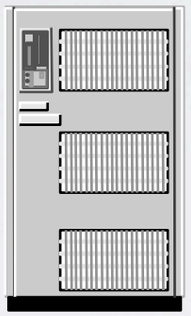
ANALYSING RA/RD BIT USAGE IN ROOT SERVER TRAFFIC

Jim Reid, RTFM LLP
jim@rfc1035.com

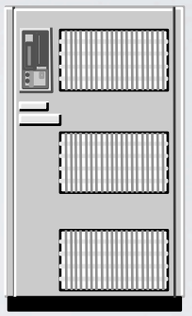
Mitigating Name Collision: ICANN's Approach

- If ***whatever.newTLD*** appears in DITL data, just arrange for the name servers to return NXDOMAIN
  - Lookups for ***whatever.newTLD*** continue to get NXDOMAIN responses, just like now
- DNS behaviour is unchanged so problem goes away
  - Not quite...
  - It used to be the root servers that return NXDOMAIN, but once ***.newTLD*** is delegated, its name servers do that
- Is this strategy prudent or not?

A conventional DNS lookup before .newTLD is delegated



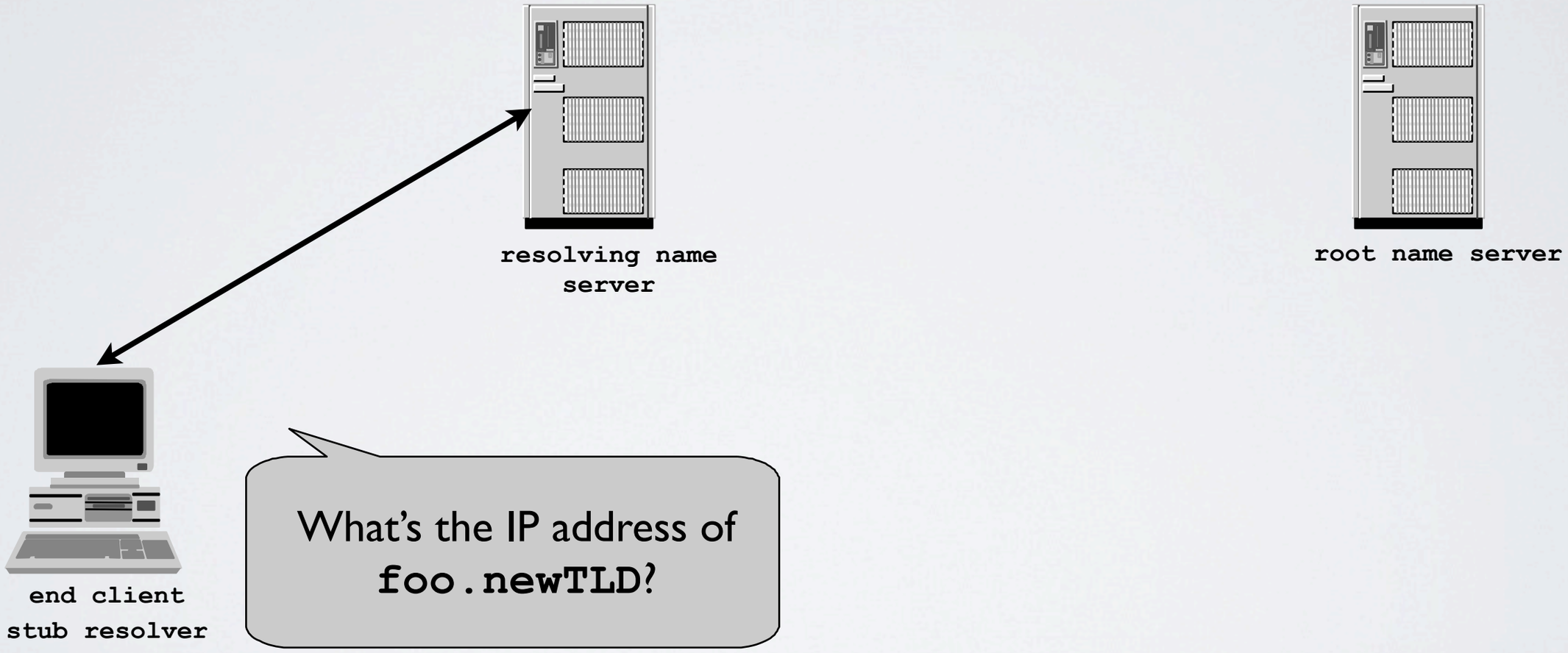
resolving name server



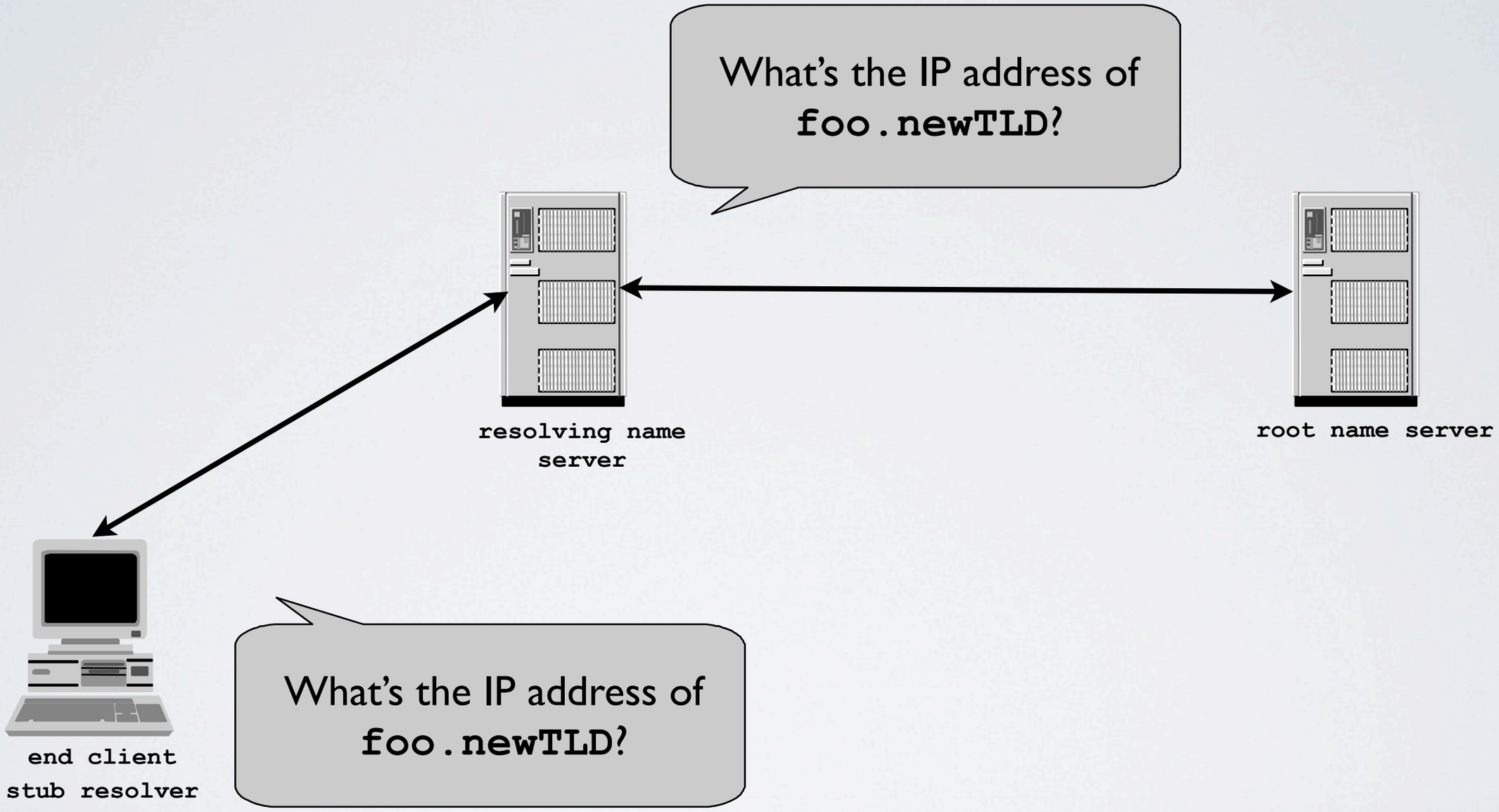
root name server

end client
stub resolver

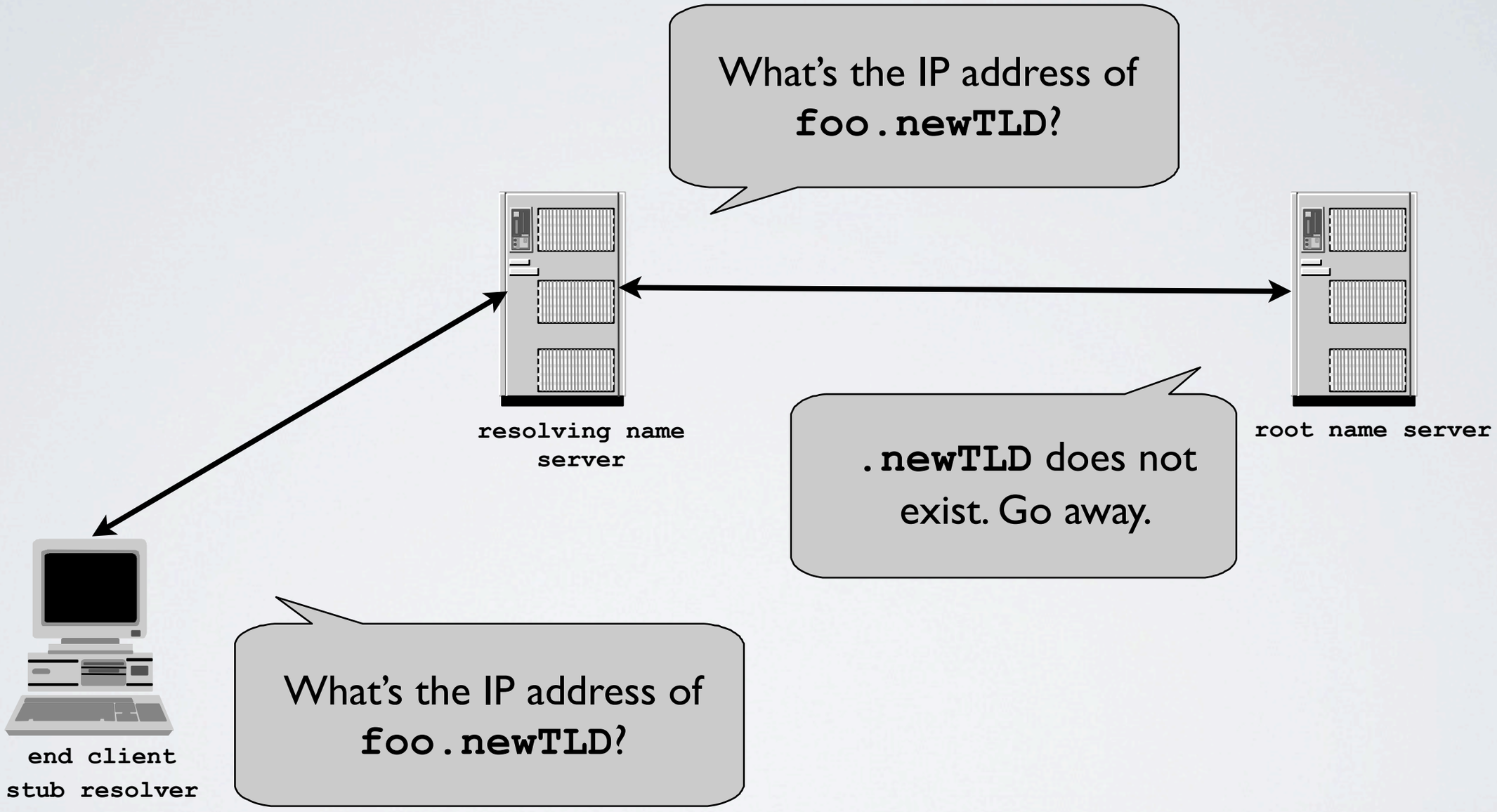
A conventional DNS lookup before .newTLD is delegated



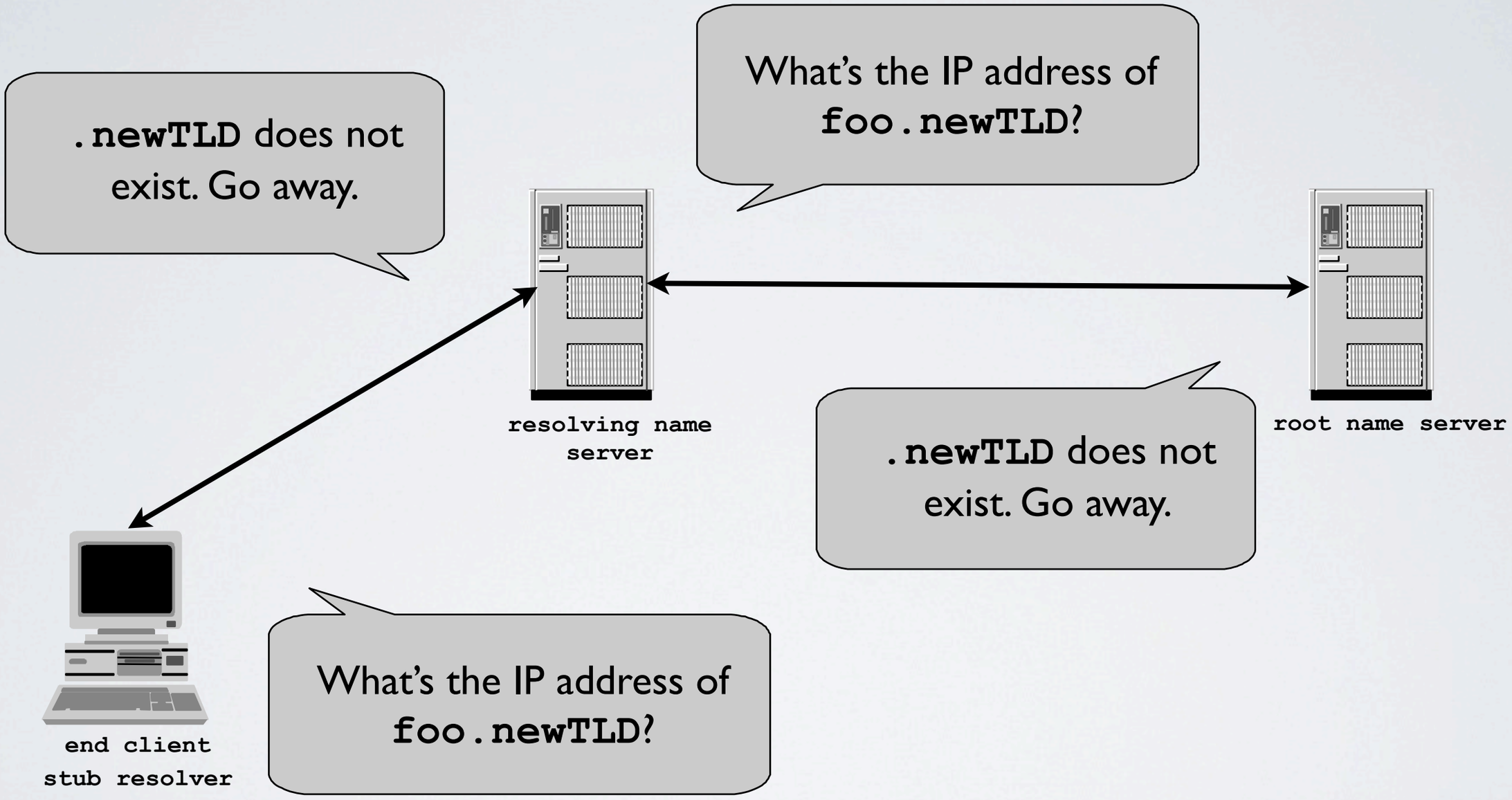
A conventional DNS lookup before `.newTLD` is delegated



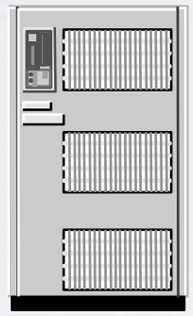
A conventional DNS lookup before .newTLD is delegated



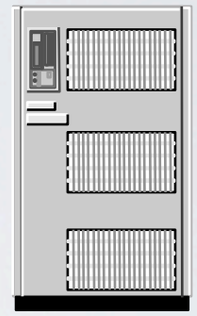
A conventional DNS lookup before .newTLD is delegated



