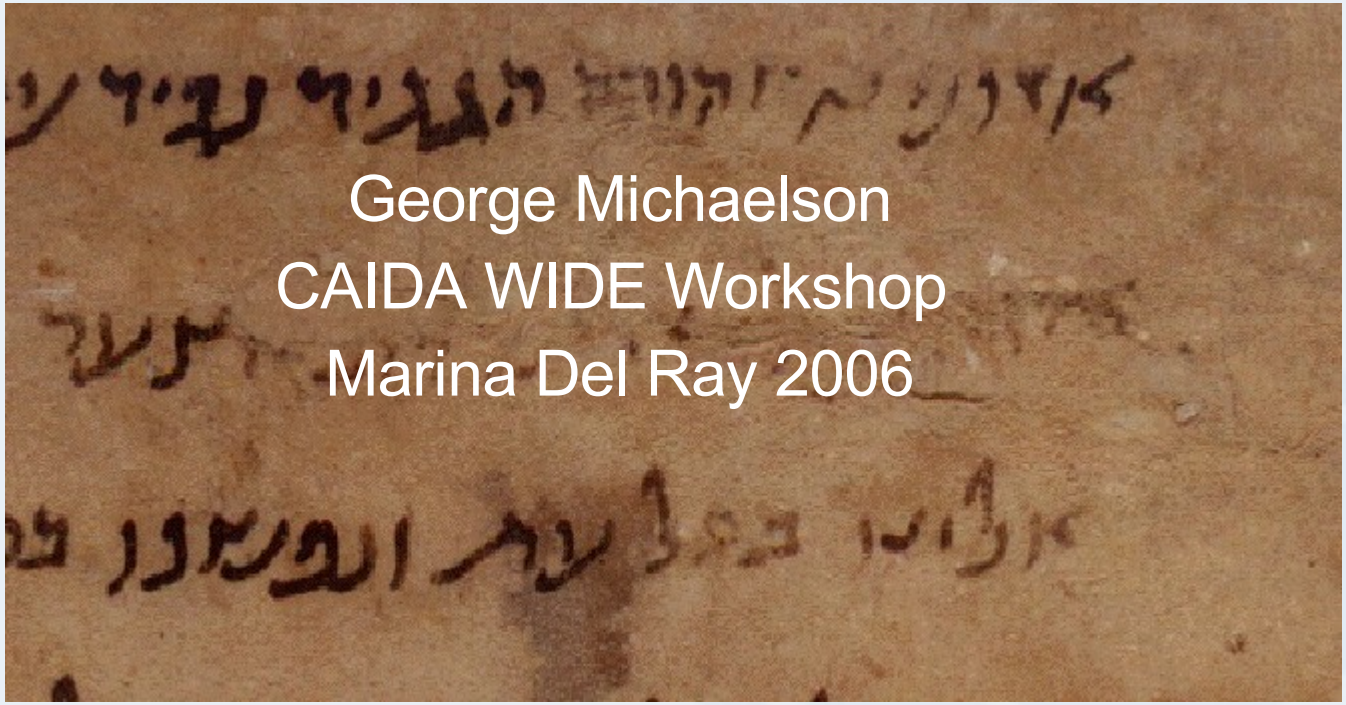
Some Lacunae in APNIC DNS Measurement



George Michaelson

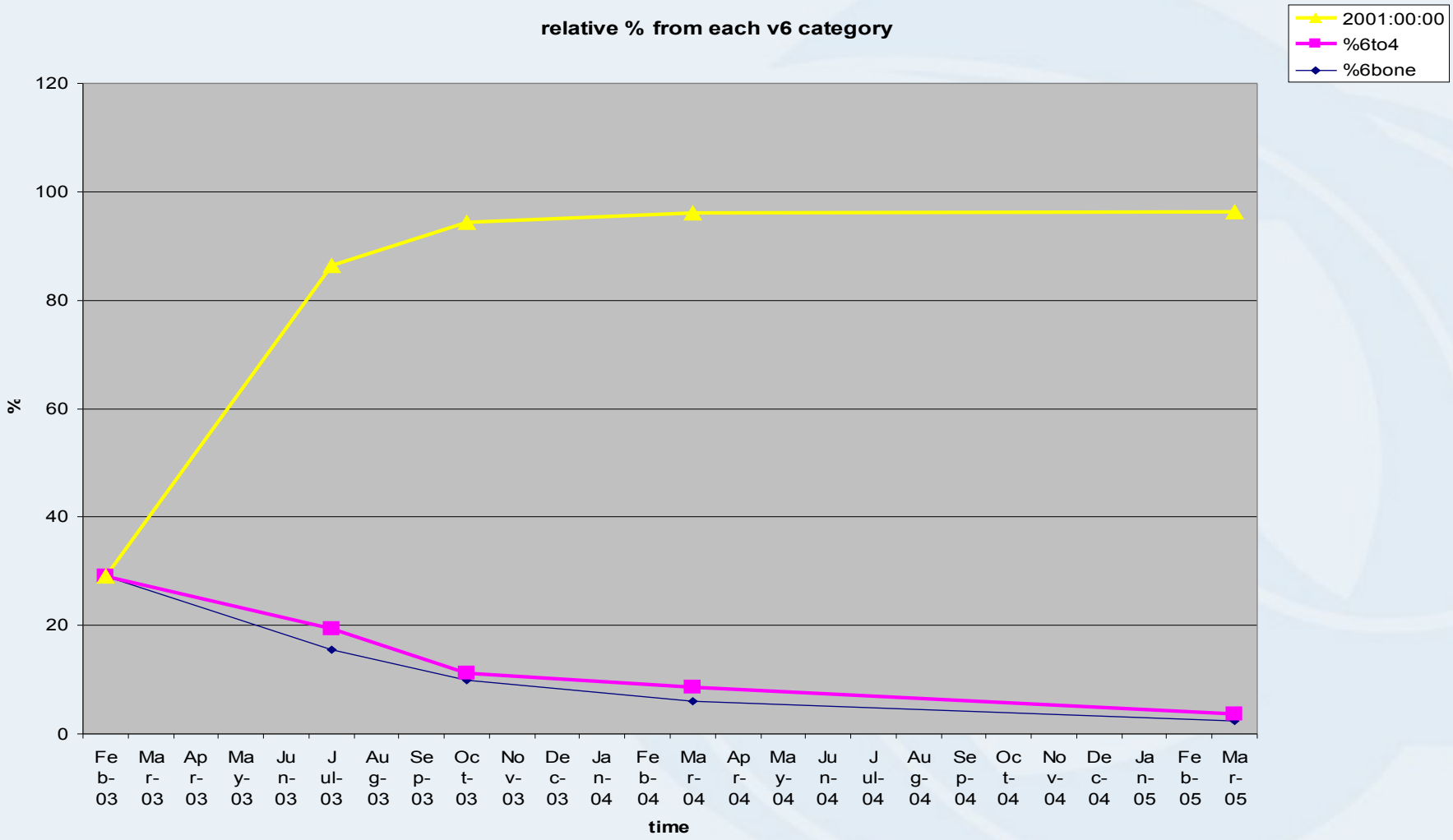
CAIDA WIDE Workshop

Marina Del Ray 2006

Backtracking…

- Same DNS measurement since 2001/2..
- Re-installation of nameservers forced re-installation of stats gather processes
  - Discovered dataloss, tanstaafl
- Reviewing held data, found 5 datasets
  - Tcpdump port 53, mostly text, some raw packets
  - 2003-2005, 6 samples
- What did I miss? What might be interesting?
  - Never measured query origin protocol family
  - Turns out we're taking Ipv6 transport query

Where in V6 do queries come from?



Where in V6 do queries come from?

