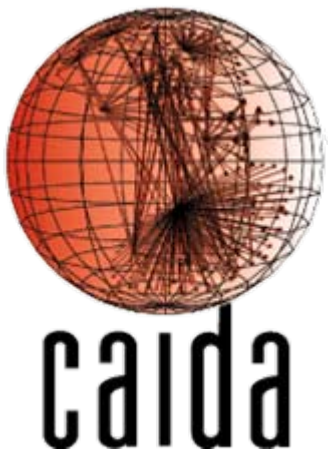
Measurements of traffic in DITL 2008

Sebastian Castro
secastro@caida.org

CAIDA / NIC Chile



2008 OARC Workshop – Sep 2008 – Ottawa, CA

Overview

- DITL 2008
- General statistics
- Query characteristics
  - Query rate comparison
  - Client rate comparison
  - Query types
  - Distribution of queries/clients
- Client classification
  - Per reverse names
  - IP TTL Histogram
  - Reputation Score
- Source Port Randomness evolution
- Invalid traffic
  - Comparison with 2007
  - Exploration of sources
    - Recursive queries
    - A-for-A
    - Invalid TLD

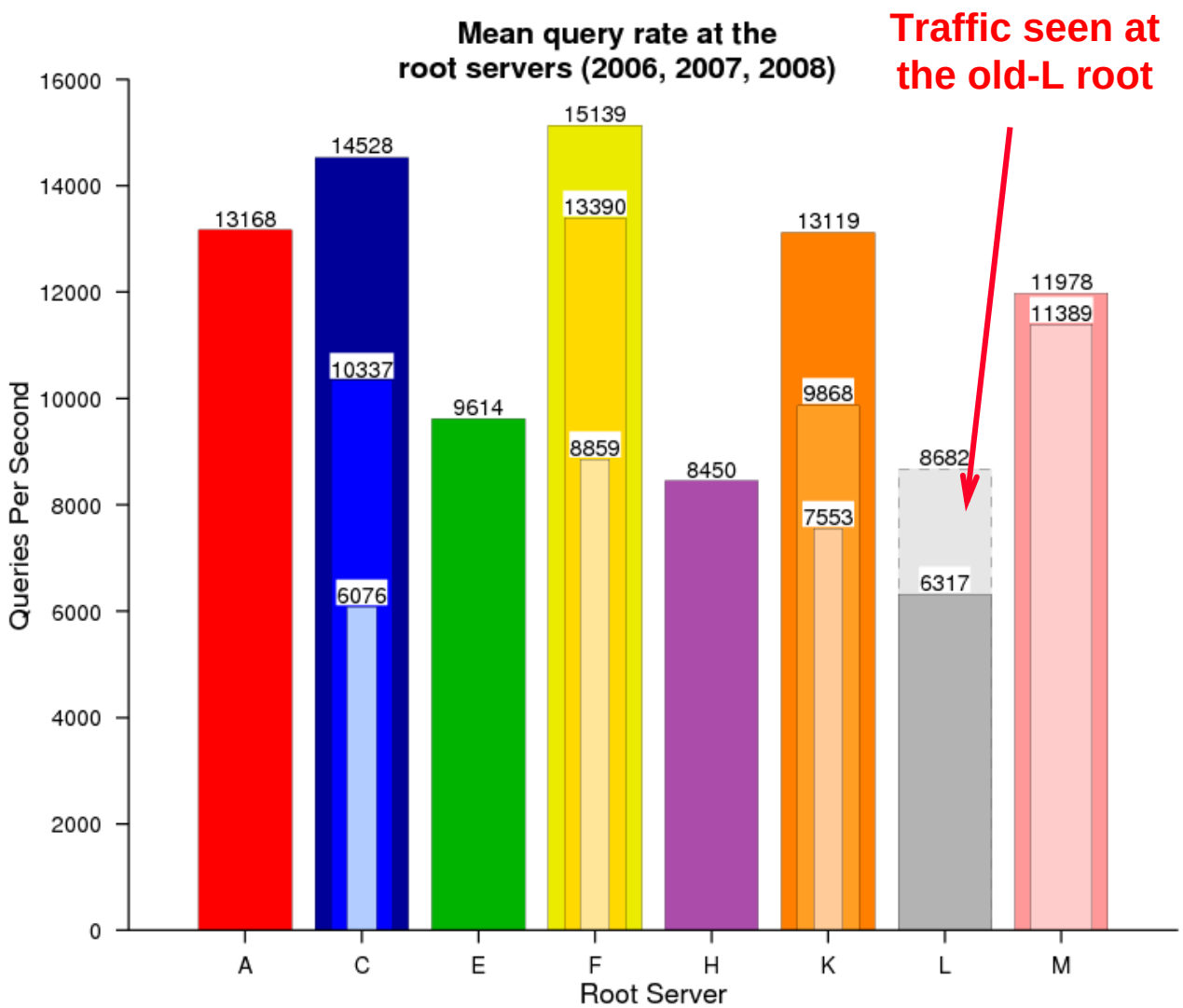
DITL 2008

- Particularly successful in terms of variety of DNS traffic
  - 8 root servers
  - 2 old root servers
  - 2 ORSN servers
  - 5 TLD (1 gTLD, 4 ccTLD)
  - 2 RIR
  - 7 instances of AS112
  - Cache traces from SIE and University of Rome
- Also includes traces and measurements

General statistics

	DITL 2007	DITL 2008	
Dataset duration	24h	24h	
Dataset start	Jan 9, noon (UTC)	Mar 19, midnight (UTC)	
Root server list and instances	C: 4/4 F: 36/38 K: 15/17 M: 6/6	A: 1/1 C: 4/4 E: 1/1 F: 40/42	H: 1/1 K: 15/17 L: 2/2 M: 6/6
Number of queries	3.84 billion	8.00 billion	
Number of unique clients	~2.8 million	~ 5.6 million	
Recursive queries	17.04%	11.99%	
TCP Bytes Packets Queries	1.65% 2.67% ~700K	N/A	
Queries from RFC1918 addresses	4.26%	N/A	

