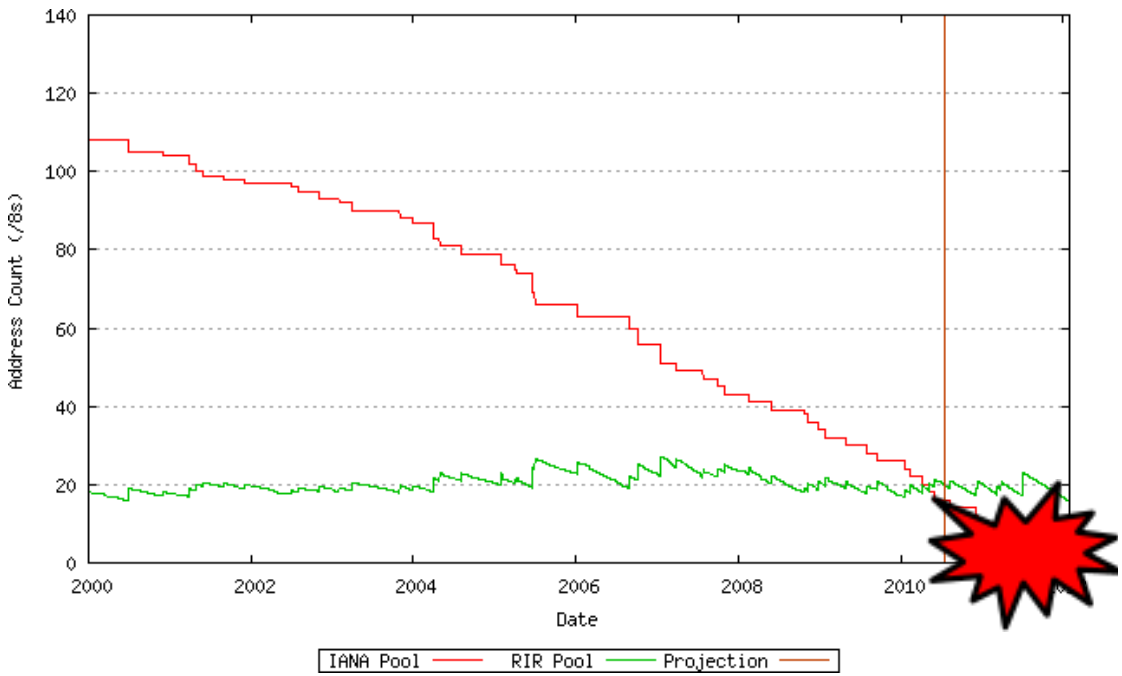
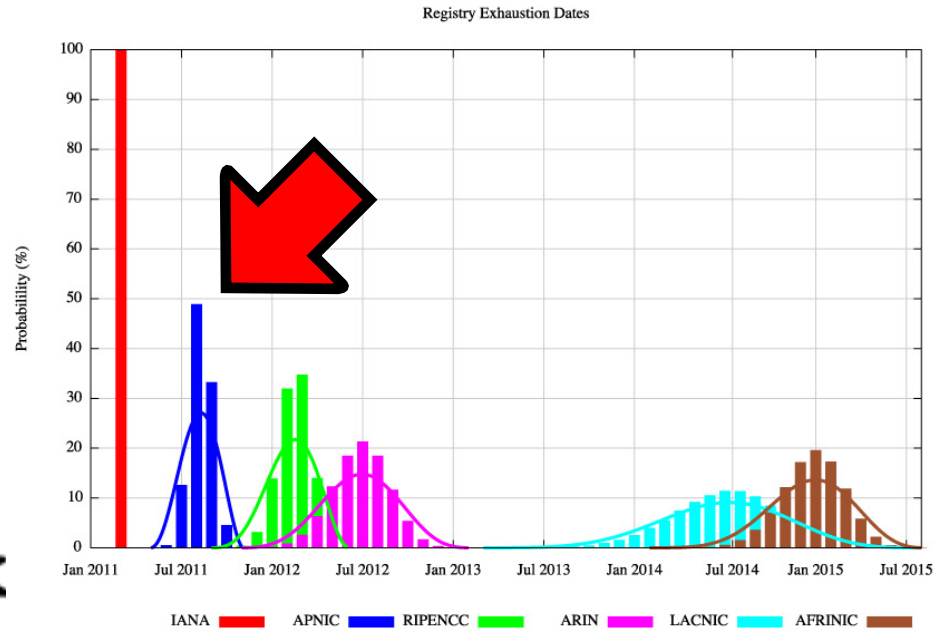
The path to IPv6

