



Traffic Distribution in 7x7 Grid



Number of Required VLANs

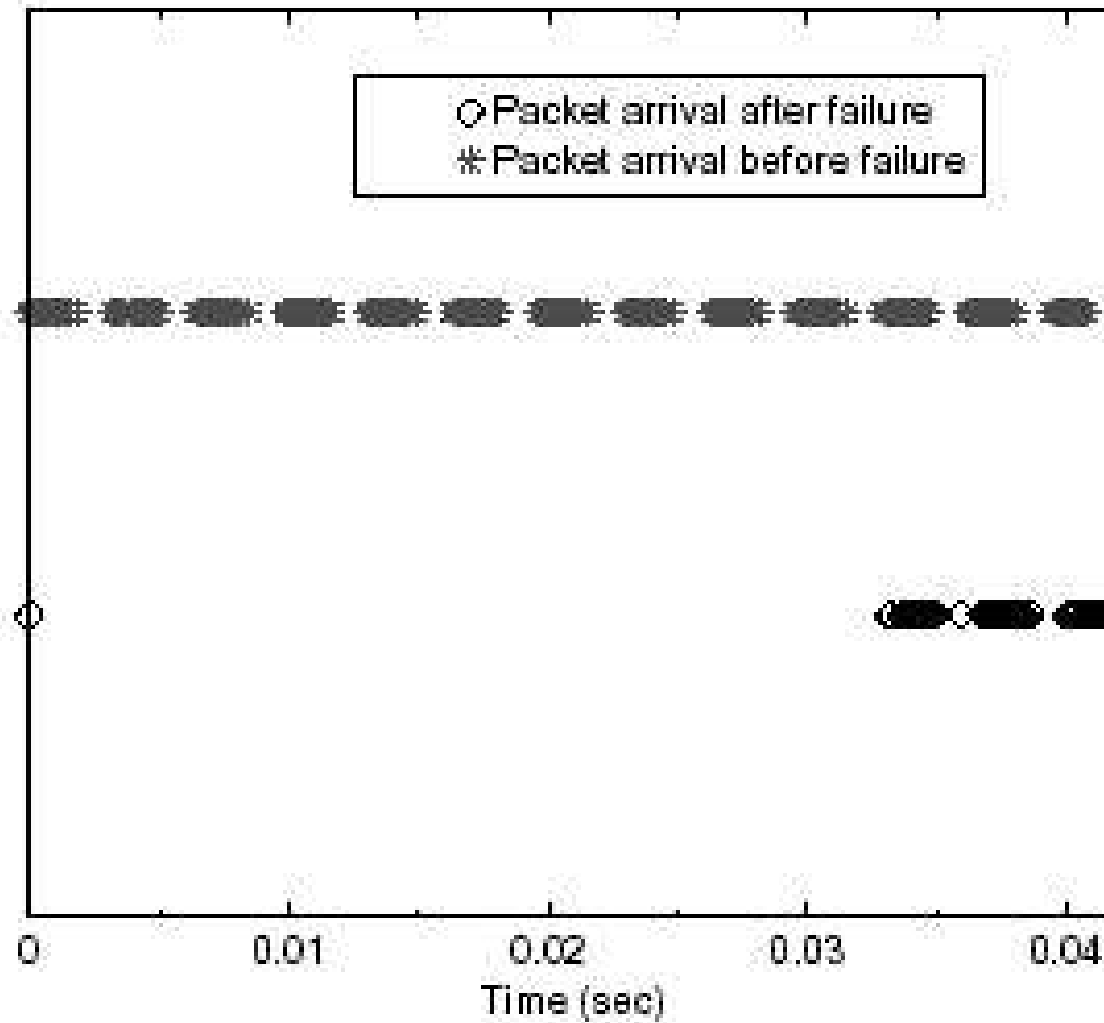
Performance Evaluation

- Empirical evaluation
 - Fault-tolerance
 - 400-600 ms detection
 - < 100 ms fail-over period after detection
 - Effect of redundant links on throughput
 - NFS performance
 - Path selection overhead