A conventional DNS lookup after .newTLD is delegated

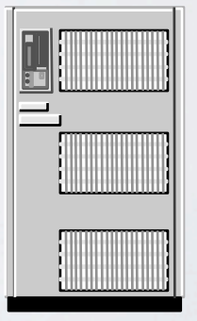


resolving name server



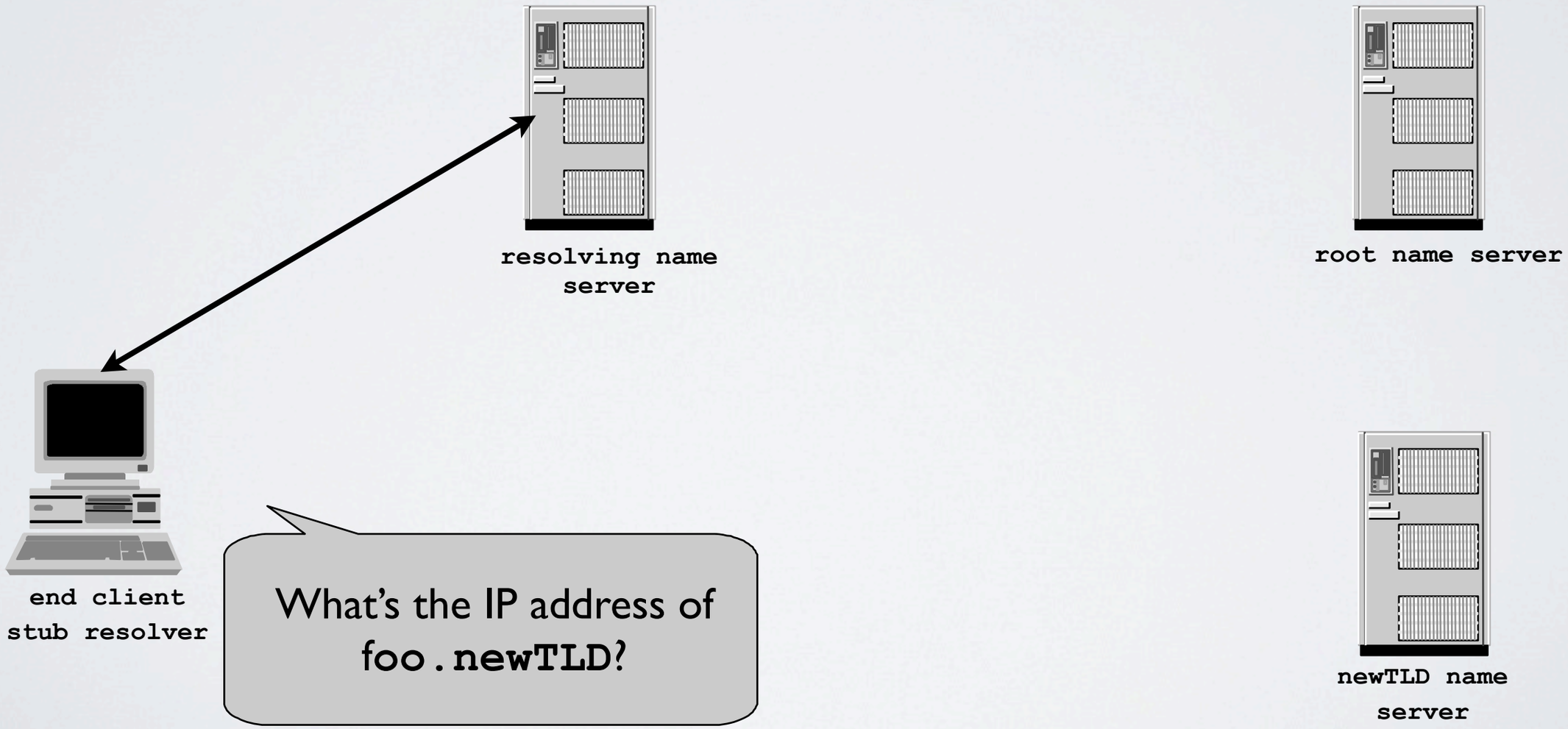
root name server

end client
stub resolver

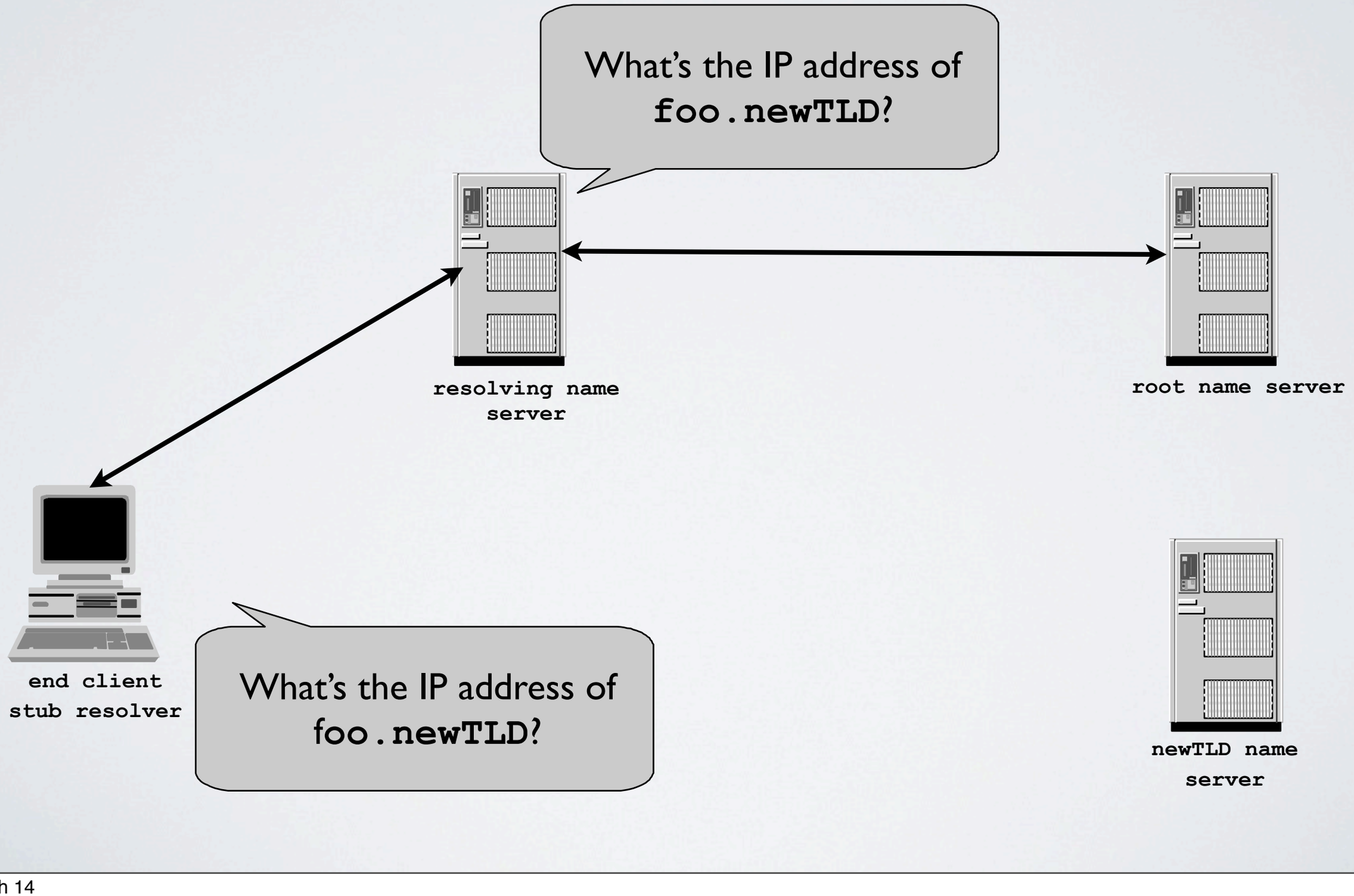


newTLD name server

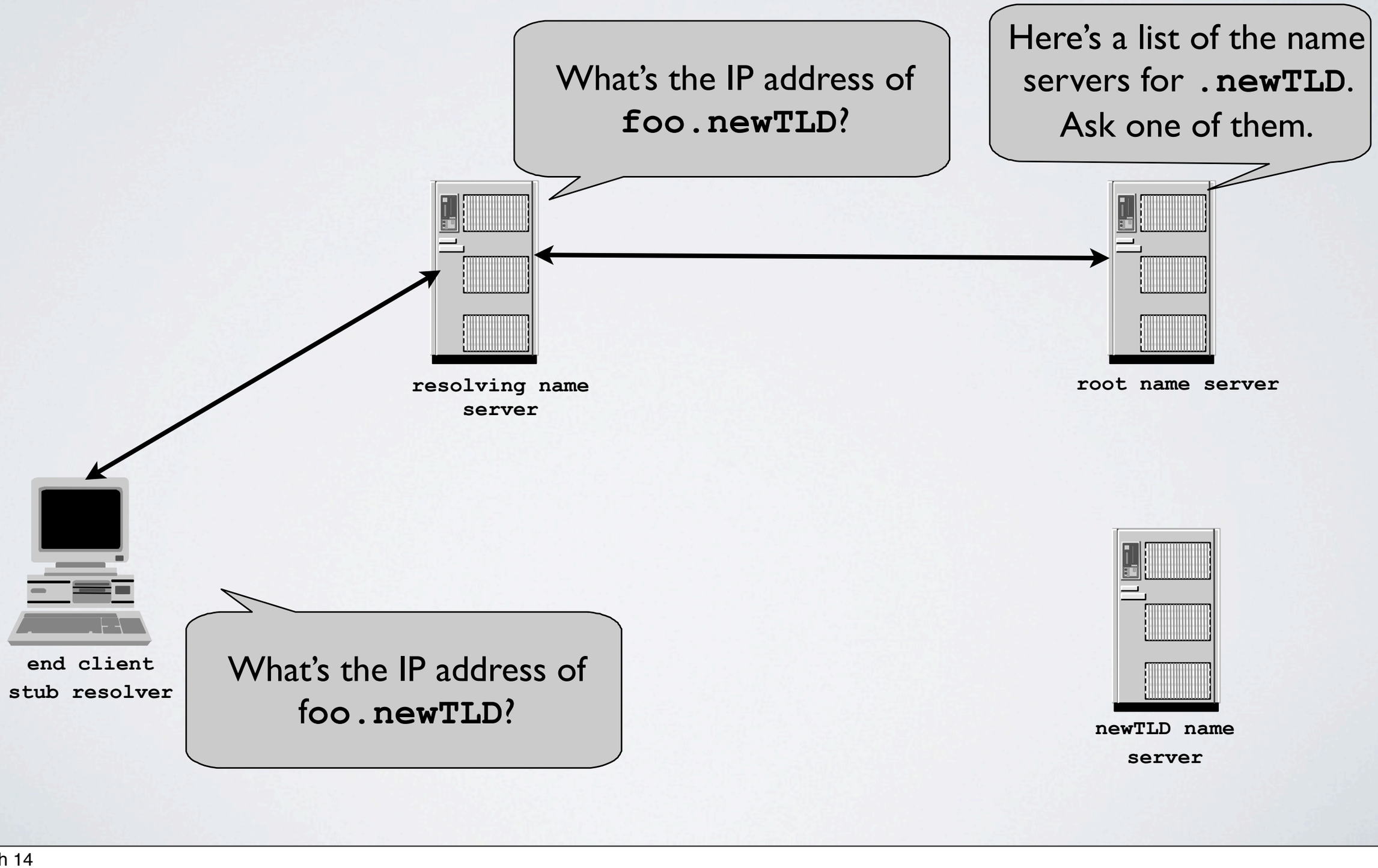
A conventional DNS lookup after .newTLD is delegated