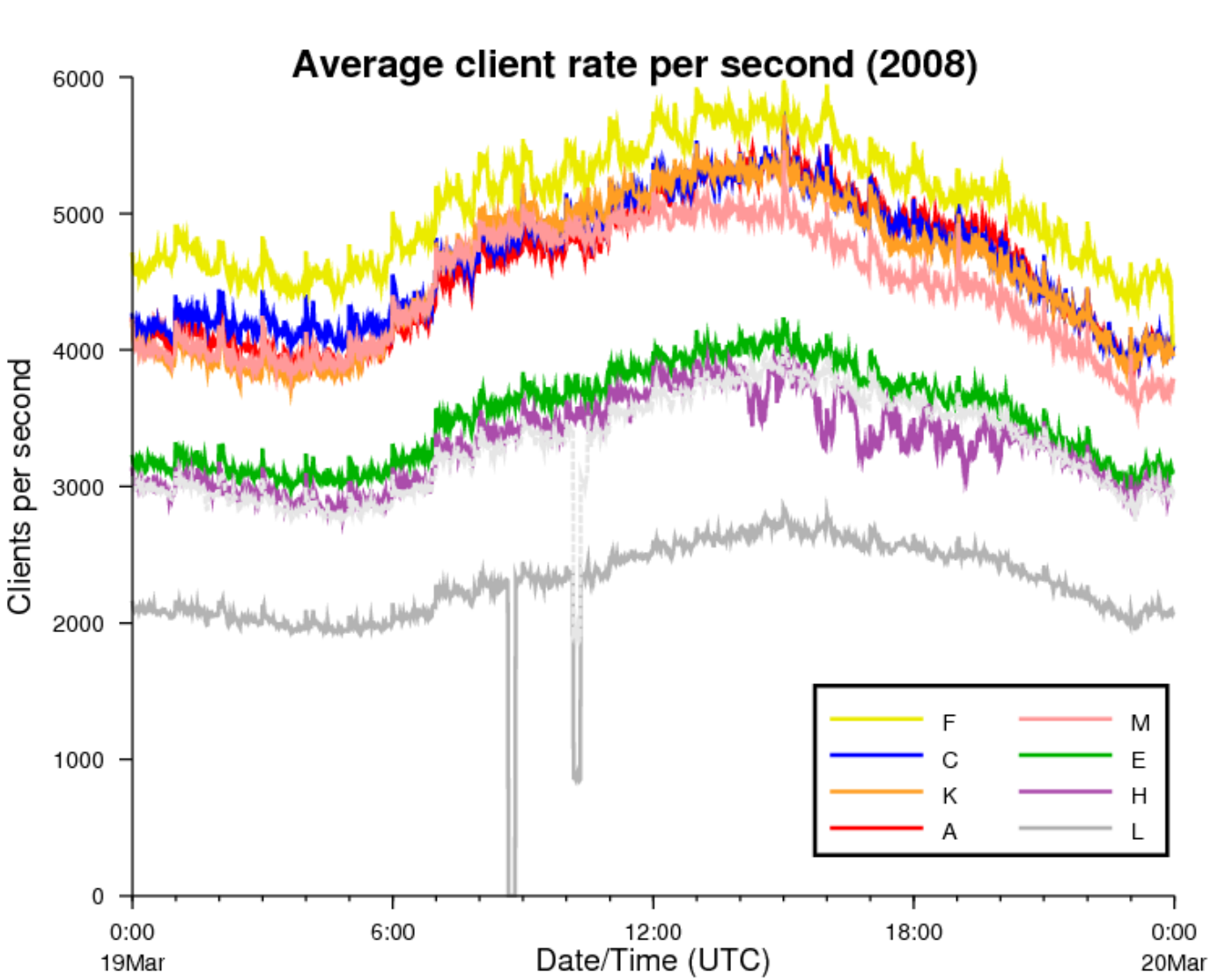
Query rates



Variation of query rates along the years

- Between 2007 and 2008, the qrate grew:
  - C: 40%
  - F: 13%
  - K: 33%
  - M: 5%
- Between 2006 and 2008:
  - C: 139%
  - F: 71%
  - K: 74%

Client rate



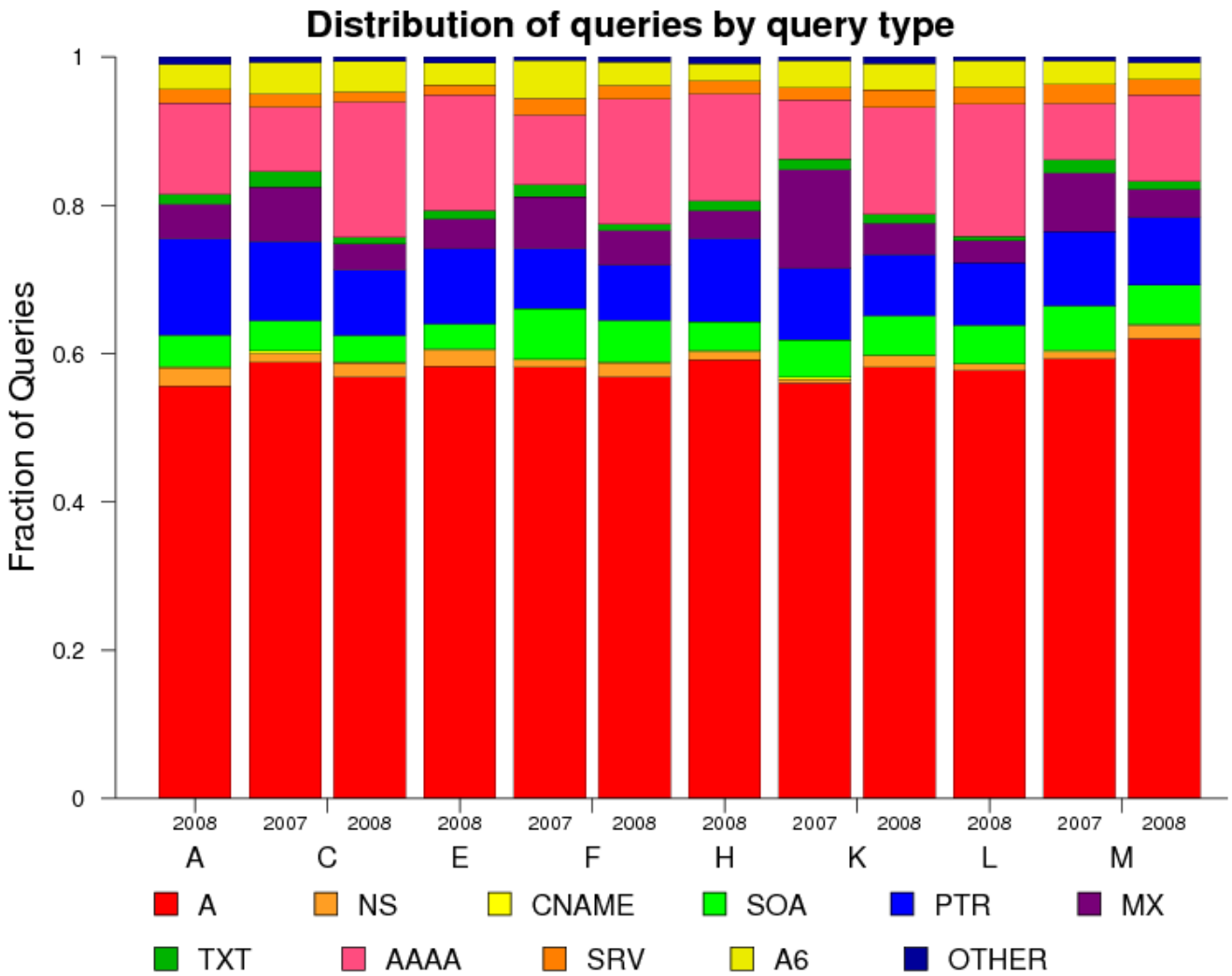
Follows the same pattern of query rates:

- A, C, F, K and M with similar behavior
- E and H
- L

But if old-L traffic is added

- E, H and L are on the same level

Distribution of queries by query type

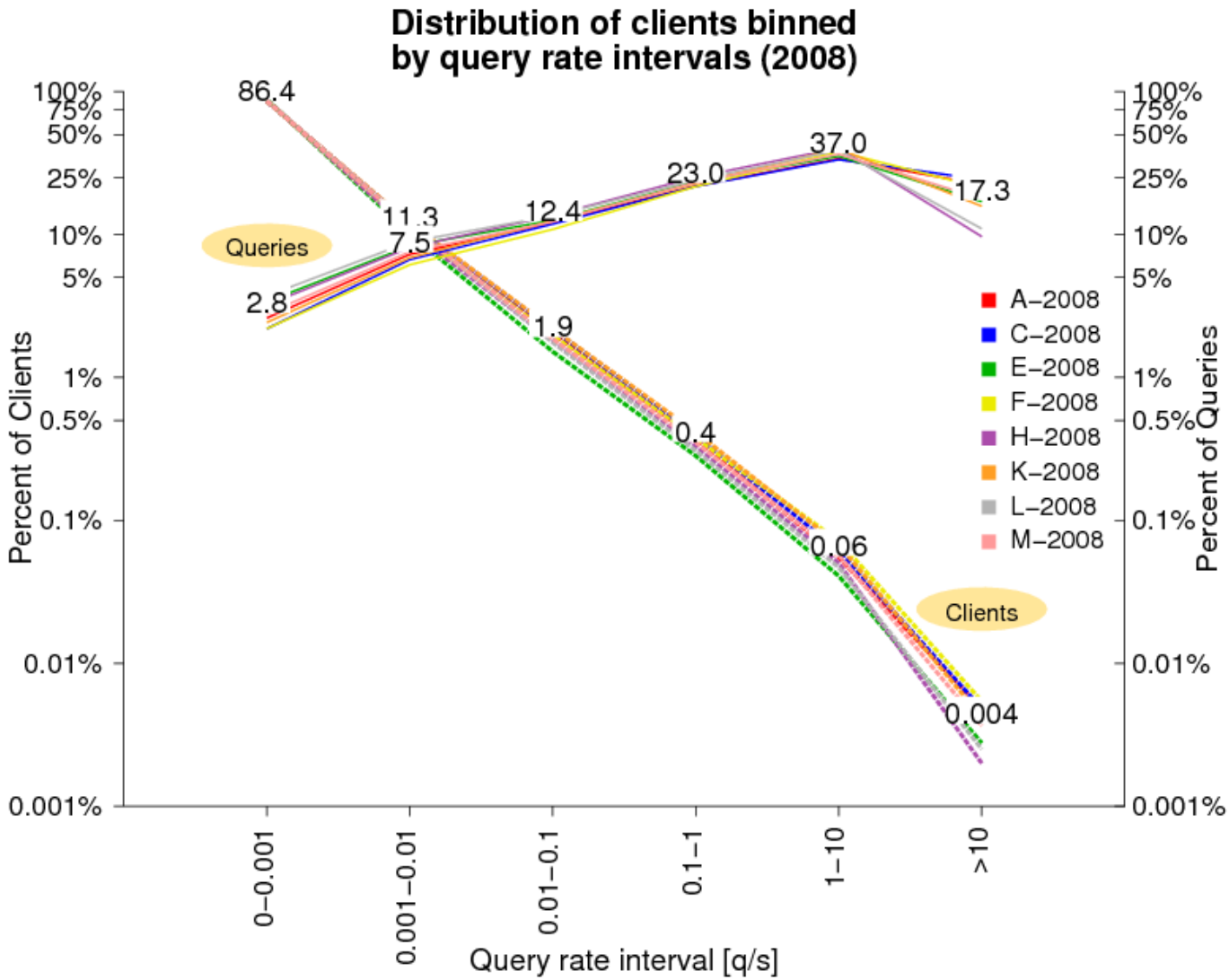


The highest fraction of queries are A queries (slightly below 60%)

Important increase on AAAA queries (pink): from around 8% in 2007 to 15% in 2008.

Reduction of MX queries (purple): K-root drop from 13% to 4%

Distribution of clients/queries



DITL 2008

Leftmost column: ~2.8% of the queries are sent by ~86.4% of clients

Rightmost column:

