Peer-to-Peer Networks

Chapter 1: Introduction

Thorsten Strufe

Note: these slides have been prepared with influence by material of Prof. Jussi Kangasharju

Lecture Outline

- Who are we?
- Course outline
- Organizational matters

- The P2P scenario
- Reprise of DS models
- Reprise of DNS
- P2P in a nutshell
- What‘s also hard?
- History of P2P
- Current state
- The future of P2P

Course Outline and Goals

- Course topic is peer-to-peer systems
- Take a look at current state in P2P systems, both in “real world” and in research work

- What does P2P mean?
- Why does P2P work?
- What makes a good P2P system, how can their quality be evaluated?
- How is the P2P principle reflected in today’s systems?
- What are new challenges / applications for P2P?

Who we are…

- Fachgebiet „Peer-to-Peer Netzwerke“
  - Prof. Thorsten Strufe (Lectures)
    - Piloty A310 (*forget it, write me an email!* ☺)
    - strufe [at] cs.tu-darmstadt.de
  - Stefanie Roos
    - roos[at]cs.tu-darmstadt.de

  - Dominik Fischer
  - Jan-Michael Heller

  - http://www.p2p.tu-darmstadt.de

Organizational matters

- Courses
  - Wed 9.50 – 11.30 (*semi c.t.*)
  - S2|02/C110
- Exercises
  - Tue 14.25 – 16.05
  - S2|02/C110
  - Both written and programming
  - Can be done in groups (of 2 ppl)
- Exams
  - Date: to be announced
  - Hopefully oral exams (if < 32 students)
  - Successfully completing exercises can buy you a bonus

- „The P2P-lecture“
    - http://www.p2p.tu-darmstadt.de/teaching/winter-term-2012/p2p-networks-lecture/
  - Mailing list: p2p-ws12 [at] informatik.tu-darmstadt.de (please subscribe)
  - Forum „D120“ at „Fachschaft“

Material

- Slides will be available on the web site

  (please send emails to push me, there are millions of other things in my head! :-/)

- Literature
  - scholar.google.com/citeseer, „p2p“,“peer-to-peer“,…
  - google (avoid wikipedia, unless you only want a very first impression)
  - Books:
    - Coulouris, Dollimore, Kindberg: „Distributed Systems“
    - Raj Jain: „The art of computer systems performance evaluation“
    - Booth, Colomb, Williams: „The craft of research“
    - Steinmetz/Wehrle „P2P systems“ (free on the web, no close relation)

Questions?

Peer-to-Peer?

- What is it?

- What does the word “peer” mean?
  - Merriam-Webster: one that is of equal standing with another **:** EQUAL; *especially* **:** one belonging to the same societal group especially based on age, grade, or status

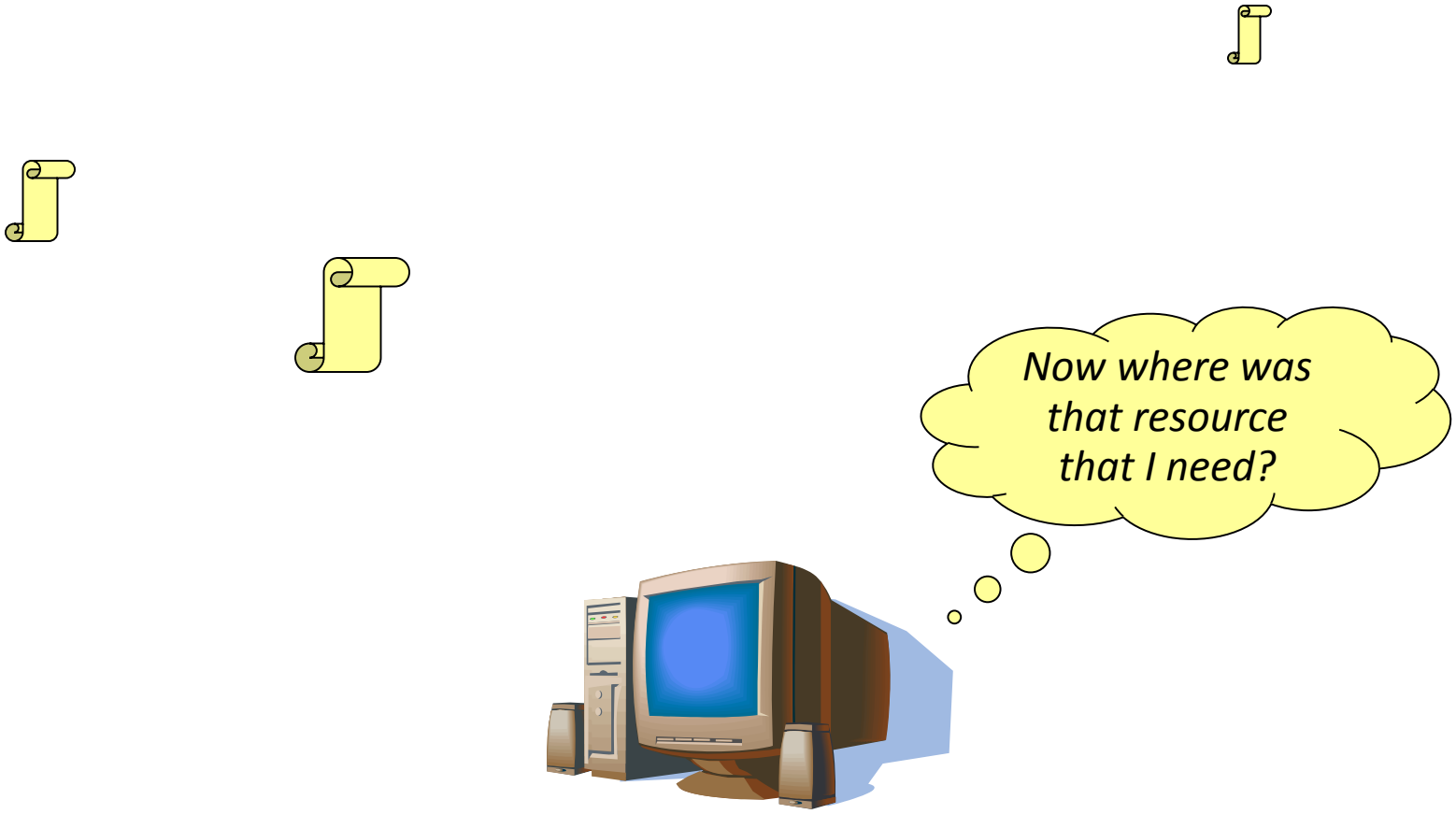
- Peer-to-peer: From one equal partner to another?