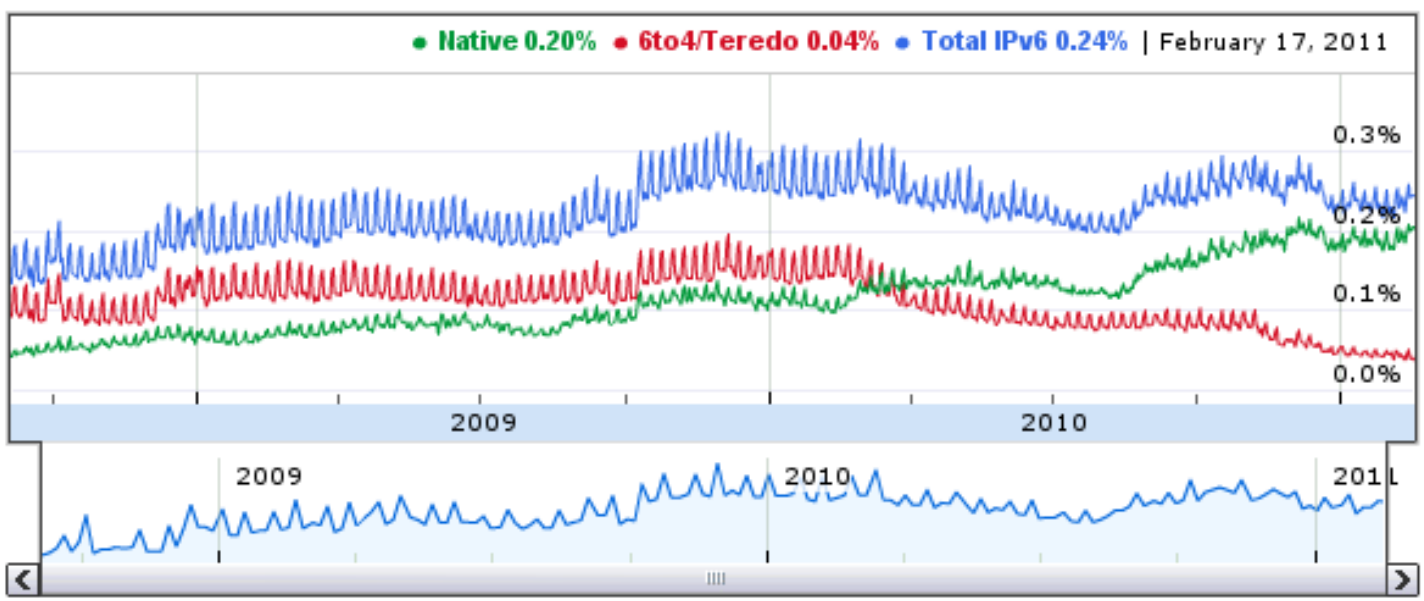
Lorenzo Colitti
lorenzo@google.com

What's the problem?

