Allocating Network Bandwidth to Match Business Priorities

Speaker

Peter Sichel
Chief Engineer
Sustainable Softworks
psichel@sustworks.com

MacWorld San Francisco 2006
Session M225
12-Jan-2006 10:30 AM - 12:00 PM

Introduction
Speaker Background

Macintosh developer specializing in kernel level TCP/IP networking for the past 10 years.

- IPNetMonitorX - Macintosh internet tools
- IPNetTunerX - network optimization tuner
- IPNetRouterX - native IP connectivity suite
- 14 years prior experience as software architect at Digital Equipment Corporation. Emphasis on user interface design and communications.

Overview

What is this session about:

- Managing Increasing Network Demand
- Improving network throughput and availability
- Tools and techniques for addressing difficult performance problems
- What happens when you push TCP/IP to the extreme?

Bandwidth Allocation

- Packet Shaping and QoS (Quality-of-Service)
- Bandwidth Allocation Examples
- Using proxies for content caching
- Dead Gateway Detection with automatic failover

TCP Tuning

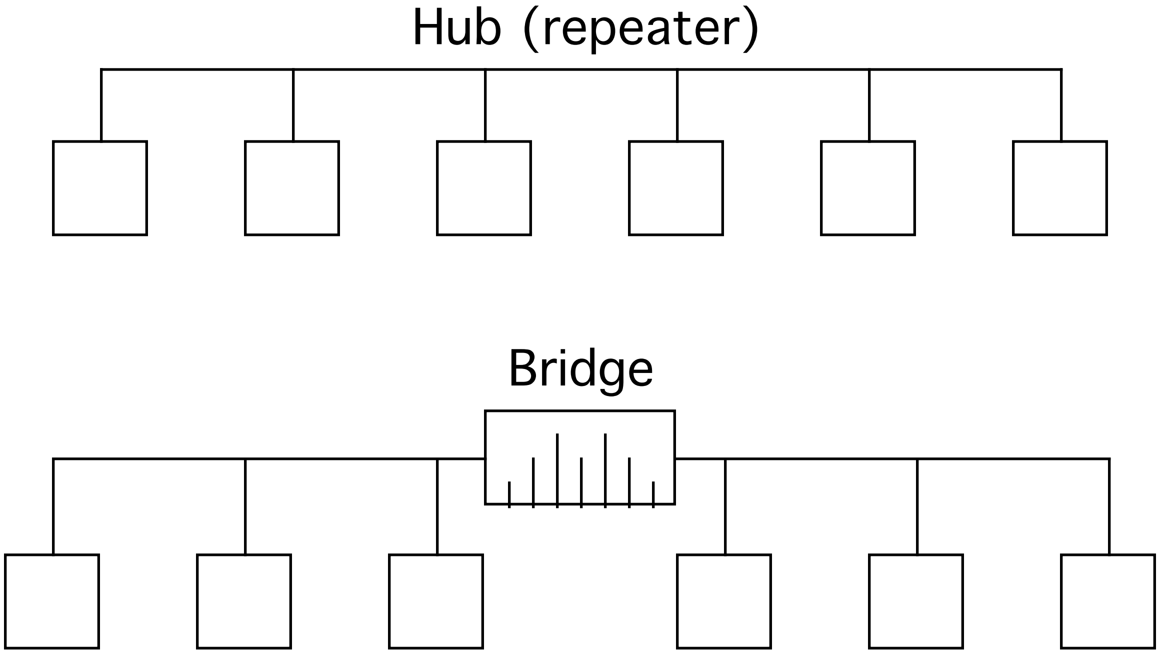
Managing Network Demand
General Techniques

- Higher speed links
  Great when available, but not always practical or cost effective
  Beware of stimulating demand

- Partition network to isolate traffic
  - Bridging / Routing / Shaping
  - Allocate or reserve bandwidth
  - Resource location and content caching
  - Police unwanted traffic

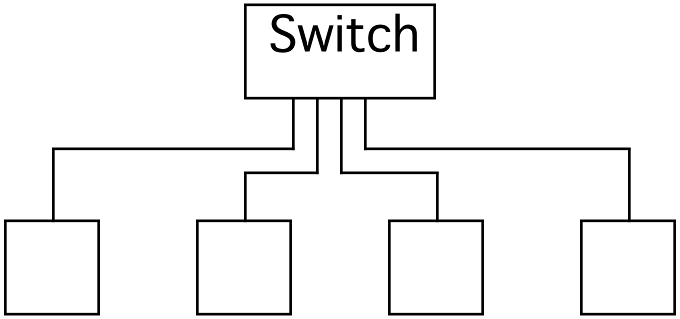
- Tune for performance
  Increasingly important for high speed links

Partition to Isolate Traffic

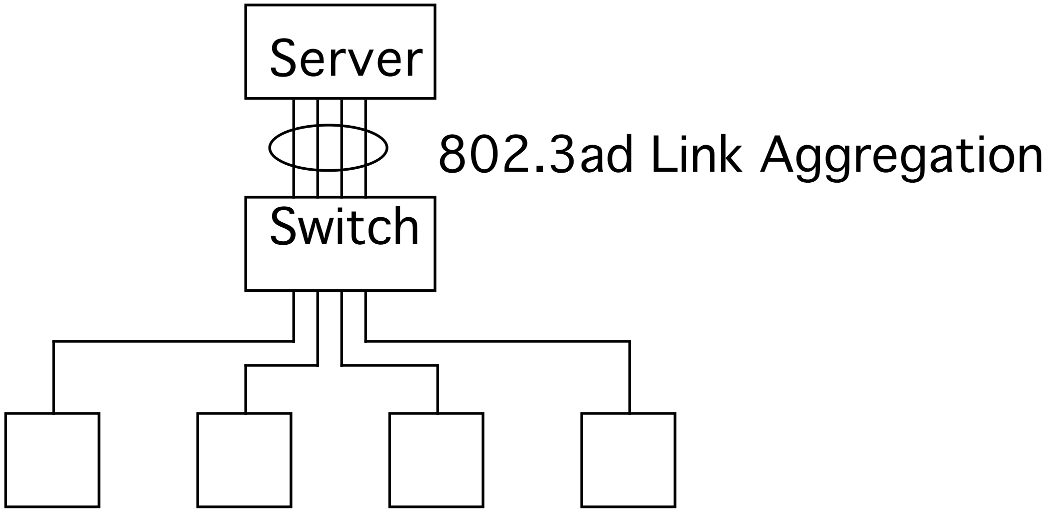


Look at Ethernet frame header

Partition (switching)