How do you define peer-to-peer?

What do you consider to be the main problems?

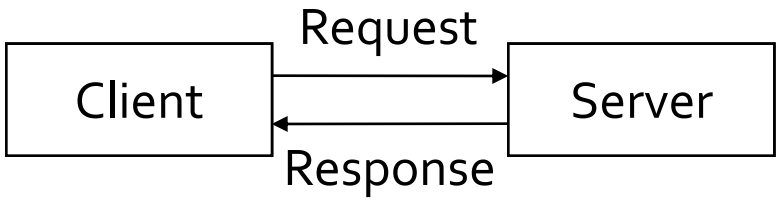
What do you think are the solutions (how does P2P work?)

The Peer-to-Peer Problem



Reprise: The client/server model

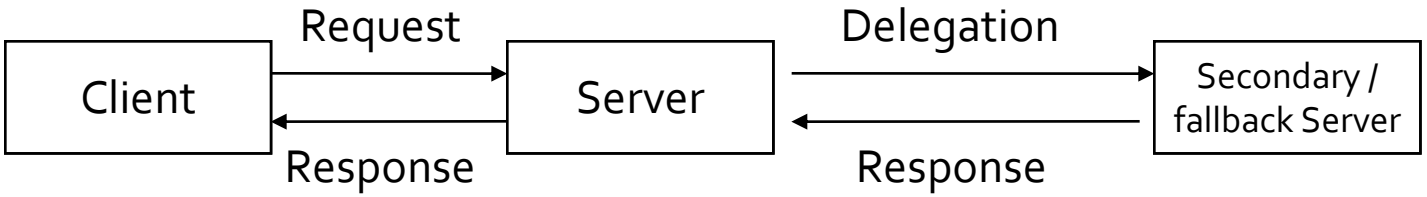
- Communication in request-response pairs
- The **role model** (the roles in the communication) is asymmetric
  - The client arrives and requests a service at any given point in time
  - The server is dedicated to the service and available and it responds immediately (well, after a processing time…)



- The **communication model** is asymmetric, too:
  - The clients send **requests** to a server
  - The server provides a solution and sends the results back in a **response**

Extended Client/Server Model

The client/server model is sometimes extended to more than two roles („client/server with delegation“)



- Definition of the clients remains the same:

  A **client** is, with respect to a considered service, the original and **initial source** of a **request** and the **sink for the response**.

