SCIENCE FOR WORK AND PLAY

Measuring IPv6 with advertisements for fun and profit

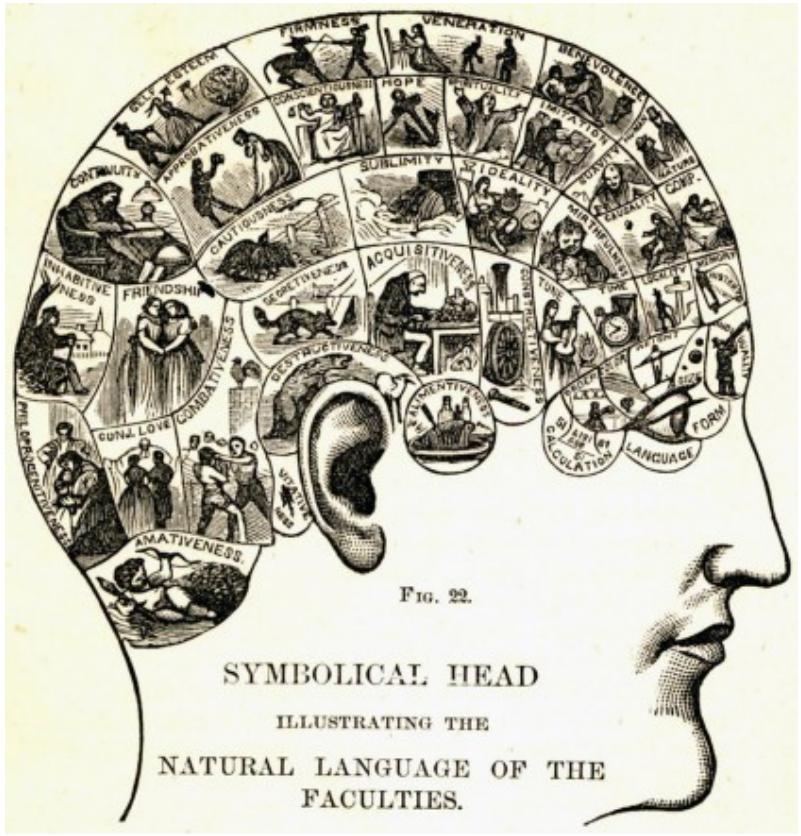
George Michaelson & Geoff Huston,
APNIC

how to measure “the Internet”

- What do we mean when we say
  “we’re measuring the Internet?”

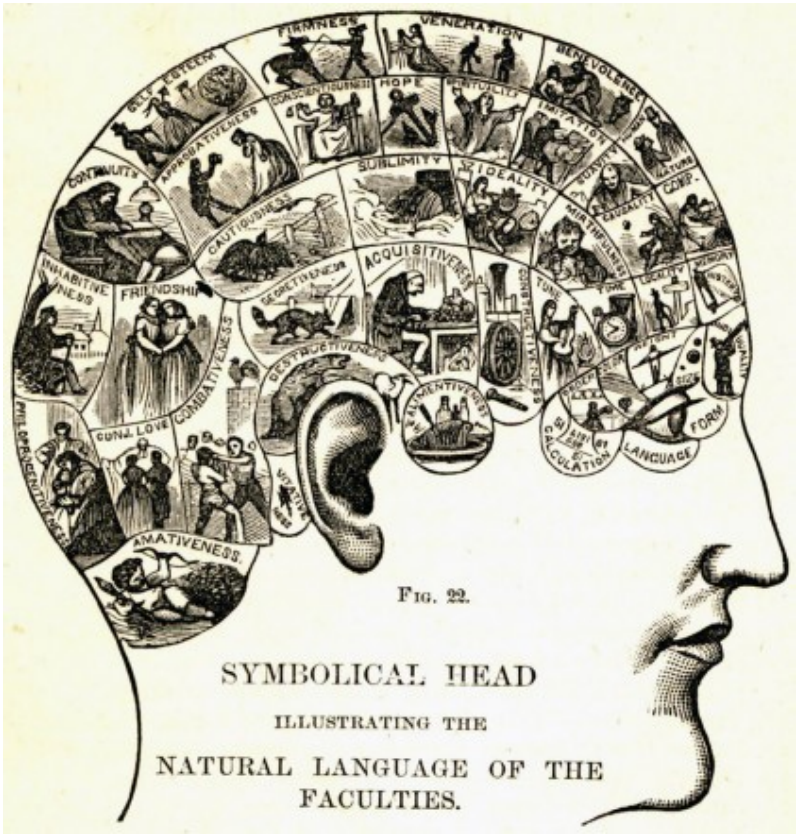
how to measure “the Internet”

- What do we mean when we say
  “we’re measuring the Internet?”



how to measure “the Internet”

- What do we mean when we say
  “we’re measuring the Internet?”



how to measure “the Internet”

- What do we mean when we say
  “we’re measuring the Internet?”
  - Number of hosts
  - Number of routes
  - Number of active routing entities
  - Number of application-events
    - Number of voip calls
    - Number of voice-over-IP in the carrier calls
    - Number of streaming TV watchers
  - ….

Clearly a multi-dimensional problem

- We're well beyond single measure
  - Routing scaling for routing professionals
  - Traffic volumes for peering/settlement
  - Voice/data mix for telephony nuts
- Finding datasets to suit the context
  - DiTL has root-DNS traffic covered
  - IX captures get local traffic mix, indicative volumes
  - ISPs track their own customer
  - Websites track their own users
- Much of this is too valuable to be shared widely

Who is measuring the end user?

Who is measuring the end user?

Measuring the end user for IPv6 readiness

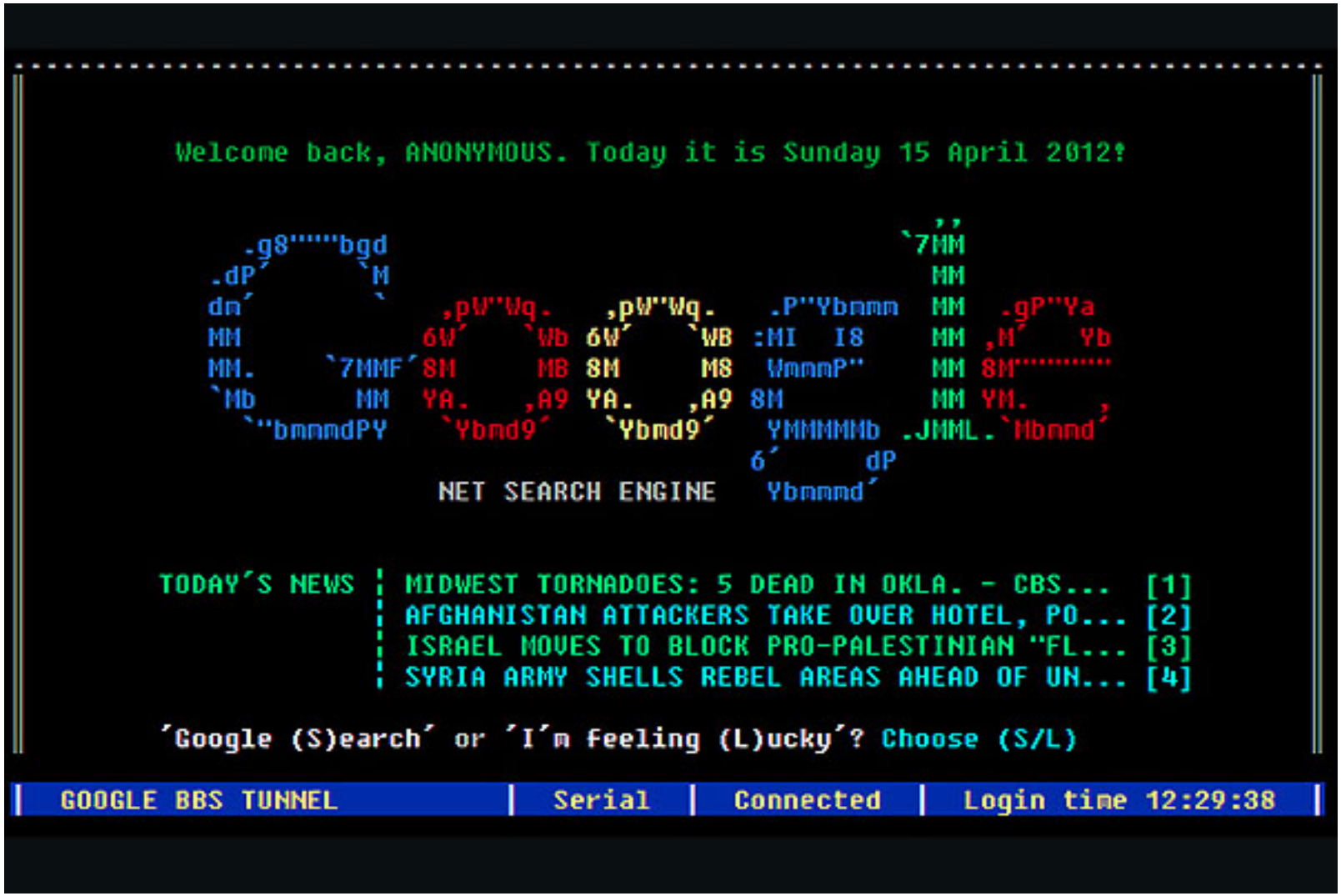
- Need a technique which is ubiquitous
  - "Hop over" missing IPv6 enabled CPE
- Reflect real-world end-user behaviour
- Avoid systemic bias 'my own web is typical'
  - Demonstrably not true for technical community
    - Access by tech end-user is 'near the core'
      - bypassing CPE
    - Often has 'special' access
      - (new/novel/experimental & pre-service release)