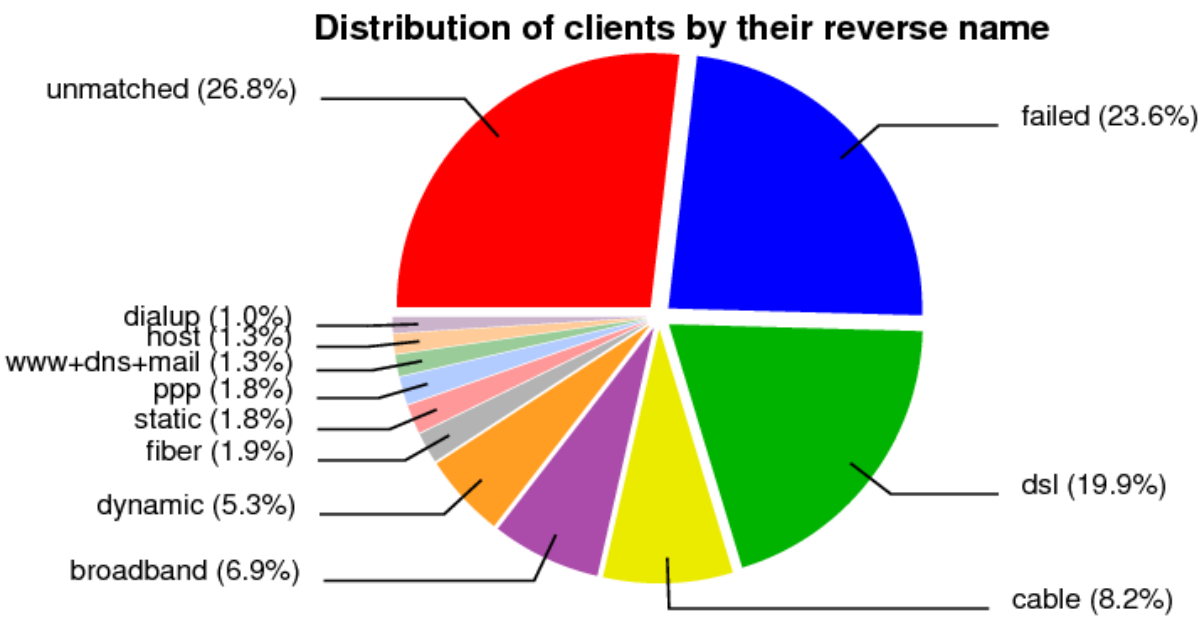
1200 clients generated ~54.3% of the queries.

Client classification

- We attempted to “classify” the clients sending queries to the roots.
  - Using the reverse names
  - Using the IP TTL of their packets
  - Using external sources of data
    - Mainly blacklists

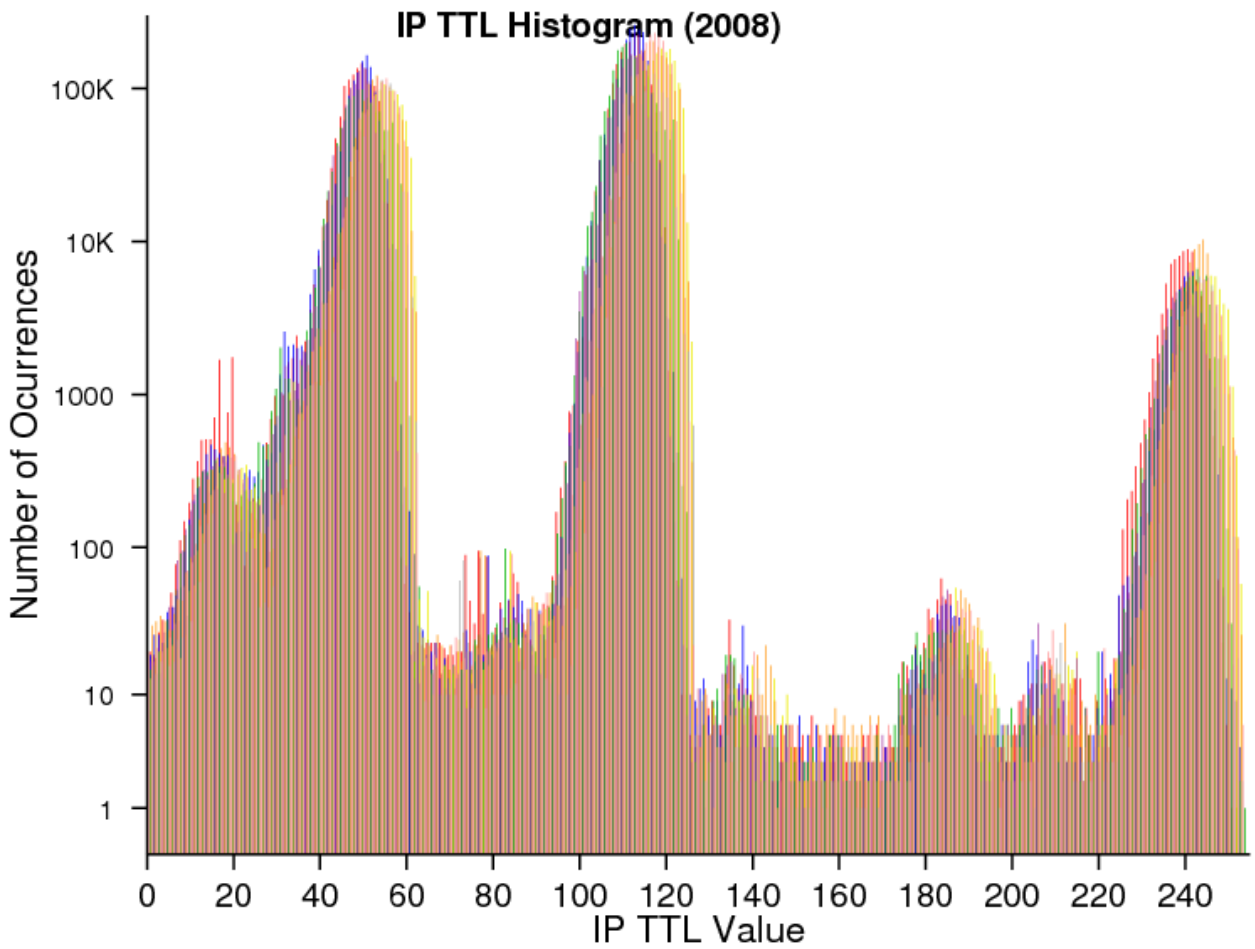
Reverse Names

- For each address, query the corresponding PTR record.
  - Using CAIDA's HostDB engine