No more IPv4... but no IPv6 yet either





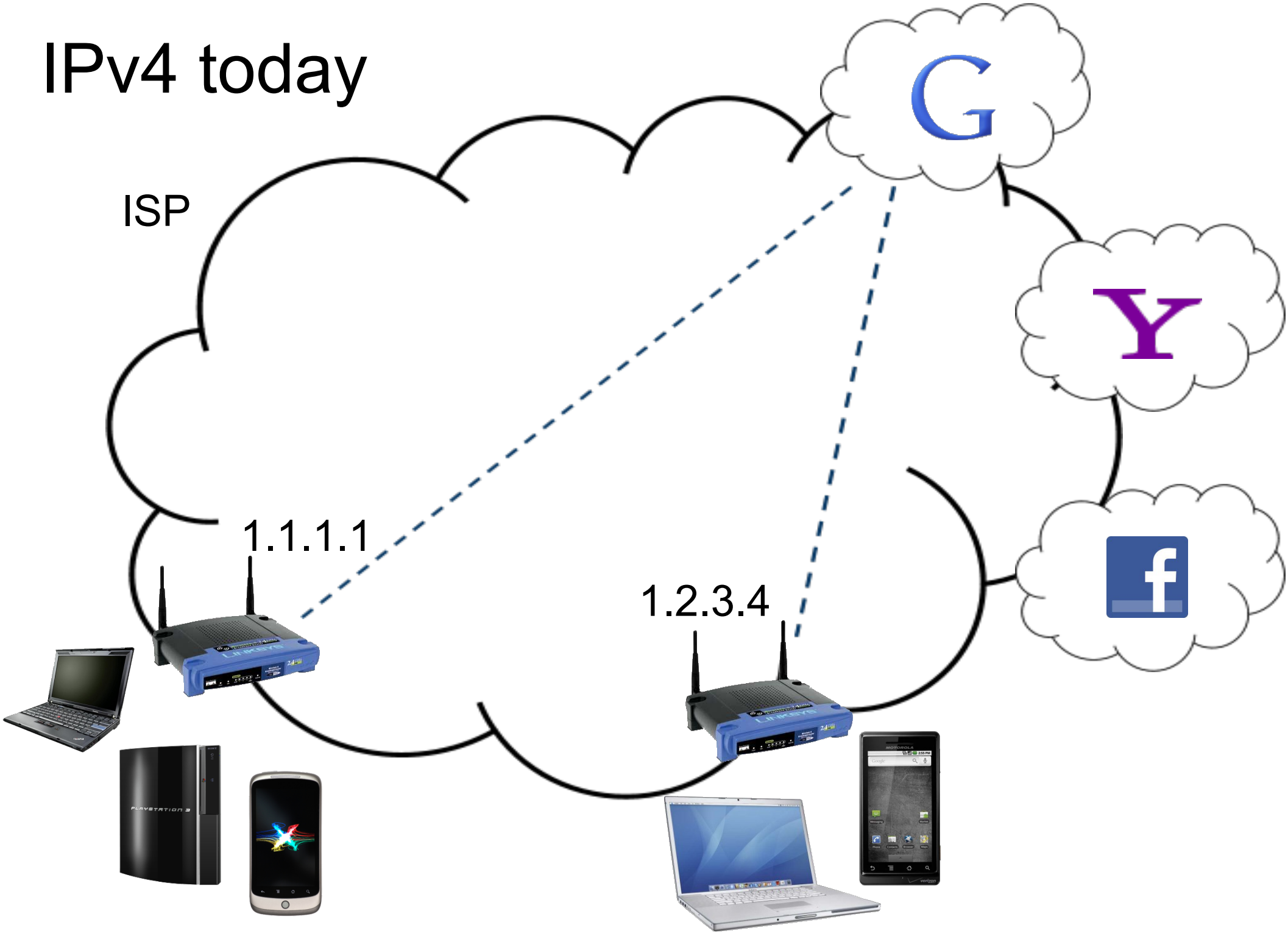


We're about to hit the wall

- APNIC running out of IPv4 space six months from now
- What next?
  - Buy IPv4 space
  - Steal IPv4 space from other parts of your network
  - Large-scale NAT
  - IPv6
- Buying will be **expensive**
  - Black market rate currently around US$4
    - US$16M per million users
    - Expect it to go up
  - Not an option for residential / mobile deployments
    - But these are the ones that need the most space

Large-scale NAT

IPv4 today



IPv4 with LSN

Impact on ISPs

- Expensive
  - Router / linecard resource use
    - Juniper: lose DPC ports, Cisco: CRS-1 blades
  - Logging TCP/UDP sessions for legal intercept
    - 5TB per month per 1M users (source: Yahoo! BB)
- Application breakage
  - VPN, VOIP, video streaming, gaming, P2P
  - UPnP doesn't work with CGN
  - Network complexity creates operation / support costs
- Will not get better over time
  - Will only get worse as more users are behind each IP