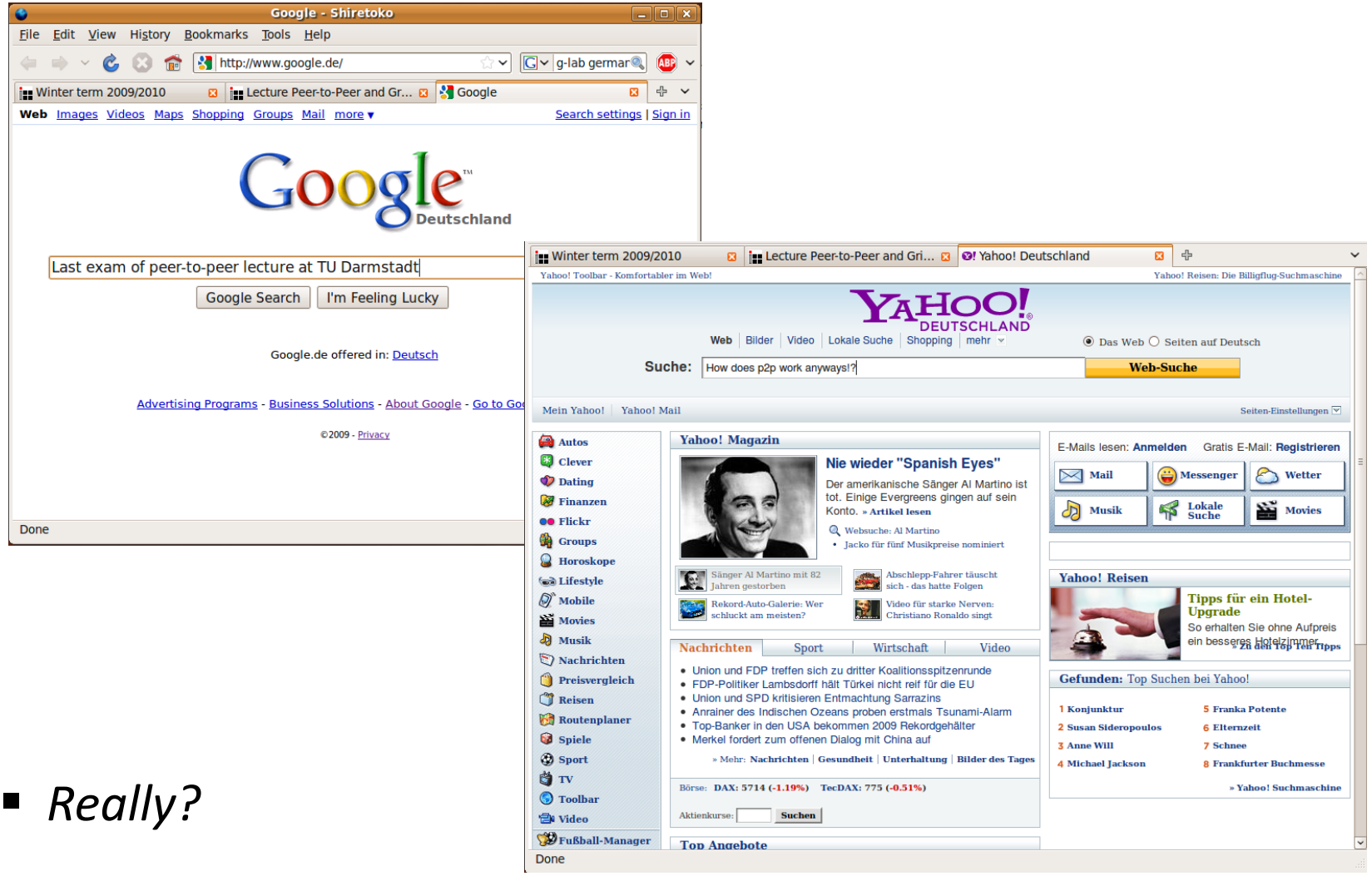
- Sidenote: Delegation may be *recursive* or *iterative*

- ***Beware!:*** *DNS calls or other services /external/ to the system are not part of the model!*

Locating Resources?

So how is it done in the Internet?

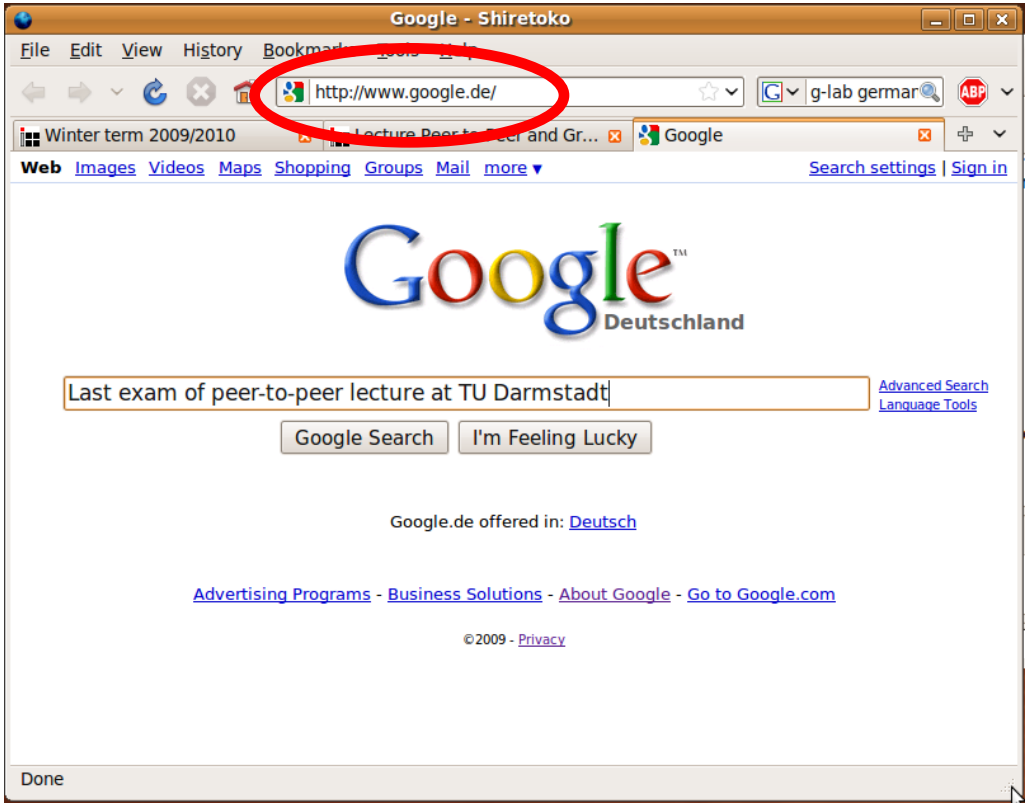
Real life resource location



- *Really?*

Reprise2: DNS – The Domain Name System

- Naming Service for (almost all) Internet traffic
- Lookup of (resolve)
  - Host-Addresses
  - Mail-Servers
  - Alias Names
  - Alternative Name Servers
  - …
- Distributed Database consisting of multitude of servers



DNS – Names

People: many identifiers:

- SSN, name, passport #

Internet hosts, routers:

- IP address (32 bit) - used for addressing datagrams
- "Name", e.g., www.yahoo.com - used by humans

Q: Map between IP addresses and name ?

