Core OS Networking

In-depth

Session 205

Anil Vempati
Manager, Core OS Networking

Session Overview

- IPv6
  - IPv6 at a glance
  - Improvements in iOS 5 and Mac OS X Lion
- Changes under the hood
  - Same APIs, different behavior
  - Backgrounded connections
- Tools
  - For network simulation and troubleshooting

IPv6 in iOS and Mac OS X

James Woodyatt
Engineer, Core OS Networking

"So, I said, 'Thirty-two bits. It is enough for an experiment.' The problem is, the experiment never ended."

Vinton G. Cerf, Internet Patrician

Current Events in the Industry

No free IPv4 addresses left

- Today is World IPv6 Day
- ISP dual-stack trials
- Home gateways and other gear

IPv6 at a Glance

Major Changes

Not many, but there are some…

- Universal routing realm
- Internet Control Message Protocol, version 6 (ICMPv6)

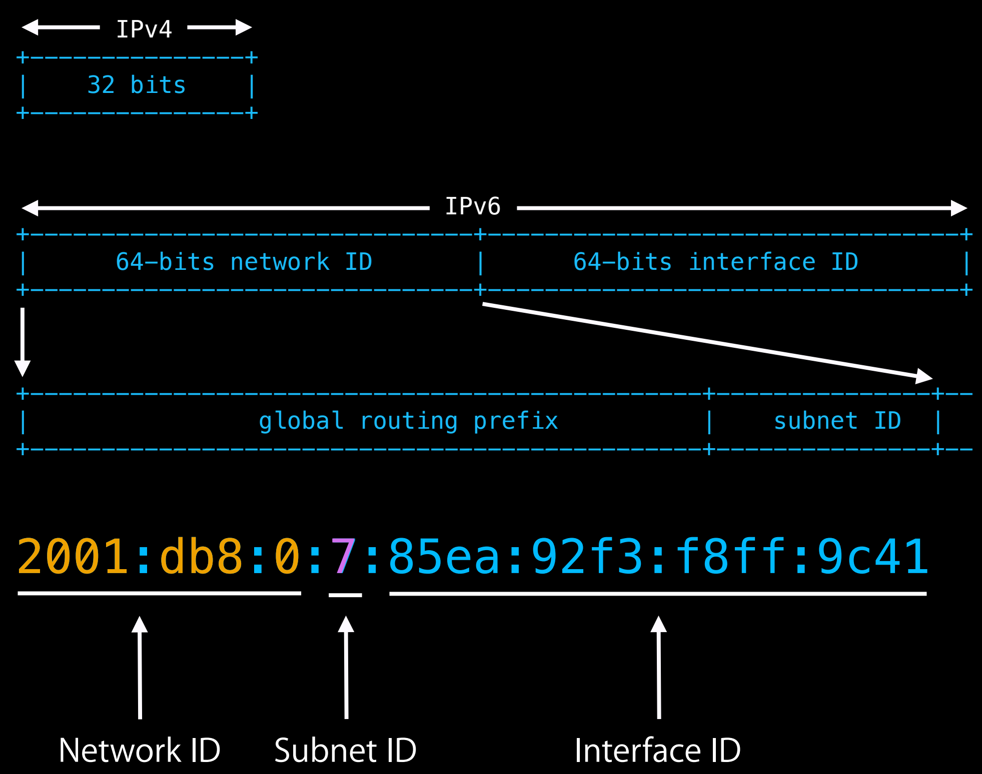
IPv6 Addresses

Improved architecture

- Hierarchical allocation model
- Well-defined scopes

IPv6 Addresses

Hierarchical Allocation Model



IPv6 Addresses

Well-defined scopes

- Link-local unicast `fe80::/10`

```
struct sockaddr_in6 s6;
s6.sin6_addr = MY_HOST_ADDRESS;
s6.sin6_port = MY_PORT;
s6.sin6_scope_id = if_nametoindex(“en0”);
```

