

Happy Eyeballs

Vaibhav Bajpai and Jürgen Schönwälder

Leone Plenary, Rome

School of Engineering and Sciences
Jacobs University Bremen
Bremen, Germany

February 2013

overview

- ⦿ precursor
 - ⦿ happy eyeballs
- ⦿ experiment
- ⦿ data analysis
- ⦿ open questions

happy eyeballs

- ⦿ `getaddrinfo(...)` behavior:

- ⦿ returns list of endpoints in an order that prioritizes IPv6 upgrade path
- ⦿ order is dictated by [RFC6724] and `/etc/gai.conf`
- ⦿ if IPv6 is broken, application is unresponsive in order of seconds

- ⦿ happy eyeballs algorithm:

- ⦿ initiate a connection with the first endpoint, give it 300ms
- ⦿ switch over to a different address family otherwise
- ⦿ the competition runs fair after 300ms

happy

- ⦿ simple TCP happy eyeballs [RFC 6555] probing tool
 - ⦿ it probes all endpoints of a service name though
 - ⦿ helps compare different happy eyeball algorithms
 - ⦿ uses getaddrinfo(...) to resolve service names to endpoints
 - ⦿ uses non-blocking connect(...) to connect to all endpoints of a service
 - ⦿ uses a short-delay between connection attempts to avoid SYN floods
 - ⦿ returns connection timings to each endpoint
 - ⦿ produce human-readable or machine-readable output
 - ⦿ file locking capability
 - ⦿ cross-compiled for openWRT, running on SK boxes.

```
>> ./happy -q 1 -m www.google.com www.facebook.com
HAPPY.0;1360681039;OK;www.google.com;80;173.194.69.105;8626
HAPPY.0;1360681039;OK;www.google.com;80;2a00:1450:4008:c01::69;8884
HAPPY.0;1360681039;OK;www.facebook.com;80;2a03:2880:10:6f01:face:b00c::8;170855
HAPPY.0;1360681039;OK;www.facebook.com;80;31.13.72.39;26665
```