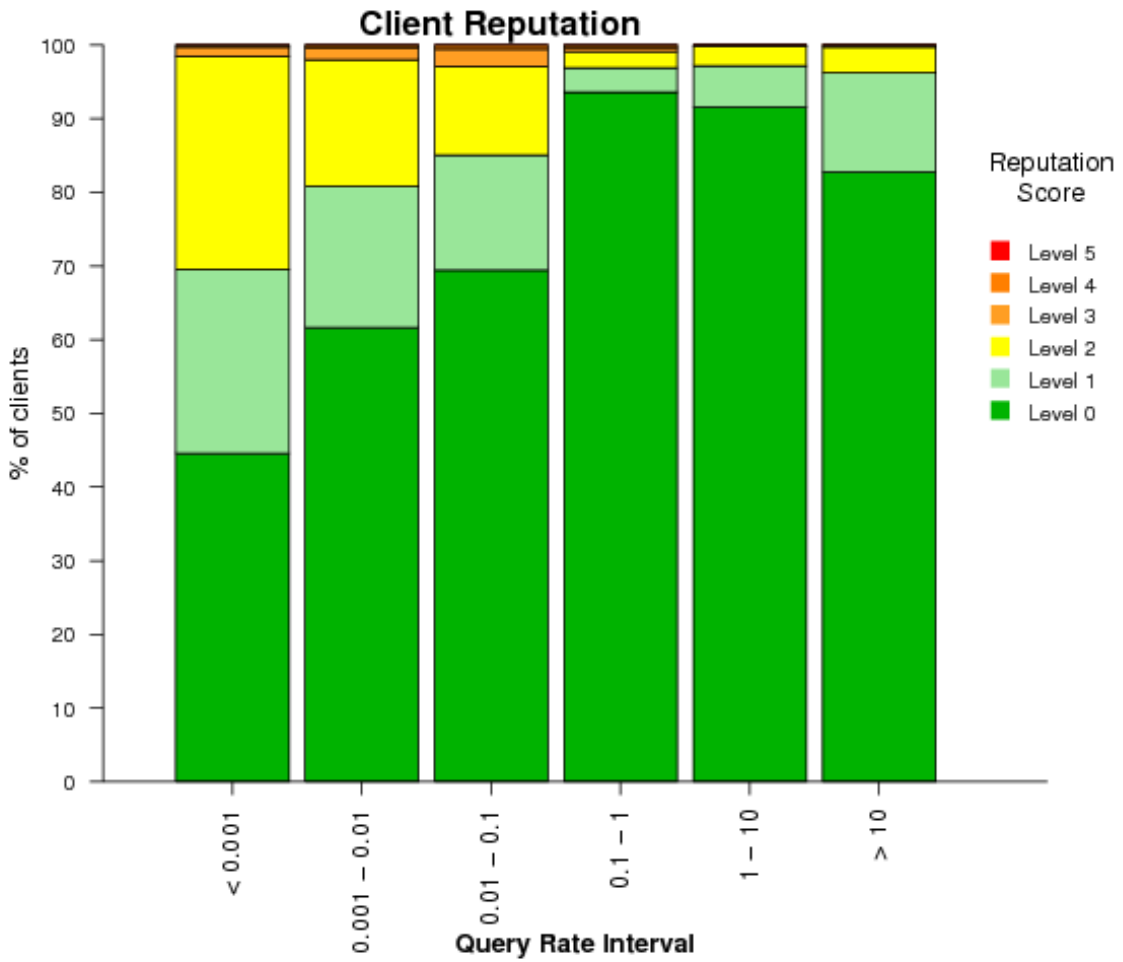
- Five major groups
  - No match found
  - Failed
  - By connection type
    - DSL, cable, fiber, dialup, etc
  - By address assignment
    - static, dynamic
  - By a "service"
    - mail, dns, resolver, fw, etc

IP TTL



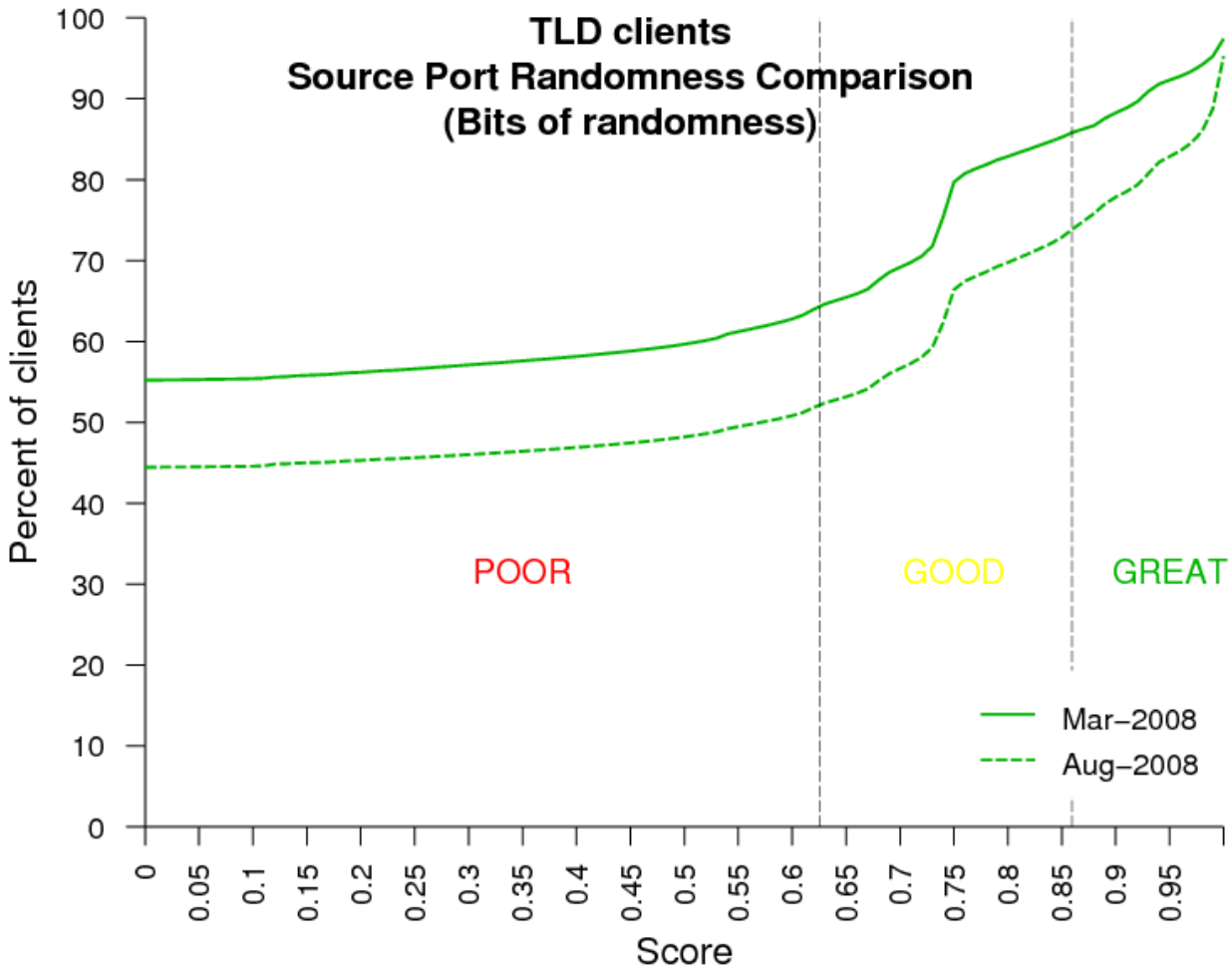
- For each sending queries to the roots, count the observed IP TTL
  - One thin line per root
- 68 clients presented more than 40 different TTL values

Client Reputation



- Sampled 1200 clients on each query rate interval bin
- Queried for the address on 5 different DNSRBL
- Assign a “reputation score” based on the number of matches found.