- Global unicast
  - Public routed `2000::/3`
  - Unique-local `fc00::/7`
  - IPv4-transitional `64:ff9b::/96, ffff::/96, 2002::/16, 2001::/32`
- Multicast `ff00::/8`

Control Messages

ICMP ➔ ICMPv6

- ARP, IPCP ➔ Neighbor Discovery
- DHCP ➔ Router Discovery + DHCPv6
- Path MTU discovery
  - IPv4: nonstandard, TCP-only
  - IPv6: standard, for all transports

Facts for Developers

Same challenges, new wrinkles

- Automatic network renumbering
- Normal for interfaces to have multiple addresses
- Middleboxes (firewalls, proxies, and NAT)

"For a successful technology, reality must take precedence over public relations, for nature cannot be fooled."

Richard Feynman, Rogers' Commission Report (on the Challenger disaster)

iOS and Mac OS X

History of IPv6 in iOS and Mac OS X

- Appeared in iOS 3.2 and Mac OS X 10.1
- Not visible in Settings on iOS
- Improved in iOS 4.1
  - Configure DNS servers with router advertisements `RFC 5006`
  - Stateless DHCPv6 client `RFC 3736`
- Improved in iOS 4.3 and coming in Mac OS X 10.7
  - Temporary addresses for privacy `RFC 3041`
  - Stateful DHCPv6 client `RFC 3315`

New in iOS 5 and Mac OS X 10.7

Upgrades

- TAHI conformance and full application readiness
- Improved address selection policy `RFC 3484bis`
- Concurrent TCP connections in CFNetwork
- IETF Advanced Sockets API `RFC 3542`
- Temporary addresses used by default in Lion `RFC 4941`

Advanced Socket API

RFC 3542

- Not fully source code compatible with older API `RFC 2292`
- Affected RFC 2292 options: `IPV6_PKTINFO, IPV6_HOPLIMIT, IPV6_NEXTHOP, IPV6_HOPOPTS, IPV6_DSTOPTS, IPV6_RTHDR`
- You may need to choose the socket API you are using

```
#define __APPLE_USE_RFC_3542
```

- For the older API…

```
#define __APPLE_USE_RFC_2292
```

Temporary Addresses

RFC 4941

- Preferred for outbound flows by default
  - Use `IPV6_PREFER_TEMPADDR` socket option to override
- Returned in `getifaddrs(3)` results
  - Marked with `ND6_IFF_TEMPORARY` flag
  - Other flags you may see: `ND6_IFF_TENTATIVE, ND6_IFF_DEPRECATED`

New

Next Steps

- Get ready for IPv6
  - Light up your home network
  - Upgrade your IPv4-only apps
- Things to remember
  - Transition is happening now
  - Success = ordinary users never know
  - High-level APIs make it easy