service names list

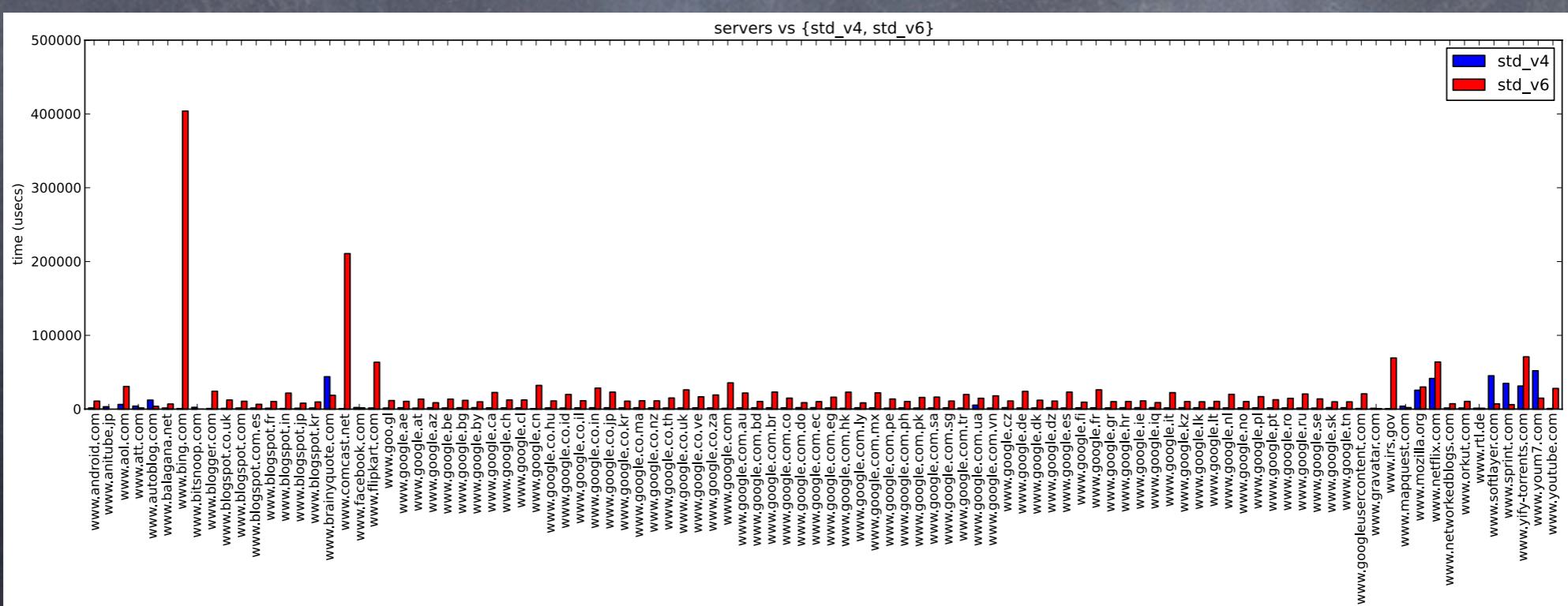
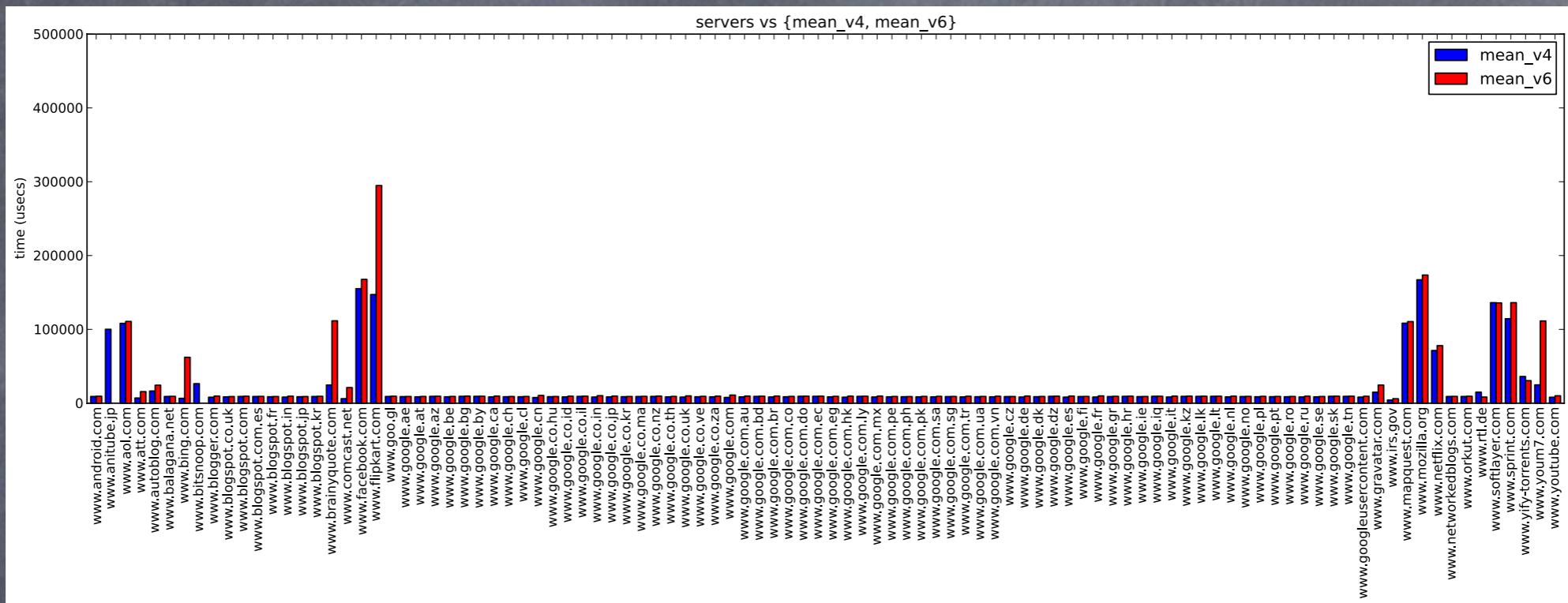
- ⦿ what is a good set of service names?
- ⦿ HE.net maintains a list of top 100 dual-stacked service names
 - ⦿ they use 1M service names from alexa
 - ⦿ prepare a top 100 dual-stacked service name list from this input
 - ⦿ some domains we expect are missing from the list
 - ⦿ some services only provide a IPv6 endpoint on prepending a www
 - ⦿ HE.net does not follow CNAMEs
- ⦿ amazon has made 1M service name list public
 - ⦿ we use it and script it ourselves to explicitly follow CNAMEs

experiment design

- ⦿ dual-stacked service name list:
 - ⦿ custom generated from amazon' publicly available top 1M service names
- ⦿ measurement points:
 - ⦿ native IPv6, 6in4, IPv6 Teredo, 4in4, native IPv4
 - ⦿ located at Bremen, Amsterdam, Braunschweig
 - ⦿ tunnels popping out at Frankfurt, Berlin, New York
- ⦿ runtime frequency:
 - ⦿ every 3 minutes and collect data in a local file
- ⦿ data collection:
 - ⦿ sync the local file with a remote data collection endpoint
- ⦿ measurement cycle length:
 - ⦿ 1 week

how does IPv6 compare in
performance to IPv4?