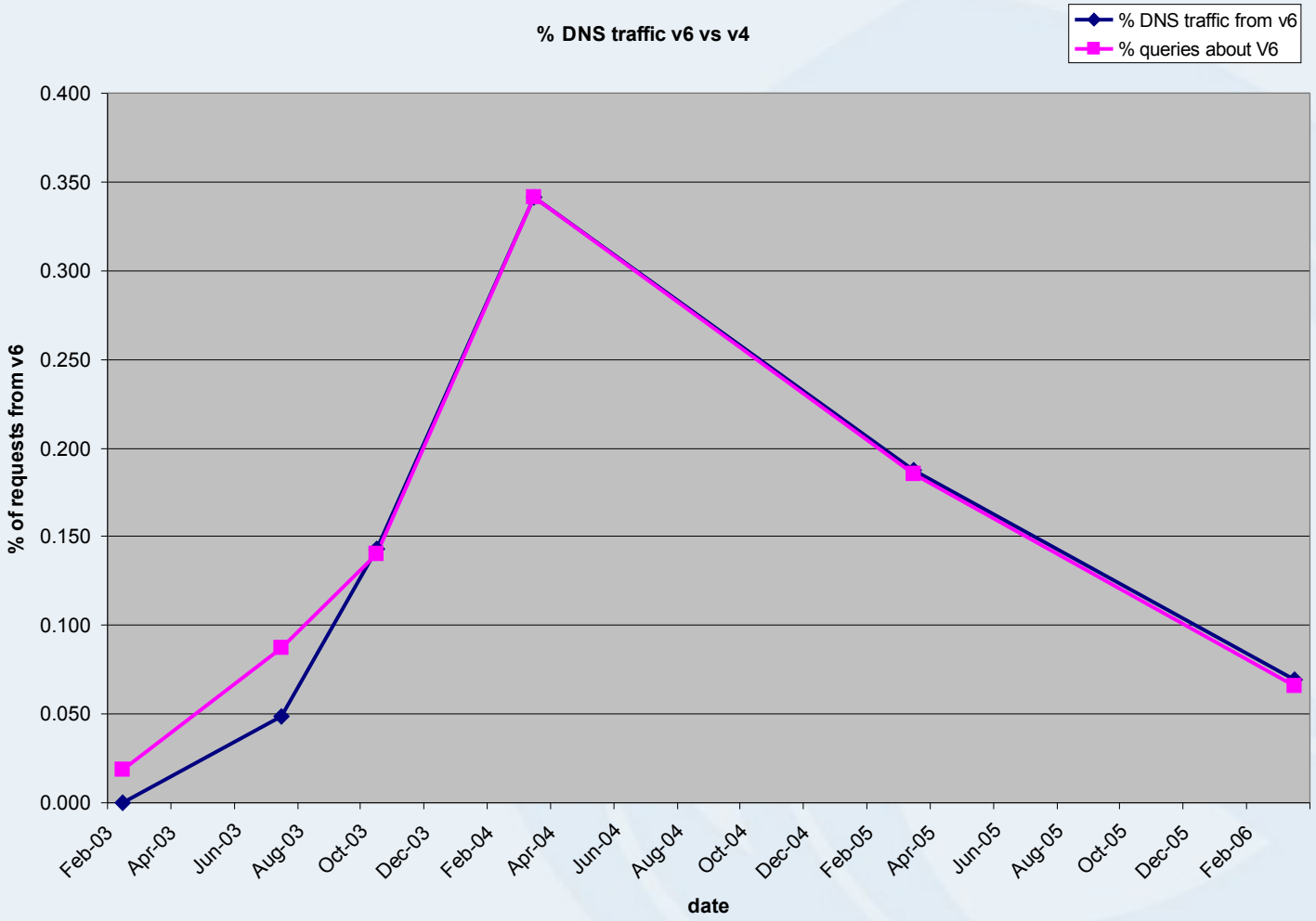
- reading from file /dev/stdin, link-type LINUX_SLL (Linux cooked)
- 01:17:21.687611 2001:40a8:3000:1:202:dead:babe:cafe.4158 > 2001:dc0:1:0:4777::140.53

3ffe may be declining but 2002 is alive and well..