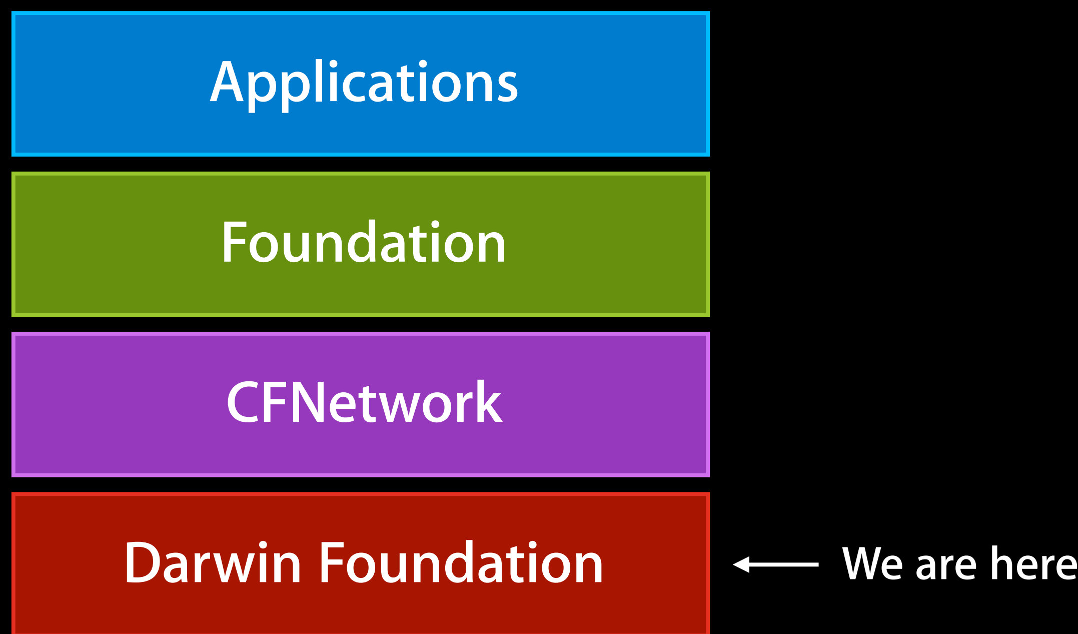
Changes Under the Hood

Old interfaces, new tricks

Josh Graessley
Engineer, Core OS Networking

Changes Under the Hood

Old interfaces, new tricks



Changes Under the Hood

- Address selection
- Connect-by-Name
- Statistics
- Throttling
- TCP buffer sizing

Changes Under the Hood

Dynamic environment

- Today
  - IPv4 primary
  - Edge Cases
    - IPv6 tunnels
    - IPv6 broken routes
    - DNS AAAA (IPv6 records) broken
- Eventually
  - IPv6 only
  - IPv4 legacy