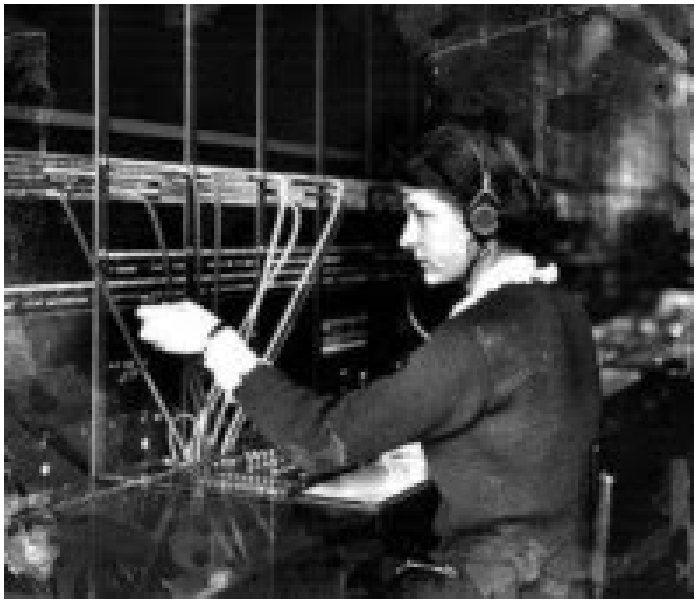
Impact on content

- With carrier-grade NAT, users share IP addresses
- Less accurate geolocation
  - Content licensing for streaming, etc.
- Abuse identification / blocking
  - If an IPv4 address is spamming/hacking/...
    - If we block it, do we take out 100 users?
- Port exhaustion and HTTP intercept
  - AJAX applications suffer

Impact on new applications

- The Internet was successful because of end-to-end
- Users *still want* end-to-end!
  - Skype, Bittorrent, cannot work in the absence of public IP addresses
- What happens if this goes away?
  - Will the Internet become like TV?
  - Will the Internet become like the phone network?
  - Will any Internet communication require ISP support?
- The killer application of IPv6 is the survival of the open Internet as we know it

The Internet without IPv6



IP address sharing

TCP port overload

Content inspection/rewriting

Barriers to innovation

IPv6 in content networks

The way forward?

- IPv4 won't go away for at least a decade
- Carrier-grade NAT is inevitable
  - Long tail of content will not be IPv6-ready for years
- But it doesn't need to carry all traffic
  - A lot of of traffic is from a few content providers
- So, make big content available over IPv6
  - Long-tail content stays on IPv4
  - The CGNs can be smaller
    - Content providers don't suffer from CGN effects
    - ISPs save money
    - IPv6 matures

IPv6 brokenness

- To put content on IPv6, it needs to be as reliable as IPv4
- Currently, about 0.03% of Internet users experience connectivity problems when IPv6 is enabled on a web site
  - Mostly due to misconfigured / broken devices in home
  - If you have 1B users, 0.03% = 300k users
  - User doesn't know what's going on
    - "Everything else works"
    - This is unacceptable
- How do we fix this?
  - DNS whitelisting
  - Engage OS vendors
  - World IPv6 Day

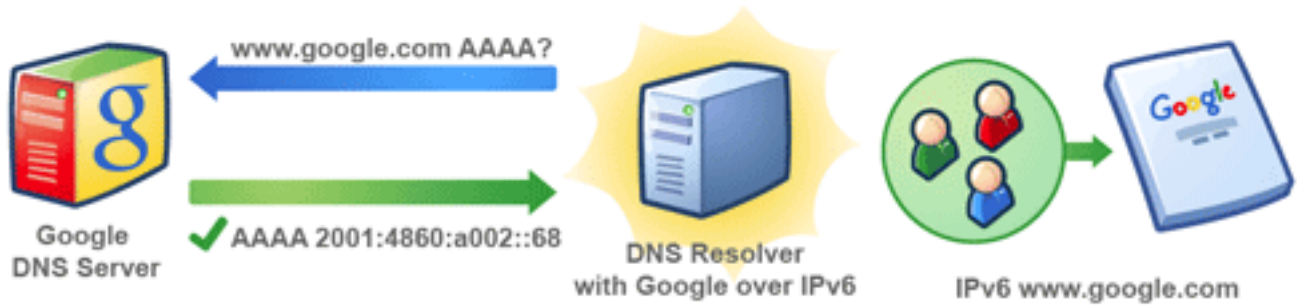
Google over IPv6

- We can't enable IPv6 for www.google.com
  - But we can enable IPv6 access for selected networks
- Most Google services are available
  - www, mail, news, docs, youtube, ...
- Requirements:
  - Good IPv6 connectivity to Google
  - Production-quality IPv6 network
  - IPv6-enabled users use separate DNS servers