JavaScript

- Small community of researchers using JavaScript to ‘tickle’ web users via undisplayed fetches
  - Additions to website, not rendered into DOM, using unique DNS names, traceable
    - *.test.domain wildcards sub-classed to provide unique per-client test names
    - JavaScript driven client-side view of delay
    - Summary sent to web by ‘fetch’ of DNS name
      - Data in the name fetched, not in the content sent
- Not substantially different to normal website tracking methodologies
  - Third party site, or alternate DNS names to main site

Anatomy of a web page fetch (1990s)

Anatomy of a web page fetch (1990s)



Anatomy of a web page fetch (1990s)

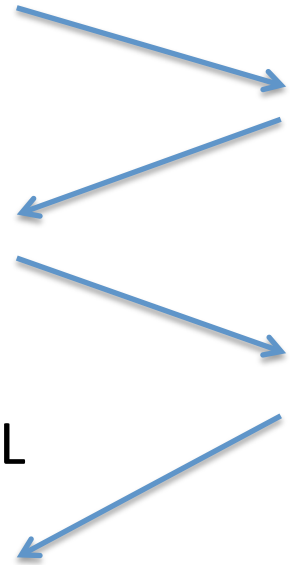
- Client issues HTTP connect

:

- Client issues “GET”

:

- Client receives HTML stream in HTTP
- Client disconnects



- Server listens
- Server accepts
- responds 200 ok
- Server receives, finds URL on disk, sends contents

:

:

- Server exits and back to listen()

Anatomy of a web page fetch (2010s)

Anatomy of a web page fetch (2010s)

Anatomy of a web page fetch (2010s)

CAT

- ~~User~~ visits URL in browser
- Cached content pre-displayed
- Cookies, JavaScript, Flash runs, decides what to display to user
- Web page viewable

:

:

:

- Maybe never Complete

- Client issues several HTTP connects in parallel

:

- Client does handshake over capabilities, header checks, tries to find cached content

:

- Client receives HTML stream in HTTP
- Client disconnects

- Server listens
- Server accepts
- responds 200 ok
- Server receives, finds URL on disk, sends contents

:

:

- Server exits and back to listen()

CAT IZ NATZ OH NOES

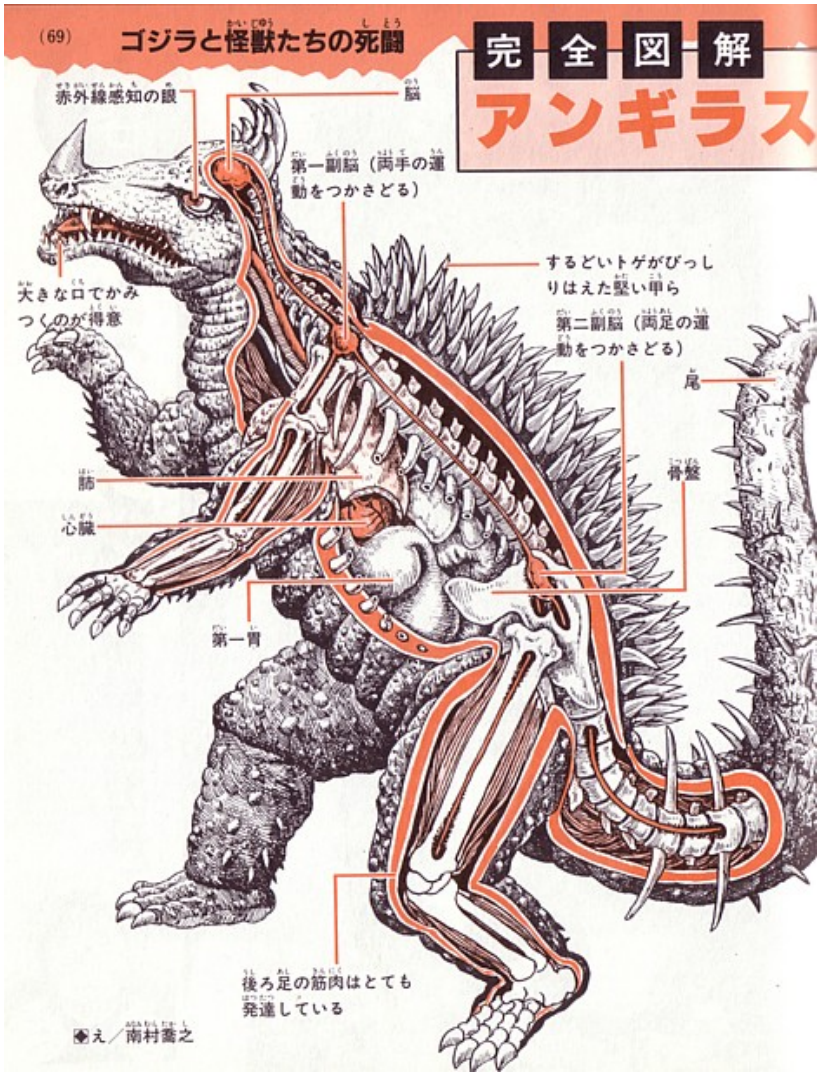
CAT IZ GON

HTTP/HTML ain't what it used to be

- A huge amount of parallelism/asynchrony has been introduced.
- Intermediate processing takes place on the stream of data, deciding what to display and what not to display
  - Adblock, flashblock, cache-optimizations
  - Document Object Model (DOM) includes material not displayed, material not added to DOM by scripts &c
- JavaScript provides rich language including timers, async fetches, string/number processing

Anatomy of a JavaScript measurement

Anatomy of a JavaScript measurement



Anatomy of a JavaScript measurement

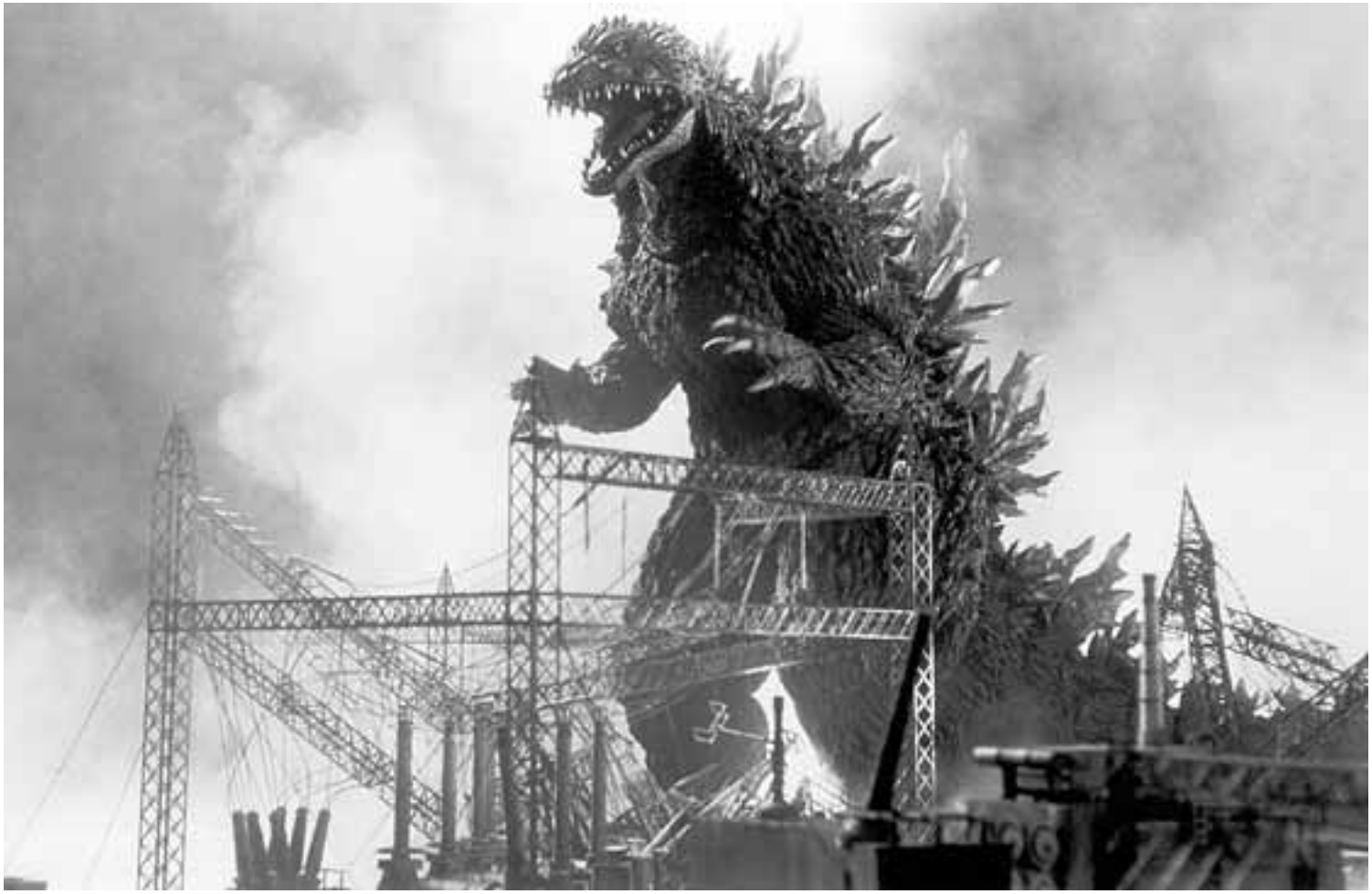
- Website markup includes .js fetch, or inline <script>…</script> block
  - JavaScript engine in browser runs asynchronously to page render
    - Web page drawn unaffected/in-parallel with fetches
  - Spin random, to identify test. All fetches include unique id in wildcard DNS name
    - test314159.*<test>*.labs.apnic.net
  - Series of test images fetched in sequence (or random)