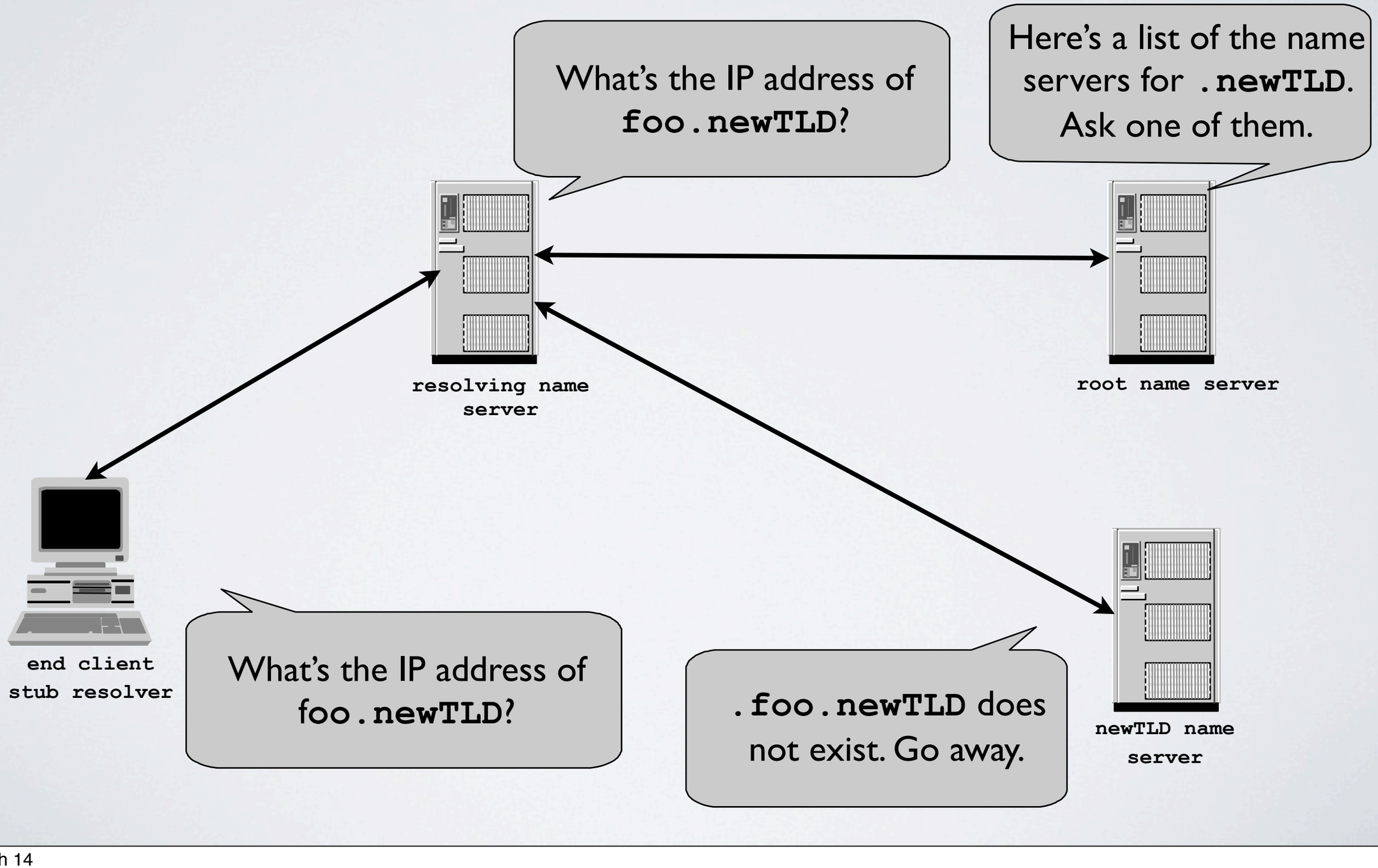
A conventional DNS lookup after `.newTLD` is delegated



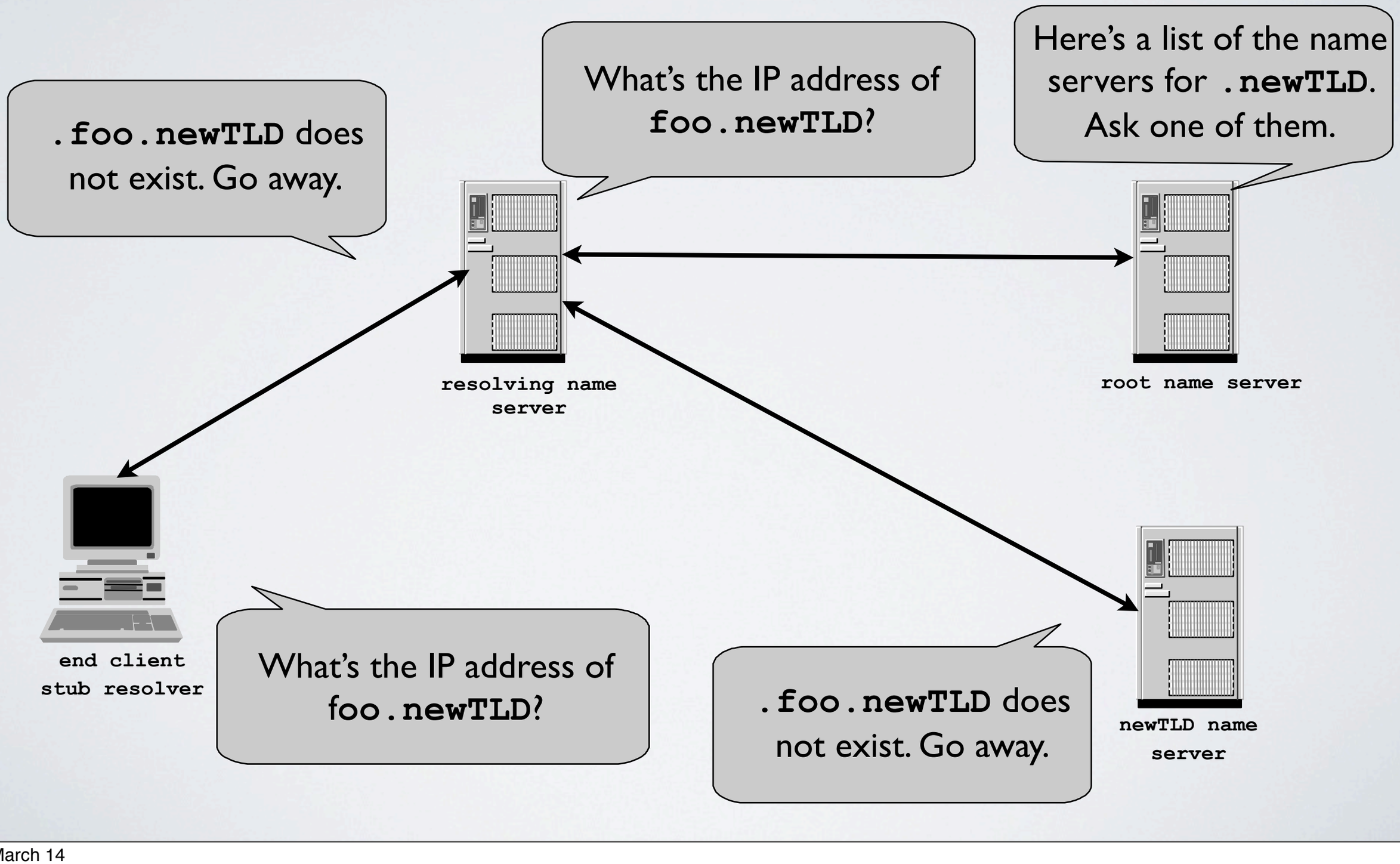
A conventional DNS lookup after .newTLD is delegated



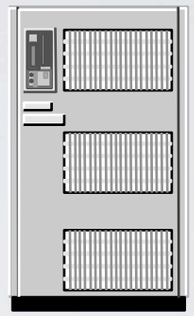
A conventional DNS lookup after `.newTLD` is delegated