Address Selection

- Same API

  `getaddrinfo`

- New rules
  - Routing statistics
  - Policy table— `RFC 3484bis`

Connect-by-Name

The old way

- Two steps
    1. Hostname to addresses

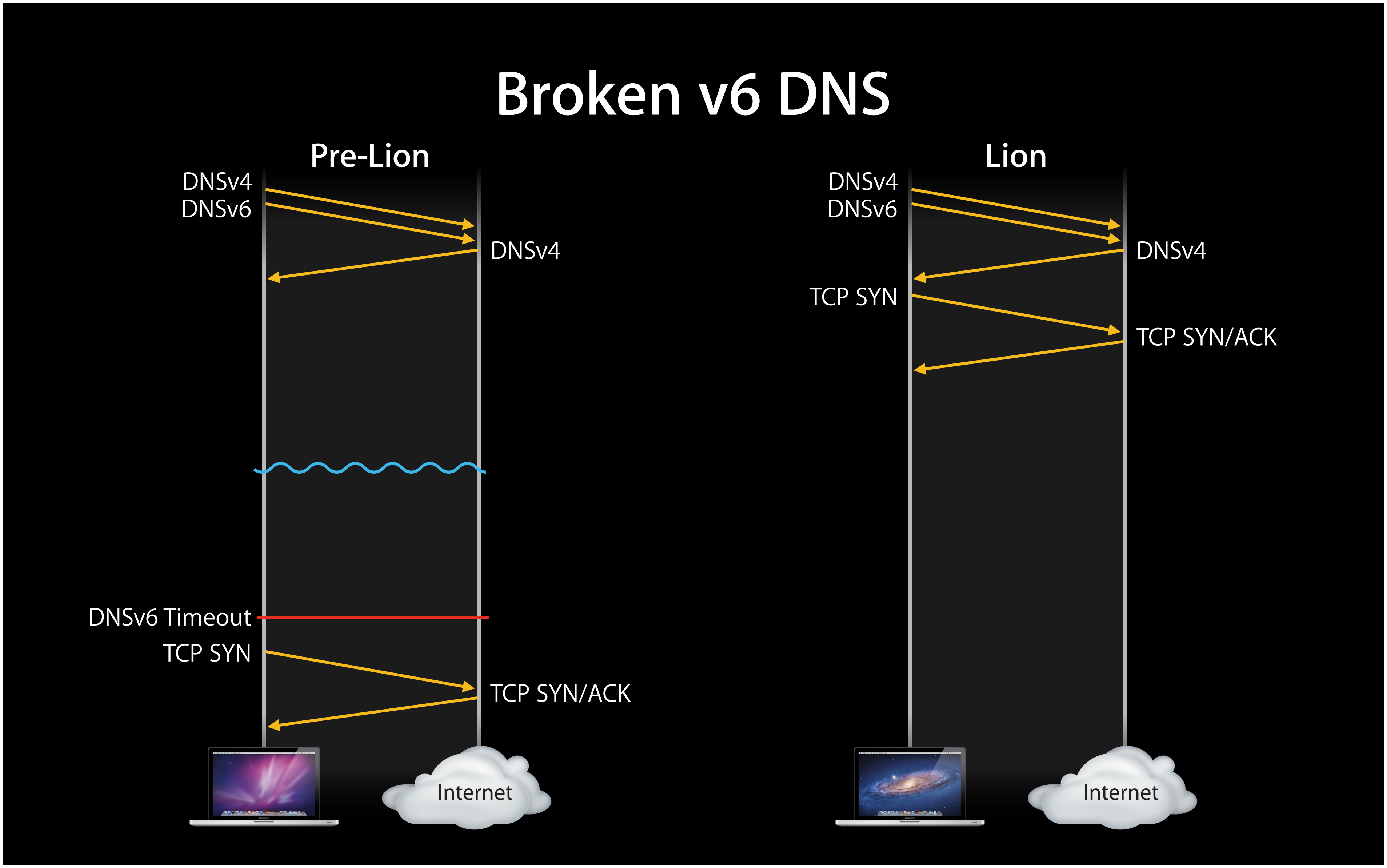
       `getaddrinfo`
    2. Addresses to connection
        - Create socket
        - Set options
        - Connect
        - Repeat on failure
- CFSocketStream abstraction