• Dual-stack	test314159.rdtd.labs.apnic.net	A and AAAA record
• IPv6 only	test314159.rdt6.labs.apnic.net	AAAA only
• IPv6 literal	[2401:2000:6660::2]	no DNS involved

  - Each fetch has its own 'sprite' like timer
    - On completion, client-side delay (ms) measured from base clock
    - "Fail" timer, to send results at hang time if tests don't complete (10s)
- Results returned with same unique test id
  - test314159.zrdtd44.zrdt6101.zv6litnull.results.labs.apnic.net
- Timer values embedded in the DNS name.

What do we get

What do we get



What do we get

- Configure DNS to have a single NS, host that NS, and turn on query logging:
  - DNS logs of clients resolver to start test now captured
- TCPdump of packetflows to webserver, dns, tunnel endpoint
  - Can detect partial connect failures, ICMP/ICMPv6 & SYN flows, TCP mss.
    - Also detailed inter-packet timings
- Web logs
  - Successful fetch logging, order not guaranteed in logfile but has server-side timing information as backup

Cross-collating the data

- Initial .js fetch on IPv4.
  - Confirms IPv4 address under test
- Final results.labs.apnic.net posting on IPv4
  - Confirms test sequence ran to completion
  - If received, also has client-side delay times
- All tests from same host carry same random ID
  - Where logged, can then cross-correlate IPv4 and IPv6
- IPv4 and IPv6 can be seen in TCPdumps
- Cross index to DNS resolver IP in DNS query also possible

Post-test processing

- Oddities
  - results line received before tests complete
  - results line never received, but individual tests run to completion
  - tests lag by extreme periods, minutes after test
    - (so contradict results line which says null for that test)
- Post-process heuristics manage this to produce ‘unified’ log of test combining data from web logs.
  - If any source said we saw the test, its included, even if results say test wasn’t run (!)
  - If results line provide times, then these times are used, otherwise server-side times are used.

What do we get?

- Outcome: measurements at the 50,000 -100,000 hits/day level across 20-30 participating websites
  - large hits from specific economies/websites, skewing data capture
- Still valid, but not yet 'global'
  - A site like wikipedia, or an international newspaper would be **EXTREMELY INTERESTING** as a collection source
  - JavaScript can be used to perform 1-in-1000 type sub-rate filters to sample highly popular sites

Solutions

- Need a way to get global coverage
- We want to leverage the JavaScript investment, use the same data collection methodology (combine datasets)
- Looking for a vehicle similar to JavaScript, but not limited to websites we can persuade to include our code.

…buy the impressions

…buy the impressions

…buy the impressions

…buy the impressions

- Daily investment of $20 buys 50,000 impressions/day
- Web advertising networks now fundamental to 'making the web pay'
  - *Lots of websites willing to have adverts placed for $*
- Well designed framework for distribution of content to websites en masse
  - *Submit once, placement as widely as possible* ***worldwide***
- Simple payment model based on impressions/clicks price point: CPM (clicks per mille)
  - *Low CPM translates to high impression count*
    - Remember: the advertising network wants your money, so if you bid too low for clicks, you get sold placements, to justify the payment

From JavaScript to flash

- Advertising using flash encoded ‘dynamic’ content
  - Flash authoring tools widely available, ubiquitous for dynamic website content
  - Advertisement presents as an ‘image’ but can use flash to download active elements, movie clips, &c
- The advertising economy is now almost completely based on flash
  - Turn off flash, or run an adblocker and visit your normal daily diet of websites…
- JavaScript and actionscript near-cousins
  - Simple to translate working JavaScript into flash

Minor Problems

Minor Problems

Minor Problems

It’s a Morris Minor …..

Minor Problems

- Advertising networks limit which flash primitives you can use (they reverse-compile the code and check)
  - Luckily, **`fetchURL()`** is basic to the flash-ad economy
  - Exclude **`random()`** library calls from code
    - the ad network has to provide information into the flash advert which is functionally highly random, which we reduce via hand-coded **`crc32()`** hash (which the reverse-compiler doesn't complain about)
- Cross site scripting demand use of crossdomain.xml fetch
  - Twice the volume of fetches required for same measurement
  - Could almost base method on this fetch alone (!)
- Flash not on all platforms (cannot measure iOS)

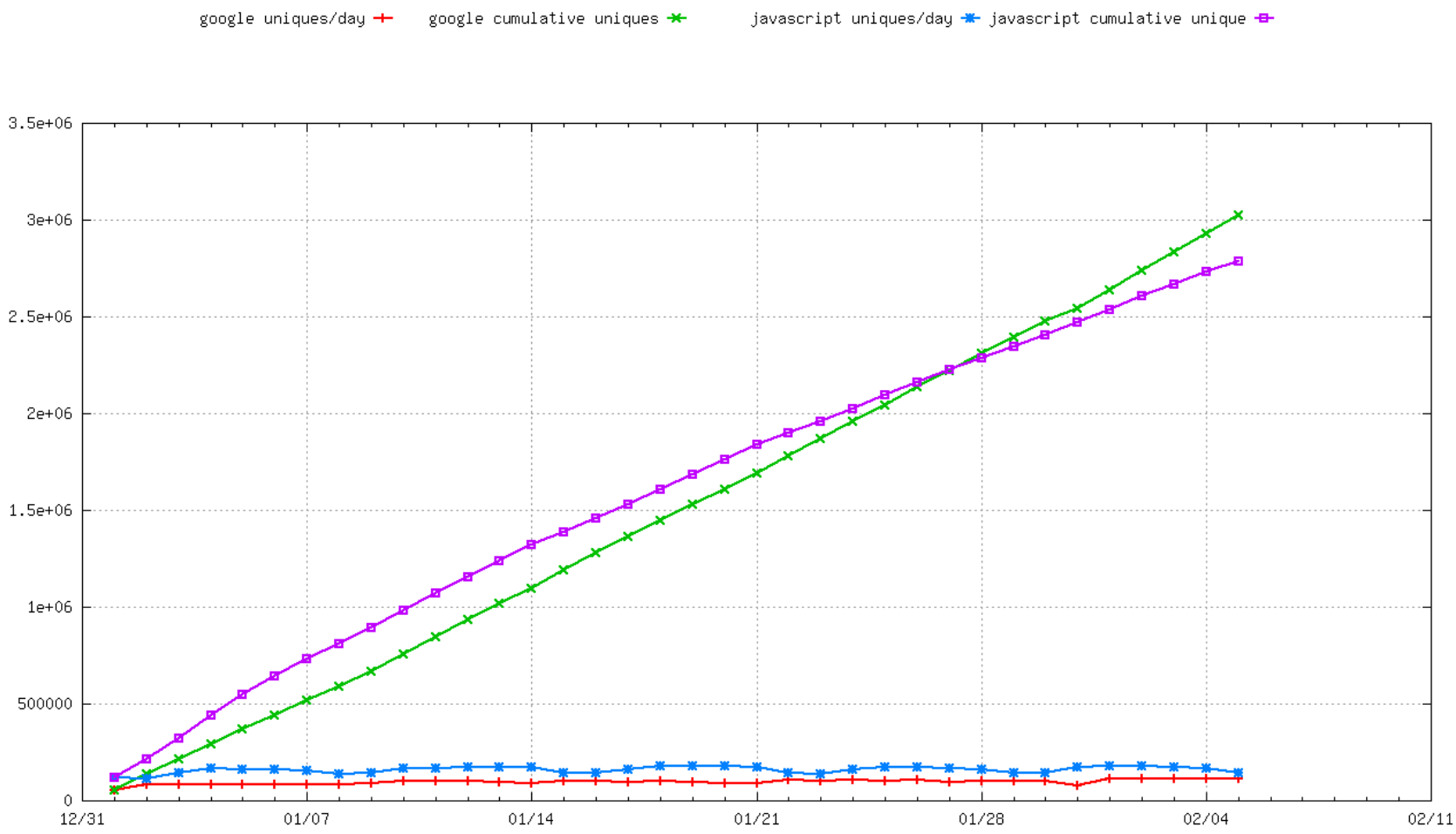
Placement

- At low CPM, the advertising network needs to present unique, new eyeballs to harvest impressions and take your money.
  - Therefore, a 'good' advertising network provides fresh crop of unique clients per day
- Language-specific selections can tune placement
  - Evidence suggests that of 250 iso3166 economies, we have secured placement into 200, with 150+ at significant volume

