IPv6 at Google

Lorenzo Colitti
Niall Murphy
{lorenzo,niallm}@google.com

Google and IPv6

Google

The need for IPv6

- IPv6 is critical for continued growth of the Internet
  - IPv4 run-out
  - Mobile devices & appliances talk to each other
  - NAT not a solution
    - Doesn't scale
    - Breaks non client-server interactions
    - Breaks end-to-end and net neutrality
    - Stifles new application development
- Early adoption critical for quality service down the road
- When our users need IPv6, we must be ready

Google involvement in IPv6

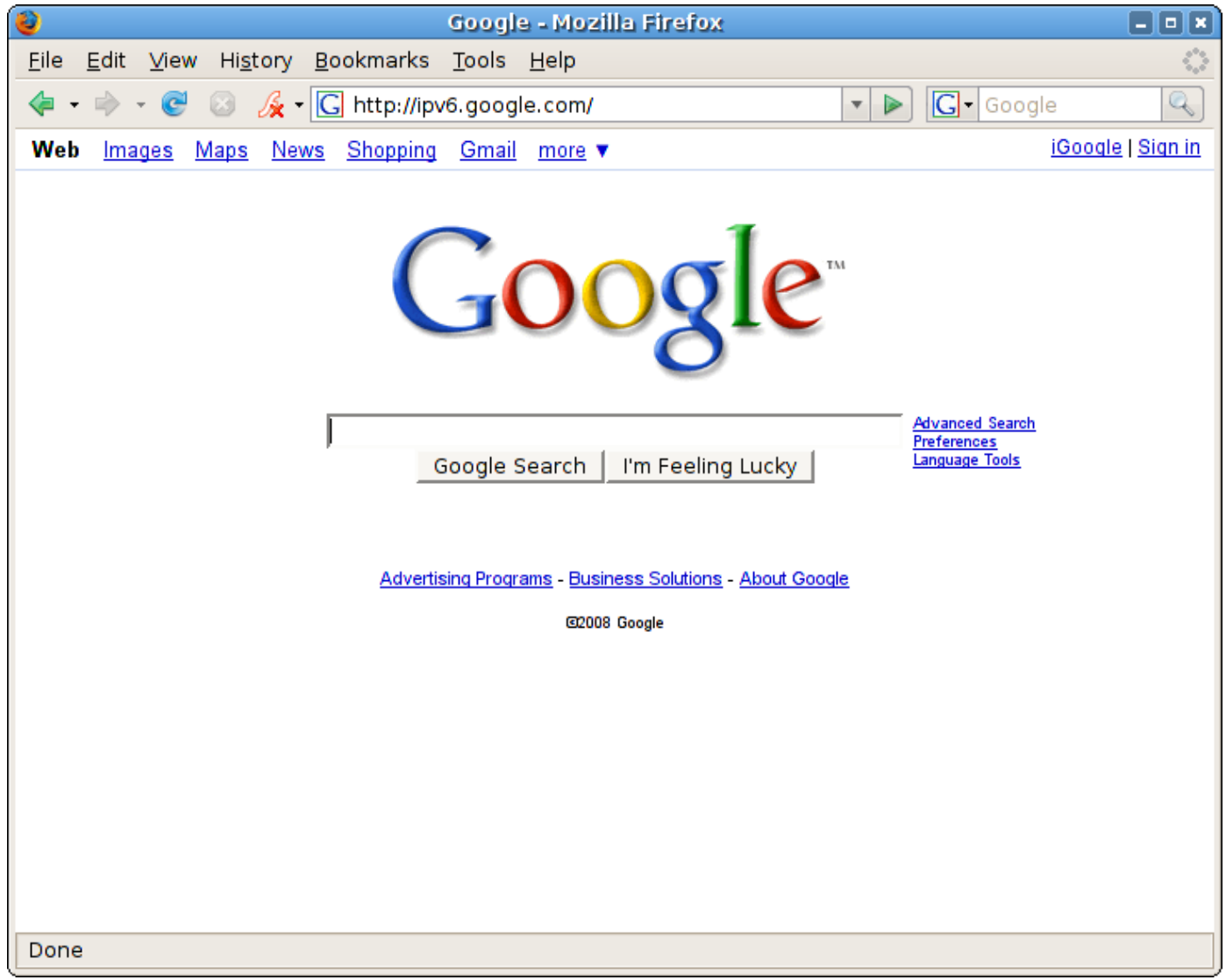
- Google IPv6 conference, January 2008
- IETF involvement
  - IPv6 WG participation
  - IETF 71 IPv4 blackout session
- IPv6-accessible websearch launch on 12 March 2008
  - Only major search engine so far
- More to come...