A conventional DNS lookup after .newTLD is delegated



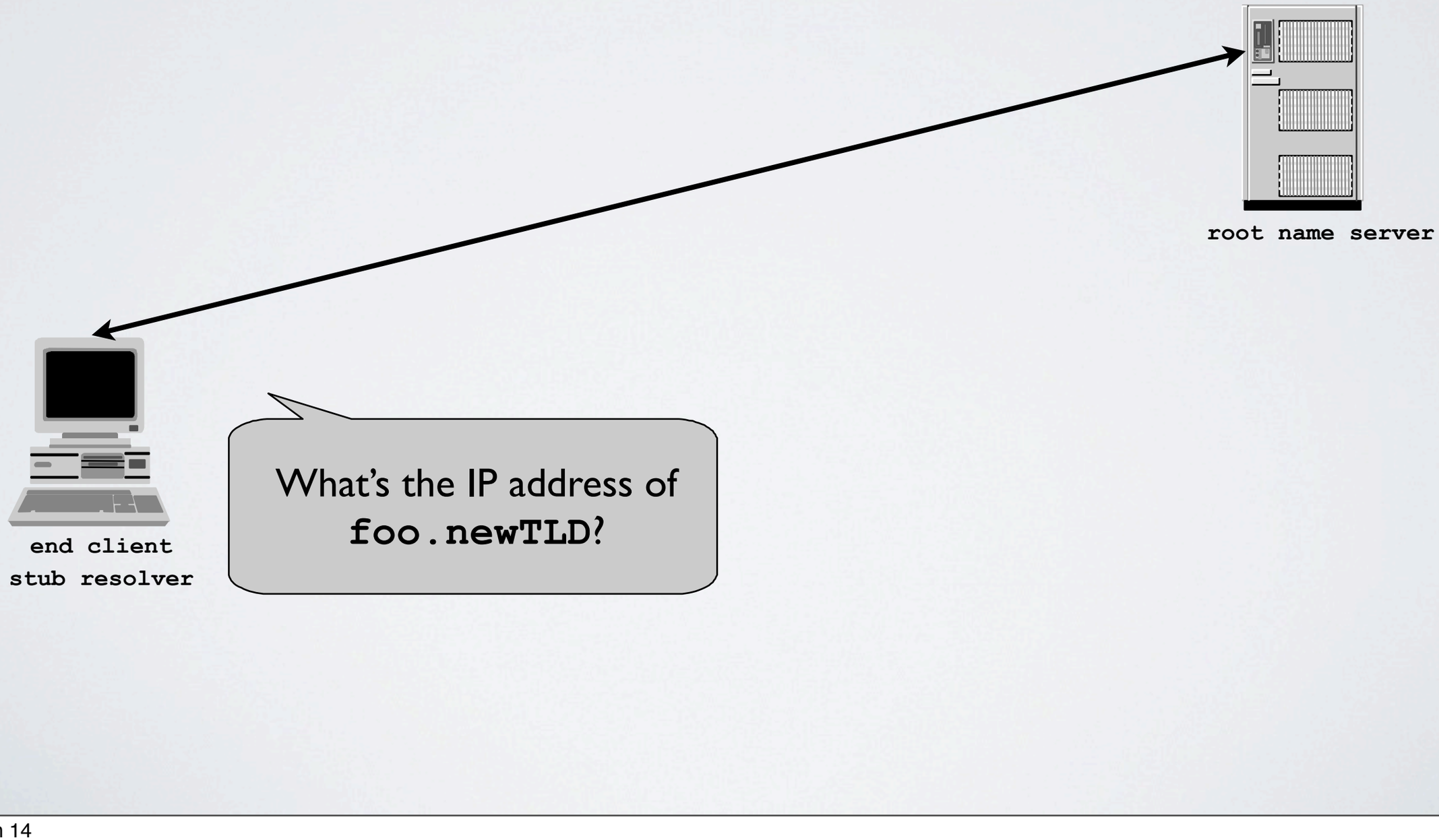
An unconventional DNS lookup before `.newTLD` is delegated



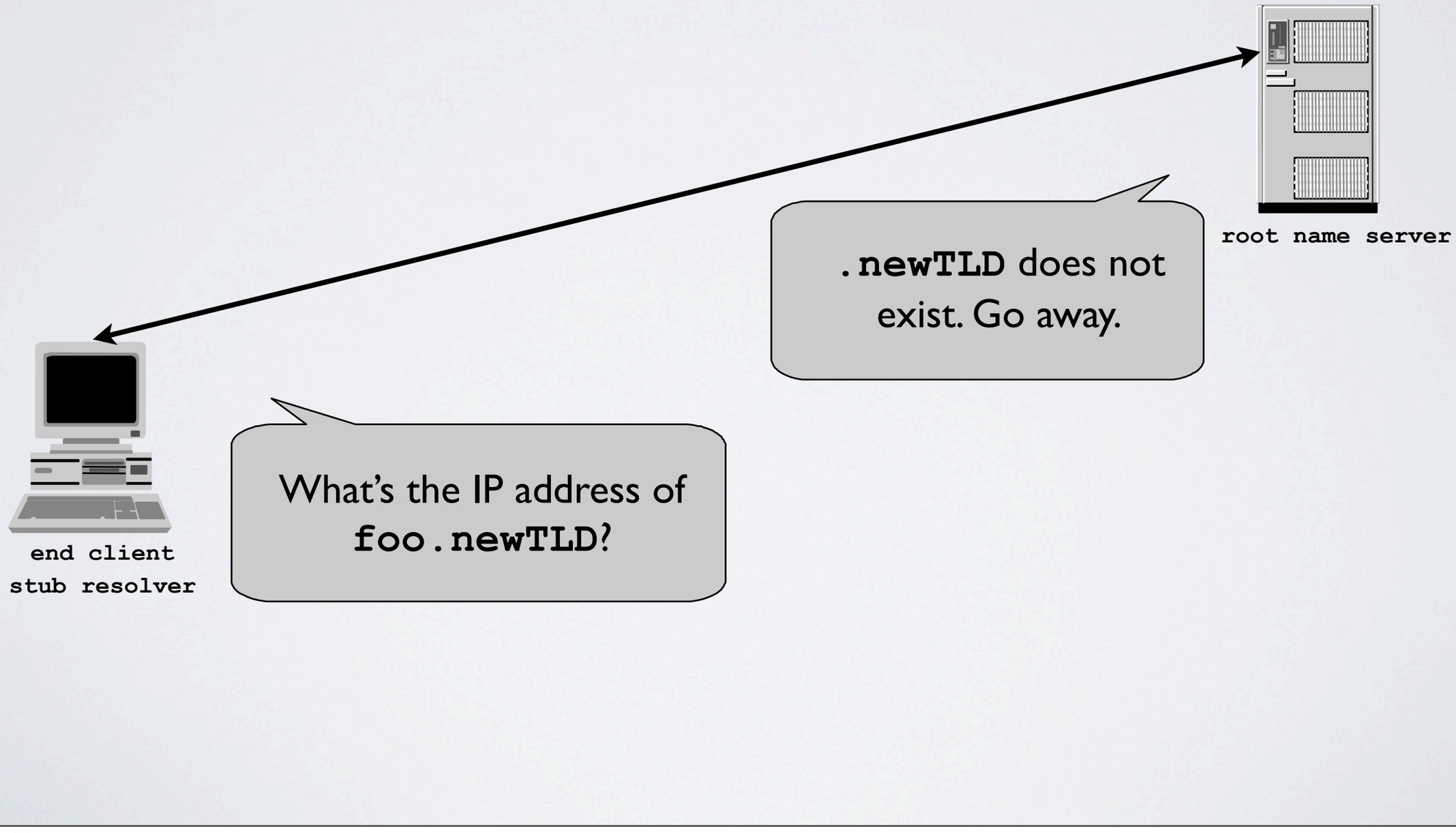
root name server

end client
stub resolver

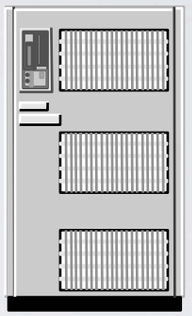
An unconventional DNS lookup before `.newTLD` is delegated