How it works

Normally, if a DNS resolver requests an IPv6 address for a Google web site, it will not receive one…

…but a DNS resolver with Google over IPv6 will receive an IPv6 address, and its users will be able to connect to Google web sites using IPv6.



http://www.google.com/ipv6/

Demo

Results so far

- Enthusiastic response:
  - O(100) organizations participating
    - ~ 75% of the native IPv6 Internet
- Feedback so far has been positive
  - Some networks see better IPv6 routing than IPv4
  - Now enough IPv6 traffic that problems get reported

"World IPv6 day"

- Organized by the Internet society
- Google, Facebook, Yahoo!, Bing, Akamai, Limelight, ...
- Turn on IPv6 for 24 hours on 2011-06-08
- Benefits
  - Users with broken IPv6 will have problems reaching all major websites, not just Google
    - May call their ISP, who will help them fix the problem
  - Rallying call for the industry
  - Common date to organize industry-wide collaboration
    - OS fixes, etc.

The IPv6 transition

How did we get here?

- No business case
  - Operators and vendors are more driven towards new revenue than away from revenue loss
- No realization of how long it takes
  - It took Google 18 months; IPv4 running out in 12
- No vendor support
  - IPv6 support != you can actually use it
  - Often there are small but critical gaps
- No demand
  - "There is no content", "There are no users"

Some misconceptions

- "We'll deploy IPv6 when users ask for it"
  - Users aren't asking for IPv4, only Internet access
    - Can we provide Internet access with no more IPv4?
- "We'll need IPv6 when IPv6-only <content|users> appear"
  - Nobody will go IPv6-only until it has >90% penetration
- "We will have to deploy IPv6 once IPv4 runs out"
  - No, carrier-grade NAT will work
  - It's just not very good, expensive, and no upgrade path
- "All our gear is IPv6 ready. We just need to turn it on"
  - "IPv6 ready" does not mean it will work
  - You'll only know it works when you turn it on

Taking the wide view

- All the elements of the value chain must have IPv6
- The weakest link is residential users
  - Don't understand the problem
  - No direct benefit until IPv4 runs out
  - Little purchasing power / leverage
- Until there is IPv6 to a substantial user base, the transition will not happen

So how do we do this?

A possible approach

- Accept that there will be no incremental revenue
  - Therefore, must be "zero" cost
  - Fortunately, surveys show that cost is not an issue
    - As long as you start early enough
- Start early, and fold IPv6 into normal upgrade cycles
- Put IPv6 in to greenfield deployments
  - Don't need to touch existing users or infrastructure
  - Easier to design
  - More leverage with vendors

Test and deploy

- Vendor claims and testbeds not enough
  - There **are** bugs lurking, and you need to find them
- The only way to find the gaps is to target real deployment
  - Same reliability as IPv4
  - Same performance as IPv4
  - Same operational complexity as IPv4
- Assume IPv6 needs to carry all traffic and start from there
  - Gather requirements and test it in the lab
  - Iterate with vendors until it works
    - Be prepared to hear "it's on the roadmap for 2012"
  - Deploy it!
    - If you don't deploy, you won't know it's really working

Google: case study

Methodology

- Tap enthusiasm
  - Started as 20% project, great influx of contributors
- Make it easy for contributors to get initial results
  - A pilot network is not expensive
  - Once network is up, internal applications follow
- Do it in stages
  - v6 needn't be as capable as v4 on day one
  - But it must be done properly
  - If it's not production-quality, it's no use to anyone
- Fold it into your normal upgrade cycles

Development strategy

- Work from the outside, move in
- First the load-balancer, then the frontend, then...
- "Address coercion" protects IPv4-only code from IPv6
  - Take IPv6 address
  - Remove user-modifiable bits
  - Hash into 224.0.0.0/3
- Sometimes not perfect
  - "Your last login was from 238.1.2.3"