Domain Name System:

Distributed database implemented in hierarchy of many *name servers*

Application-layer protocol: hosts, routers, name servers communicate to *resolve* names (address/name translation)

- Note: core Internet function, implemented as application-layer protocol
- Complexity at network's "edge"

DNS – what does it do?

DNS services

- Hostname to IP address translation
- Host aliasing
  - Canonical and alias names
- Mail server aliasing
- Load distribution
  - Replicated Web servers: set of IP addresses for one canonical name

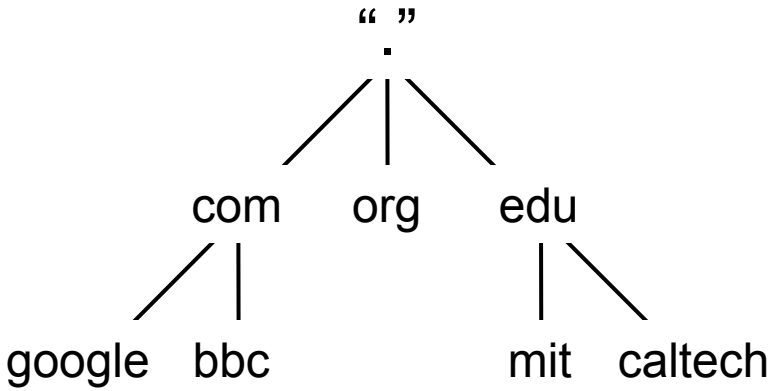
Why not centralize DNS?

- Single point of failure
- Traffic volume
- Distant centralized database
- Maintenance

does not *scale!*

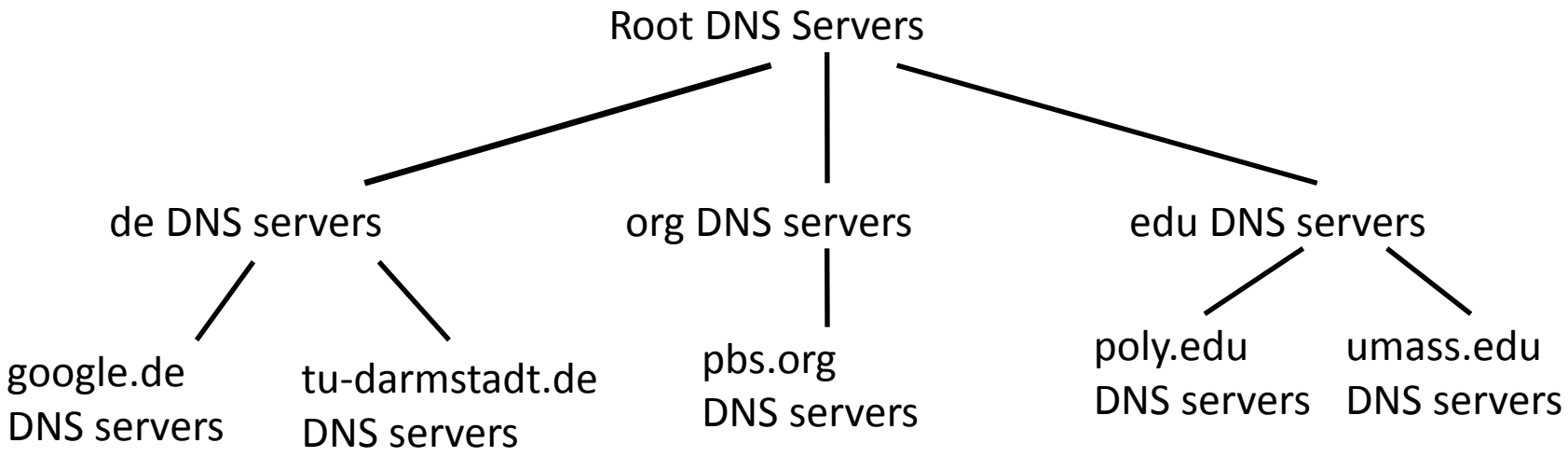
What does this „it scales“ mean anyways!?

DNS – Data Organization: Domains / Zones



- Structured Namespace
- Hierarchical organization in sub domains/zones
- Sourced at “root zone” (“.”)
- Parent zones maintain pointers to child zones (*“zone cuts”*)
- Zone data is stored as “Resource Records” (RR)