Switch

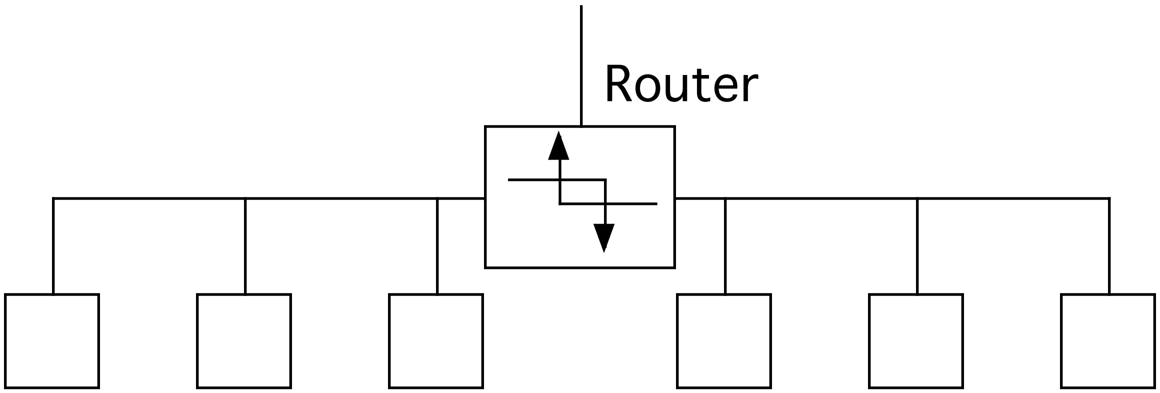


Server

802.3ad Link Aggregation

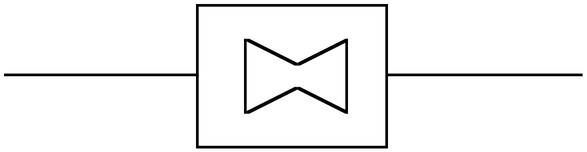
Switch

Partition (routing)



Router

Look at IP header



Shaper

Look at TCP/UDP header to classify and rate limit data flows

Packet Shaping and QoS (1)

Each TCP connection tries to use as much bandwidth as it can get, and backs off when packets are lost or delayed.

When a data link becomes heavily saturated by multiple competing applications:

- ACKs can be delayed limiting throughput
- Efficiency drops as data is retransmitted
- Connections back off to avoid congestion

The purpose of Packet Shaping and QoS is to adapt network behavior to incorporate your traffic priorities and increase TCP/IP efficiency over the “slow link” or “last mile”.

Packet Shaping and QoS (2)

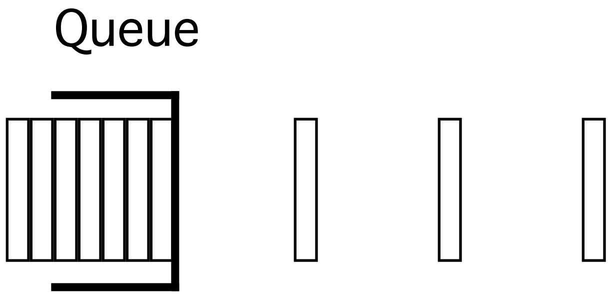
Three basic techniques:

- Queuing (buffering and waiting) to limit send rate. Since TCP is adaptive, it will self adjust to the impaired data link.
  `Examples: throttled, LinkSys WRT-54, Xincom, WonderShaper (Linux), pf and ALTQ (FreeBSD)`

- Expedite higher priority traffic by sending out of order or prioritizing among queues. Handle ACK prioritization as a special case.
  `Example: throttled, pf and ALTQ`

- Use TCP's built-in flow control mechanism to modulate the send rate for maximum efficiency.
  `Examples: Packeteer, IPNetRouterX (IPNetSentryX)`
  `No open source implementation at this time`

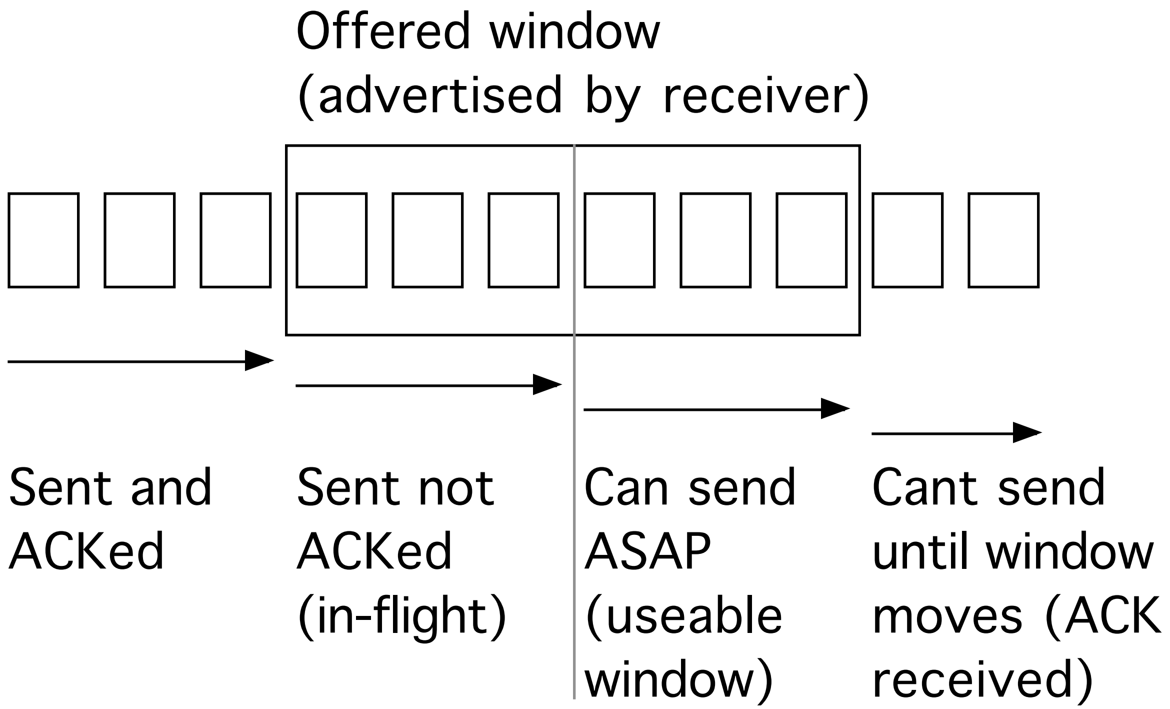
Packet Shaping Techniques (1)



Queuing (buffering and waiting)

- Easiest to implement, can be applied to any protocol
- Only rate limits send traffic (directly)
- Limited number of queues (compared to TCP itself)
- “Shaping” amounts to delaying, re-ordering, and dropping packets
- Important to prioritize empty TCP ACKs.
- Can be very effective depending on the problem, but not as powerful.