- 6to4 sourced data as a % is dropping, but it is active, and growing. (just not as fast)
- APNIC also runs the 6to4 reverse-DNS registry: 184 entries

AT: 1	CH: 6
CZ: 1	PL: 6
ES: 1	SE: 7
IL: 1	UK:11
LV: 1	EU:12
AP: 2	FI:12
CA: 3	FR:14
LT: 3	DE:19
NL: 4	SI:20
AU: 5	US:24
JP: 5	IT:26

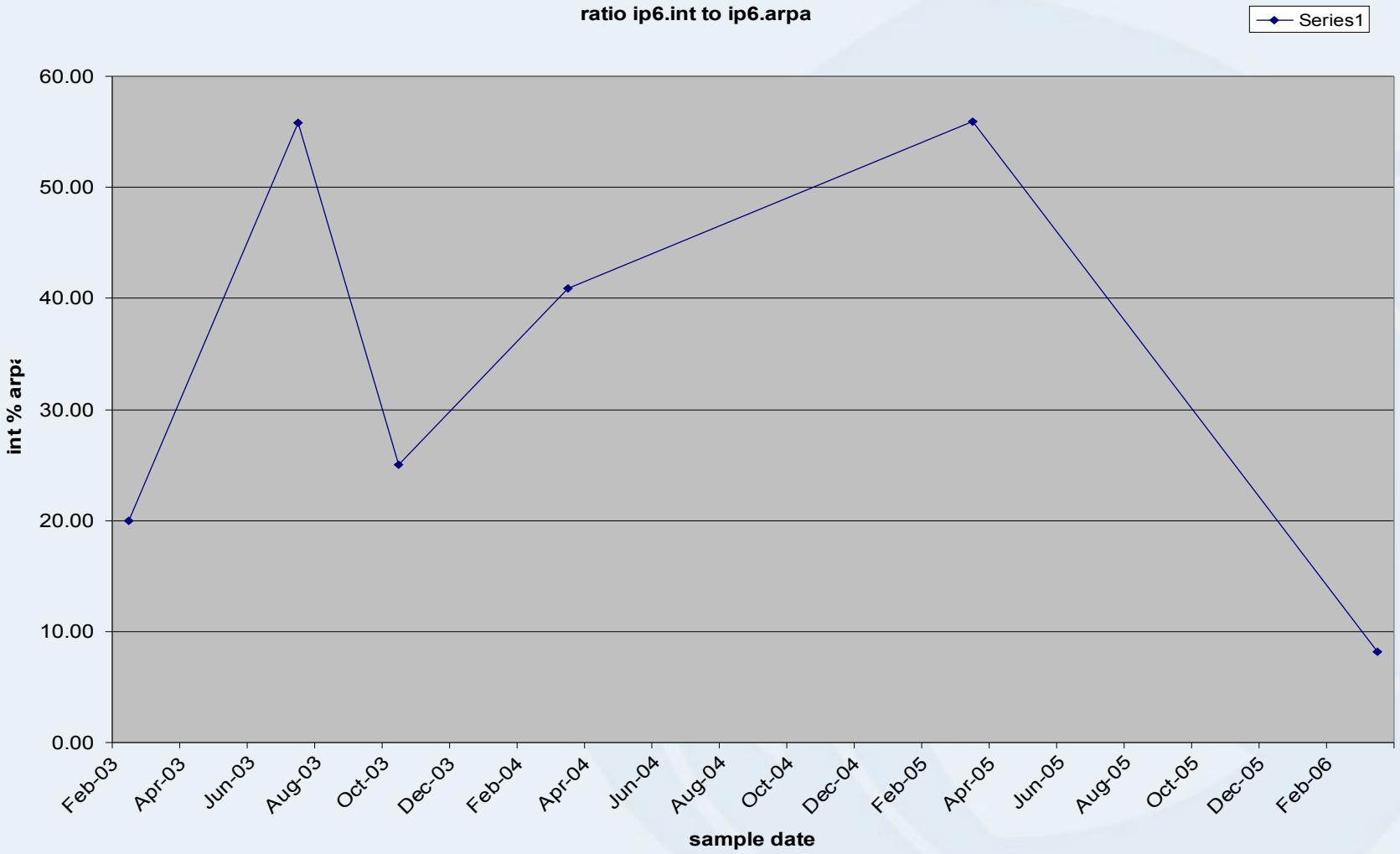
V6 as % of V4, V6 query as % of V4



V6 as infrastructure

- Nothing until Feb 2006
- In Feb, I see 15 (!) instances of:
  - V6 infrastructure doing V6 transport to query about a V6 client (reverse-DNS lookup)
  - But the majority V6 use is still using V6 transport to query V4 reverse-DNS
- Side effect of dual-stack with preference for V6 ‘automatically’ applying?
- Direct from server? Resolver?
- Worth tracking..

