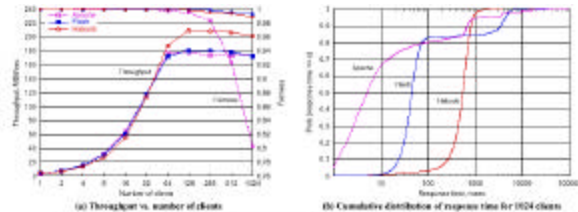




# SEDA web server



	1024 clients				1024 clients			
	Throughput	90% delay	95% delay	99% delay	Throughput	90% delay	95% delay	99% delay
Apache	174.50 req/s	1.53 s	2.00 s	2.50 s	174.50 req/s	1.53 s	2.00 s	2.50 s
Apache	174.50 req/s	1.53 s	2.00 s	2.50 s	174.50 req/s	1.53 s	2.00 s	2.50 s
Flash	174.50 req/s	1.53 s	2.00 s	2.50 s	174.50 req/s	1.53 s	2.00 s	2.50 s
Flash	174.50 req/s	1.53 s	2.00 s	2.50 s	174.50 req/s	1.53 s	2.00 s	2.50 s

Figure 12: **Baseline Web server performance.** This figure shows the performance of the Apache Web server compared to Apache and Flash. (a) shows the throughput of each server using a baseline of 1024 clients as the number of clients increases from 1 to 1024. (b) shows the cumulative distribution function for each server. A server is considered to be faster than the server to its right if its delay is smaller. The above table summarizes the performance of the servers at several points.

# SEDA Summary

- **Structure**
  - Event queues handle high loads well
  - Thread pools allow event handlers to block
- **Framework**
  - General event queue scheduler
  - Controllers to manage queues & threads
    - ◆ Adjust number of threads per stage
    - ◆ Adjust number of events processed at once
    - ◆ Sched load in overload situation
- **Implementation**
  - All written in Java
  - Special non-blocking I/O library

# RONs

■ **“Resilient Overlay Networks”, D. Andersen, H. Balakrishnan, F. Kaashoek, R. Morris, SOSP 2001.**

- Routing around Internet problems
- “Doing what BGP won’t do...”

■ **Motivation**

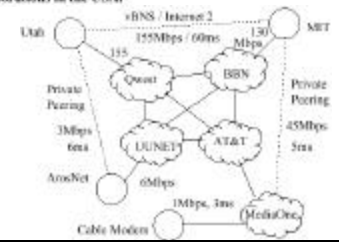
- Enable a group of nodes to communicate with each other despite internet failures
  - ◆ Connect company offices
  - ◆ Connect server sites (replication, load bal, etc...)
  - ◆ Multi-person collaboration/conferencing
- BGP doesn’t suffice
  - ◆ Optimized for large-scale Internet route stability
  - ◆ No performance metrics (just AS hops)

# Set-up

- **Design goals**
  - Failure detection and recovery on <20 secs
  - Application specific failure detection and recovery
  - Expressive routing policy



Figure 1: The current sixteen-node RON deployment. Five sites are at universities in the USA, two are European universities (not shown), three are “broadband” home Internet hosts connected by Cable or DSL, one is located at a US ISP, and five are at corporations in the USA.



# RON Architecture

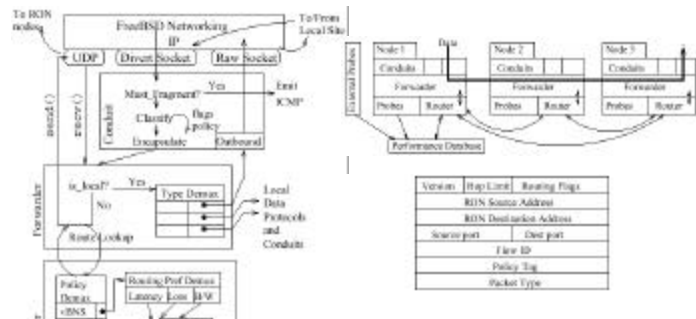


Figure 6: The RON packet header. The routing flags and flow ID are set by the input conduit. The packet type is a demultiplexing key indicating the appropriate conduit or protocol at the receiver.