Packet Shaping Techniques (2)



Offered window
(advertised by receiver)

Sent and
ACKed

Sent not
ACKed
(in-flight)

Can send
ASAP
(useable
window)

Cant send
until window
moves (ACK
received)

TCP Rate Limiting

- Adjust receive window on the fly to control when and how much the window moves.
- Can rate limit send and receive traffic.
- Avoids congestion for best use of available bandwidth.

Implementation Matters
Network Stack Order in Tiger

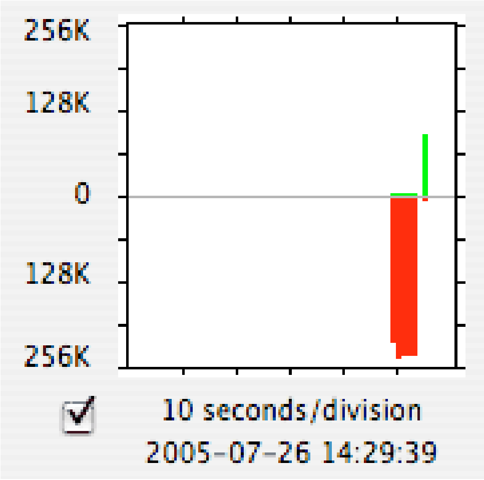
TCP/UDP

IP filter (NKE)
    IPSec (if enabled)
    IP filter (if IPSec enabled)
fragment reassembly
IP forwarding
ipfw (including divert used for natd, throttled)

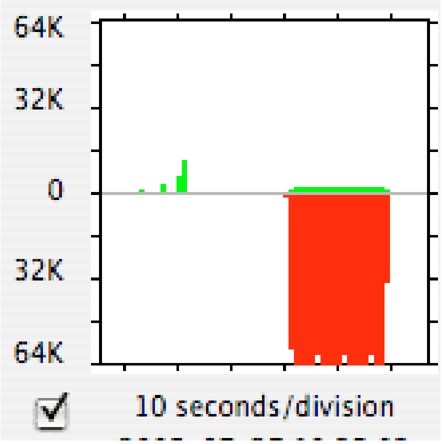
protocol and interface filters (NKE)