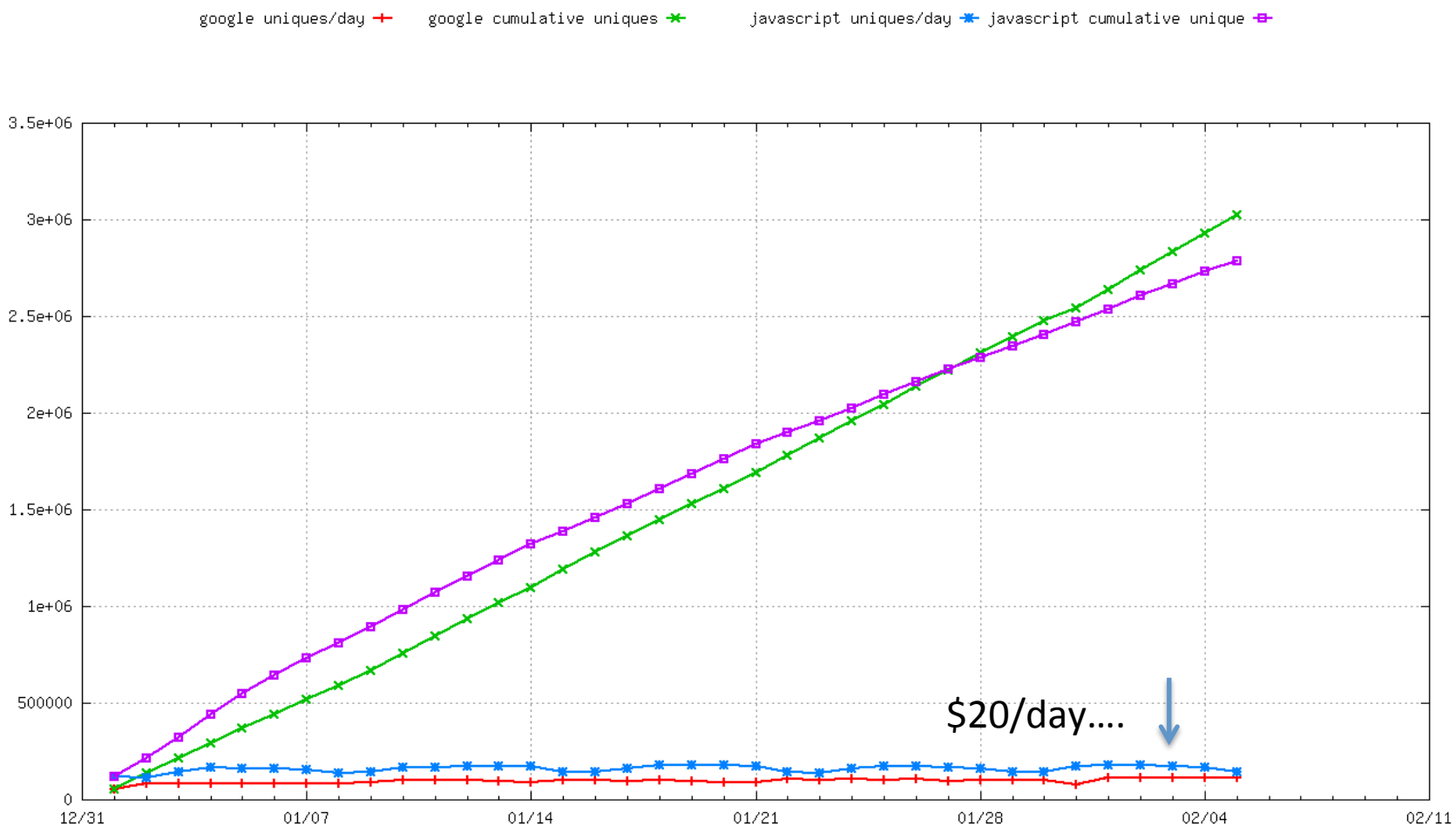
Unique IPS?

- Collect list of unique IP addresses seen
  - Per day
  - Since inception
- Plot to see behaviours of system
  - Do we see ‘same eyeballs’ all the time?

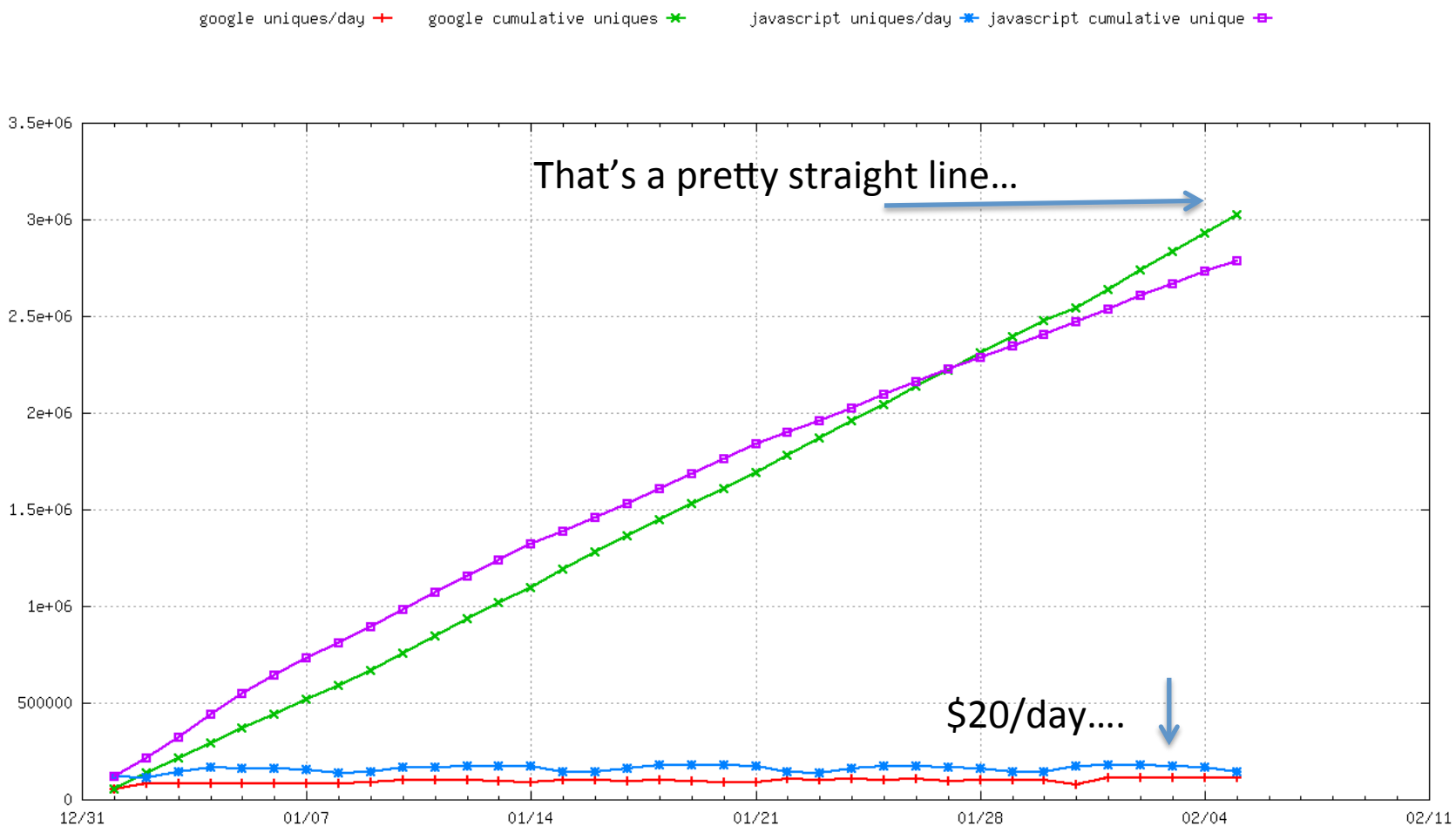
Lots of Unique IP’S



Lots of Unique IP'S



Lots of Unique IP'S



Lots of Unique IP'S

- Both JavaScript and Flash delivering consistent, variant IP address sources every day
  - (Variances now coming in with higher volumes, changes to the experiment)
- Slight sign of bias in JavaScript (same users coming back to same website)
- Google Ads placement demands 'fresh eyeballs' to justify charge of placement based on impressions