An unconventional DNS lookup before `.newTLD` is delegated

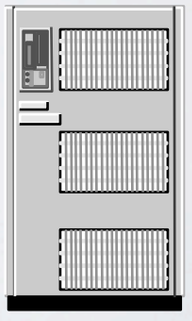


An unconventional DNS lookup after `.newTLD` is delegated



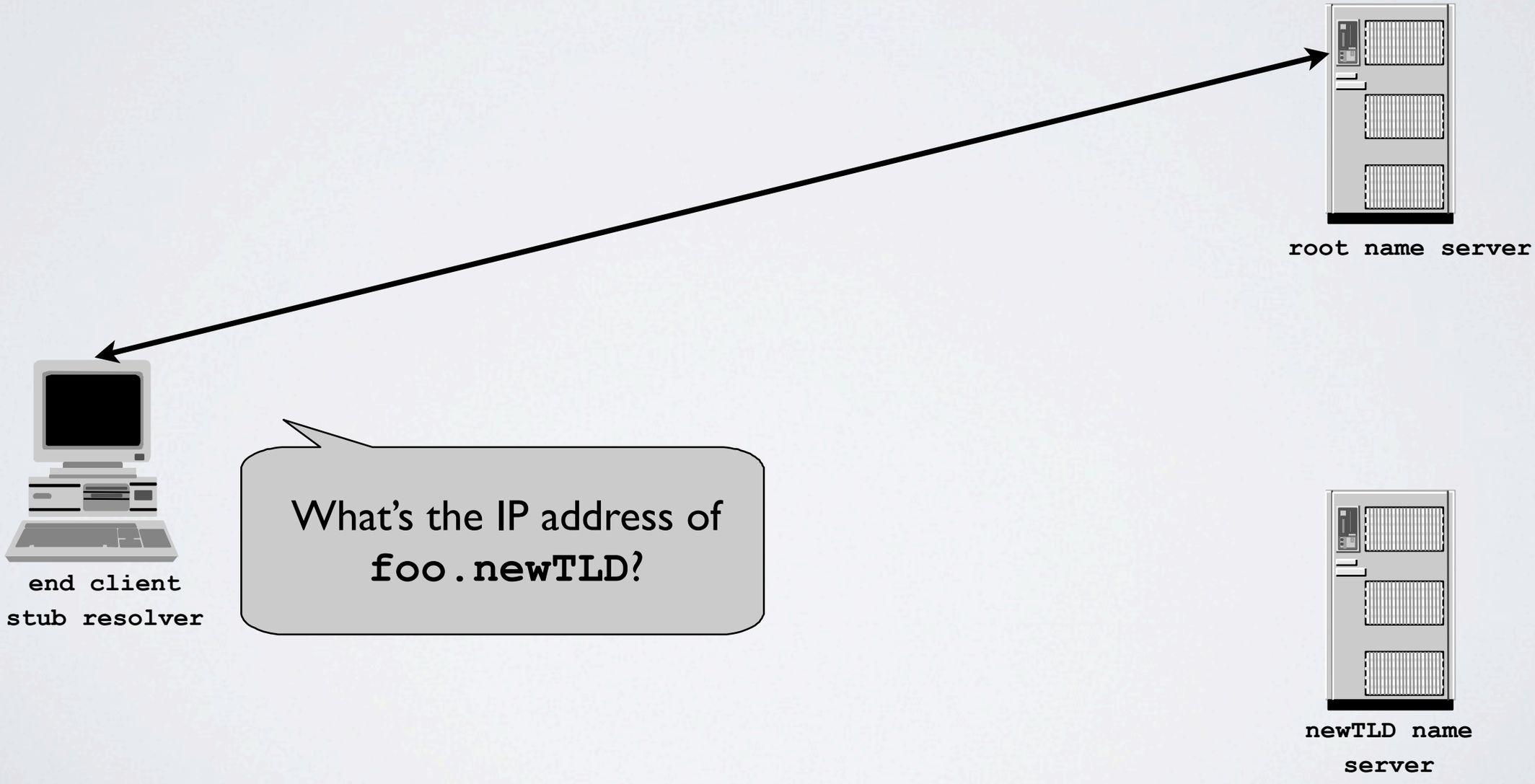
root name server

end client
stub resolver

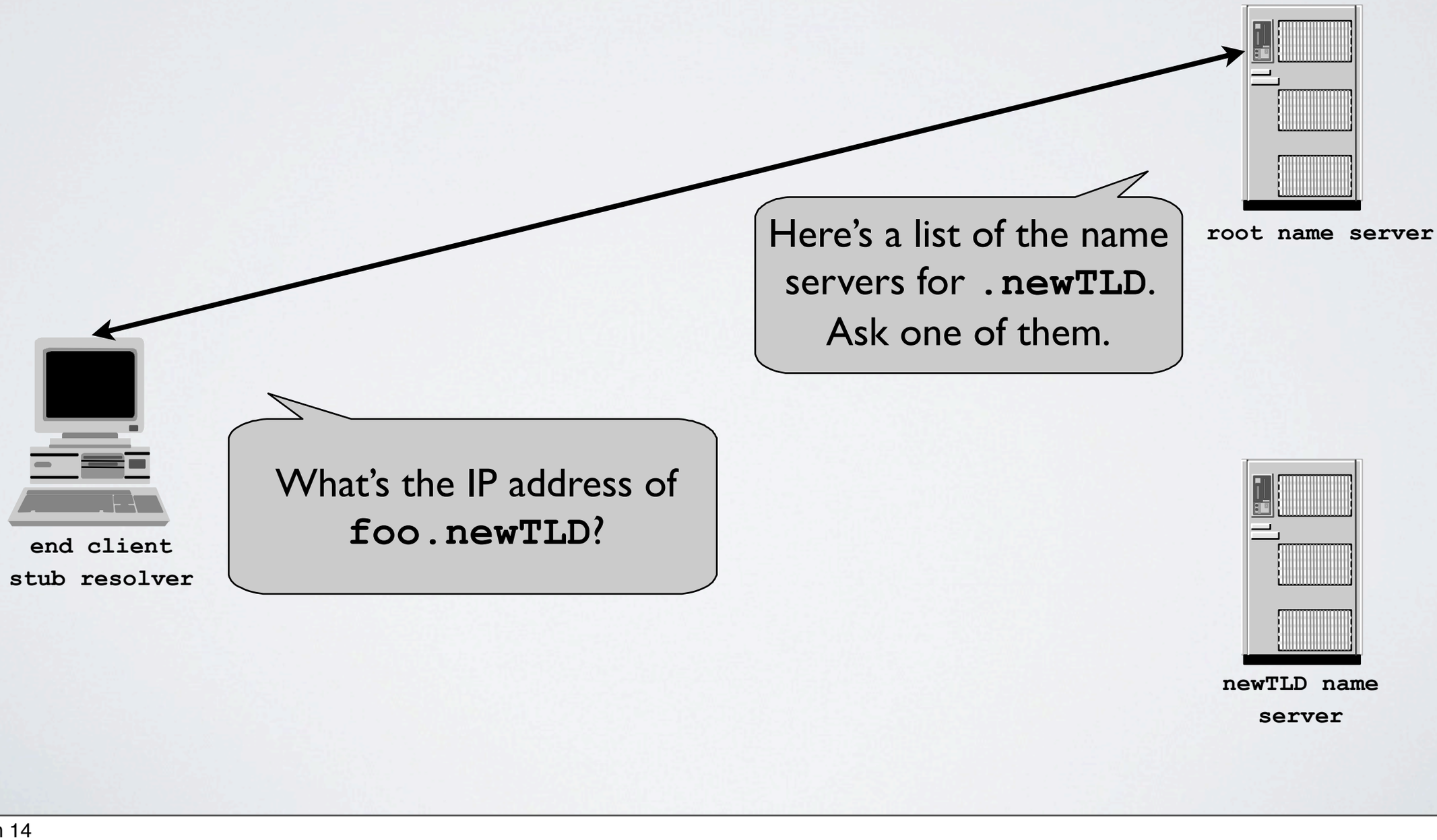


newTLD name server

An unconventional DNS lookup after `.newTLD` is delegated



An unconventional DNS lookup after `.newTLD` is delegated



An unconventional DNS lookup after `.newTLD` is delegated

Naive DNS Clients

- Stub resolvers, proxies & forwarding-only servers cannot handle referral responses
- Undefined behaviour when they get referrals:
  - Give up, report an error, try another name, fail, crash....
- These devices sometimes mistakenly query the root
  - How often does this happen?
  - Is it a problem or not?
  - Which TLDs are most/least at risk?

Analysis & Crunching

- Chewed through ~9 TB of DITL data: ~250Bn requests
  - Contributing root server pcaps from 2006-2013
  - Made three passes over that data
  - Qualitative analysis
  - Comparitive analysis
  - Historical analysis
  - Qualitative analysis

Quantitative Analysis

- There's quite a lot of RD=1 request traffic already
  - Around 12% ± 5% of current root server requests
  - This "cannot happen"
    - Only resolving name servers should be querying the root
  - Does this appear to be causing any operational problems?
- Almost nothing does RA=1
  - No surprise: only **answering** servers are expected to set this header bit