bpf tap (IOKit)
IOKit driver

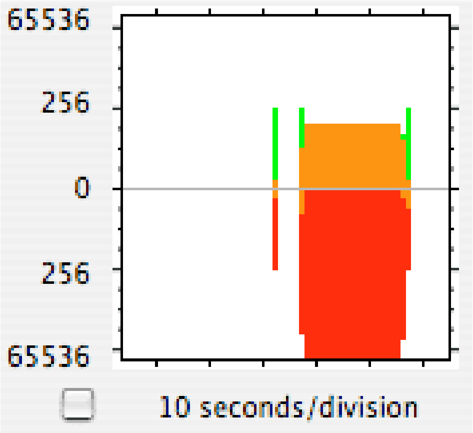
Packet Shaping Example

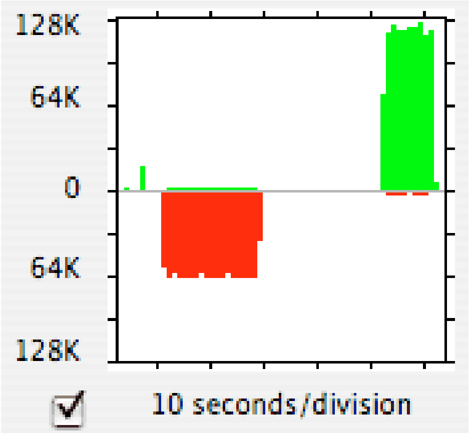


Send 1.1 MB file at 2 Mpbs



512 Kbps send limit



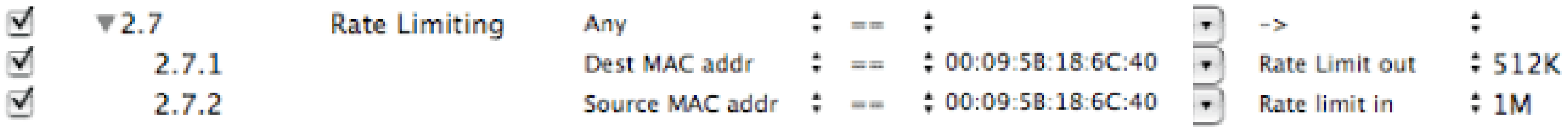


1 Mbps receive limit

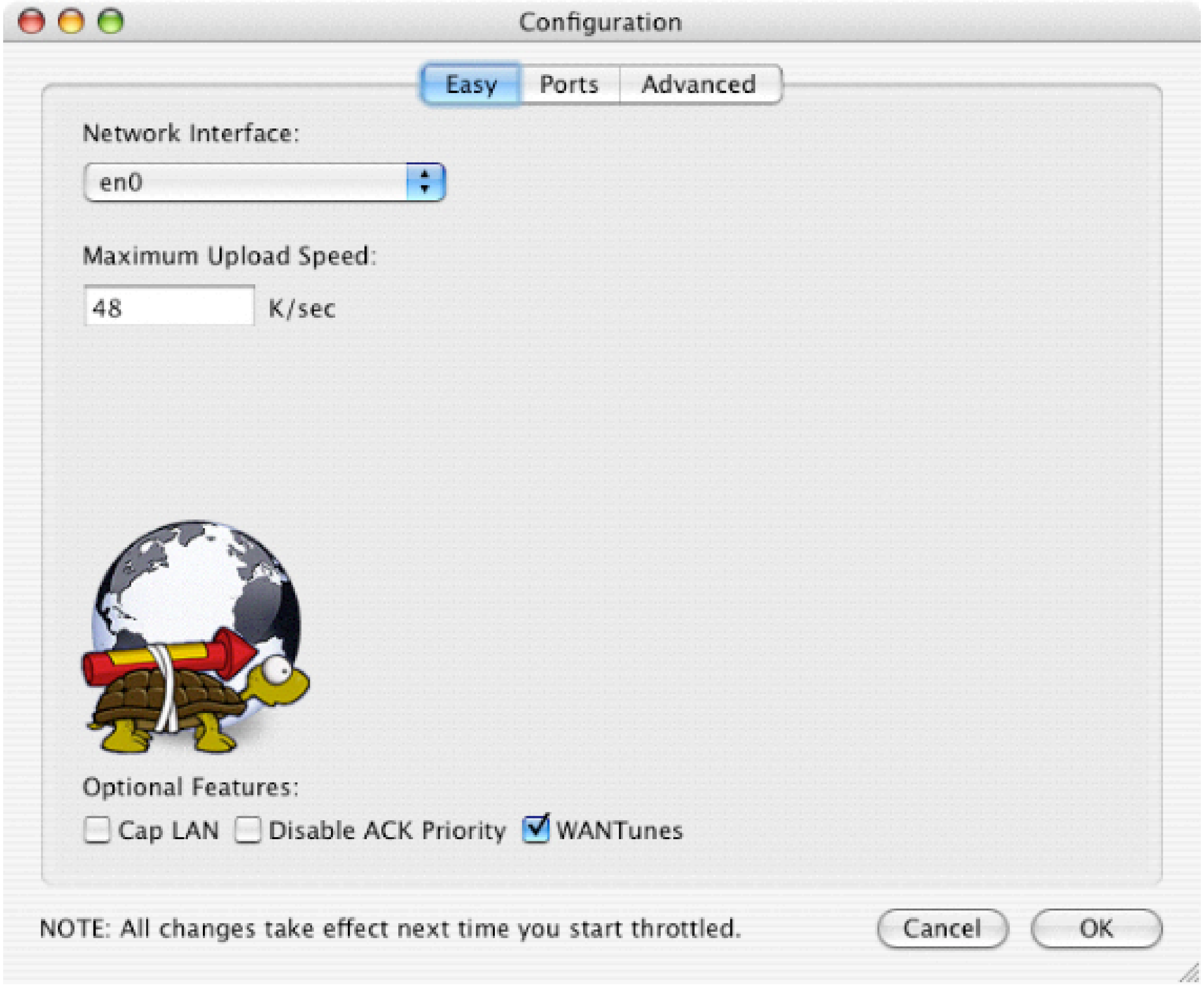
Bandwidth Allocation (1)

Classify and monitor traffic to identify data flows of interest.

- Devices (Internet router, server, host,...)
- Services (Web, Email, File Transfer, VOIP)



☑	▼2.7	Rate Limiting	Any	==		->	
☑	2.7.1		Dest MAC addr	==	00:09:5B:18:6C:40	Rate Limit out	512K
☑	2.7.2		Source MAC addr	==	00:09:5B:18:6C:40	Rate limit in	1M



Network Interface:

en0

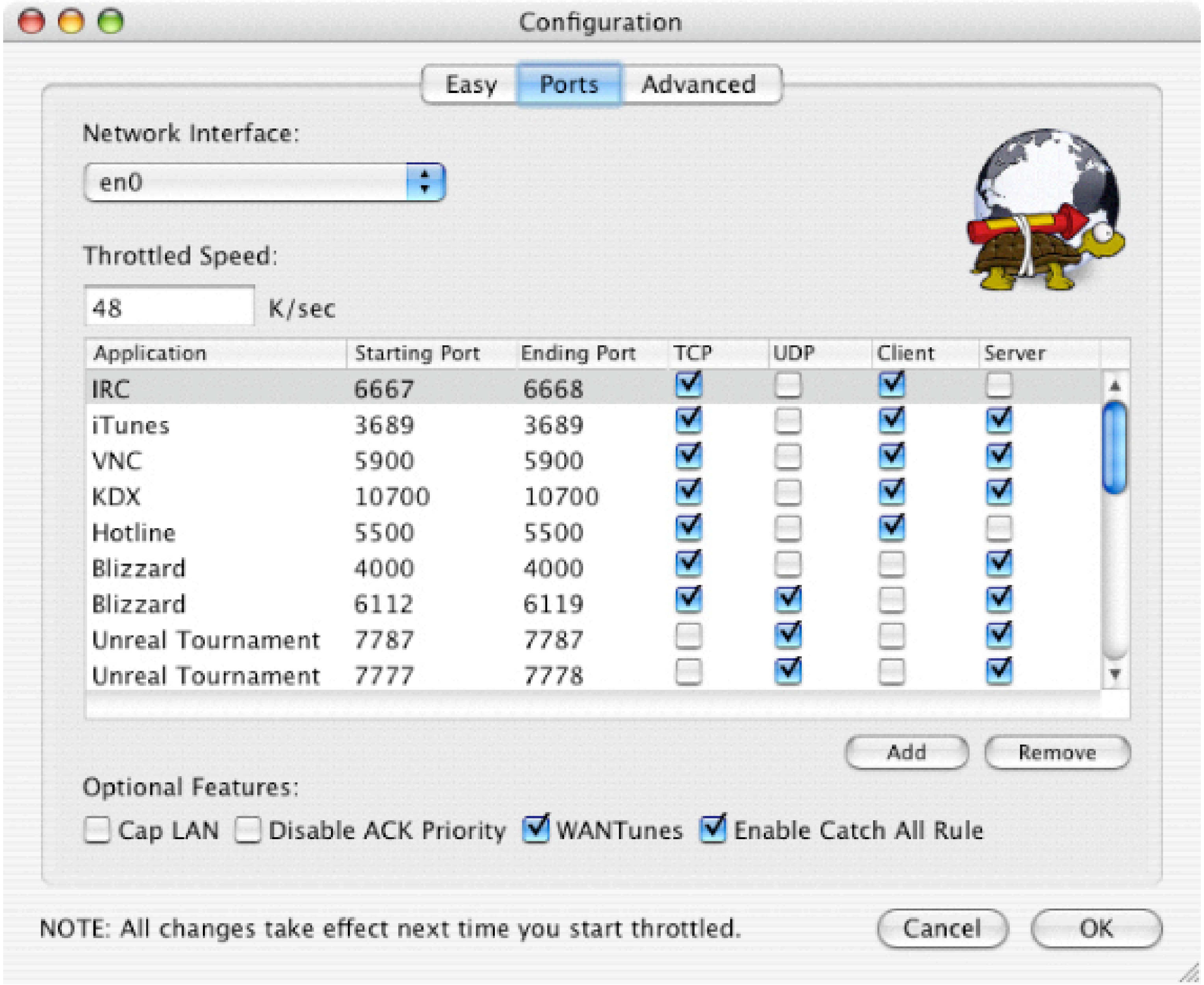
Maximum Upload Speed:

48 K/sec

Optional Features:

□ Cap LAN □ Disable ACK Priority ☑ WANTunes

NOTE: All changes take effect next time you start throttled.