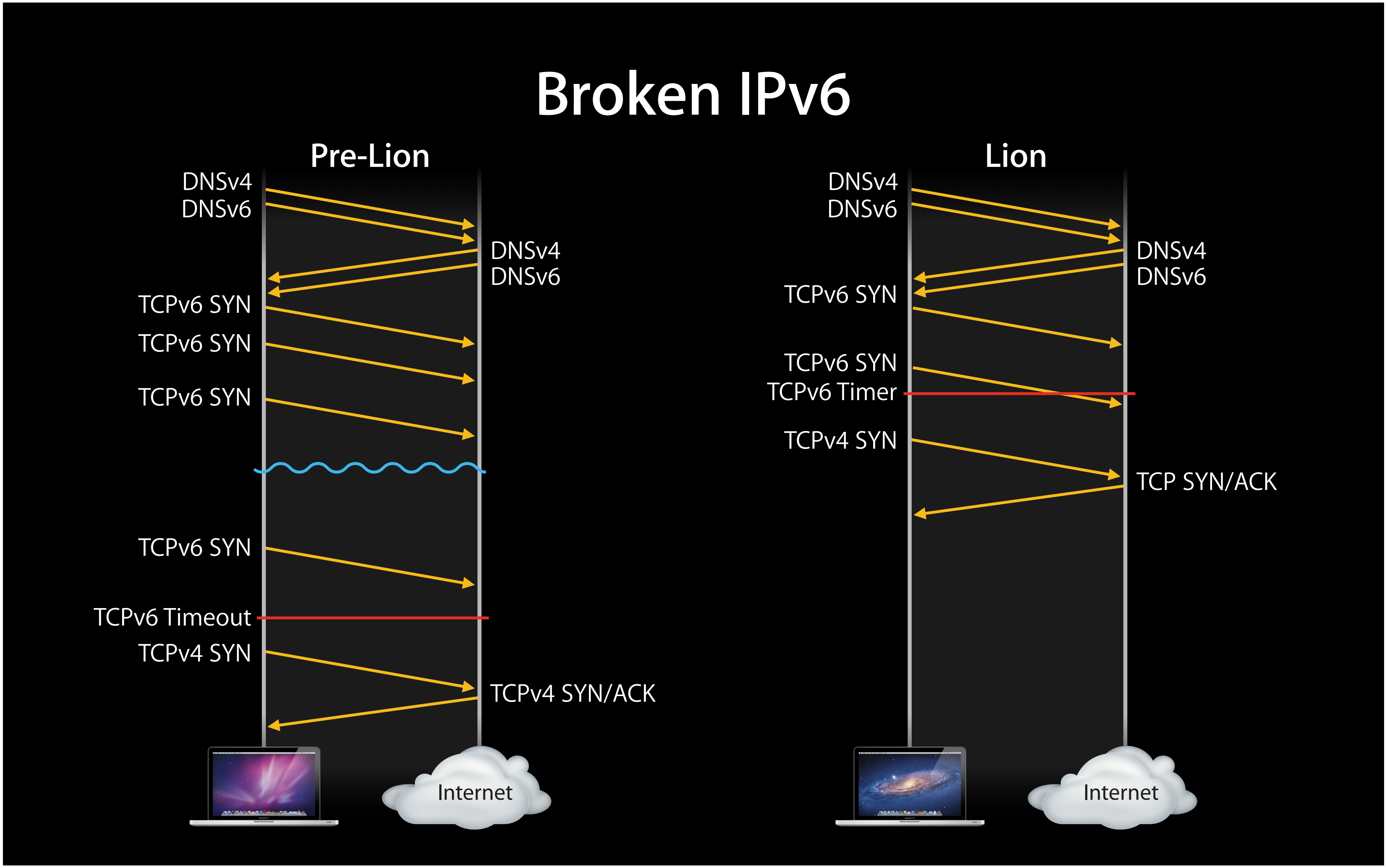
Connect-by-Name

The new way

5 New

- Same API
  - CFSocketStream
- New behavior
  - Separate DNS
    - IPv4 (A)
    - IPv6 (AAAA)
  - Sort destinations
  - Start best destination
  - Start next best after short timer
    - Repeat until established
  - Clean up incomplete connections


Broken v6 DNS
Pre-Lion
DNSv4
DNSv6
DNSv4
DNSv6 Timeout
TCP SYN
TCP SYN/ACK
Internet
Lion
DNSv4
DNSv6
DNSv4
TCP SYN
TCP SYN/ACK
Internet


Broken IPv6
Pre-Lion
DNSv4
DNSv6
DNSv4
DNSv6
TCPv6 SYN
TCPv6 SYN
TCPv6 SYN
TCPv6 SYN
TCPv6 Timeout
TCPv4 SYN
TCPv4 SYN/ACK
Internet
Lion
DNSv4
DNSv6
DNSv4
DNSv6
TCPv6 SYN
TCPv6 SYN
TCPv6 Timer
TCPv4 SYN
TCP SYN/ACK
Internet

Connect-by-Name

APIs

- CFSocketStream and above
  - WebKit
  - NSURLConnection
  - CFHTTPStream
  - And so on…

Network Statistics

- In-Kernel statistics
  - TCP sockets
  - UDP sockets
  - Routes
- Clients
  - Connect-by-Name
  - nettop

Demo

Network Throttling

- Goals
  - User-initiated operations fast
  - Zero-to-low sacrificed throughput
- Conditions
  - Background application sockets
  - Non-user-initiated iOS services
  - Foreground networking application