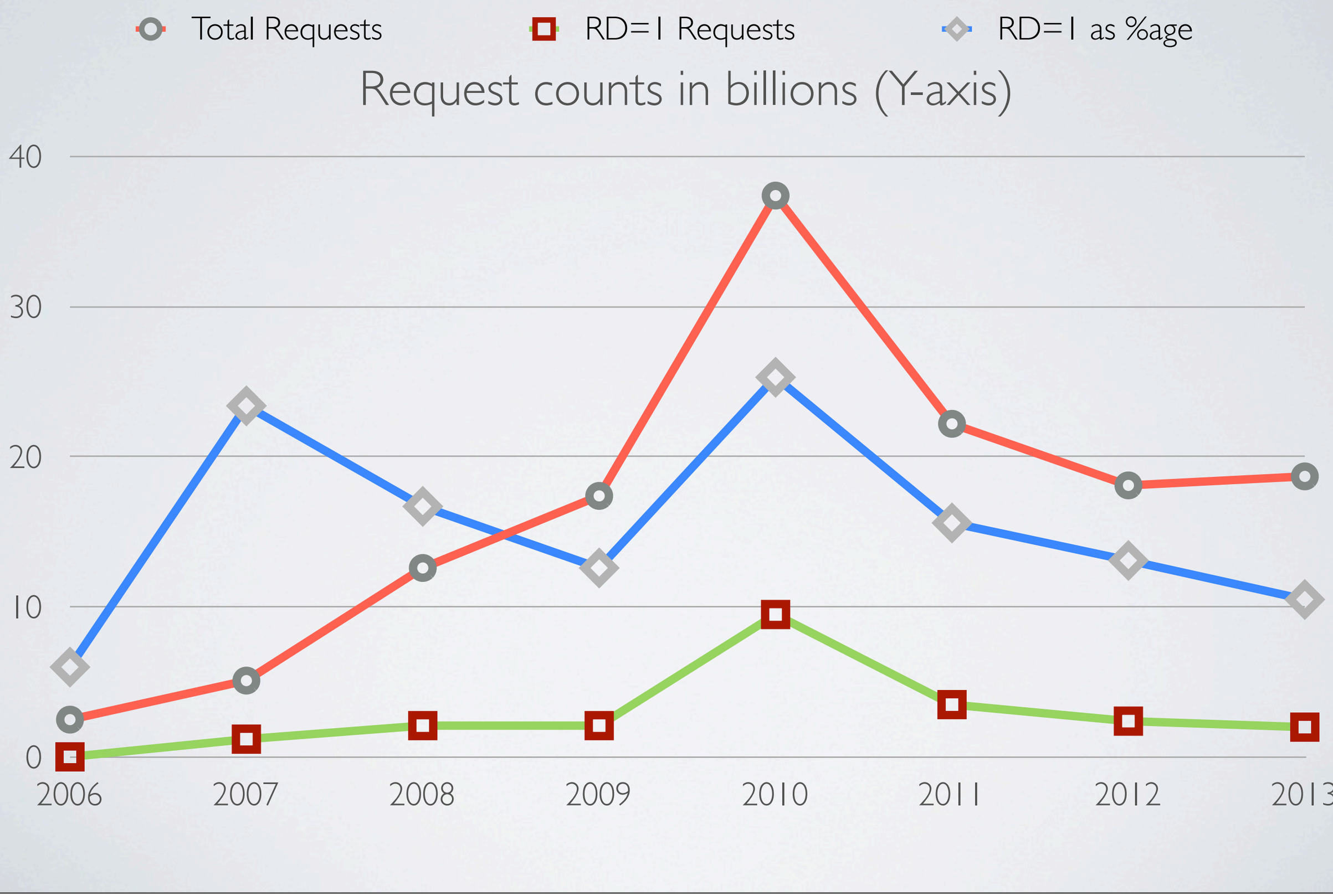
Comparitive Analysis

- Usual suspects amongst existing TLDs responsible for the majority of RD=1 requests:
  - **`.com`**, **`.net`**, **`.arpa`**, **`.org`**, **`.uk`**, **`.de`**, **`.cn`**, **`.jp`**
- Very few new gTLDs have RD=1 requests
- **`.home`** and **`.corp`** are by far the biggest source
- Most have none
- Rates for the others are usually 1-2 orders of magnitude lower than existing TLDs
- **`.google`** seems to get more than its fair share

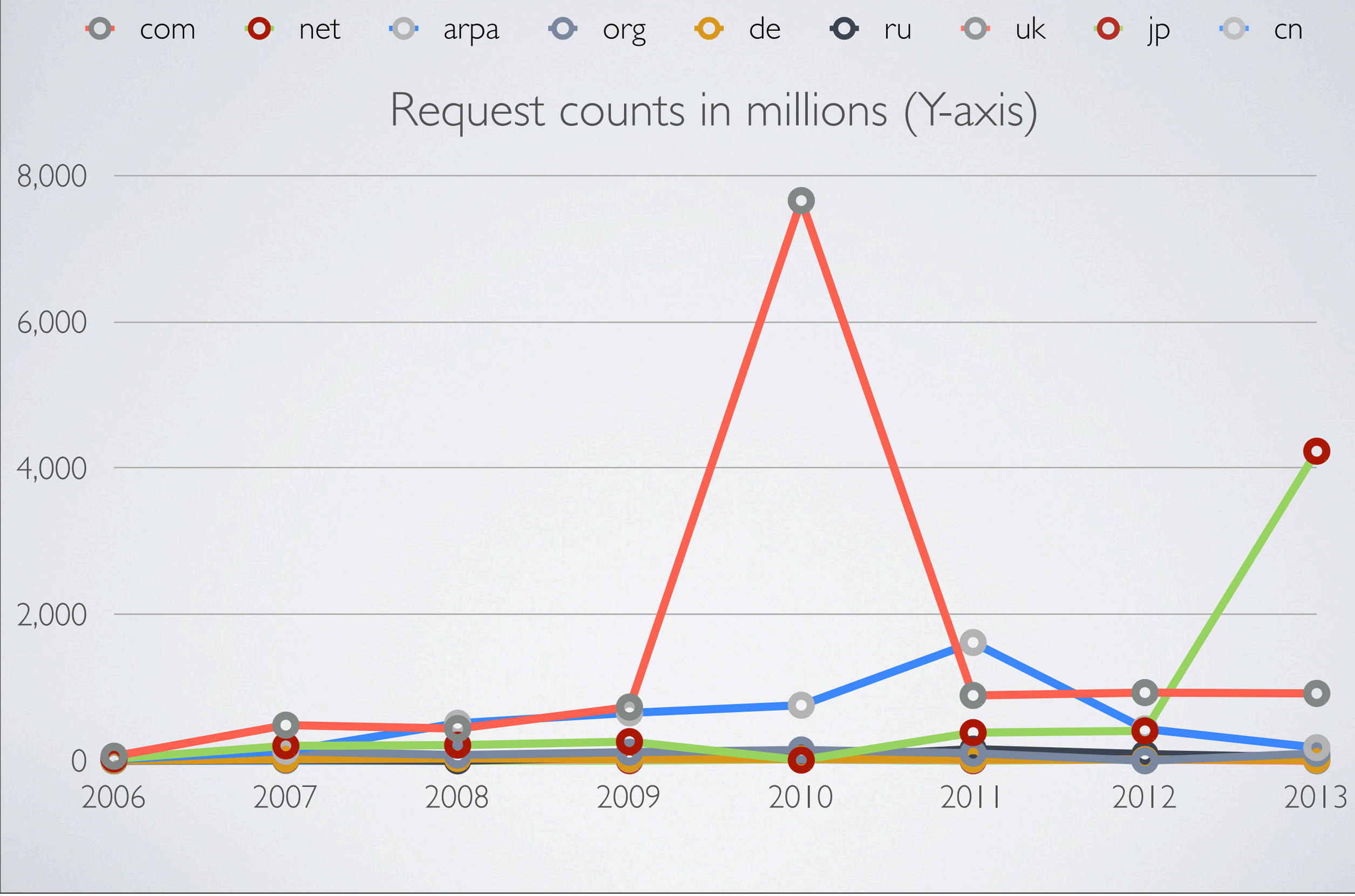
Historical Analysis

- Overall traffic patterns seem stable
- Little variation in each year's DITL data
  - Same TLDs appear in broadly the same position each year
- Behaviour of the DNS as a whole seems consistent
  - A few outliers
- Not much sign of "new/changed stuff" perturbing the observed traffic in the DITL data sets