Timeline

July 2007	Network architecture and software engineering begin (20%)
December 2007	Mark Townsley challenges Google to serve IPv6 by IETF 73
January 2008	First pilot router. Google IPv6 conference, Google over IPv6 for attendees
March 2008	ipv6.google.com (IETF 72)
January 2009	Google over IPv6 publicly available
March 2009	Google maps available over IPv6, 3x increase in traffic
August 2009	IPv6 enabled in Android (available on Droid and Nexus One)
February 2010	Youtube available over IPv6, 10x increase in traffic
March 2010	Backbone fully dual-stack, AppEngine available on IPv6
June 2010	Hosted AppEngine domains available over IPv6. Start crawling IPv6-only websites.
October 2010	Blogger available over IPv6

And all this with a small core team

Lessons learned

Testing and iteration

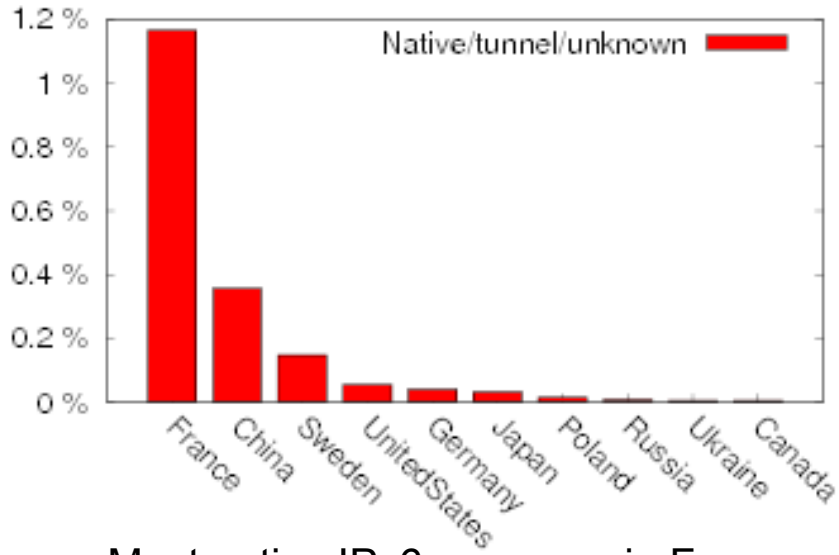
- Implementations mostly work, but will have bugs
  - Nobody has really kicked the tyres
- Don't expect something to work just because it's supported
- If you find a bug in the lab:
  - Report it
  - .. and keep testing!
  - There are many more bugs to find
- Work around it in the design
  - If you get to something that is supportable, trial it
  - That will help you find the hard bugs

For example...

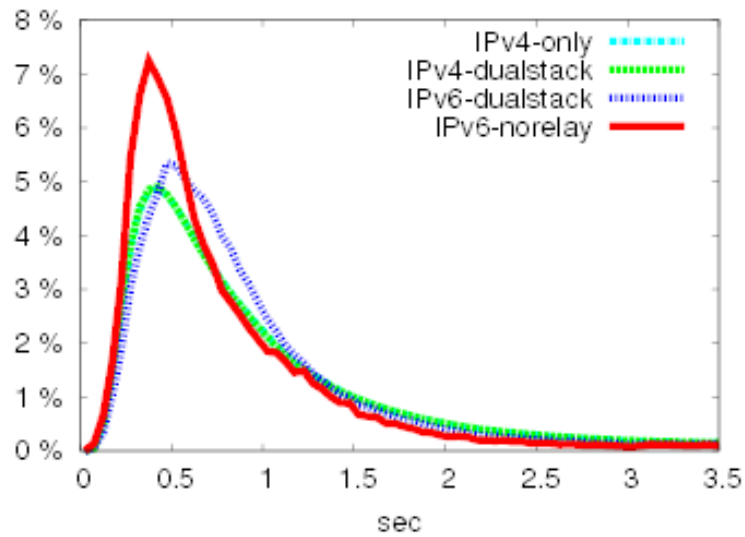
- If a firewall filter term has a 1-bit match in bits 32-64, and then term with a 2-bit match on bits 64-96, the second term will not match
- In particular circumstances, FIB and RIB may get out of sync due to race conditions in pushing updates
- If DAD triggers due to an interface loop, it requires removing config from the interface and putting it back
- If a linux gets a packet too big on a receive-only interface with no route, it ignores it
- Are you going to find these in the lab?
  - We only saw the race condition after months in production in a fair number of datacenters