Background Network Throttling

Techniques

- Transmit—TCP congestion window
  - Normal—shrink on packet loss
  - Throttled—shrink on packet loss or latency
- Receive—TCP advertised receive window
  - Normal—open as data is read
  - Throttled—interpacket delay variation based

Background Process

- Throttled
  - Short period after backgrounding
  - Only if foreground app uses networking `(UIRequiresPersistentWiFi)`
- Suspend
  - Sockets shutdown
  - Bonjour records unregister
- Resume
  - All socket operations fail
  - Close socket—try again

TCP Automatic Socket Buffer Sizing

- Autosizing
  - Requires TCP timestamps
  - Optimal size
  - Minimal waste
- Do not set
  - Send buffer size `SO_SNDBUF`
  - Receive buffer size `SO_RCVBUF`

Take-Away

- Use CoreFoundation/Foundation network APIs
  - Highest-layer API
- Network Apps use `UIRequiresPersistentWiFi`
- Handle network errors
- Let system pick socket buffer size
- Eliminate blocking operations on main thread

Tools

Simulation and troubleshooting

Vincent Lubet
Engineer, Core OS Networking

Network Link Conditioner

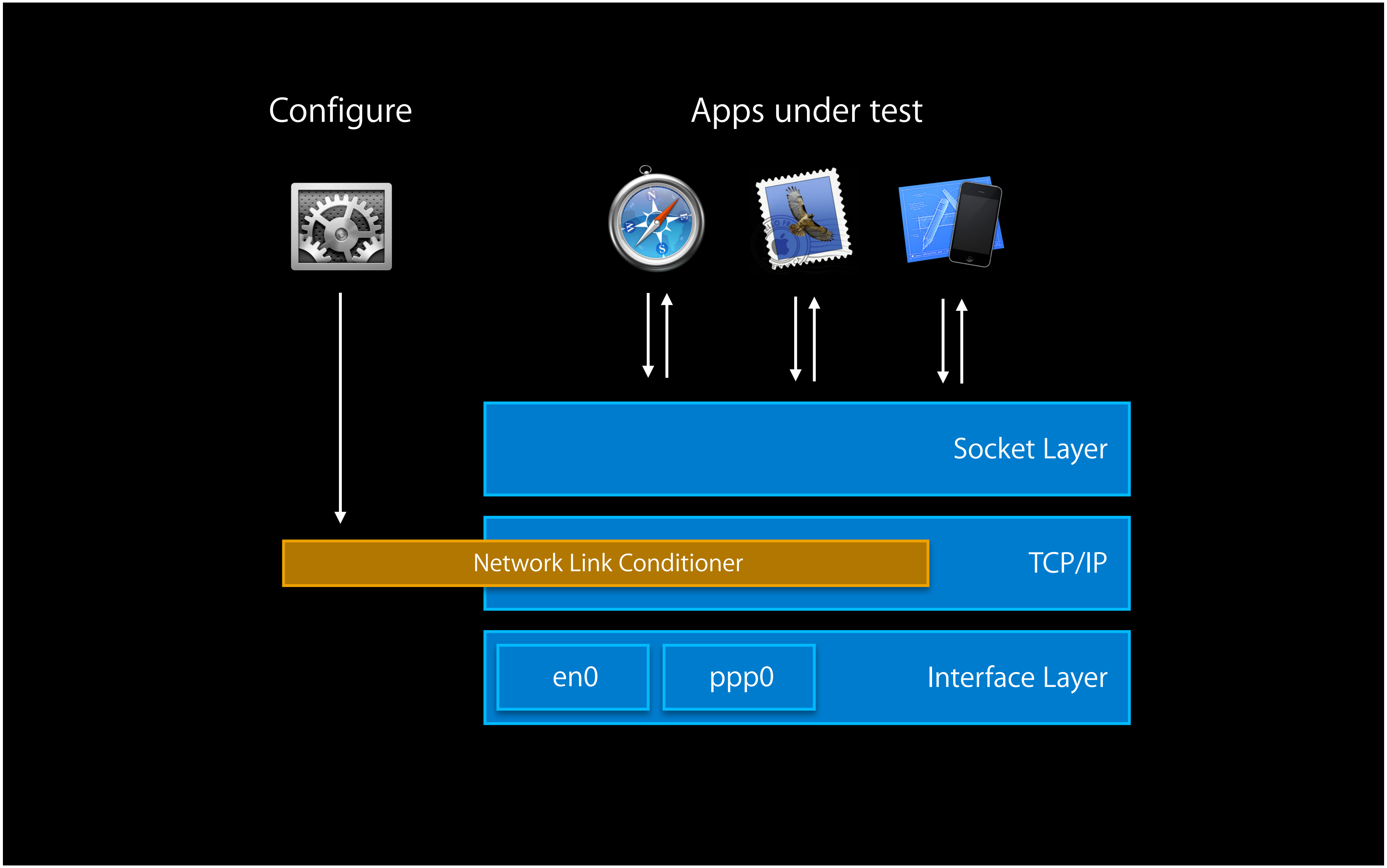
Network Link Conditioner

New

- Tool to emulate realistic network conditions on your Mac
  - Presets for 3G, Edge, Wi-Fi, DSL…
- Tune your networking code from the start
- NLC works for all applications running on the Mac
  - Includes iOS apps running in the iOS Simulator

What NLC Can Do

- Affects all IPv4 traffic on the Mac
- Limits bandwidth
- Adds delay
- Adds packet loss
- Adds DNS delay
- Separate uplink and downlink settings
    - Support custom configurations


Configure
Apps under test
Socket Layer
Network Link Conditioner
TCP/IP
en0
ppp0
Interface Layer

Demo

Network Link Conditioner

Network Link Conditioner Take-Away

- Test networking code early and often in realistic conditions
