

Fat-Tree Overlays for Scalable Multisource Multicast

**Stefan Birrer, Fabián E. Bustamante,
Dong Lu, Peter Dinda and Yi Qiao**
 Department of Computer Science
 Northwestern University

FatTrees for Overlay Multicast

Leiserson's fat-trees

- Minimal and scalable diameter
- Maximal and scalable bisection bandwidth

Resemble Leiserson's fat-trees on the overlay

Data Forwarding in FatNemo

FatTree Resilience

Conventional Trees

- Dependent on reliability of non-leave nodes
- Overlay nodes are highly transient Median session time 1 hr - 1 min.

Forwarding under Failures

Simulation – Failures

1 Publisher, 256 end hosts, MTTF=60 minutes high bandwidth scenario

Protocol	No Failures [%]	With Failures [%]
FatNemo	~1.0	~1.0
Nice	~1.0	~1.0
Narada	~1.0	~0.85
Overcast	~1.0	~0.95
SplitStream	~1.0	~0.7

Handles failures well

The need for efficient group communication

- Multiplayer online gaming
- Video conferencing
- Cooperative Virtual Environments
- Content distribution

Our Approach

Fat Trees for Root Bottleneck Problem

We bypass the root bottleneck problem found in conventional tree-based systems by emulating Leiserson's fat-trees. The resulting overlay fat-trees have peers with higher bandwidth capacity located higher up in the hierarchy.

Our Goal
 Enabling large scale multisource multicast applications.

The Challenge
 Handling highly transient populations and leverage/ respect heterogeneity.

Nemo – Resilient P2P Multicast

Structural robustness through high path diversity [MMCN'05]

PlanetLab – Failures

1 Publisher, 50 end hosts, MTTF=60 minutes

Protocol	No Failures [%]	With Failures [%]
FatNemo	~1.0	~0.95
Nice	~0.9	~0.95
Overcast	~0.5	~0.5
SplitStream	~0.7	~0.7

Simulation – Delivery Latency

1 publisher, 256 end hosts, low bandwidth scenario

Protocol	Mean [s]	Std [s]
FatNemo	~0.2	~0.1
Nice	~0.2	~0.1
Narada	~0.3	~0.2
Overcast	~0.15	~0.1
SplitStream	~0.5	~0.2
MST-BW	~1.1	~0.5

Low delivery latency

Simulation – Delivery Latency CDF

6 publisher, 256 end hosts, low bandwidth scenario

Protocol	Lowest delivery latency with multiple publishers
FatNemo	~0.1
Nice	~0.2
Narada	~0.3
Overcast	~0.4
SplitStream	~0.5
MST-BW	~0.6

Lowest delivery latency with multiple publishers

PlanetLab – Delivery Latency

1 publisher, 50 end hosts

Protocol	Mean [s]
FatNemo	~0.5
Nice	~1.4
Overcast	~0.5
SplitStream	~3.8

Low delivery latency

FatTree Scalability

Scalability of tree-based protocols

cluster degree d of 3 with expected cluster size of 5.5 (Nice, Nemo)

Protocol	$E[\text{Layers}]$	$E[\text{Outdegree}]$
Nice	$\log_{5.5}(N)$	$4.5 E[\text{Layers}] = 4.5 \log_{5.5}(N)$
Nemo	$\log_{5.5}(N)$	$1.5 E[\text{Layers}] = 1.5 \log_{5.5}(N)$
FatNemo	$\frac{-2.5 + \sqrt{6.2 + 8.8 \ln(N)}}{2.2}$	$E[\text{Outdegree}] = \frac{-2.5 + \sqrt{6.2 + 8.8 \ln(N)}}{2.2}$

Scalability of tree-based protocols

cluster degree d of 3

# Peers	$E[\# \text{Layers}]$	$\max E[\# \text{Neighbors}]$
200k	~5	~50
400k	~5.5	~50
800k	~6	~50
1M	~6.5	~50

FatNemo is highly scalable

- Low expected number of layers
- Num. of neighbors grows only slowly

Simulation – Outdegree CDF

8 publisher, 256 end hosts, high bandwidth scenario

Protocol	Small maximal outdegree	Scalability is limited by forwarding responsibility
FatNemo	~0.5	~0.5
Nice	~0.8	~0.8
Narada	~0.7	~0.7
SplitStream	~0.6	~0.6
MST-BW	~0.5	~0.5

Ideally most peers should work in this region

Outdegree defines the forwarding load of the peers

High outdegrees leads to overloaded peers