Statistics

Some statistics

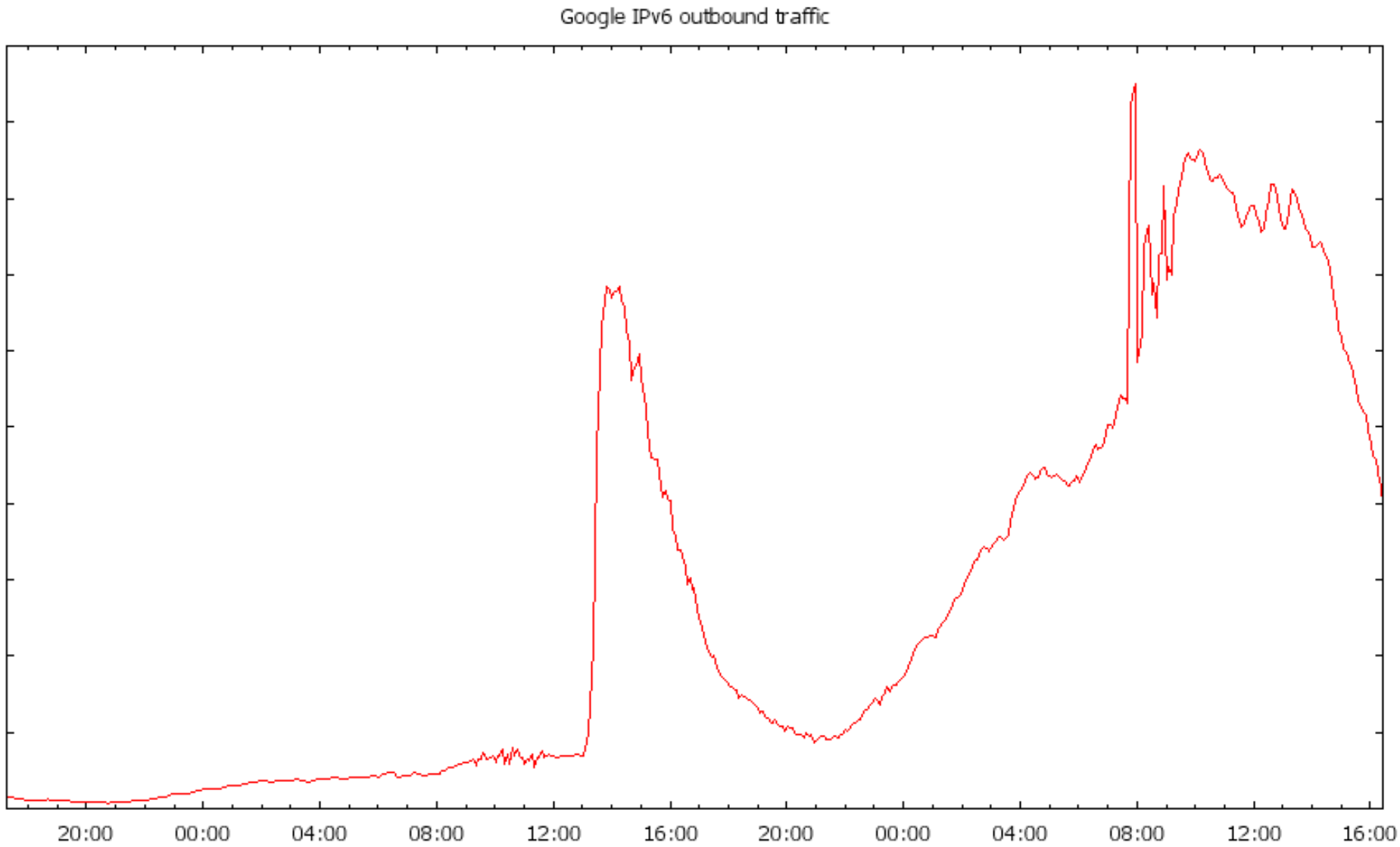


Most native IPv6 users are in France



6to4 / teredo latency penalty > 50ms

Traffic can appear overnight



(IPv6 video launch 2010-01-28)

Questions?

Lorenzo Colitti
lorenzo@google.com