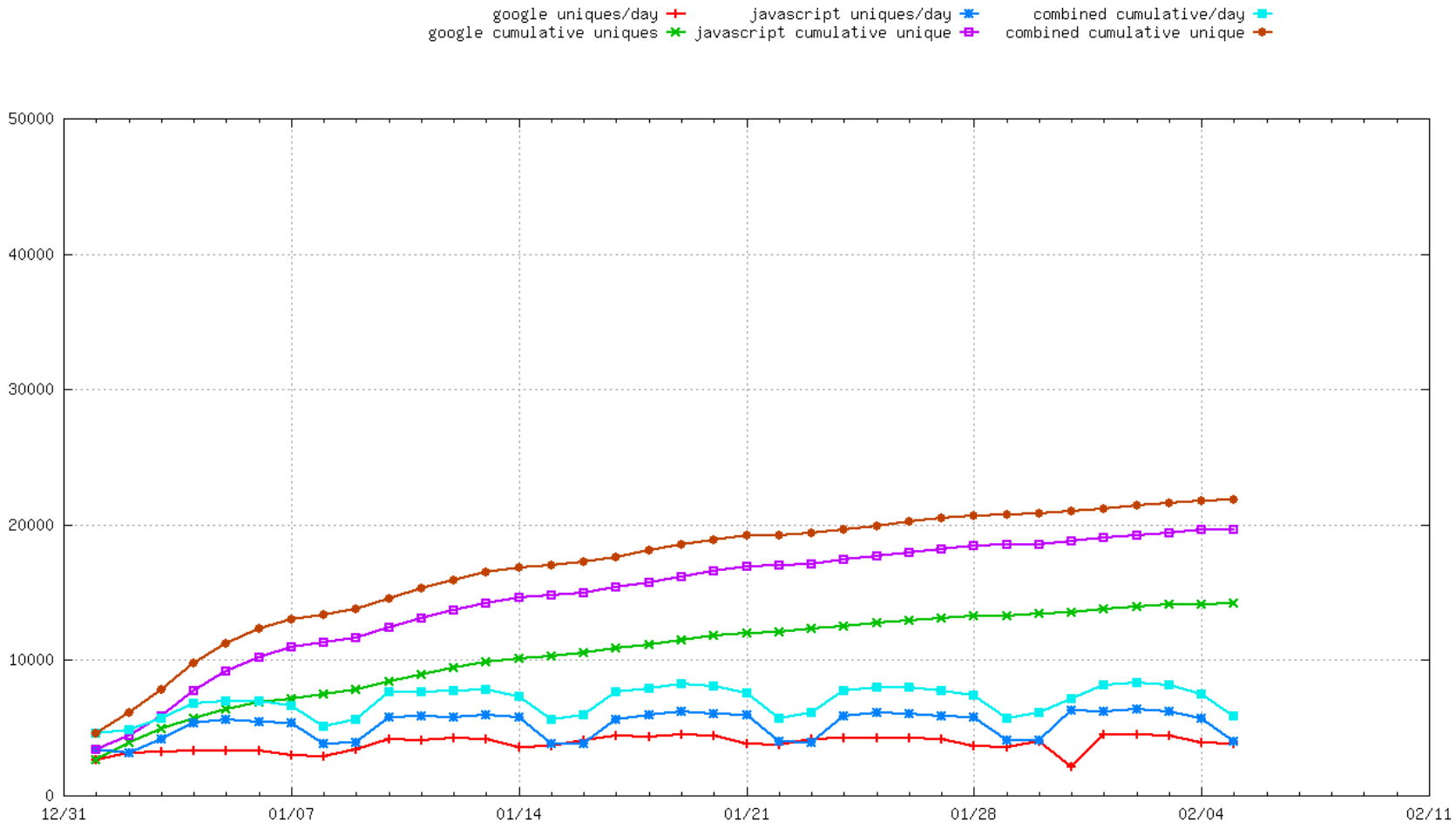
How do the adwords people know we haven’t seen the IP addresses before?

THEY ASK GOOGLE

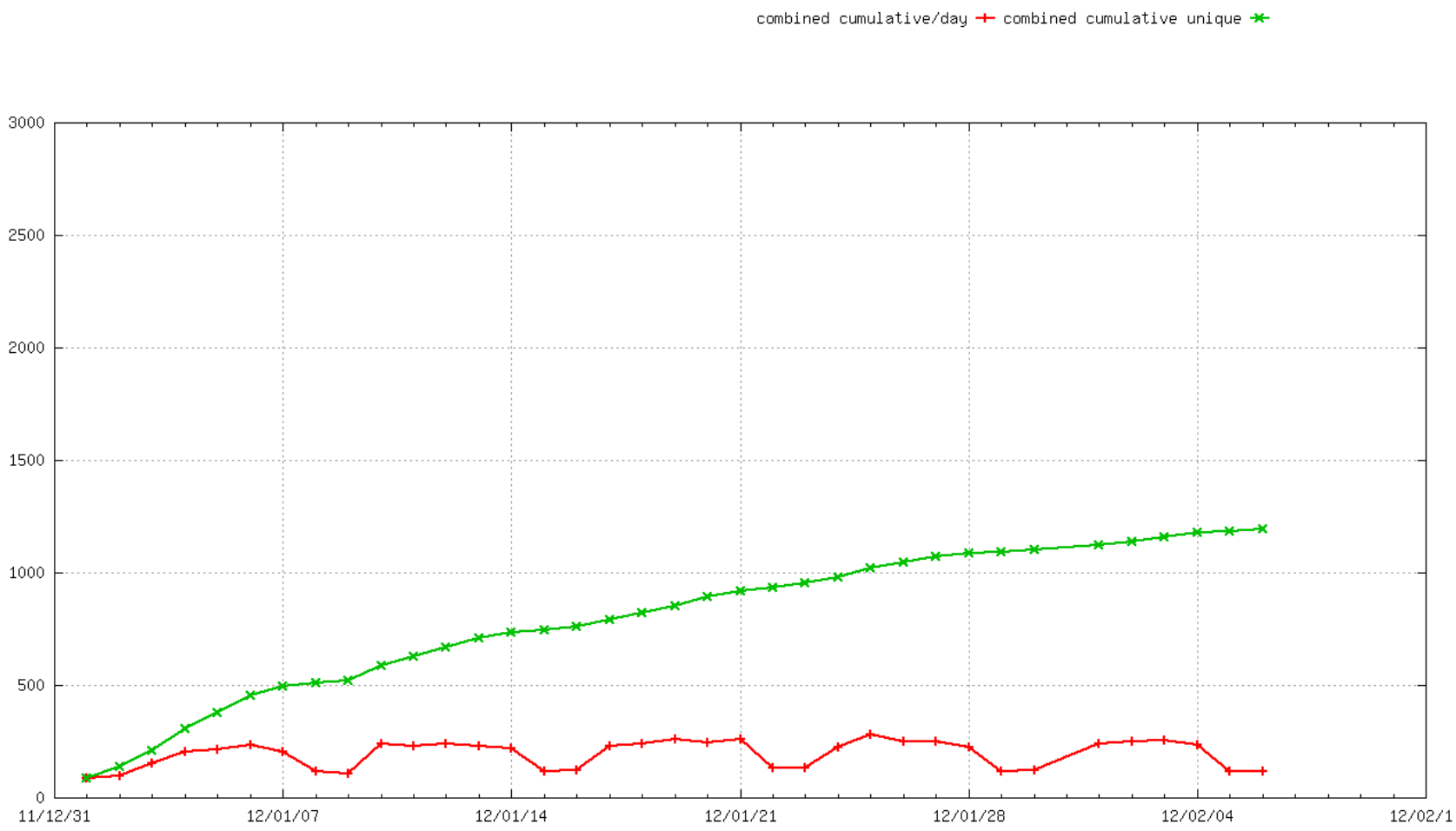
ASN Coverage

- Collect list of unique AS for both IPv4 and IPv6 seen each day
- Collate since Inception
  - Plot to see behaviours

IPv4 ASN Coverage by time



IPv6 ASN Coverage by time



AS Range by time

- JavaScript shows more signs of ‘weekend droop’
  - Google aim to supply consistent load over time so artificially ‘inflate’ traffic against normal usage
    - Remember we’re 1/nth for a very small n of total advertising, so they can make us ‘constant’ when the JavaScript reflects real-world load per website and so can’t mask the ‘drop-off’ of weekend load
- Trending to 25,000+ AS seen in IPv4
- vs 1400 in IPv6
  - Few AS unique to either collection method (js/flash)

AS Range is Representative

- 25,000 ASN in IPv4 is a SIGNIFICANT amount of the global DFZ routed address space
- We believe there is a lot of headroom yet to tap in the unique IP being served to us by the advertising network
- We believe that we can use this data to make observations about global/internet-wide behaviours, as seen at the end-user.
  - Worldwide reach (thin at the margins)
  - We've already applied this method to 'debogon' checks on returned v4, found bogon filter matches.

RTT

- One HTTP fetch of a 1x1.png is around 7 packets
  - Candidates for syn-ack matches in tcpdump
- Each 1x1 fetch requires a crossdomain.xml
  - more syn-ack matches
- Results line requires a crossdomain/1x1
- At least 4, rising 10+ RTT measurements available off-the-wire, per tested IP
  - From 500,000+ distributed points in the network per day
- High variances being seen in some parts of the network (c/f Geoff's presentation)
  - Complements ATLAS and other measures