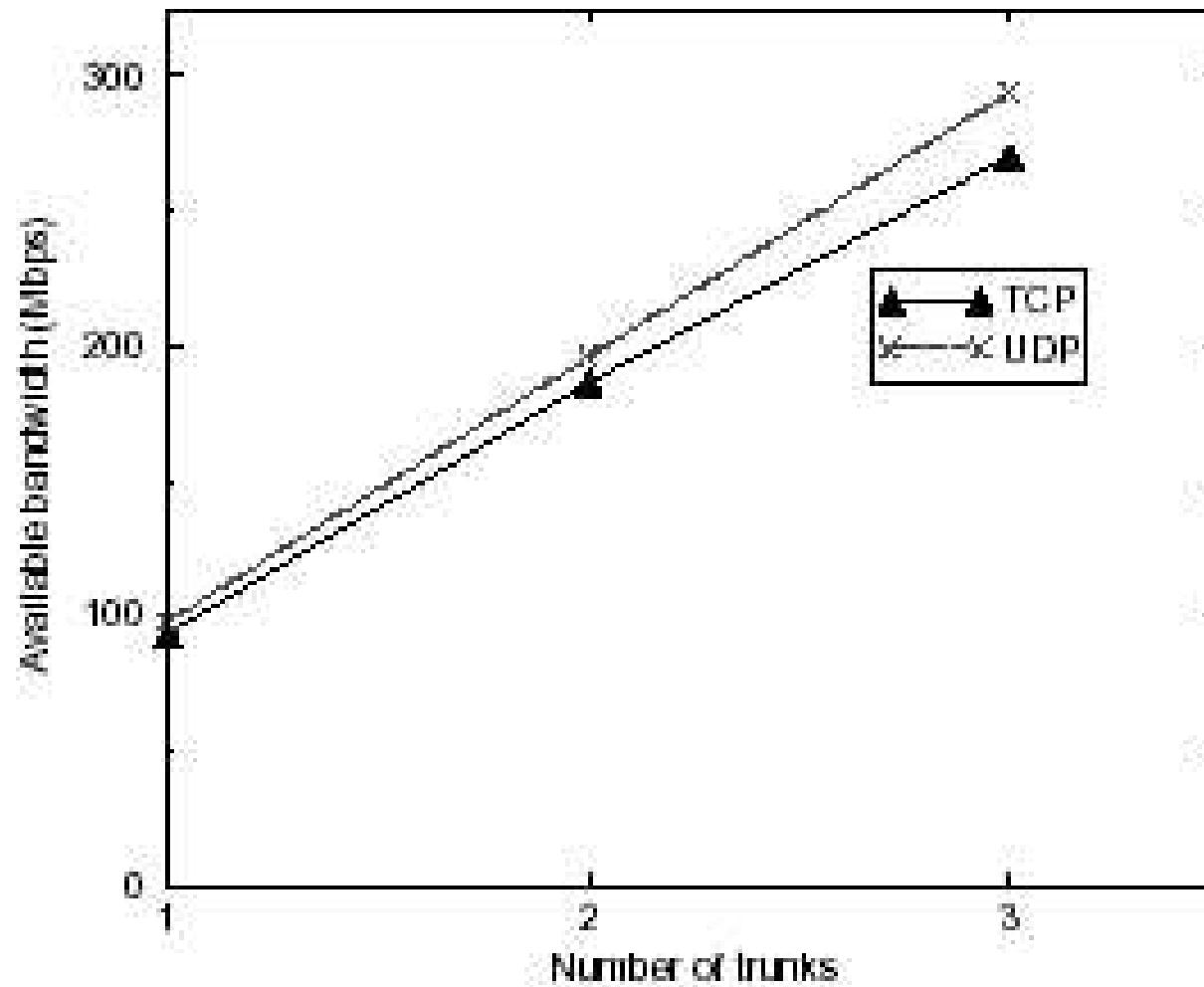
Summary

- Viking gives a multi-spanning-tree architecture for MAN and Cluster Networks
- Makes use of VLANs in a novel way
- Performance increase is multi-fold
- Sub-second level fault-tolerance
- Load characterization aids network tuning