SPR Measurements



- For each client sending queries to .BR, .ORG and .UK
  - At least 20 queries in total

  Two datasets: Mar-19 and Aug-9

  Three metrics
  - Port changes/queries ratio
  - # different ports/queries ratio
  - Bits of randomness
    - Presented by Duane Wessels at CAIDA/WIDE/CASFI workshop (using standard deviation as a metric of randomness)

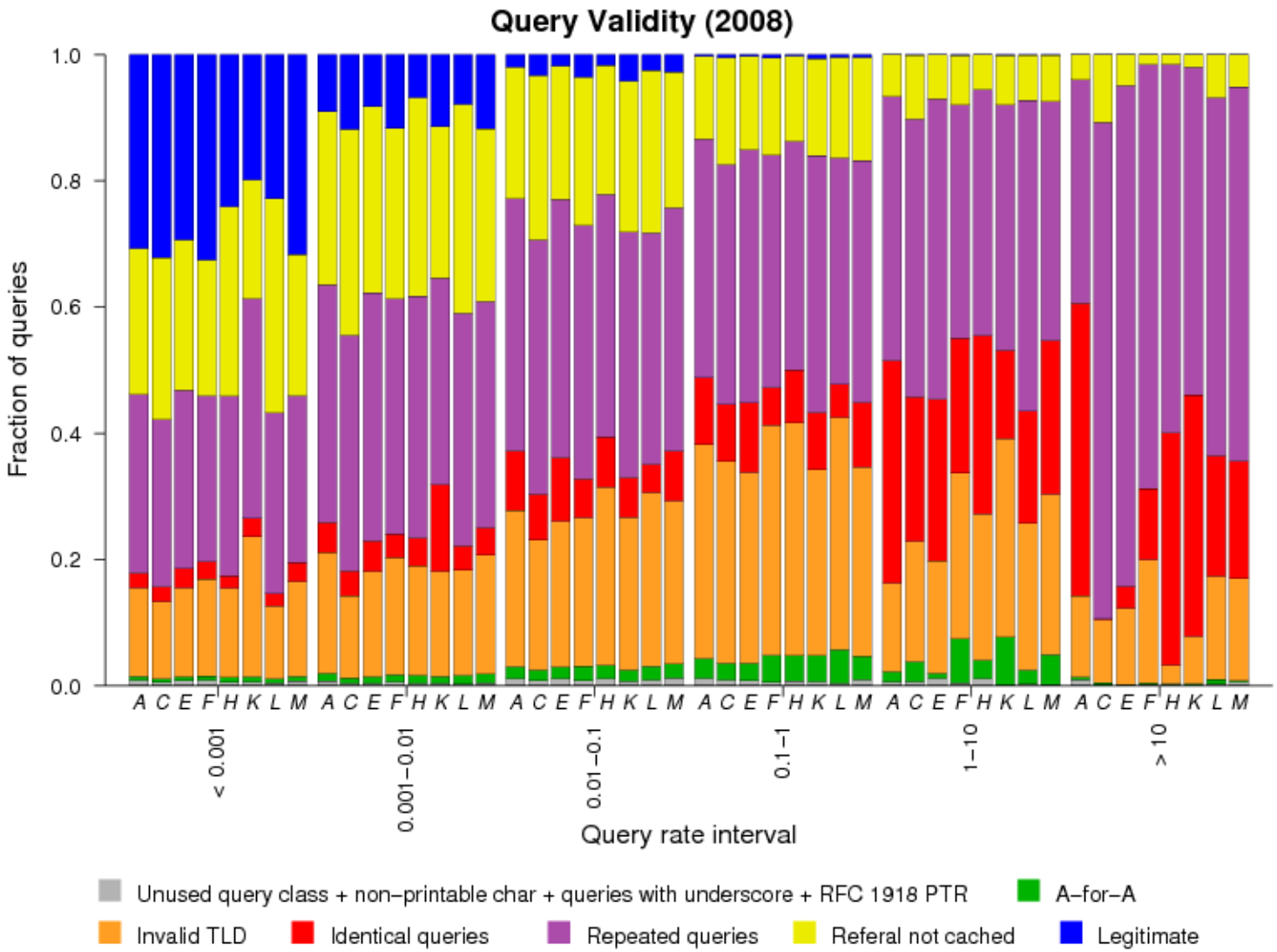
Invalid queries analysis

- To prepare the invalid queries analysis we required to split the traces per source address.
  - We sampled 10% of the unique source addresses observed on each root
- Each query could fit in nine categories of invalid queries
  - The match was done sequentially
  - If none matched, was counted as **valid query**

Invalid queries categories

- Unused query class:
  - Any class not in IN, CHAOS, HESIOD, NONE or ANY
- A-for-A: A-type query for a name is already a IPv4 Address
  - <IN, A, 192.16.3.0>
- Invalid TLD: a query for a name with an invalid TLD
  - <IN, MX, localhost.lan>
- Non-printable characters:
  - <IN, A, www.ra^B.us.>
- Queries with '_':
  - <IN, SRV, _ldap._tcp.dc._msdcs.SK0530-K32-1.>
- RFC 1918 PTR:
  - <IN, PTR, 171.144.144.10.in-addr.arpa.>
- Identical queries:
  - a query with the same class, type, name and id (during the whole period)
- Repeated queries:
  - a query with the same class, type and name
- Referral-not-cached:
  - a query seen with a referral previously given.