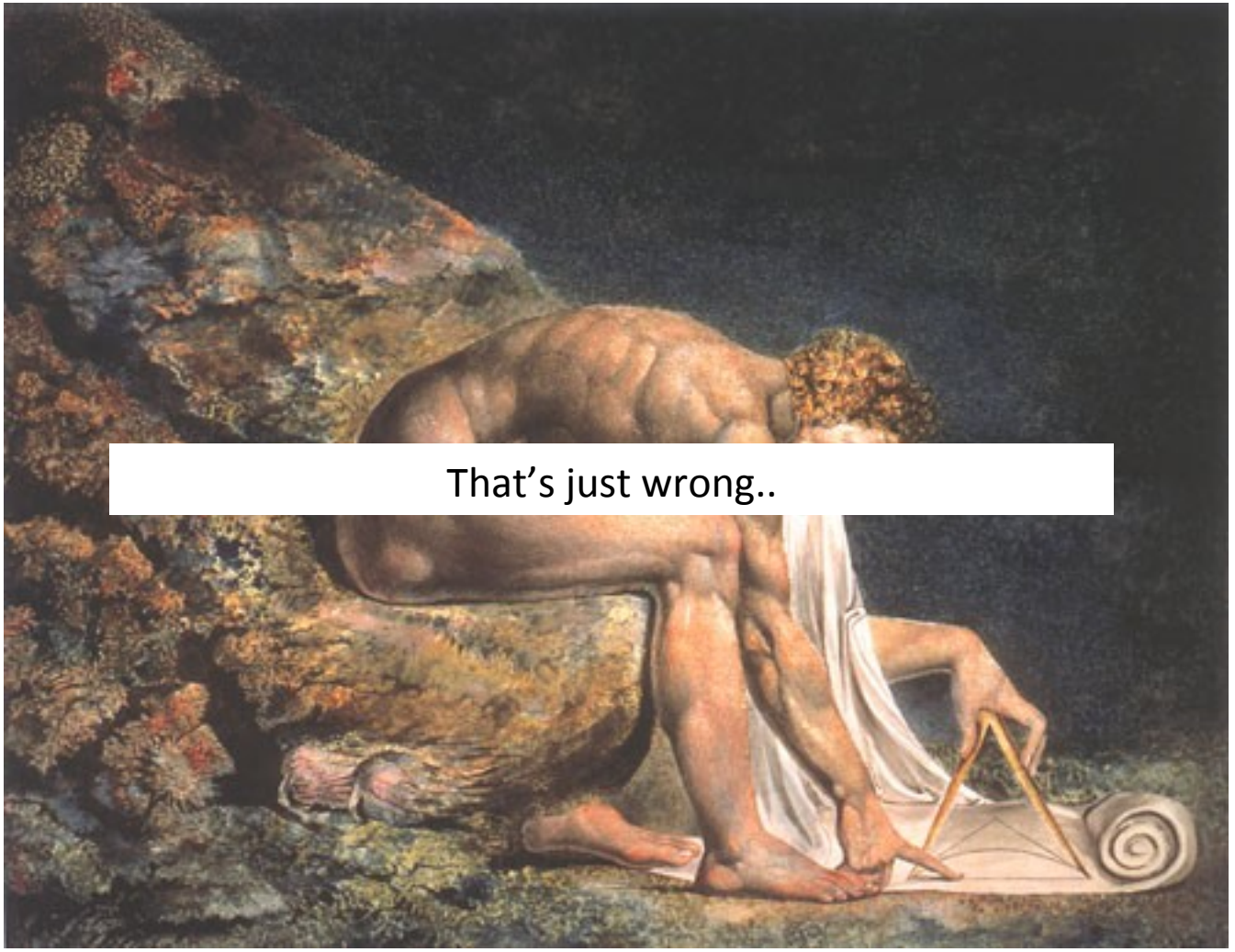
Dealing with the data

- Per-day, unified web, dns, tcp dumps
  - Separate per collection head-end .bz2
- Reduce to Single-line per IPv4/IPv6 instance
  - (client being tested)
  - Add times of dual-stack, IPv6 literal, relative to IPv4 fetches
  - Approx 10Mb per day, 800,000 experiments/day
    - As of April 2012
- Post-process to add
  - Economy of registration (RIR delegated stats)
  - Covering prefix and origin-AS (bgp logs for that day)
- Combine into weekly, monthly datasets (<10Mb)
- Map to publicly visible dataset forms (JSON/CSV)

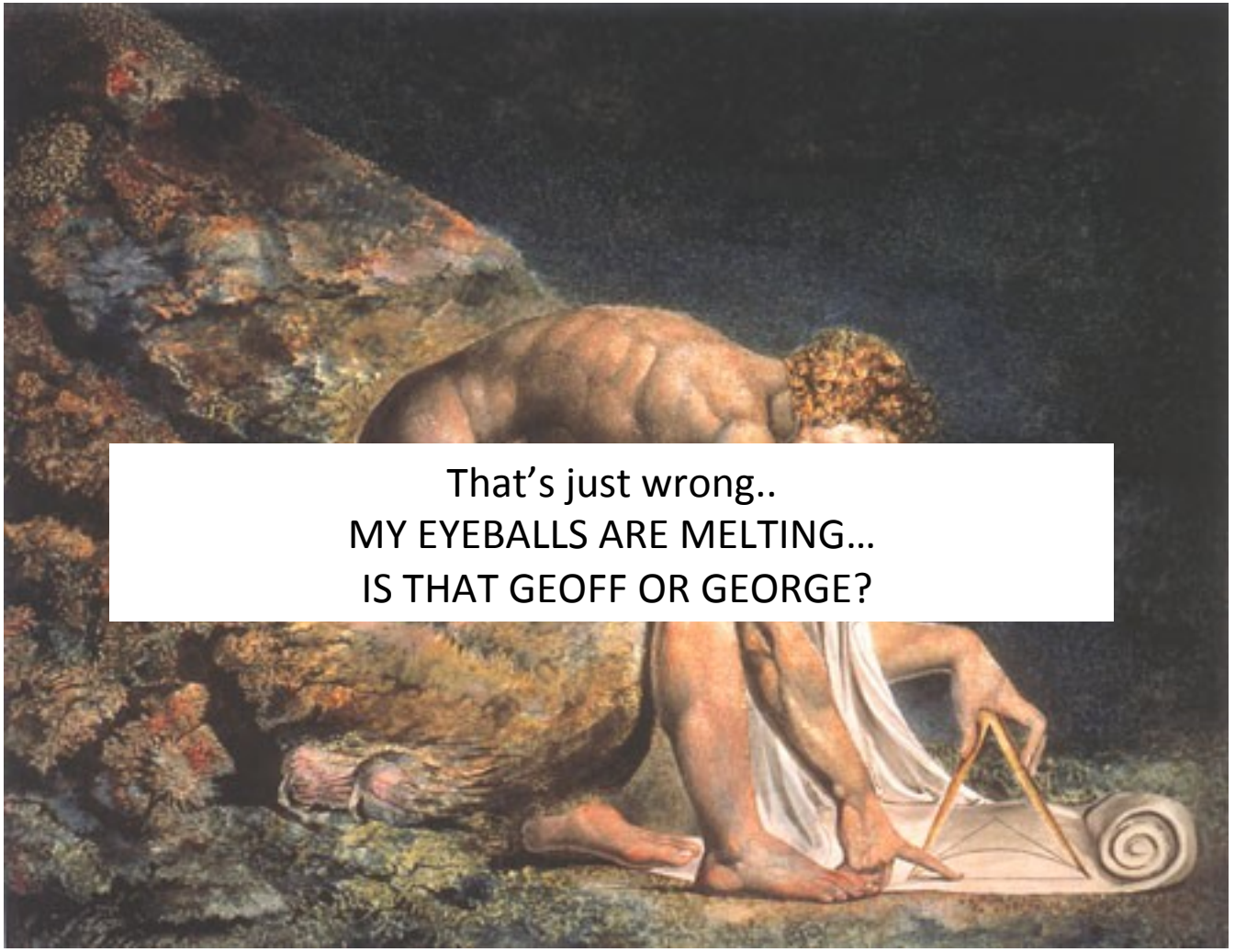
What are we finding?

What are we finding?

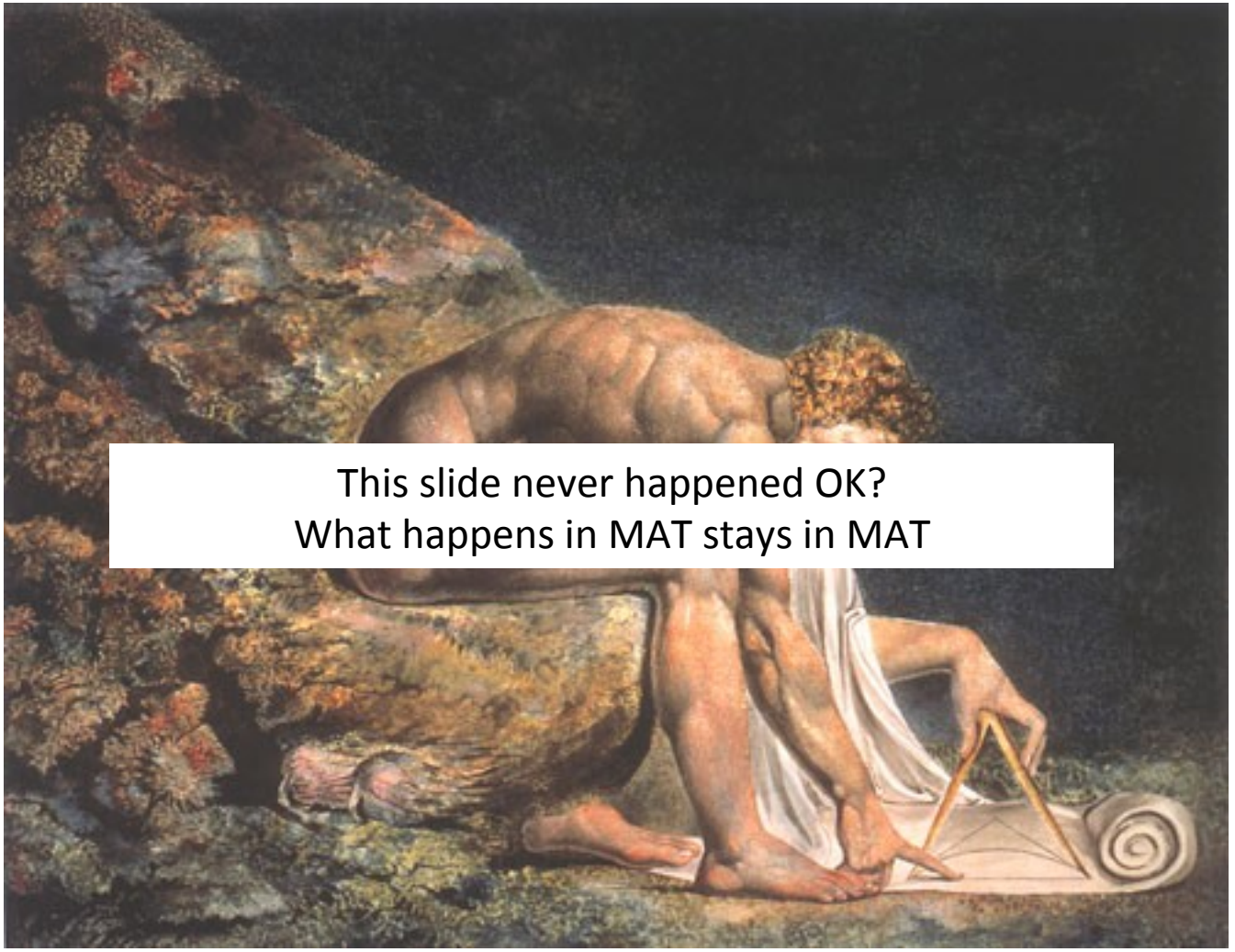
What are we finding?