Distributed, Hierarchical Database



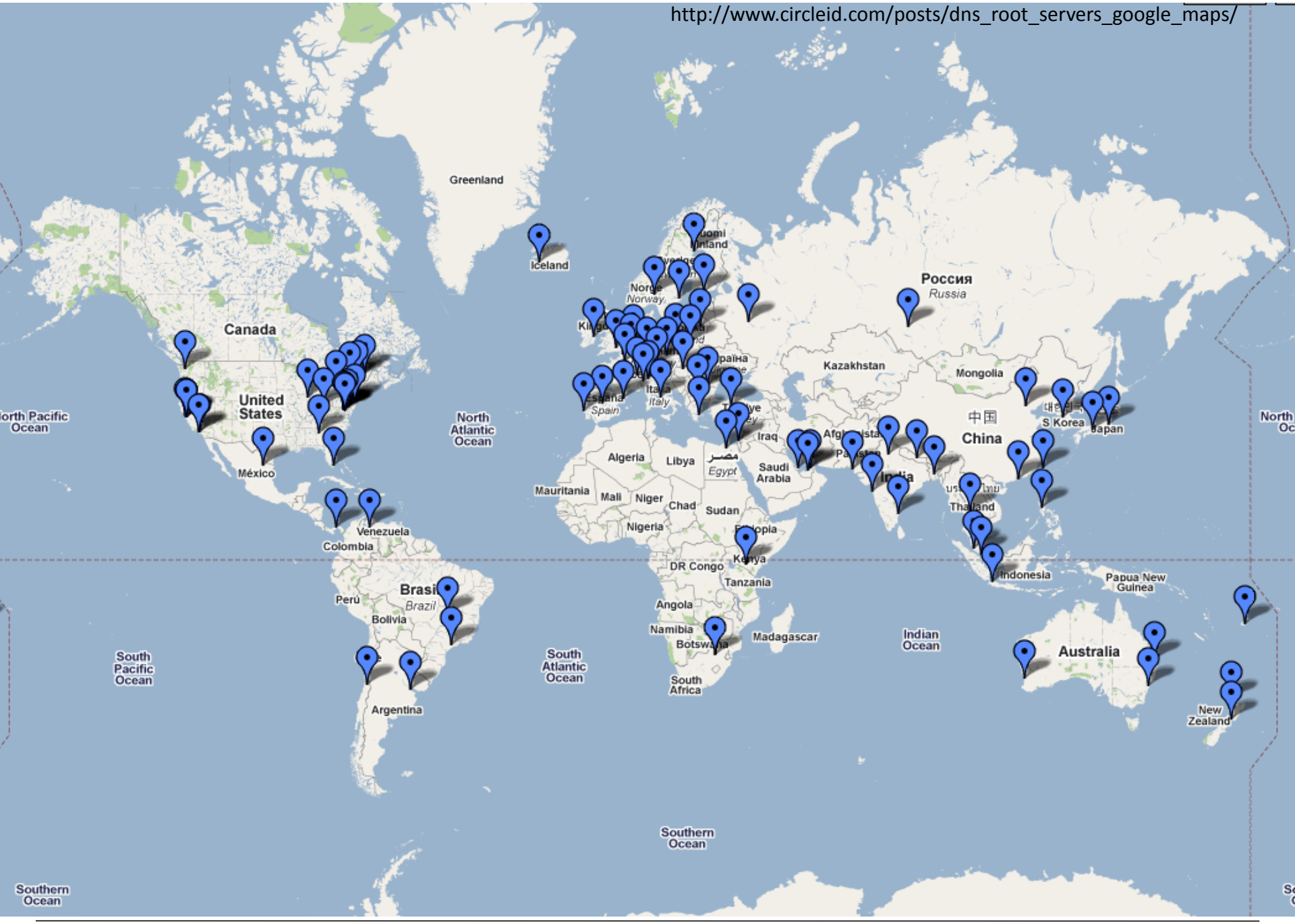
Client wants IP for www.p2p.tu-darmstadt.de; 1st approx:

- Client queries a root server to find **de** DNS server
- Client queries de DNS server to get **tu-darmstadt.de** DNS server
- Client queries tu-darmstadt.de DNS server to get IP address for www.p2p.tu-darmstadt.de

DNS: Root Name Servers

So, how many root nameservers are there actually? (physically)