Network Interface:

en0

Throttled Speed:

48 K/sec

Application	Starting Port	Ending Port	TCP	UDP	Client	Server
IRC	6667	6668	☑	□	☑	□
iTunes	3689	3689	☑	□	☑	☑
VNC	5900	5900	☑	□	☑	☑
KDX	10700	10700	☑	□	☑	☑
Hotline	5500	5500	☑	□	☑	□
Blizzard	4000	4000	☑	□	□	☑
Blizzard	6112	6119	☑	☑	□	☑
Unreal Tournament	7787	7787	□	☑	□	☑
Unreal Tournament	7777	7778	□	☑	□	☑

Optional Features:

□ Cap LAN □ Disable ACK Priority ☑ WANTunes ☑ Enable Catch All Rule

NOTE: All changes take effect next time you start throttled.

Bandwidth Allocation (2)
What Problems does this solve

Maintain network responsiveness during heavy Email or file transfers.

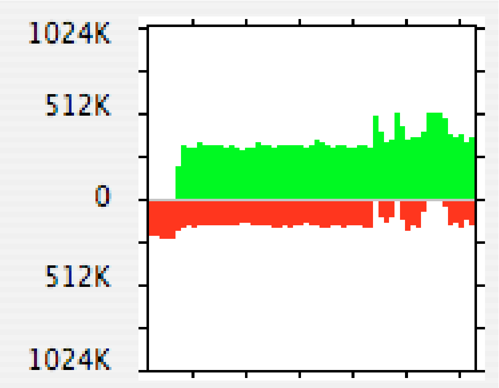
Improve network performance during simultaneous upload and download.

Improve server responsiveness when frequently accessed.

Reserve bandwidth for VOIP.

Good fences make good neighbors.

Simultaneous Upload & Download



Cable modem 5M up, 2M down.
First half shows rate caps at 4M up, 2M down.
Second half with rate caps turned off.

Server Pacing

Consider a web server behind a T1 line (1.544 Mbps) that recieves two requests for a large web page.

On Mac OS X, the TCP send window defaults to 32K.

The sender ramps up to 32K in flight for each request.

At T1 speeds it takes over 300ms to send the 64K in flight, any other requests will need to wait behind this backlog.

With rate limiting, the total send window is limited to 1/10 the specified rate limit. If we use 1.5 Mbps, the total send window will be 20K or about 100ms.

The server still responds immediately and pages arrive just as quickly. We just parcel out the data more slowly to avoid creating a backlog.

Reserve Bandwidth for VOIP

Does your VOIP phone sound worse than your cell phone?

If downstream is saturated by other file transfers, need to slow these down.

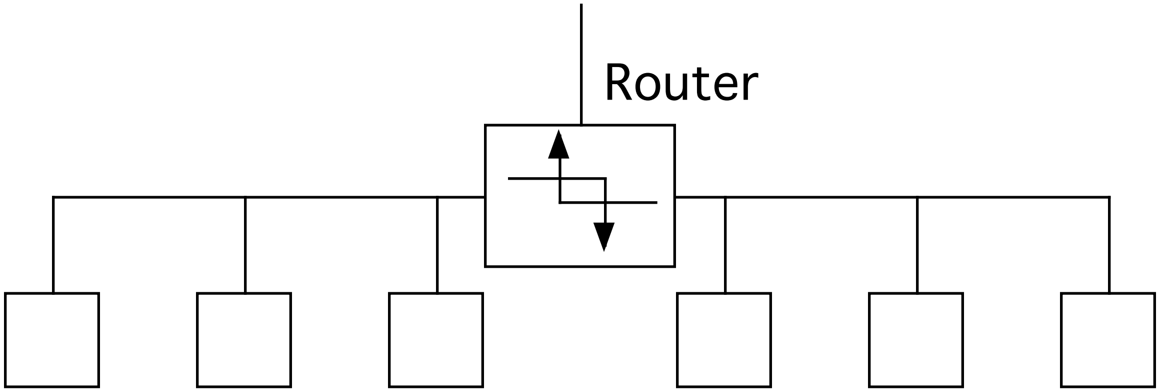
Simple queue and drop won’t work since the traffic is inbound and we need to slow the individual data flows, not outbound packets

Good Fences Make Good Neighbors

Want to share your WiFi but limit how much bandwidth it actually gets?

Do your kids sometimes hog your connection?

Back to Partitioning



The intent of partioning is to keep unwanted traffic out of the way.

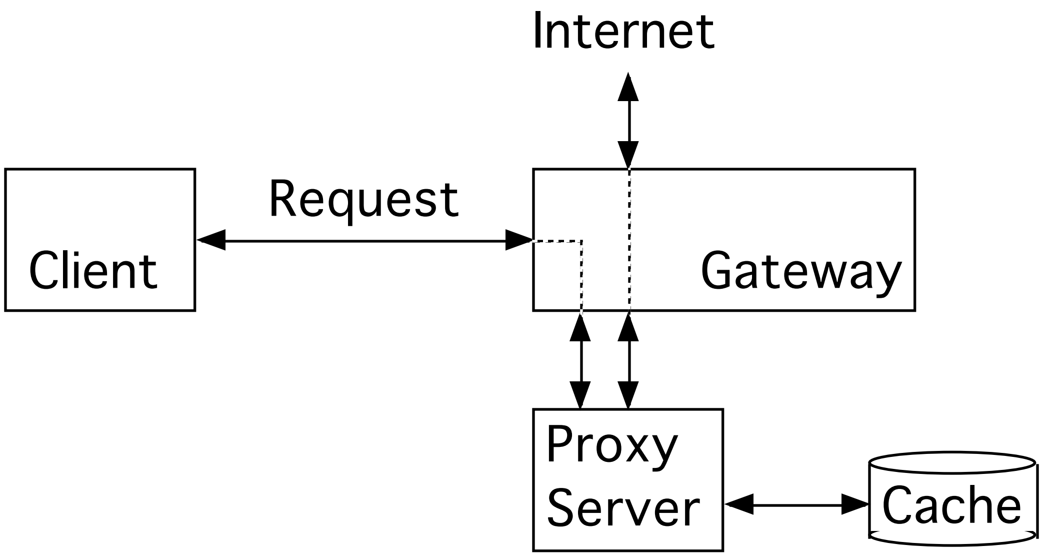
Another way of doing this is to relocate resources to where they are needed and police unwanted traffic.

Transparent Proxy

Redirect client requests without any special configuration or knowledge at the client.

Cache frequently requested pages or files to reduce WAN traffic.

Police even the most aggressive peer-to-peer protocols.