Overall RD=1 Rates/Percentages



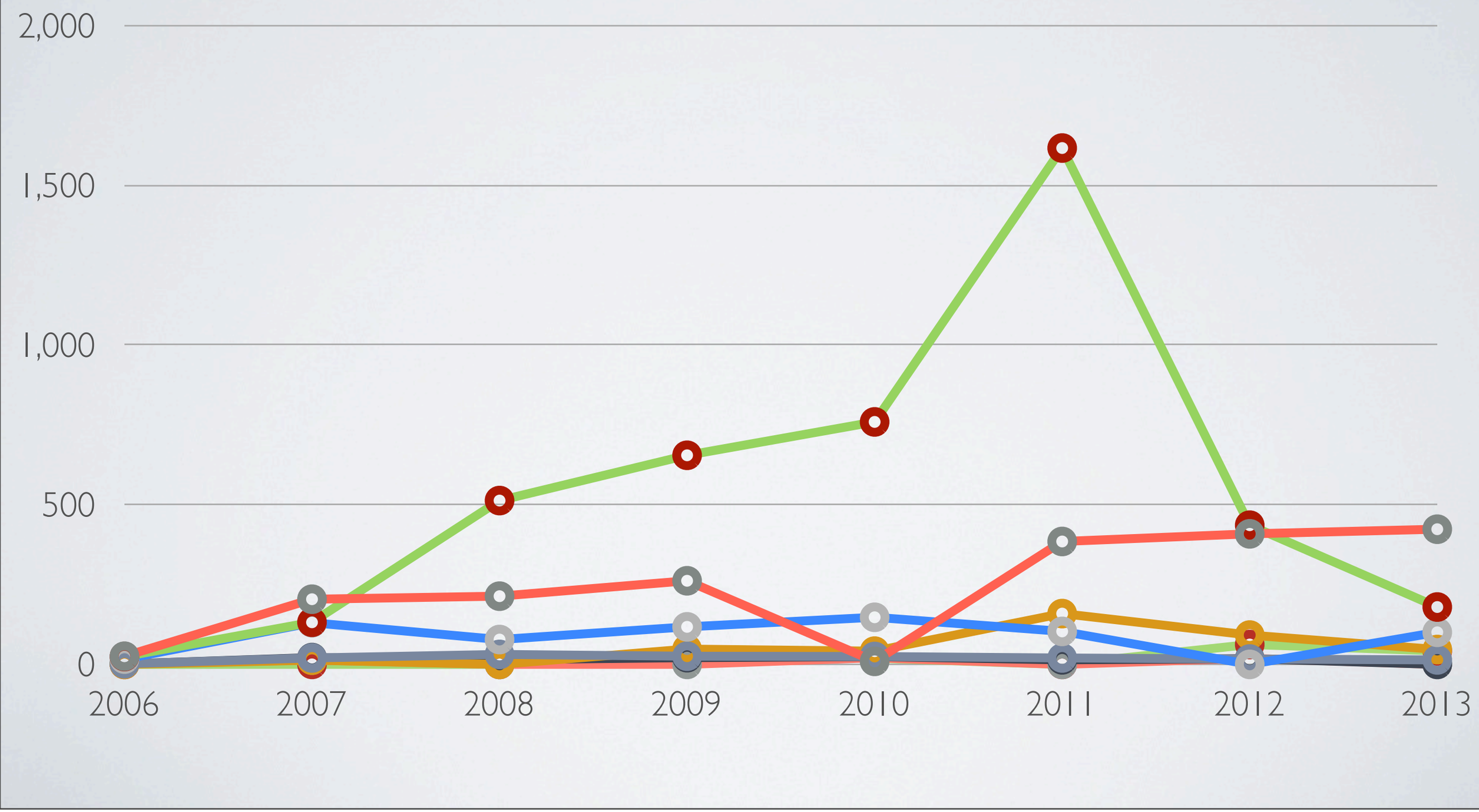
RD=1 Rates for Current TLDS



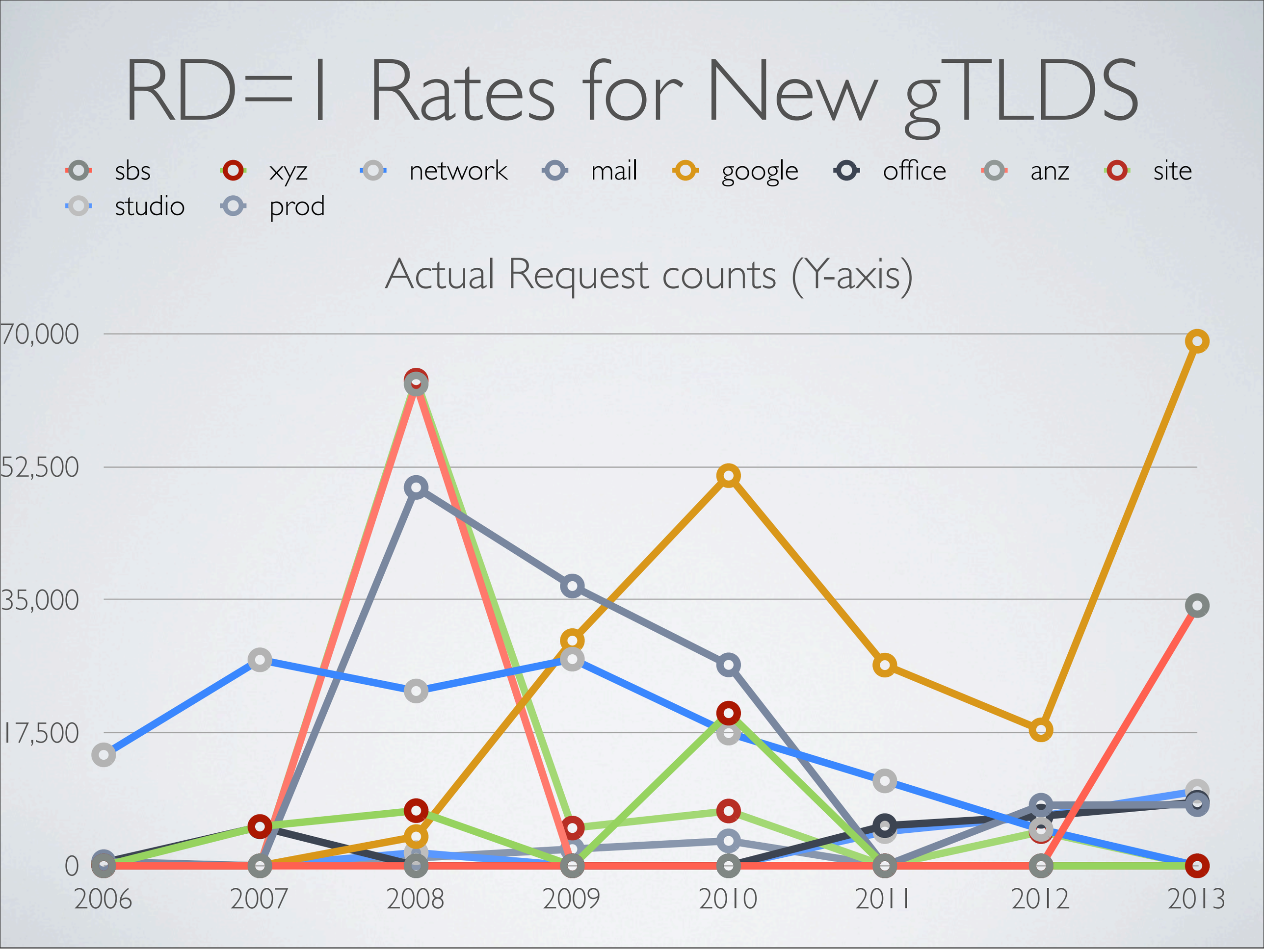
RD=1 Rates excluding .com

net arpa org de ru uk jp cn

Request counts in millions (Y-axis)



RD=1 Rates for New gTLDS



Qualitative Analysis

- In-depth analysis of everything would take forever and probably wouldn't unearth anything new
- Needed to make some simplifications:
  - Just looked at the glaringly obvious outliers
  - Ignored traffic levels below ICANN's "safe" threshold - except when there was something interesting to look at
- High-level summary: nothing to see here, move along

2013 Data

- 57,000 of 70,000 RD=1 queries for **`.google`** came from one IP address, a Californian school (***`something`*`.k12.ca.us`**)
- One IP address at a US ISP generated almost all the RD=1 lookups for **`.statefarm`**
  - Remainder had RFC1918 source addresses
  - Similar patterns for **`.thd`** and **`.sbs`** traffic
- Probably looking at isolated examples of rogue applications or misconfigured CPE
  - Unable to identify root cause(s) - so far

2012 Data

- Diffuse data sources for **`.google`** lookups:
  - ~600 /24s each generating ~600 queries
  - Some RFC1918 addresses again
- Probably not worth further investigation
  - QNAMEs generally for google's mail servers without a valid TLD suffix: e.g. **`gmail-smtp-in.l.google`**
- Transient stub resolver or mail server misconfiguration?

2008 Data - I

- Single /24 at a Florida ISP generated half the **`.anz`** RD=1 queries

- Gloriously bizarre QNAMEs:

- **`asad86158676.adeli.aks4you.irmr.maliblog.sina.virusgro.ups.iranmy.sharvin.lionel00.kooliver.2game2.aminpidofsh.2mb.rozmaregi.anz`**

- Clearly nothing to do with ANZ Bank

2008 Data - 2

- RD=1 queries for **`.mail`** were too diffuse to analyse/trace
  - Few hundred source /24s, each generating 300-500 requests

- Probably not worth further investigation either
  - Can anybody account for and explain a few hundred DNS queries for one day 6 years ago?
  - Could that info, if available, be meaningful or relevant today?

2008 Data - 3

- ~60,000 RD=1 queries for **`klingon.site`**
- All had the same query id - 0 - and source port
- All from the same IP address
  - Prefix assigned to University of Toronto
  - No reverse DNS
- Probably a student programming exercise gone wrong
  - Mr. Spock can't code? :-)

Botnet DDoS Considerations

- Details of a particular DDoS attack emerged during the analysis
  - Generates lots of spoof traffic with RD=1
  - Traffic had/has a distinctive footprint
- Re-examined the DITL data to see if this pattern was present
  - Didn't appear to be an issue:
    - No significant deviation in the distribution of source port numbers and query-ids
- Attack probably targets (signed) TLD name servers, not the root

Findings/Conclusions - 1

- There's a **lot** of RD=1 traffic going to the root already: ~12%
  - Probably always has been and always will be...
  - This doesn't seem to be breaking anything significant
    - Naive resolvers are either failing safe or working around referral responses somehow
- Billions of referrals from the root to **`.com`**, **`.net`**, **`.arpa`**, etc. do not seem to be causing problems for naive DNS clients today

Findings/Conclusions - 2

- RD=1 traffic for new gTLDs is **much** lower in absolute and relative values than the rates found for existing TLDs
  - Whatever generates these requests for new gTLDs should somehow cope OK with referral responses - probably
- Traffic for **`.google`** might be a concern if rogue clients are not isolated incidents
- Fairly stable (but low) rate of RD=1 requests for **`.mail`**
  - Could mean some mail gets delayed or bounced
- ICANN's name blocking strategy shouldn't cause harm

QUESTIONS?