Query validity (the graph)



Query validity (the numbers)

Category	A	C	E	F	H	K	L	M	Total
Unused	0.1	0.0	0.1	0.0	0.1	0.0	0.1	0.1	0.1
A-for-A	1.6	1.9	1.2	3.6	2.7	3.8	2.6	2.7	2.7
Invalid TLD	19.3	18.5	19.8	25.5	25.6	22.9	24.8	22.9	22.0
Non-print char	0.0	0.1	0.1	0.1	0.1	0.0	0.1	0.0	0.0
Queries with _	0.2	0.1	0.2	0.1	0.2	0.1	0.1	0.1	0.1
RFC 1918 PTR	0.6	0.3	0.5	0.2	0.5	0.2	0.1	0.3	0.4
Identical queries	27.3	10.4	14.9	12.3	17.4	17.9	12.0	17.0	15.6
Repeated queries	38.5	51.4	49.3	45.3	38.7	42.0	44.2	43.9	44.9
Referral not cached	10.7	15.2	12.1	10.9	12.9	11.1	14.3	11.1	12.4
Valid 2008	***1.7***	***2.0***	***1.8***	***1.9***	***1.8***	***2.0***	***1.8***	***1.8***	***1.8***
Valid 2007		***4.1***		***2.3***		***1.8***		***4.4***	***2.5***

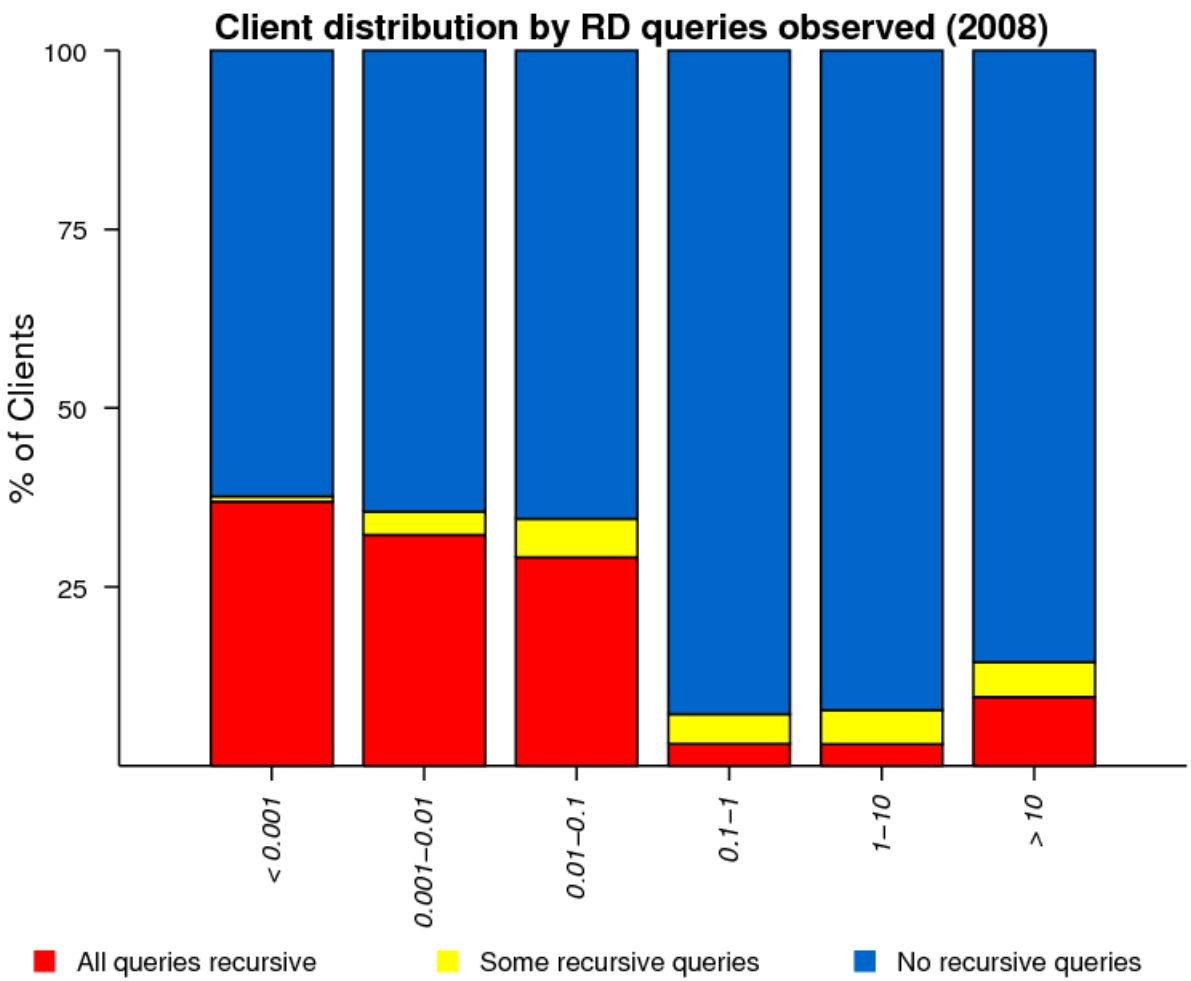
Query validity (the words)

- Based on our first graphs, the query load keeps increasing
  - So the pollution
- The fraction of valid traffic is decreasing
- The pollution is dominated by “invalid TLD”, repeated and identical queries.