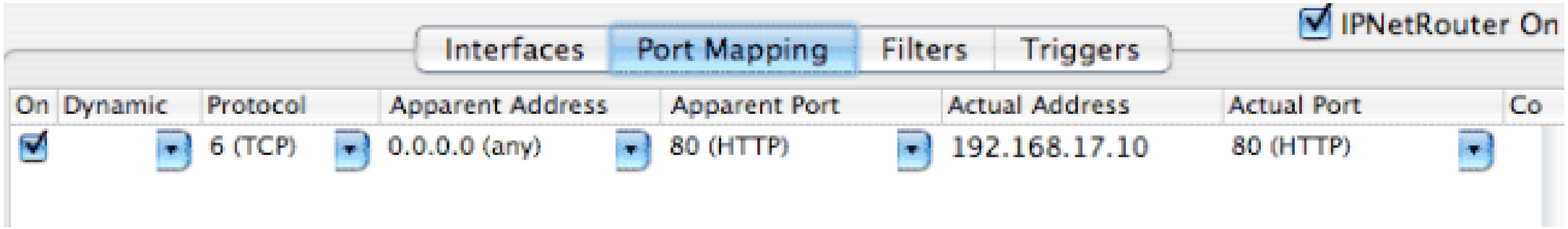


Content Caching

Squid is the most popular proxy server in use today and even has Macintosh friendly front ends such as SquidMan or Maxum iAssist to keep installation and configuration simple.

Typically, caching can save 20-30% of the bandwidth and improve browsing speeds, especially over slow or congested network connections.

UNIX bind supports local caching DNS server. See “bind9arm” for details or use a GUI front end.

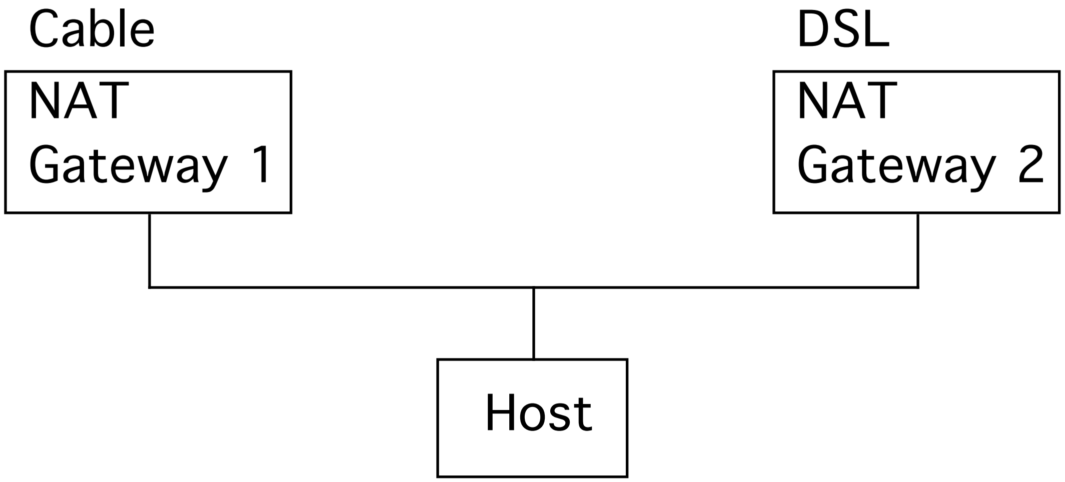


☑ IPNetRouter On

On	Dynamic	Protocol	Apparent Address	Apparent Port	Actual Address	Actual Port	Co
☑		6 (TCP)	0.0.0.0 (any)	80 (HTTP)	192.168.17.10	80 (HTTP)	

Dead Gateway Detection

- Similar to Windows behavior
- When a connection attempt is retransmitted for the 3rd time, select an alternate gateway by creating a new host route on the fly.
- If more than half of active connections have failed over to alternate gateway, make it the default gateway by selecting a new location.
- Transparent fail over before connection attempt times out.



Tuning for Performance
Why is it necessary to tune TCP/IP?

- TCP/IP is not controlled by any manufacturer and is designed to work on almost any kind of underlying network with widely varying characteristics.

- To meet this challenge, the protocol designers made TCP adaptive. TCP is self monitoring and optimizes its own behavior to match the network environment. Adaptation takes time and the default settings cannot optimize for every possible kind of network.

- There are many independent implementations, the protocols continue to evolve in response to practical experience, some implementations have peculiar compatibility constraints.

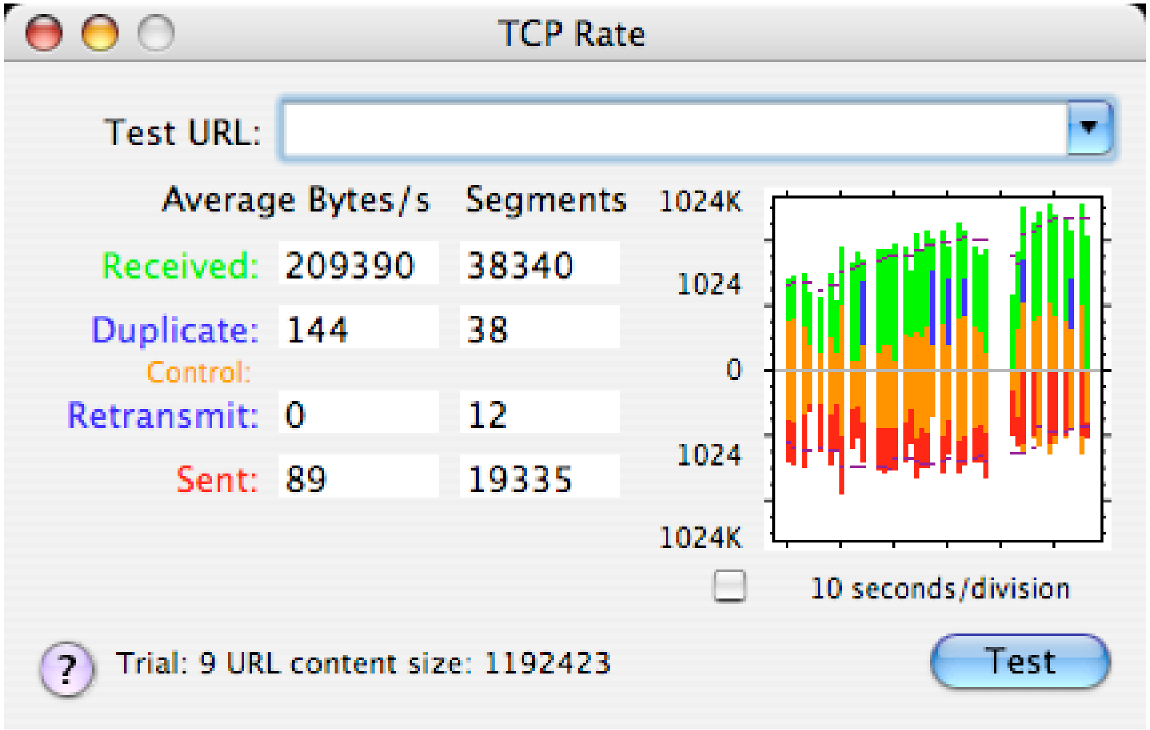
Typical Tuning Problems

- Asymmetric connections that are much faster downstream than upstream.