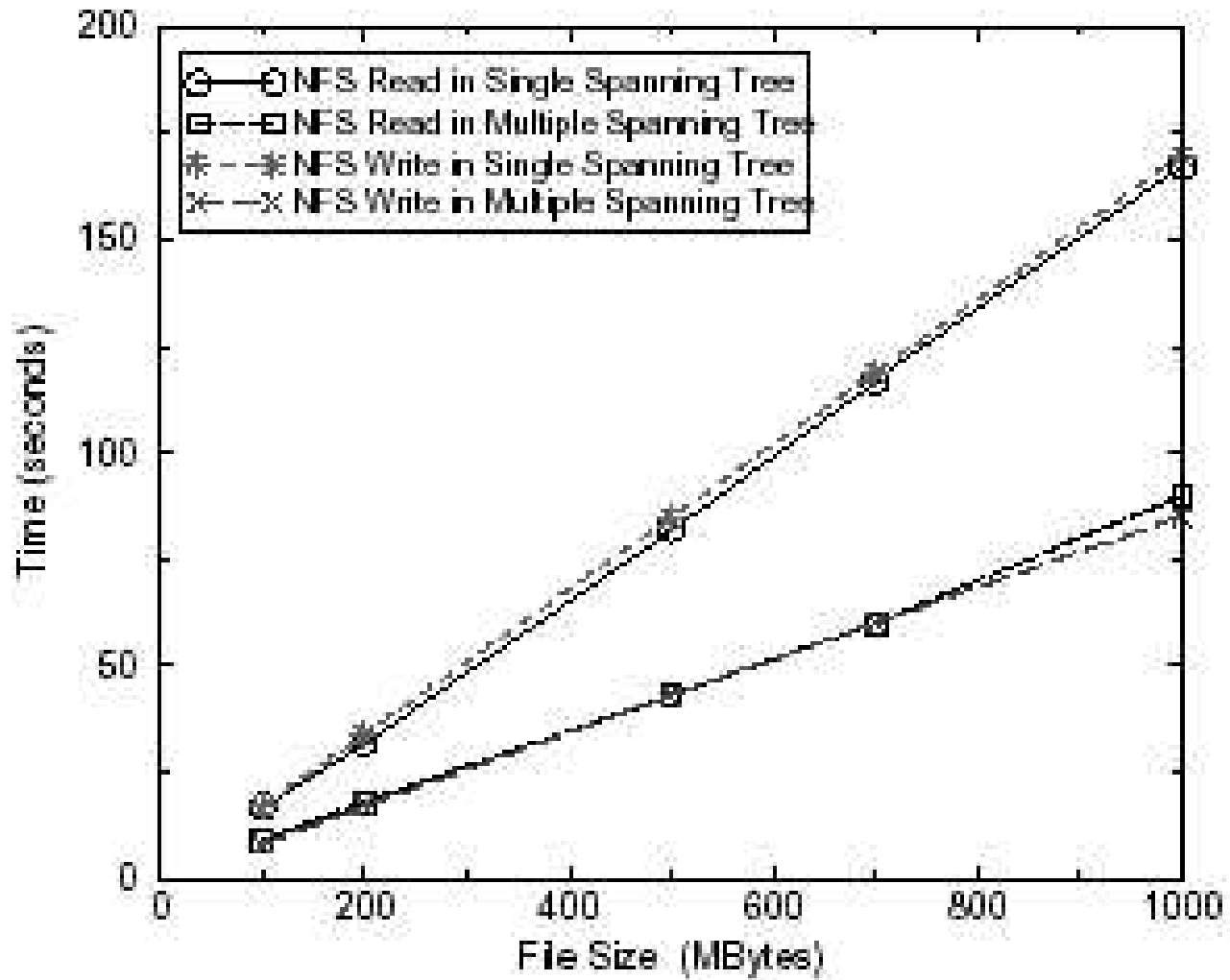
Thanks !!! Questions???