Looking some of the sources of pollution

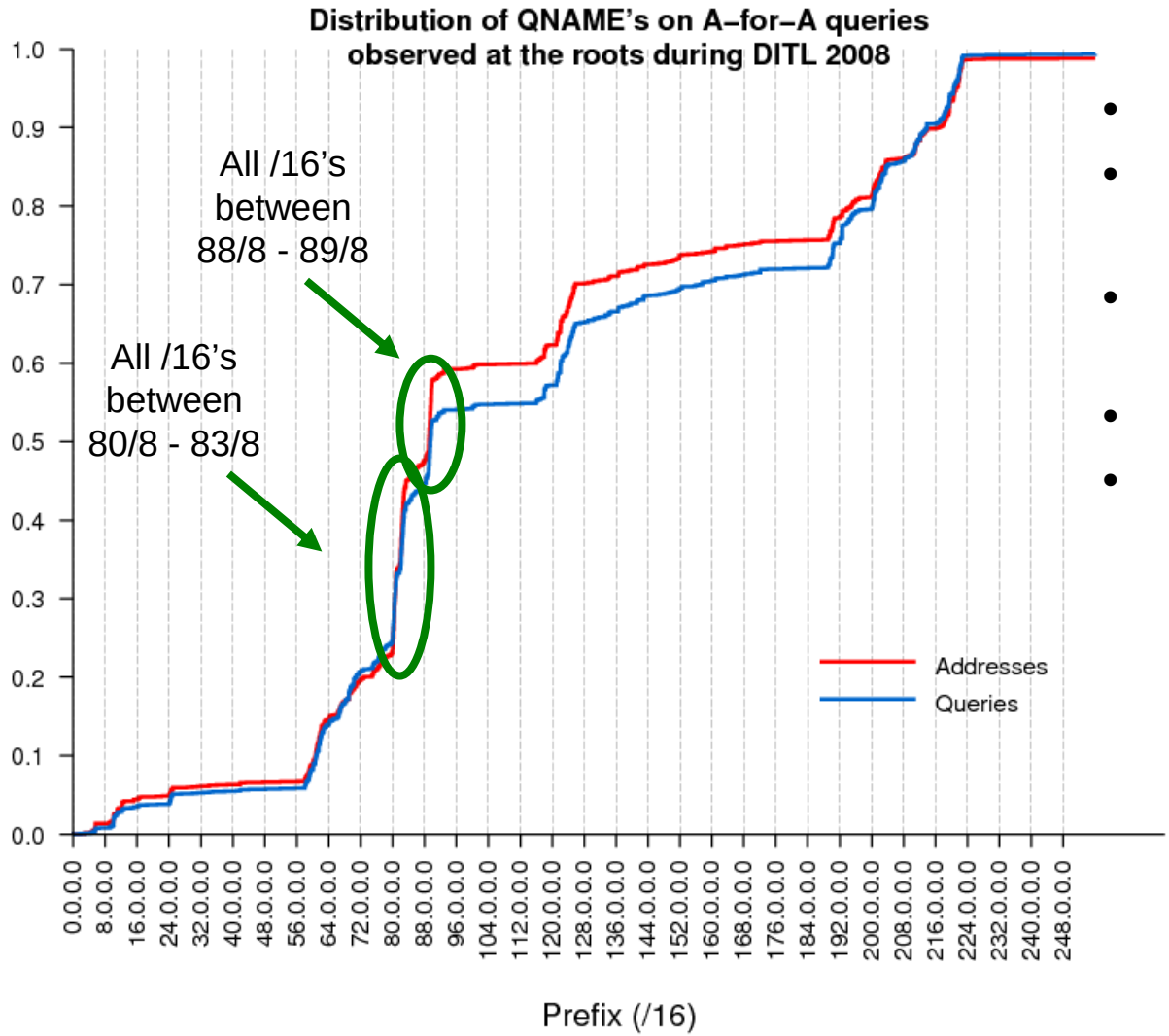
- We explored more details on the sources of pollution
  - Recursive queries
  - A-for-A queries
    - Including some evidence of address space scanning and a new type of trash.
  - Invalid TLD
- … and propose some solutions

Recursive Queries



- During 2008 the number of recursive queries reduced compared to 2007
  - 2008: 11.99%; 2007: 17.04%
- But the number of sources increased
  - 2007: 290K (11.3%)
  - 2008: 1.97M (36.4%);
- What to do?
  - Return a REFUSED
    - Bad Idea
  - Drop the query?
    - Even worst
  - Delay the query?
  - Do nothing

A-for-A: Address space scanning



- Took all QNAME and convert them to addresses
- Group them by /24 and /16
- 18270 sources sent queries for the 80/8 – 83/8
- 8845 sources sent queries for the 88/8 – 89/8
- 8115 sources in common
- Seemed coordinated: different sources sent queries for different partitions, iterating over the third octet.

A6-for-A? AAAA-for-A?

- Originally this category included A-queries with a query name in the form of an IPv4 address
  - What about the other query types for addresses?
  - The result: 3.32% of this type of queries were for A6/AAAA queries

```
00:04:03.347275 IP 195.2.83.107.5553 > 12.0.0.2.53: 40248 [1au] A? 221.0.93.99. (40)
00:04:03.347392 IP 195.2.83.107.5553 > 12.0.0.2.53: 1887 [1au] AAAA? 221.0.93.99. (40)
00:04:03.347642 IP 195.2.83.107.5553 > 12.0.0.2.53: 2737 [1au] A6? 221.0.93.99. (40)
00:04:59.579904 IP 195.2.83.107.5553 > 6.0.0.30.53: 40723 [1au] A? 84.52.73.160. (41)
00:05:36.016886 IP 195.2.83.107.5553 > 11.0.0.8.53: 28473 [1au] A? 148.240.4.32. (41)
00:05:36.016902 IP 195.2.83.107.5553 > 11.0.0.8.53: 27782 [1au] AAAA? 148.240.4.32. (41)
00:05:36.016908 IP 195.2.83.107.5553 > 11.0.0.8.53: 1175 [1au] A6? 148.240.4.32. (41)
00:06:58.022212 IP 195.2.83.107.5553 > 13.0.0.1.53: 28596 [1au] A? 61.143.210.226. (43)
00:06:58.022647 IP 195.2.83.107.5553 > 13.0.0.1.53: 10748 [1au] AAAA? 61.143.210.226. (43)
00:06:58.023381 IP 195.2.83.107.5553 > 13.0.0.1.53: 12721 [1au] A6? 61.143.210.226. (43)
```

Invalid TLD

- Queries for invalid TLD represent 22% of the total traffic at the roots
  - 20.6% during DITL 2007
- Top 10 invalid TLD represent 10.5% of the total traffic
- RFC 2606 reserves some TLD to avoid future conflicts
- We propose:
  - Include some of these TLD (local, lan, home, localdomain) to RFC 2606
  - Encourage cache implementations to answer queries for RFC 2606 TLDs locally (with data or error)

TLD	**Percentage of total queries**	
	2007	2008
local	5.018	5.098
belkin	0.436	0.781
localhost	2.205	0.710
lan	0.509	0.679
home	0.321	0.651
invalid	0.602	0.623
domain	0.778	0.550
localdomain	0.318	0.332
wpad	0.183	0.232
corp	0.150	0.231

Repeated/identical queries

- Minas Gjoka at CAIDA found 50% of the repeated/identical queries arrived within a 10-sec time window
- The use of *Bloom filters* was proposed to detect if a query reaching a server has been seen within the last *k* seconds
  - Using a hash of <QNAME, QCLASS, QTYPE>
  - If seen, take some action (discard? delay?).
- Probably we will work on an implementation to test effectiveness and performance.

Conclusions

- The traffic grows, the pollution grows
- We don’t know much about the sources of unwanted traffic
  - But we do learn a little bit more every time
  - And we will continue looking for answers
    - By simulating combinations of elements that might create pollution
- More brain power is needed to analyze this huge amount of data

Questions? Suggestions?

Thanks for your time