What are we finding?



What are we finding?



What are we finding?

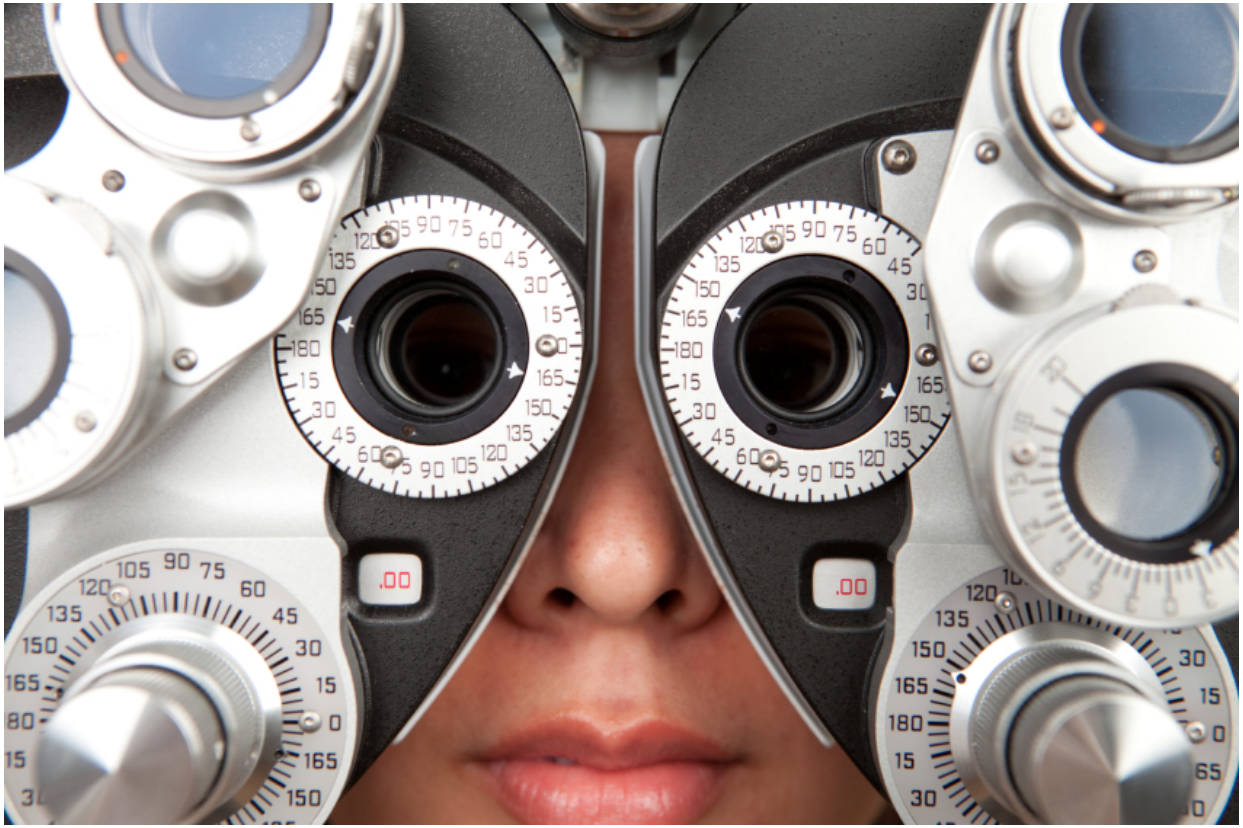
This slide never happened OK?
What happens in MAT stays in MAT

What are we finding?

- http://labs.apnic.net/ipv6_measurement
  - Breakdowns by ASN, Economy, Region, Organisation
  - JSON and CSV datasets for every graph produced, on a stable URL
  - Coming soon: single fetch of the dataset for bigtable map/reduce
- 125+ economies provide >200 samples/interval consistently in weeklies, 150+ at monthlies.
  - Law of diminishing returns as more data collected
  - 200 is somewhat arbitrary, but provides for 0.005 level measure if we get one-in-200 hit.
  - Beyond this, data is insufficient to measure lowside IPv6 preference
- It doesn't match google!
  - Yes, we differ significantly from the published google IPv6 statistics
  - This is interesting! Different measures tell us different things?