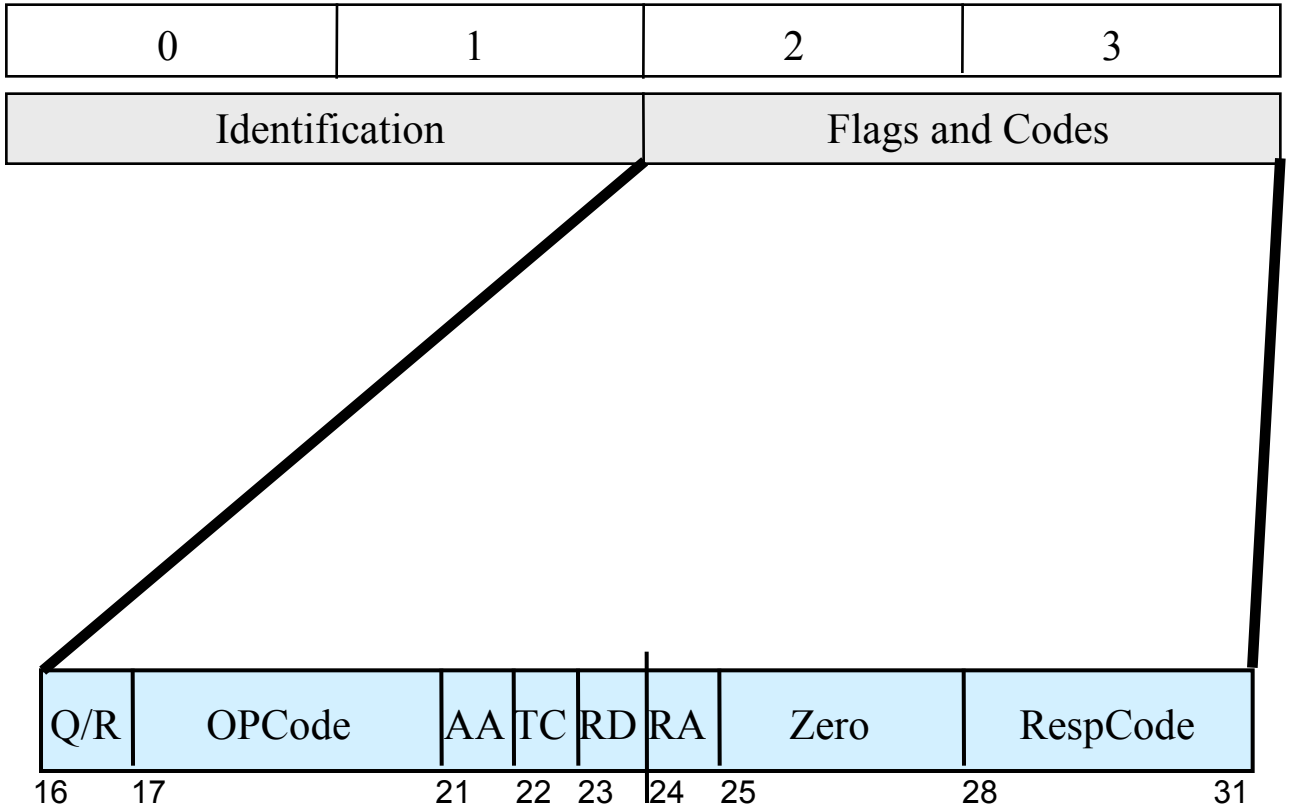
http://www.circleid.com/posts/dns_root_servers_google_maps/



DNS – Components

- Authoritative Server
  - Server maintaining authoritative content of a complete DNS zone
  - Top-Level-Domain (TLD) servers & auth servers of organization's domains
  - Pointed to in parent zone as authoritative
  - Possible load balancing: master/slaves
- Recursive (Caching) Server
  - Local proxy for DNS requests
  - Caches content for specified period of time (soft-state with TTL)
  - If data not available in the cache, request is processed recursively
- Resolver
  - Software on client's machines (part of the OS)
  - Windows-* and *nix: Stub resolvers
    - Delegate request to local server
    - Recursive requests only, no support for iterative requests

DNS – Message Format



- Q/R *Query/Response Flag*
- *Operation Code*
- AA *Auth. Answer Flag*
- TC *Truncation Flag*
- RD *Recursion Desired Flag*
- RA *Recursion Available Flag*
- Zero (three resv. bits)
- *Response Code*

DNS – Resource Records

- Atomic entries in DNS are called “Resource Records” (RR)
- Format:

  <name> [<ttl>] [<class>] <type> <rdata>

  - `name` (domain name of resource)
  - `ttl` (Time-to-live)
  - `class` (used protocol): IN (Internet), CH (Chaosnet)…
  - `type` (record type): A (Host-Address), NS (Name Server), MX (Mail Exchange), CNAME (Canonical Name), AAAA (IPv6-Host-Address), DNAME (CNAME, IPv6)
  - `rdata` (resource data): Content! (What did we want to look up?)

DNS Records

DNS: Distributed DB storing resource records (RR)

RR Format: **(name, value, type, ttl)**

- Type=A
  - **`name`** is hostname
  - **`value`** is IP address
- Type=NS
  - **`name`** is domain (e.g. foo.com)
  - **`value`** is IP address of authoritative name server for this domain
- Type=MX
  - **`value`** is name of mailserver associated with **`name`**
- Type=CNAME
  - **`name`** is alias name for some "canonical" (the real) name

    `www.ibm.com` is really

    `servereast.backup2.ibm.com`
  - **`value`** is canonical name

DNS: Caching and Updating Records

- Once (any) name server learns mapping, it *caches* mapping
  - Cache entries timeout (disappear) after some time
  - TLD servers typically cached in local name servers
    - Thus root name servers not often visited

- Update/notify mechanisms under design by IETF
  - RFC 2136
  - http://www.ietf.org/html.charters/dnsind-charter.html