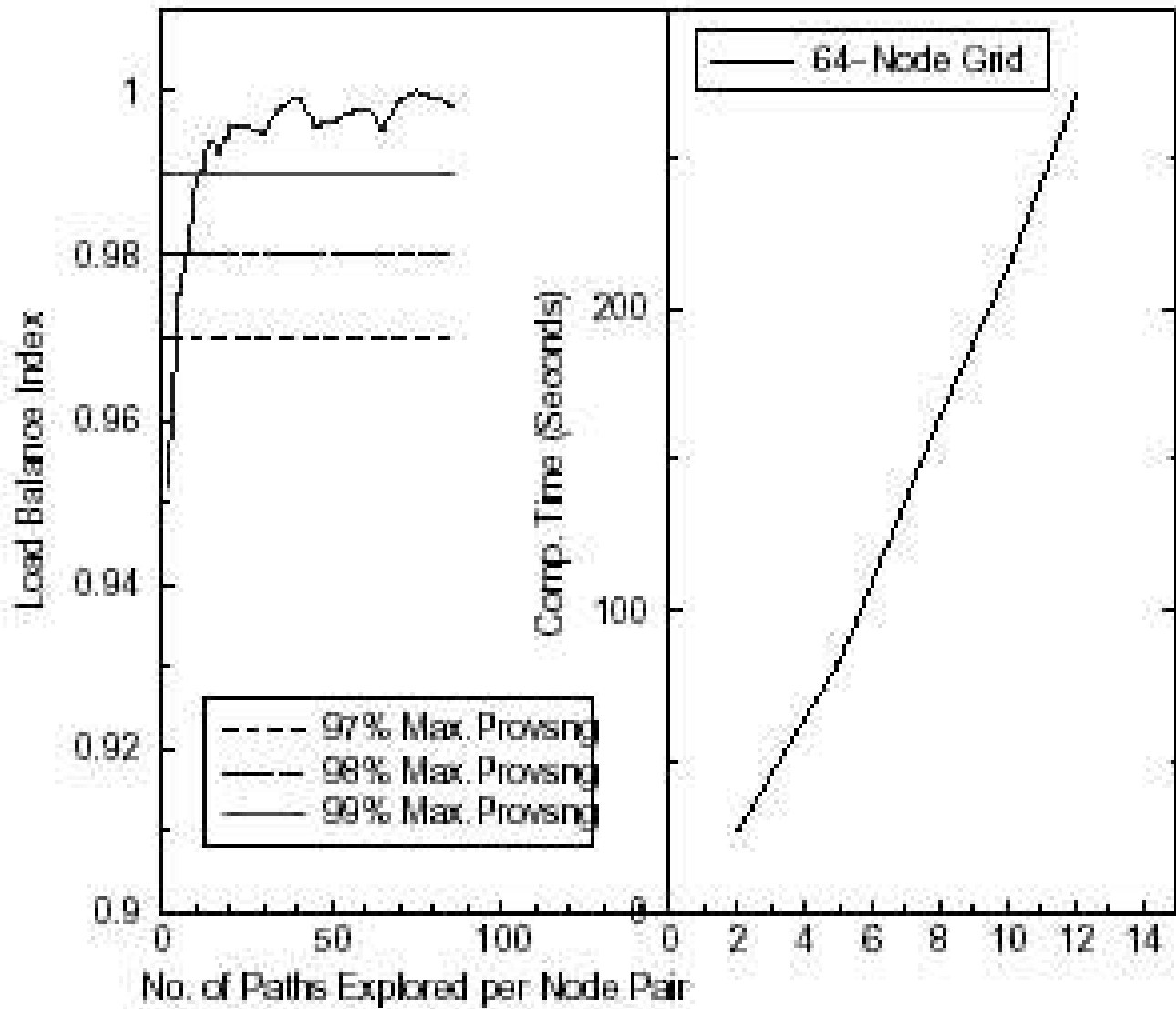
File transfer behavior across failures



Effect of redundant links on throughput



NFS performance Evaluation



Path Computation load

9/28/2004

ECSL, Stony Brook University