Loss Rate	RON Win	No Change	RON Loss
10%	557	165	113
20%	168	112	33
30%	131	84	18
40%	110	75	7
50%	106	69	7
60%	100	62	5
70%	93	57	1
80%	87	54	0
90%	85	48	2
100%	67	45	1

# Path outages

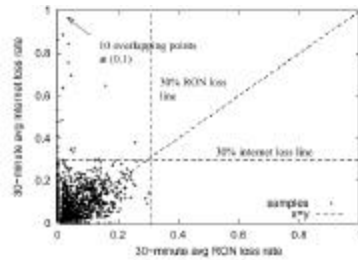
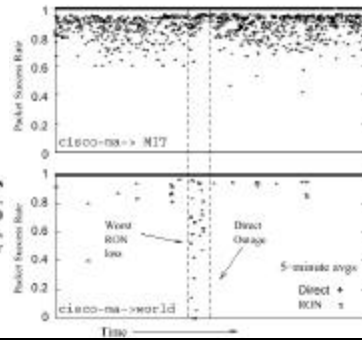


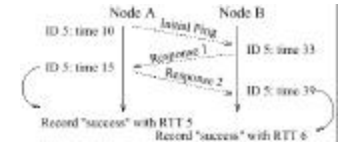
Figure 9: Packet loss rate averaged over 30-minute intervals for direct Internet paths vs. RON paths for the *RON* dataset. There are 32 points above the  $p = 0.3$  horizontal line, and 20 points above  $p = 0.5$ , including overlapping points. In contrast, RON's loss optimizing router avoided these failures and never experienced a 30-minute loss-rate larger than 30%.



# Monitoring

## Failure Detection

- UDP pings
- Low frequency (12 secs)
- Hi freq if failure (3 secs)



## Latency estimation

- 3-packet RTT meas.
- One-way vs. RTT

$$lat_i \leftarrow \alpha \cdot lat_i + (1 - \alpha) \cdot new\_sample_i \quad (1)$$

We use  $\alpha = 0.9$ , which means that 10% of the current latency estimate is based on the most recent sample. This number is similar to the values suggested for TCP's round-trip time estimator [20]. For a RON path, the overall latency is the sum of the individual virtual link latencies:  $lat_{path} = \sum_{i \in path} lat_i$ .

## Loss estimation

- Avg loss of last 100 probes

$$loss = \frac{\sqrt{13}}{rtt \cdot \sqrt{p}} \quad (2)$$

## Throughput estimation

- Calculate from latency & loss
- Shortest-path not necessarily best

where  $p$  is the one-way end-to-end packet loss probability and  $rtt$  is the end-to-end round-trip time estimated from the hop-by-hop samples as described above.

# Latency & Throughput

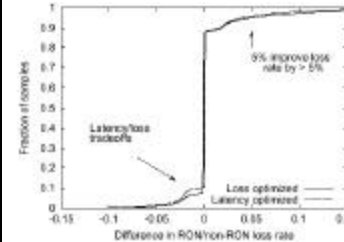


Figure 11: The cumulative distribution function (CDF) of the improvement in loss rate achieved by RON. The samples detect unidirectional loss, and are averaged over  $\tau = 1800$  intervals.

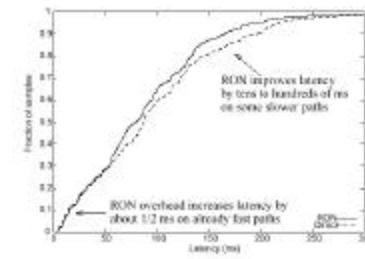


Figure 13: 5-minute average latencies over the direct Internet path and over RON, shown as a CDF.

## Summary

### ■ Overlay Network can Improve connectivity

- Within 20 seconds
- Majority with one extra hop

Hysteresis	# Changes	Avg	Med	Max
0%	26205	19.3	3	4607
5%	21253	24	5	3438
10%	9436	49	10	4607
25%	4557	94	17	5136
50%	2446	138	25	4703
Random process	260,909	2	1	<.16

Table 5: Number of path changes and run-lengths of routing persistence for different hysteresis values.

### ■ Items to consider

- Routing policies
  - ◆ Means to specify which traffic is allowed to use which links
- Application-specific optimization metrics
  - ◆ "total outage" vs.
  - ◆ Packet loss
  - ◆ Latency
  - ◆ Throughput

13