Inserting Records Into DNS

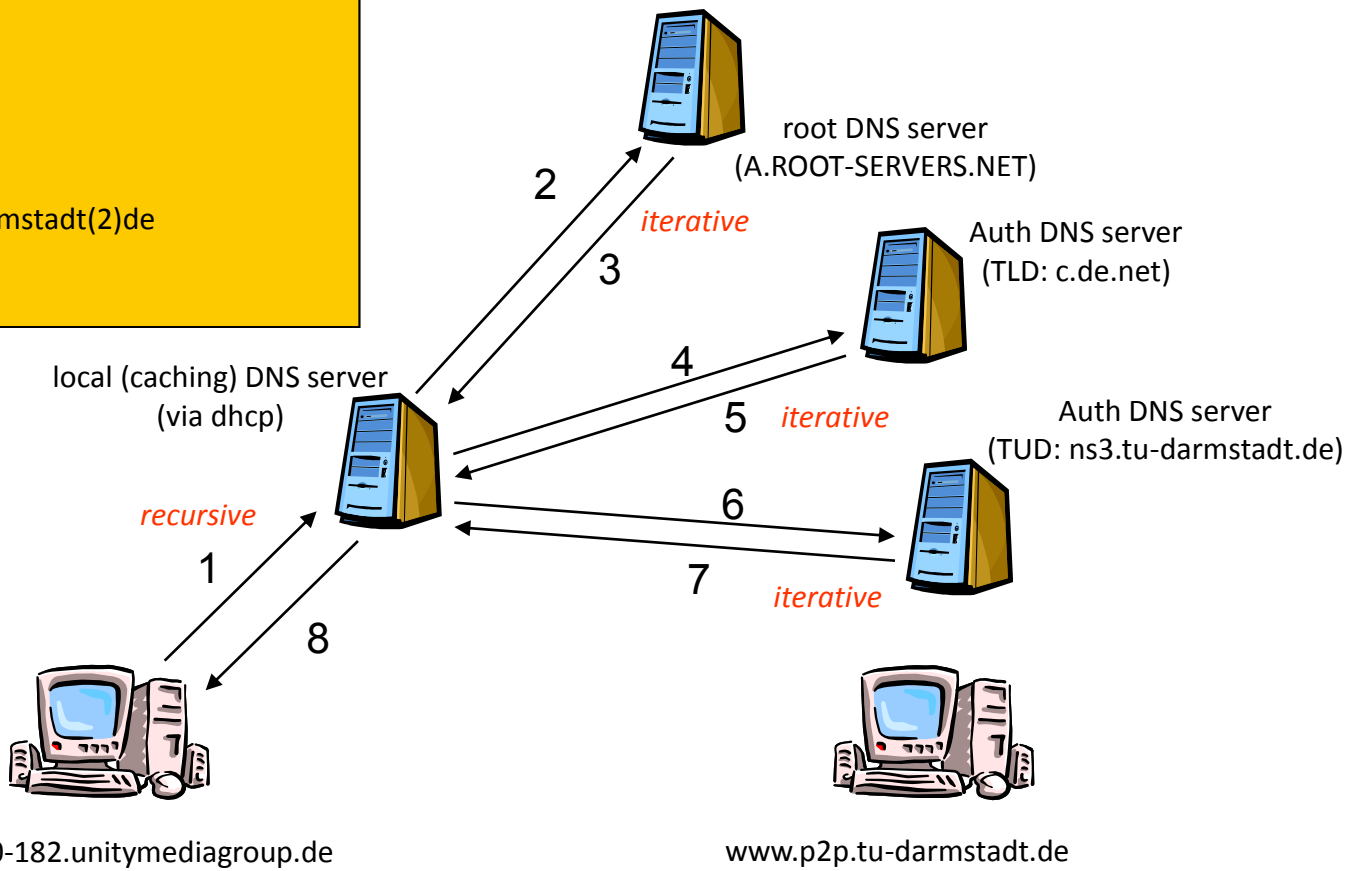
- Example: just created startup “Network Utopia”
- Register name networkutopia.com at a registrar (e.g., Network Solutions)
  - Need to provide registrar with names and IP addresses of your authoritative name server (***primary*** and ***secondary***)
  - Registrar inserts two RRs into the com TLD server:

```
(networkutopia.com, dns1.networkutopia.com, NS)
(dns1.networkutopia.com, 212.212.212.1, A)
```

- Put in authoritative server Type A record for www.networkuptopia.com and Type MX record for networkutopia.com

DNS – Recursive and Iterative Queries

```
S HEADER (send)
lentifier:        0x3116
ags:              0x00 (Q )
pcode:            0 (Standard query)
eturn code:       0 (No error)
umber questions:  1
umber answer RR:  0
umber authority RR: 0
umber additional RR: 0
UESTIONS (send)
ueryname:         (3)www(3)p2p(12)tu-darmstadt(2)de
ype:              1 (A)
lass:             1 (Internet)
```



A Quick Example…

```
strufe@eris:~$ dnstracer -v www.p2p.tu-darmstadt.de
Tracing to informatik.tu-darmstadt.de[a] via 130.83.163.141, maximum of 3 retries
130.83.163.141 (130.83.163.141) IP HEADER
-Destination address:  130.83.163.141
-DNS HEADER (send)
-- Identifier:         0x3116
-- Flags:              0x00 (Q )
-- Opcode:             0 (Standard query)
-- Return code:        0 (No error)
-- Number questions:   1
-- Number answer RR:   0
-- Number authority RR:  0
-- Number additional RR: 0
-QUESTIONS (send)
-- Queryname:          (3)www(3)p2p(12)tu-darmstad
-- Type:               1 (A)
-- Class:              1 (Internet)
-DNS HEADER (recv)
-- Identifier:         0x3116
-- Flags:              0x8080 (R RA )
-- Opcode:             0 (Standard query)
-- Return code:        0 (No error)
-- Number questions:   1
-- Number answer RR:   2
-- Number authority RR:  0
-- Number additional RR: 0
-…….
```

```
QUESTIONS (recv)
- Queryname:          (3)www(3)p2p(12)tu-darmstadt(2)de
- Type:               1 (A)
- Class:              1 (Internet)
ANSWER RR
- Domainname:         (6)charon(7)dekanat(10)informatik(12)tu-darmstadt(2)de
- Type:               1 (A)
- Class:              1 (Internet)
- TTL:                1592 (26m32s)
- Resource length:    4
- Resource data:      130.83.162.6
ANSWER RR
- Domainname:         (3)www(3)p2p(12)tu-darmstadt(2)de
- Type:               5 (CNAME)
- Class:              1 (Internet)
- TTL:                49817 (13h50m17s)
- Resource length:    28
- Resource data:      (6)charon(7)dekanat(10)informatik(12)tu-darmstadt(2)de
Got answer [received type is cname]
```