  - Avoid to discover pitfalls late in development cycle
- Follow best networking programming practices
  - Use asynchronous operations instead of blocking
  - Large I/O are best for bulk data transfer
  - Pipeline transactions

Remote Packet Capture

Remote Packet Capture

- Captures network traffic off an iOS device
  - Wi-Fi and cellular data
  - IPv4 and IPv6 traffic
- Helps diagnose all kinds of networking issues
- An OS X virtual network interface represents the networking stack of the iOS device
- Works with any tool that uses BPF or libpcap on OS X
  - tcpdump, Wireshark…

Remote Packet Capture Howto

Mac with iOS 5 SDK

iOS 5 device

USB cable

Remote Packet Capture Cheat Sheet

- Create the remote virtual interface

```
rvictl -s <UDID>
```

- ifconfig to check the remote virtual interface name
- ifconfig

```
...
rvi0: flags=3005<UP,DEBUG,LINK0,LINK1> mtu 0
```

- Run tcpdump

```
tcpdump -n -i rvi0
```

- When done, delete the remote virtual interface

```
rvictl -x <UDID>
```

Demo

Remote Packet Capture

Summary

- IPv6
  - Transition is under way! Test your apps with IPv6
- Changes under the hood
  - Connect-by-Name
  - Network statistics with nettop
  - Handling of backgrounded connections
- Tools
  - Network Link Conditioner
  - Remote Packet Capture

More Information

Paul Danbold
Evangelist
danbold@apple.com

Documentation
Networking
http://developer.apple.com/library/mac/navigation
http://developer.apple.com/library/ios/navigation

Apple Developer Forums
http://devforums.apple.com

Related Sessions

Core OS Networking, Key Principles	Marina Tuesday 9:00AM
Bonjour Network Discovery and Connectivity	Pacific Heights Thursday 11:30AM

Labs

Network Lab	Core OS Lab A Tuesday 11:30AM
Network Lab	Core OS Lab A Wednesday 2:00PM
Network Lab	Core OS Lab A Thursday 4:30PM

WWDC2011