connection times (mean/std)

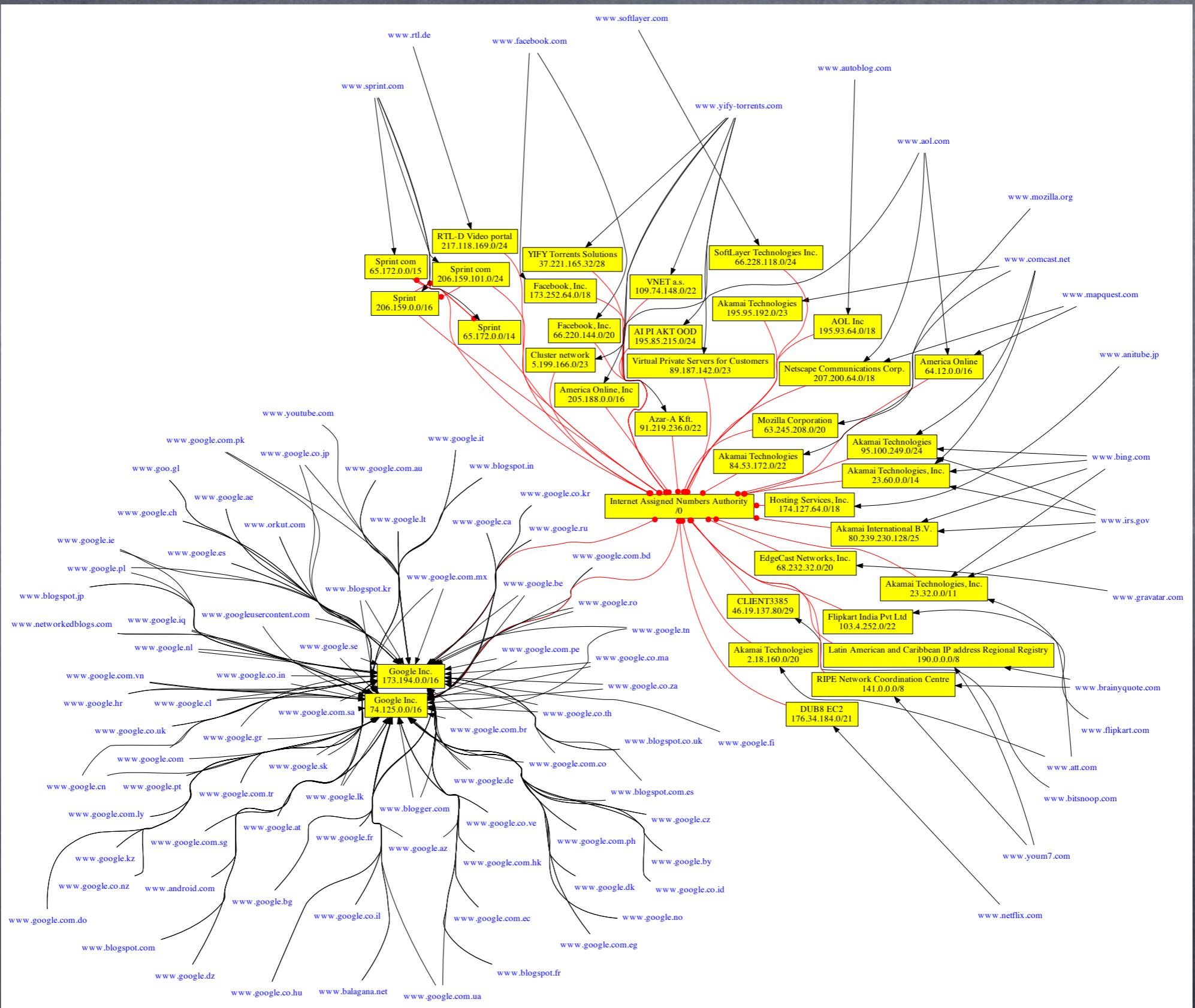


data analysis

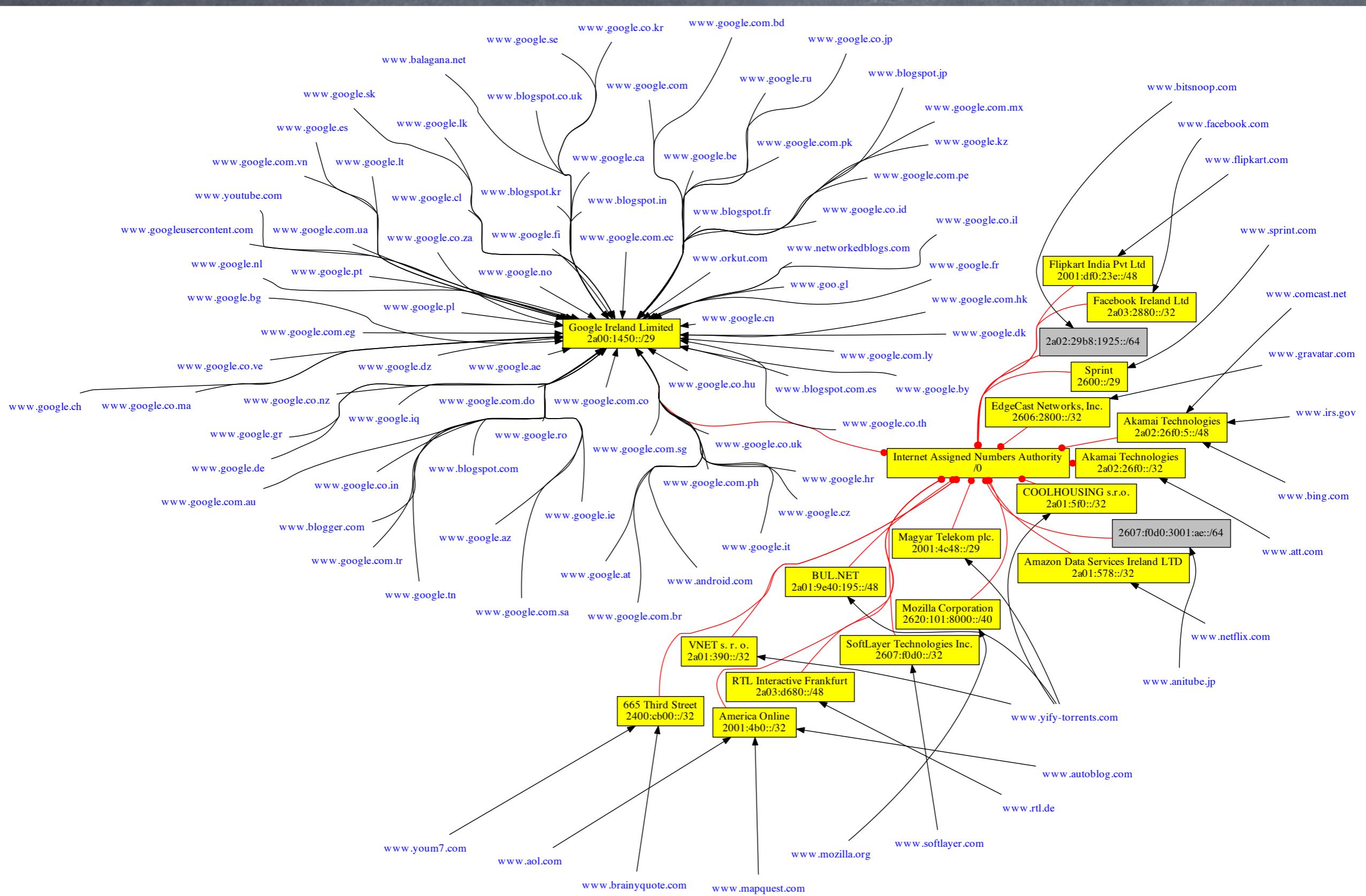
- ⦿ a number of dissimilar services show similar mean/std
 - ⦿ maybe services are either resolving to the same endpoint or set of endpoints belonging to the same allocated prefix
 - ⦿ for each endpoint make a whois REST call to whois.arin.net
 - ⦿ if the endpoint was allocated by a different RIR, make a REST call to whois.ripe.net. RIPE API can also delegate requests to APNIC and AFRINIC
 - ⦿ grab allocated prefix, organization and regional registry

do a major portion of services
centralize on CDNs?