So where is the Info?

```
strufe@eris:~$ dnstracer -v -qns tu-darmstadt.de
Tracing to tu-darmstadt.de[ns] via 130.83.163.130
130.83.163.130 (130.83.163.130) IP HEADER
- Destination address:  130.83.163.130
DNS HEADER (send)
- Identifier:          0x4C45
- Flags:               0x00 (Q )
- Opcode:              0 (Standard query)
- Return code:         0 (No error)
- Number questions:    1
- Number answer RR:    0
- Number authority RR: 0
- Number additional RR: 0
QUESTIONS (send)
- Queryname:           (12)tu-darmstadt(2)de
- Type:                2 (NS)
- Class:               1 (Internet)
DNS HEADER (recv)
- Identifier:          0x4C45
- Flags:               0x8080 (R RA )
- Opcode:              0 (Standard query)
- Return code:         0 (No error)
- Number questions:    1
- Number answer RR:    5
- Number authority RR: 0
- Number additional RR: 9

.....
```

```
QUESTIONS (recv)
- Queryname:           (12)tu-darmstadt(2)de
- Type:                2 (NS)
- Class:               1 (Internet)
ANSWER RR
- Domainname:          (12)tu-darmstadt(2)de
- Type:                2 (NS)
- Class:               1 (Internet)
- TTL:                 70523 (19h35m23s)
- Resource length:     6
- Resource data:       (3)ns1(3)hrz(12)tu-darmstadt(2)de
ANSWER RR
- Domainname:          (12)tu-darmstadt(2)de
- Type:                2 (NS)
- Class:               1 (Internet)
- TTL:                 70523 (19h35m23s)
- Resource length:     5
- Resource data:       (2)ns(6)man-da(2)de
ANSWER RR
- Domainname:          (12)tu-darmstadt(2)de
- Type:                2 (NS)
- Class:               1 (Internet)
- TTL:                 70523 (19h35m23s)
- Resource length:     6
- Resource data:       (3)ns2(3)hrz(12)tu-darmstadt(2)de

.....
```

Answer ctd…

…….
ADDITIONAL RR
- Domainname: (3)ns1(3)hrz(12)tu-darmstadt(2)de
- Type: 1 (A)
- Class: 1 (Internet)
- TTL: 17335 (4h48m55s)
- Resource length: 4
- Resource data: 130.83.22.63
ADDITIONAL RR
- Domainname: (2)ns(6)man-da(2)de
- Type: 28 (unknown)
- Class: 1 (Internet)
- TTL: 38386 (10h39m46s)
- Resource length: 16
- Resource data: 2001:41b8:0000:0001:0000:0000:0000:0053
ADDITIONAL RR
- Domainname: (2)ns(6)man-da(2)de
- Type: 1 (A)
- Class: 1 (Internet)
- TTL: 38386 (10h39m46s)
- Resource length: 4
- Resource data: 82.195.66.249
ADDITIONAL RR
- Domainname: (3)ns2(3)hrz(12)tu-darmstadt(2)de
- Type: 28 (unknown)
- Class: 1 (Internet)
- TTL: 17335 (4h48m55s)
- Resource length: 16
- Resource data: 2001:41b8:083f:0022:0000:0000:0000:0063
…….

…….
ADDITIONAL RR
- Domainname: (3)ns2(3)hrz(12)tu-darmstadt(2)de
- Type: 1 (A)
- Class: 1 (Internet)
- TTL: 17335 (4h48m55s)
- Resource length: 4
- Resource data: 130.83.22.60
ADDITIONAL RR
- Domainname: (3)ns2(6)man-da(2)de
- Type: 1 (A)
- Class: 1 (Internet)
- TTL: 38386 (10h39m46s)
- Resource length: 4
- Resource data: 217.198.242.225
ADDITIONAL RR
- Domainname: (3)ns3(3)hrz(12)tu-darmstadt(2)de
- Type: 28 (unknown)
- Class: 1 (Internet)
- TTL: 17335 (4h48m55s)
- Resource length: 16
- Resource data: 2001:41b8:083f:0056:0000:0000:0000:0060
ADDITIONAL RR
- Domainname: (3)ns3(3)hrz(12)tu-darmstadt(2)de
- Type: 1 (A)
- Class: 1 (Internet)
- TTL: 17335 (4h48m55s)
- Resource length: 4
- Resource data: 130.83.56.60
Got answer

DNS – Lessons Learned

1. Structure name space (divide et impera)
2. Simple „routing“ b/c of structured (hierarchical) namespace

1. Store information at multiple locations
2. Maintain multiple connections
3. Be redundant! (Replicate…)
   - primary and secondary server, multiple TLD servers

4. Delegation using iterative or recursive forwarding

   (Btw: what are the pros and cons of each?)

What does this „routing“ mean anyways!?