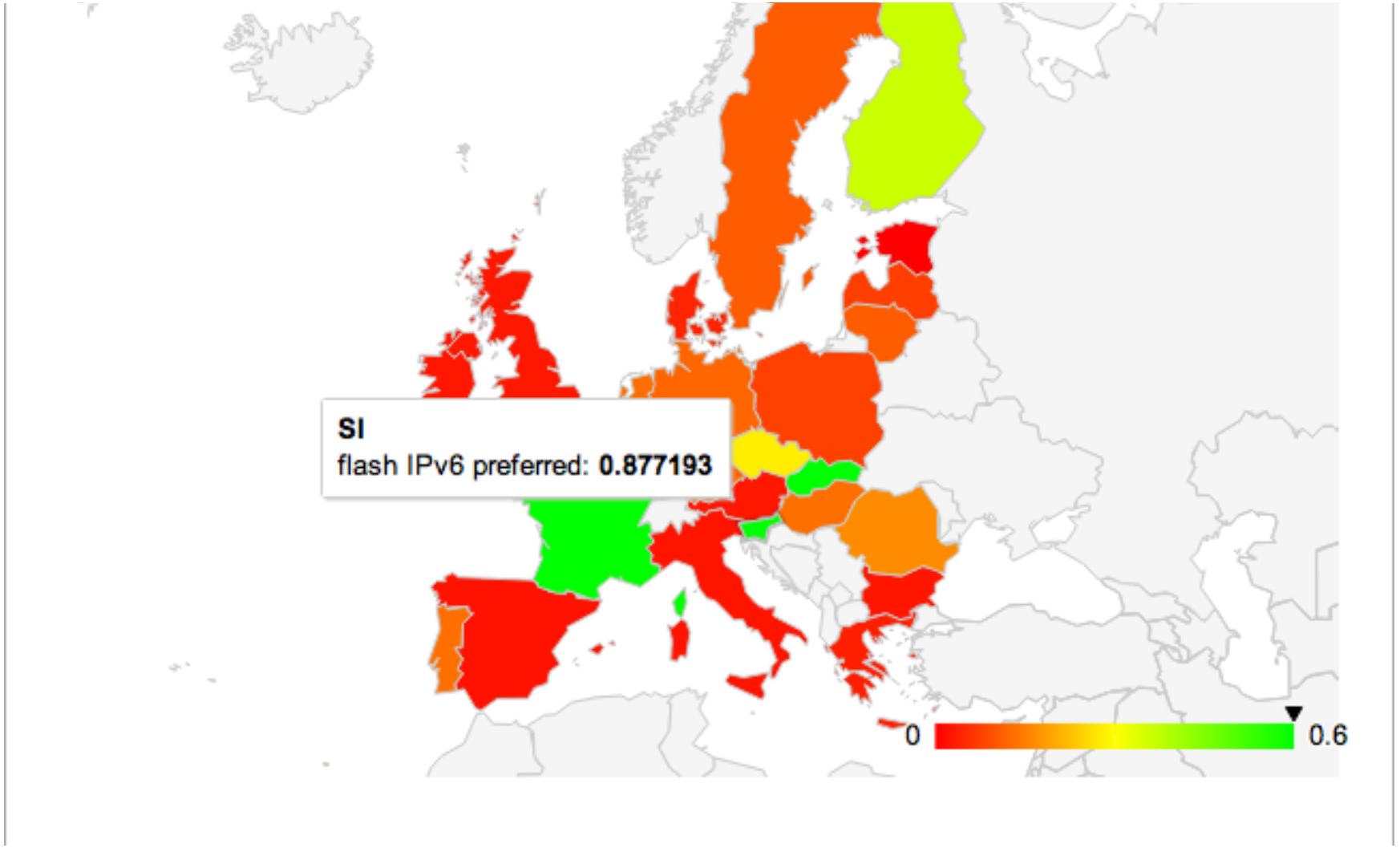
Google visualization API

Google visualization API

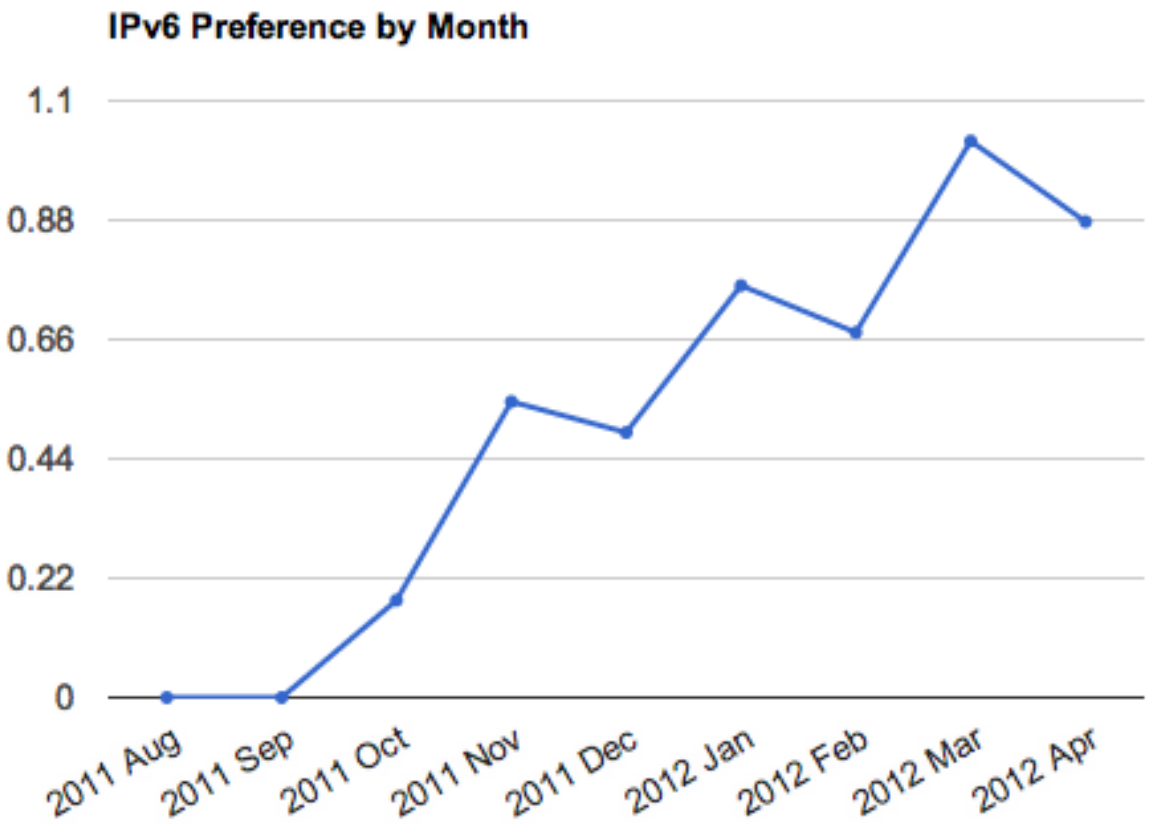


Google visualization API

Google visualization API



Google visualization API

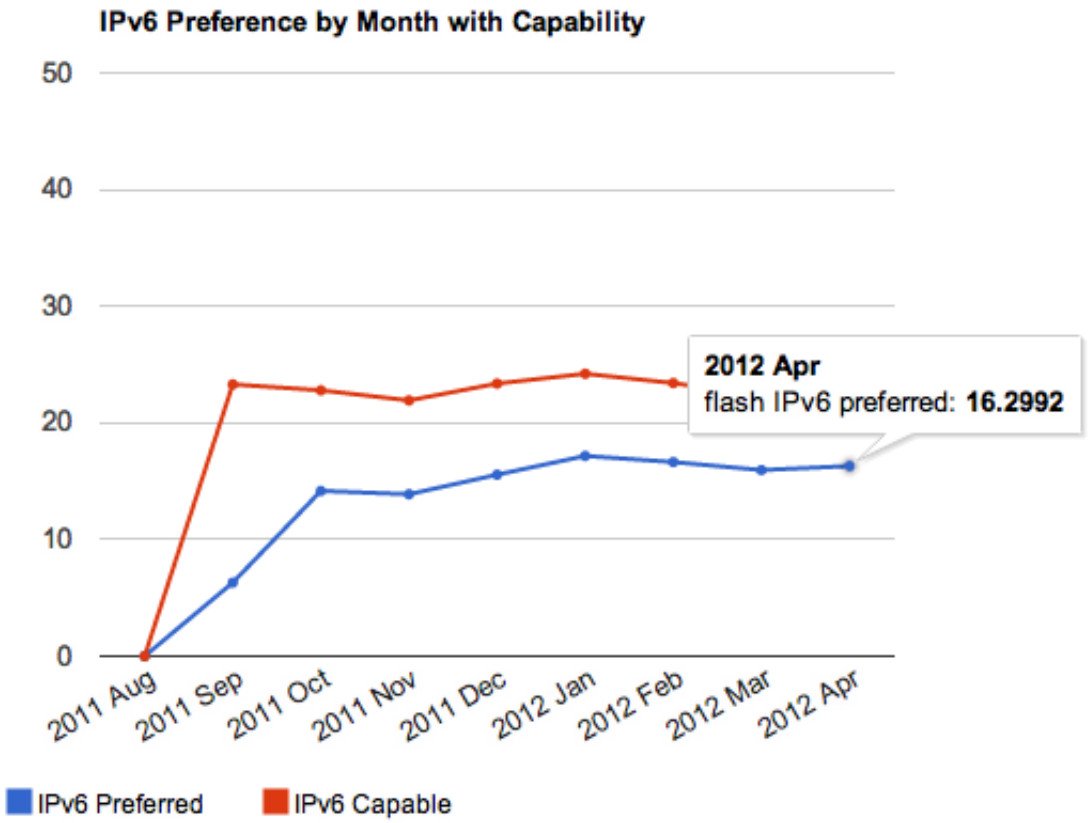


APNIC

IPv6 measurements for AS12322 PROXAD Free SAS

back

IPV6 Preference by Month | IPV6 Capability by Month | IPV6 Preference by Week | IPV6 Capability by Week | Sample Count by Month



JSON CSV

Observations

- The world is 'lumpy' for IPv6 preference
  - We can detect regional/economy-specific variances against world 0.3%
  - France up at 4%, Japan at 1.6%, US at 0.56
- Sample sizes for OECD economies, UN regions are high enough to be statistically useful
  - Emerging Internet economies more marginal
  - Increased uptake would help extend coverage for iso3166, observing we're already at 229/250 for some data, and over 100 economies for 'good' data volumes

AS views

- Take per-day BGP views (AS4608)
- Process IP stream by longest-match covering prefix, emit prefix and Origin AS
  - Side benefit: (cleanroom strip end-user IPs)
- Result: per Origin-AS views of IPv6 preference
  - ~1500 pass the 200-minimum samples test
    - Out of a population of ~5000 ASN in the IPv6 DFZ
  - More data not raising more V6 ASN significantly
    - Law of diminishing returns on who provides V6 to end-users at this time.
- Interesting trend in hop-overs, and which ASN used
  - Signs we are detecting CGN/NAT/Economy-border filters?

IPv6 measurement

- Penetration rate of IPv6 into the global AS economy is slow
- No signs of 'game changer' flip to IPv6 at the end-user yet
- Widely distributed hop-over for IPv6 being seen.
  - due to the CPE gap ?
  - Even IPv6 enabled ISPs have customers tunnelling over the air-gap
- Much more information about IPv6, global internet behaviour is in the data
  - "watch this space" –long-term investment in measurement, ongoing.
  - Better datasets, BigTable map/reduce
  - Collaborations with "the usual suspects" to extend the experiment

Conclusions

- JavaScript on web pages, and Flash in advertising networks are viable tools for a broad-range, high volume data collection of simple construction
- Low cost of entry, high return on investment for measurement activity
- Many Internet-wide, unique IP Addresses are visible. 33,000 ASN (v4) to 1600 ASN (v6) visible (April 2012)
  - 75%+ coverage of V4, 25%+ of IPv6 ASN In DFZ
- We have a LOT more information to get out of this investment. RTT, MTU/MSS, pMTU &c &c

Acknowledgements

Thanks to RIPE NCC, Google and ISC

Acknowledgements

Thanks to RIPE NCC, Google and ISC

Acknowledgements

Thanks to RIPE NCC, Google and ISC

Acknowledgements

Thanks to RIPE NCC, Google and ISC

Acknowledgements

Thanks to RIPE NCC, Google and ISC

More info, JSON/CSV data (daily updates)

http://labs.apnic.net/ipv6-measurement/

?

APNIC