Agenda

1. Google and IPv6
2. ipv6.google.com
3. Lessons learned
4. Where do we go from here?

ipv6.google.com

Google



"Virtually none of the better known web destinations were reachable over IPv6. That changed when ipv6.google.com popped into existence."

-- Iljitsch van Beijnum on the IETF71 blackout

An important first step

- Currently search only
  - ... but users have already hacked around this
- Crawls IPv4 sites only
  - ... but not a lot of content on IPv6 out there now
- Doesn't display perfectly on an IPv6-only connection
  - ... but search results are IPv4-only anyway
- Separate hostname
  - `www.google.com IN AAAA` would break users!

User response

- Slashdot, blog posts
- "My IPv6 connection is faster than my IPv4 connection"
- "Here's how to hack ipv6.google.com to read gmail"
- "Here's how to use IPv6 in the Firefox search box"
- "Can I have <insert Google service here> over IPv6?"
- ...

Lessons learned

Device support: features

- Feature parity not there yet
  - No MPLS traffic engineering
  - Extension header filtering in hardware problematic
  - Temperamental (broken?) NAT-PT implementations
  - No hardware support for 6to4 or Teredo
  - Load-balancer support not mature yet
    - VRRP
    - Even Path MTU discovery didn't work at first!
- Adequate for initial deployment
  - We can live without all this today
  - But not if we need to serve IPv6 at high volume

Device support: reliability

- Load balancer memory leaks
- Router crashes
  - On eve of launch, three routers in two continents crash within a minute of each other
  - "*In certain rare conditions, <X> routers may crash when finding the best match for a specified prefix* ."
  - So three at the same time is "rare"?
  - "*This crash is more likely to happen with IPv6 because the prefixes are longer* "
- You might want to consider dedicated IPv6 devices :-)

Internetworking

- Rejecting extension headers causes MTU black holes
  - Lucky the minimum IPv6 MTU is 1280...
- IPv6 interdomain routing patchy
  - Indiscriminate transit
    - Slows convergence, increases RTT
  - Blackholing
    - Our /32 not visible from IETF on day of launch
    - "Tier-1" networks with incomplete BGP tables
  - Rich peering interconnections essential!
- IPv6 interdomain performance unknown, assumed < v4

Tunnels

- Tunnels increase latency and complicate debugging
  - Avoid them wherever possible
    - Particularly for interdomain traffic!
- 6to4 and Teredo
  - Suboptimal performance
    - Outgoing path can be optimized by deploying relays close to content
    - Incoming path still bad if relay not close to user
  - Do not provide stable addresses
    - For HTTP, might as well use IPv4...

Operations

- Dispel notion that IPv6 is "experimental"
- IPv6 must be a production service
  - Monitored
  - Supported
  - Designed to the same quality standards as IPv4
- How to achieve this?
  - Make NOC aware of IPv6
  - Scale down, but don't skimp
  - Design as closely to IPv4 as possible
    - Make the principle of least surprise work for you

Where do we go from here?

The road ahead?

- Rich connectivity will increase performance & reliability
  - Peering, peering, peering
  - Avoid tunnels
- NAT-PT and v6-only networks essential
  - Ease address crunch
    - A lot of the Internet is behind NATs anyway
  - Decouple clients from content!
    - Content can move to IPv6 as appropriate
      - When the other end has v6, NAT goes away
  - Requires mature NAT-PT implementations...

So, what do we need?

- Backbone:
  - MPLS traffic engineering
    - 6PE not a solution
    - Don't like blackholing traffic if tunnels go down
  - Extension header filtering in hardware
    - MTU black holes are bad
- Datacenter
  - VRRP
    - NUD not fast enough for production quality failover

... and what else?

- User sites:
  - NAT-PT that works
    - Need a bare-bones, *non* all-singing-all-dancing NAT-PT standard
      - NAT is broken anyway
      - Making it work like in v4 is good enough
      - Undeprecate RFC 2766?
- User connectivity:
  - 6to4, Teredo boxes, or hardware support in routers

The real challenge

- How do we adopt IPv6 while maintaining Google quality of service?
- `www.google.com IN AAAA` not the solution today
  - Lower reliability and higher latency for many users
  - Partial/total breakage for small percentage of users
    - Our users rely on us
    - Breakage is unacceptable!

A possible solution?

- Get a handle on the problem
  - Measure the the IPv6 Internet
    - Size?
    - Performance?
    - How many users have suboptimal connectivity?
- Bilateral cooperation
  - Where two IPv6 networks directly peer:
    - QoS can be guaranteed, problems can be fixed
    - Both networks gain operational experience
    - Production-quality services can be provided
    - Any takers?

Questions?

Lorenzo Colitti
Niall Murphy
{lorenzo,niallm}@google.com