- High performance connections with relatively long latency (measured in bytes) such as fiber optic, satellite, or PPP via cellular network.

- PPPoE implementations that restrict the network MTU without support for “Path MTU Discovery” or fragmentation.

Optimization 101 - measure, adjust, repeat



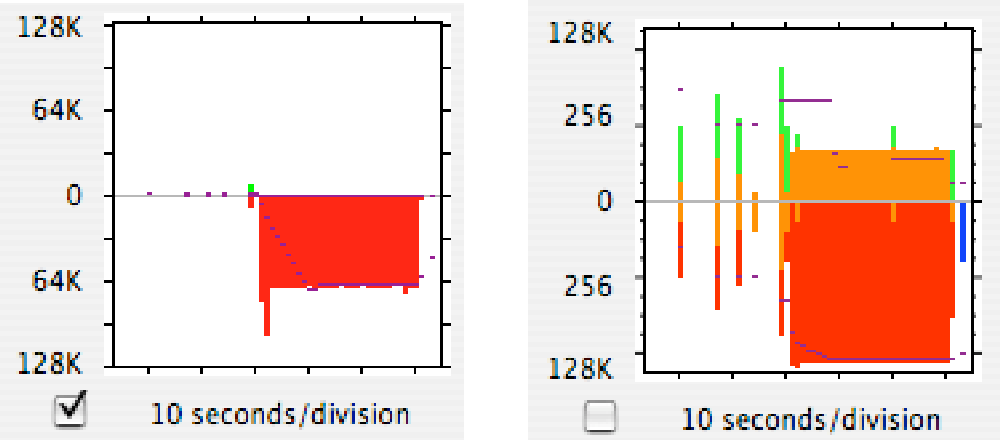
TCP Rate

Test URL:

	Average Bytes/s	Segments
Received:	209390	38340
Duplicate:	144	38
Control:		
Retransmit:	0	12
Sent:	89	19335

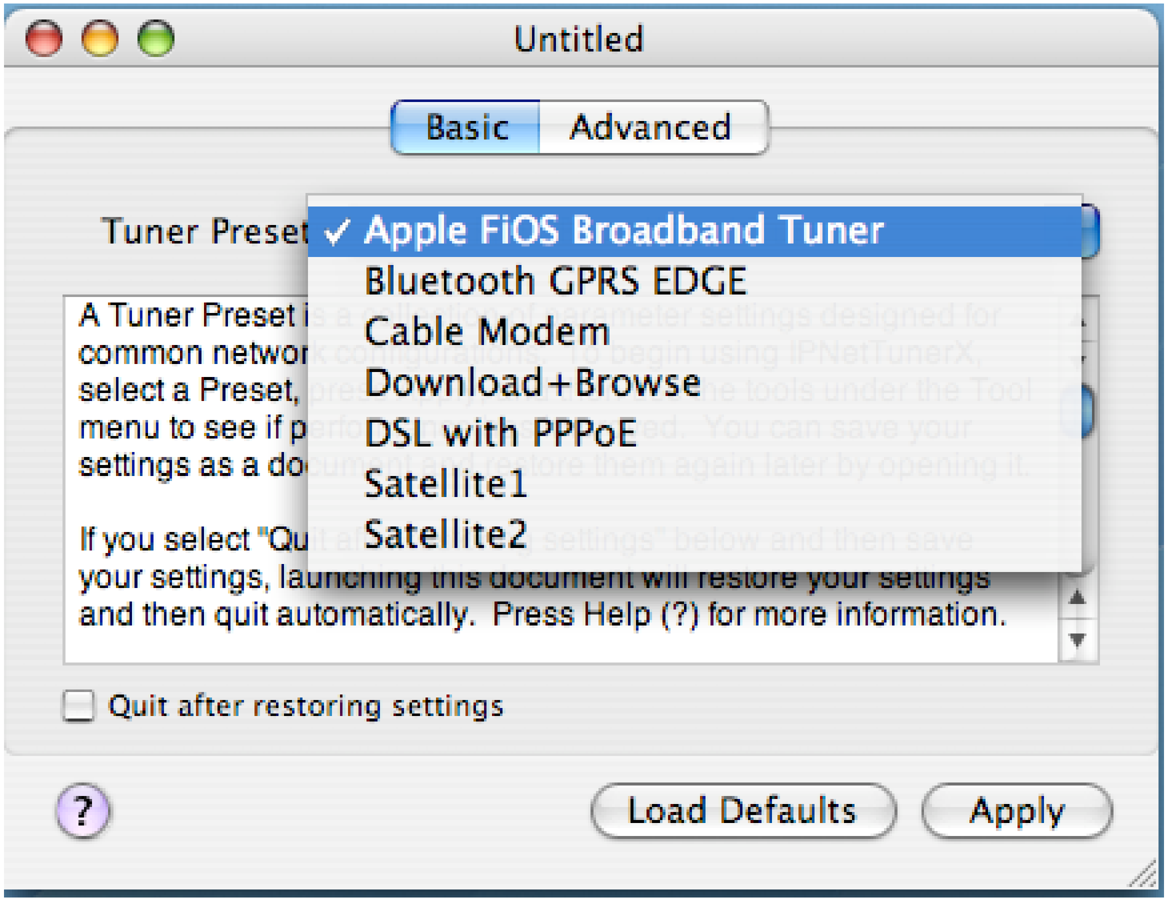
Trial: 9 URL content size: 1192423

Problematic transfer with duplicate/retransmit data



Efficient transfer (Linear and Log scale)

Don’t need to find perfect settings, just get close enough that TCP can work efficiently.