IPv4 aggregation cloud



IPv6 aggregation cloud



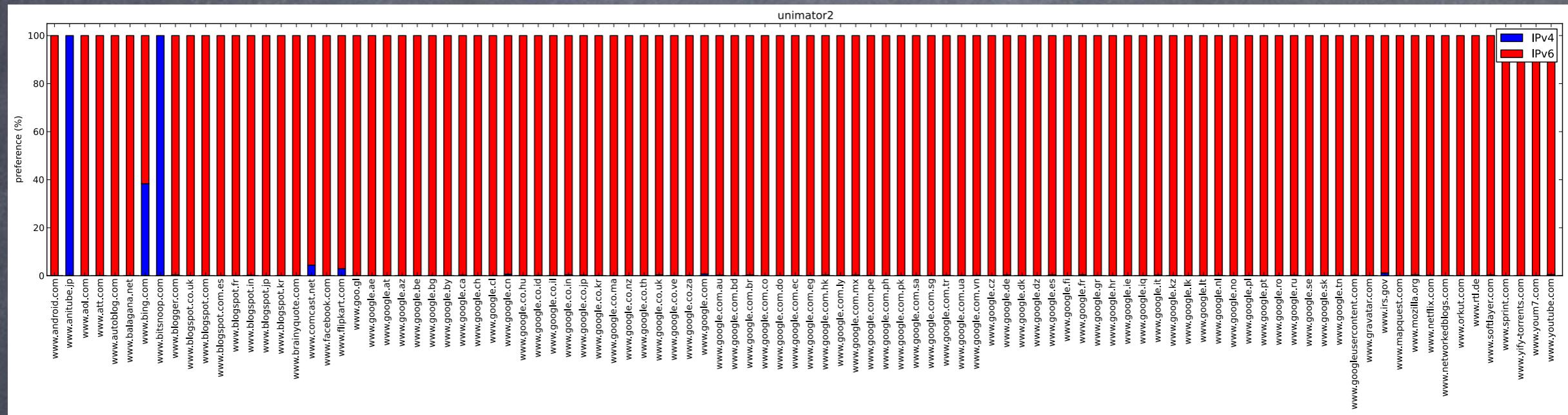
data analysis

- major portion of the services map to allocated prefixes owned by popular organizations: Google, Akamai, Amazon

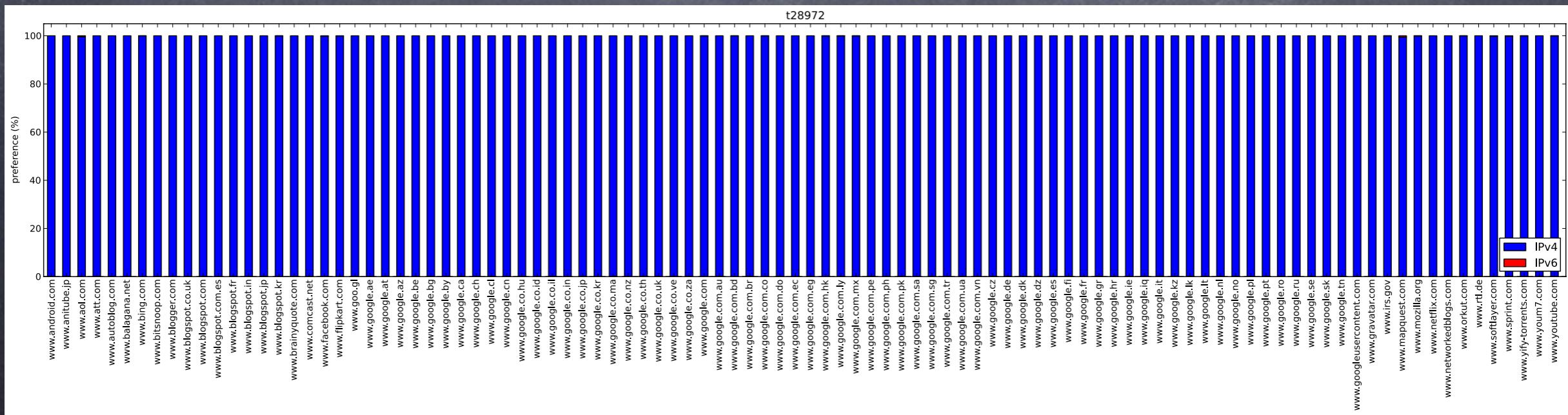
to what extent is IPv6 preferred
when connecting to a dual-
stacked service?

IPv6 preference %

Native IPv6



IPv6 teredo tunnel



data analysis

- ⦿ IPv6 is preferred hands-down on a native IPv6 and 6in4 connection
- ⦿ even if one has Teredo, a happy-eyeballed application will NEVER use IPv6 to go to a dual-stacked service
- ⦿ Teredo is only useful to connect to a IPv6-only endpoint

open questions

- ⌚ to what extend does the CDN aggregation depend on regionalization?
- ⌚ how slow (or fast) a happy-eyeballed winner is compared to the loser?
- ⌚ endpoint aggregation based on announced BGP prefixes instead of using address blocks allocated by the RIR

on a separate note

- ⌚ can we identify a CG-NAT from a home gateway?
- ⌚ can we identify multiple layers of NAT from a home gateway?