ip6.int is finally dying?



Bitlabel sickness in the wild…

- Target servers were any NS of ip6.arpa
  - Not the subs, the delegation point itself
- One instance, March 2005, from Italy...
  - (1 1min sample)
- 5950 instances Nov 2005
  - 638 discrete Ipv4 addrs, 3 day sample
- 5154 instances Feb 2006
  - 432 discrete Ipv4 addrs,1 day sample
- 23 instances Mar 2006
  - 10 discrete Ipv4 addrs, 10 min sample
  - looks to have peaked (~ 3000 per day?)
- Was worldwide, tracked specific linux glibc deployment

DNSSEC in the wild

- Seeing instances of DS & DNSKEY
- Two servers now bind9.3, enabled, secondary RIPE-NCC reverse DNS
  - **Significant increase in on-disk, network, memory and CPU cost**
  - At least as measured in userspace. On the wire, its not yet so clear.
- Sec1 (au) (1 eu, 1 nl, 1 ru) DS, DNSKEY
- Sec3 (jp) (1 uk, 2 eu) DS
- Why are they coming to me from Europe?
  - 1 packet RTT alg not finding best server?
  - Could the cost of DNSSEC make me look good?

Next Steps

- Re-implement stats gather
  - Try to keep more raw samples for more backtracking
  - Count query protocol, matrix of proto:query
  - DNSSEC needs more attention
- Tools
  - Tcpdump needs updating. (newer DNS Type and QType codes)
  - ‘sample for <n> seconds’ would be useful
    - Currently using packetcount limits
- Continue ip6.int measure beyond 6/6/6
  - How long will it take the old code to die?
  - De-listing will make it hard to track this…

Why these measurements?

- APNIC does other measurements for capacity planning, load, service reliability using bind logs, bindstats, munin/nagios etc
- Harder to answer some queries from these logs:
  - Who comes to me to ask questions?
  - Where are they asking about inside the zones?
  - What <odd> traffic am I taking?
- Randy Bush suggested an ip6.int/arpa measurement
- Was interested in prefix-by-economy measures from prior work before APNIC

Why might reverse-DNS be interesting?

- Its Server-side query:
  - Server backtracking on connecting clients
  - Corporate Entities with resolvers, ISPs
- Natural chokepoint as a function of the 5 RIR and their listed secondaries
  - Less busy than roots
  - More interesting than IX snarf? Probably not
  - Possibility of reasonably complete view?
- Appears to have strong correlators to real-user activity
  - Analysis by economy follows diurnal trend
  - Midnight log rollover effects skew this
  - Evidence of RTT preferencing by economy

Odd measures: economy by time

- Intrinsic vs Extrinsic DNS
- Map src, dst economy in 2D
- Render as time-series
- Colourcode 'density' of queries in time to show hotspots
- Animation shows (I think)
  - Intrinsic (own-cc to own-cc) traffic patterns
  - Strong lines for specific economies
    - China, Japan, USA
  - Potentially interesting hotspots of inter-economy traffic

Odd Measures: Timezone by time

- Map economy to timezone
  - ok. fudge china (crosses 10hrs of TZ)
  - Fudge USA/Canada (cross 4)
- Render as time series with some indication of where daytime is
- Can you see any timezone specific behaviours?
- Is GMT midnight a significant time worldwide?
- *<this is deeply painful to watch for any length of time>*

Nevils packetsize distributions in APNIC

- Ns3: no DNSSEC. Sec3 has DNSSEC