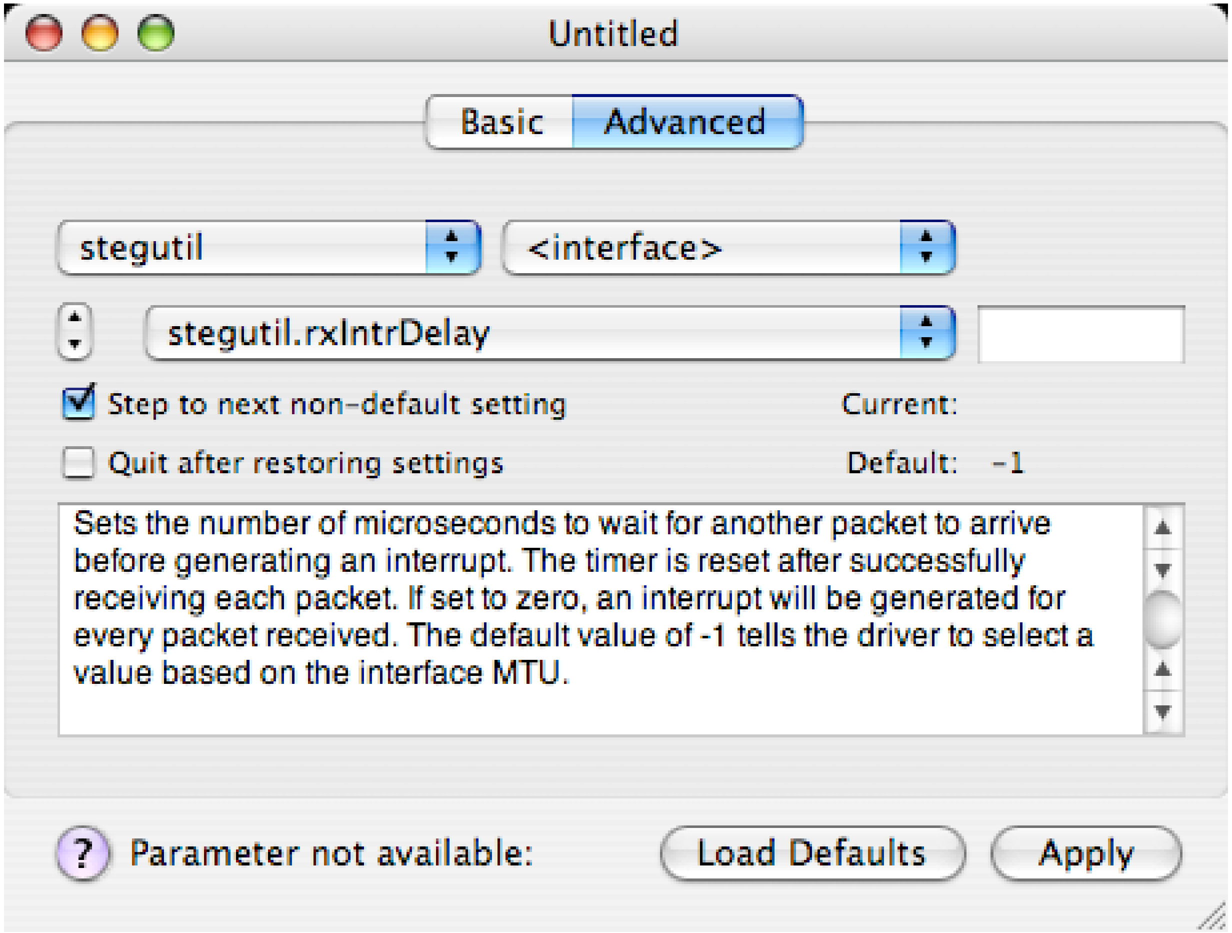


Tuner Preset ✓ Apple FiOS Broadband Tuner

A Tuner Preset is a collection of parameter settings designed for common network configurations. To begin using IPNetTunerX, select a Preset, press Apply, and then use the tools under the Tool menu to see if performance has improved. You can save your settings as a document and restore them again later by opening it.

If you select "Quit after restoring settings" below and then save your settings, launching this document will restore your settings and then quit automatically. Press Help (?) for more information.

□ Quit after restoring settings



stegutil

<interface>

stegutil.rxIntrDelay

☑ Step to next non-default setting

□ Quit after restoring settings

Current:

Default: -1

Sets the number of microseconds to wait for another packet to arrive before generating an interrupt. The timer is reset after successfully receiving each packet. If set to zero, an interrupt will be generated for every packet received. The default value of -1 tells the driver to select a value based on the interface MTU.

Parameter not available:

Q & A

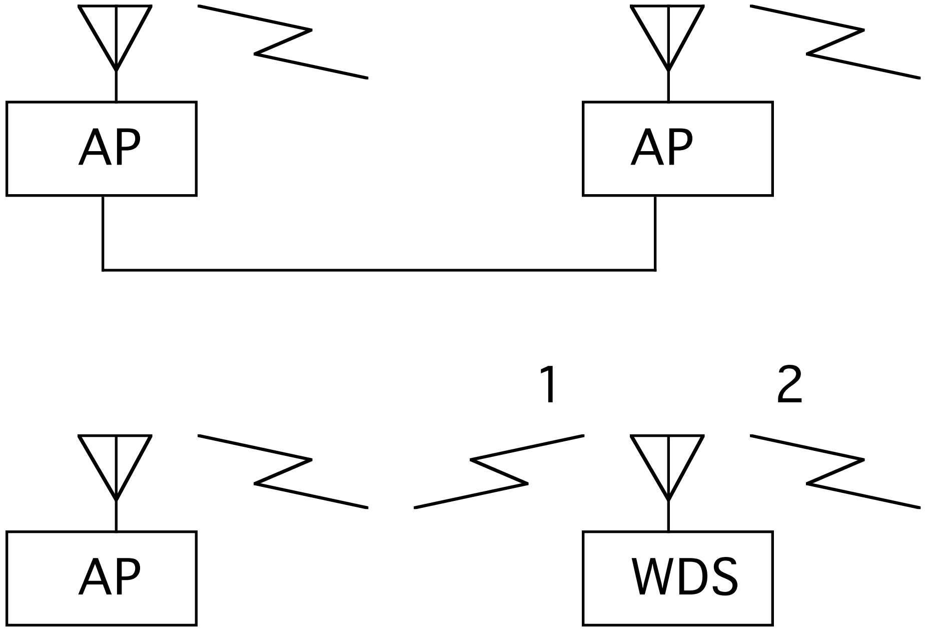
More Information:

http://www.sustworks.com
psichel@sustworks.com

Updated Slides:

ftp://sustworks.com/MWSF06_M225.pdf
(or send Email)

Partition (wireless)



Switching is not widely available.
As power or cell size increases, so does congestion
(total available bandwidth decreases)

WDS - time division relay using a single radio