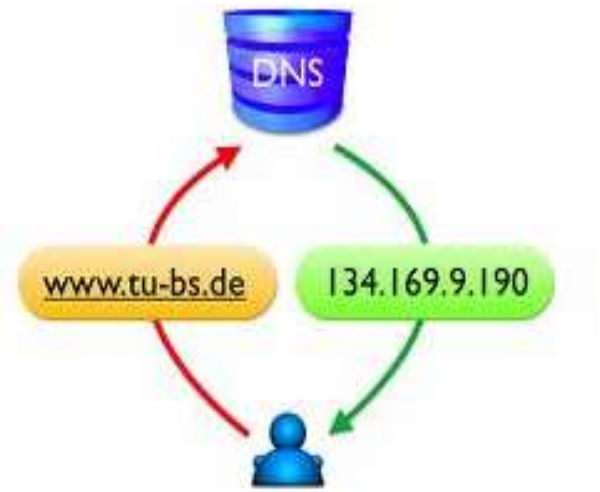
THU
Technische
Hochschule
Ulm

Rechnernetze

Kapitel 5: IP-Hilfsprotokolle

Hochschule Ulm
Prof. Dr. F. Steiper



Rechnernetze, INF2, 2022

- *Urheberrechte*
  - *Die Vorlesungsmaterialien und Vorlesungsaufzeichnungen zum Kurs „Rechnernetze (INF2)" dürfen nur für private Zwecke im Rahmen Ihres Studiums an der Technischen Hochschule Ulm genutzt werden.*
  - *Eine Vervielfältigung und Weitergabe dieser Materialien in jeglicher Form an andere Personen ist untersagt.*
  - *© Copyright. Frank Steiper. 2022. All rights reserved*

5.1 Address Resolution Protocol (ARP)

[Ref 1] Kapitel 5, Seite 505-508
[Ref 2] Kapitel 5, Seite 533-535

- *ARP (Address Resolution Protocol)*
  - *Vermittlungsprotokoll zwischen Adressen der Vermittlungs- und der Sicherungsschicht*
  - *Aufgabe des ARP-Protokolls in Verbindung mit IP*
    - *Ermittlung der MAC-Adresse eines Zielrechners aus dessen IP-Adresse*
    - *Besonderheit:*
      ARP-Meldungen werden in Sicherungsschicht-Rahmen transportiert!
  - *RARP: Reverse ARP-Protokoll in Verbindung mit IP*
    - *Ermittlung der IP-Adresse zu einer MAC-Adresse*
    - *Benötigt einen RARP-Server*
    - *Vorläufer-Protokoll von DHCP (Dynamic Host Configuration Protocol)*
      → *Wichtig bei Rechnern ohne Festplatte, die ihre eigene IP-Adresse nicht speichern können*

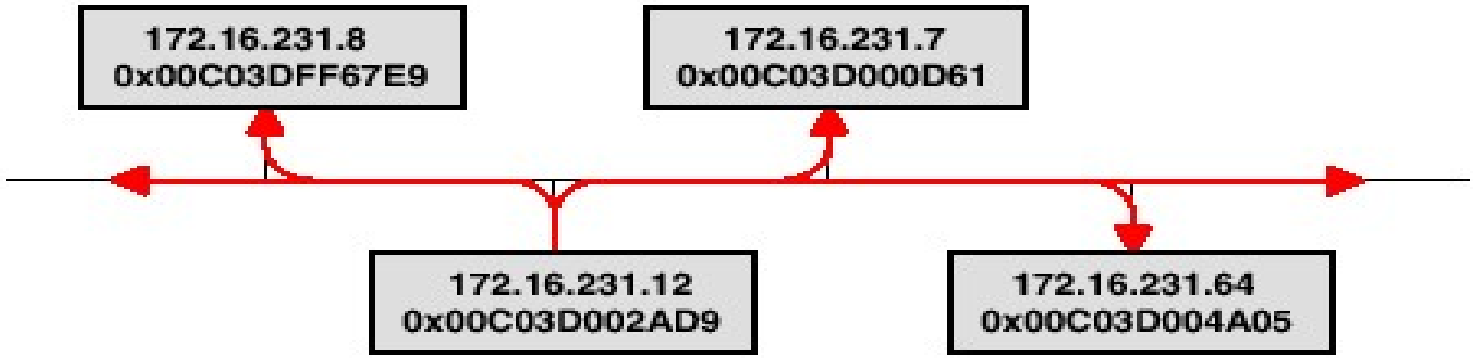
5.1 Address Resolution Protocol (ARP)

- *ARP-Tabelle*
  - *Jeder Rechner im LAN baut eine ARP-Tabelle (=ARP-Cache) auf*
    - *Die ARP-Spezifikation definiert das Frage-/Antwort-Protokoll, über das ein Rechner diese Tabelle füllen/modifizieren kann*
    - *Einträge altern nach einer konfigurierbaren Zeit aus*
      - *→ Typisch: 3-5 Minuten*
    - *Zweck: Reduktion der Senderate von ARP-Meldungen*

IP Address	Hardware Address
197.15.3.2	0A:07:4B:12:82:36
197.15.3.3	0A:9C:28:71:32:8D
197.15.3.4	0A:11:C3:68:01:99
197.15.3.5	0A:74:59:32:CC:1F
197.15.3.6	0A:04:BC:00:03:28
197.15.3.7	0A:77:81:0E:52:FA

5.1 Address Resolution Protocol (ARP)

- *Ablauf: ARP-Request/RARP-Request*
  - (R)ARP-Request wird i.d.R. an die MAC-Broadcast-Adresse gesendet:



ARP Request: **Wem gehört die IP-Adresse 172.16.231.64?**

Beispiel:

Quell-Hardware-Adresse:	0x00C03D002AD9
Quell-IP-Adresse:	172.16.231.12
Ziel-Hardware-Adresse:	0x0:0:0:0:0:0
Ziel-IP-Adresse:	172.16.231.64

} Adress-Felder **in(!) der** ARP-Nachricht

RARP-Request: **Ich nenne meine physikalische Adresse. Wer kennt meine IP-Adresse?**

Beispiel:

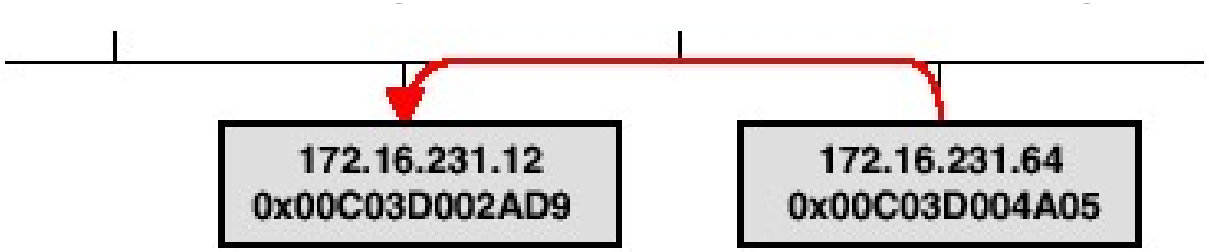
Quell-Hardware- Adresse:	0x00C03D002AD9
Quell-IP-Adresse:	0.0.0.0
Ziel-Hardware-Adresse:	0x0:0:0:0:0:0
Ziel-IP-Adresse:	255.255.255.255

} Adress-Felder **in(!) der** RARP-Nachricht

5.1 Address Resolution Protocol (ARP)

- *Ablauf: ARP-Reply/RARP-Reply…*

  - (R)ARP-Reply wird gezielt an die MAC-Adresse des Anfragers geschickt



ARP / RARP-Response:

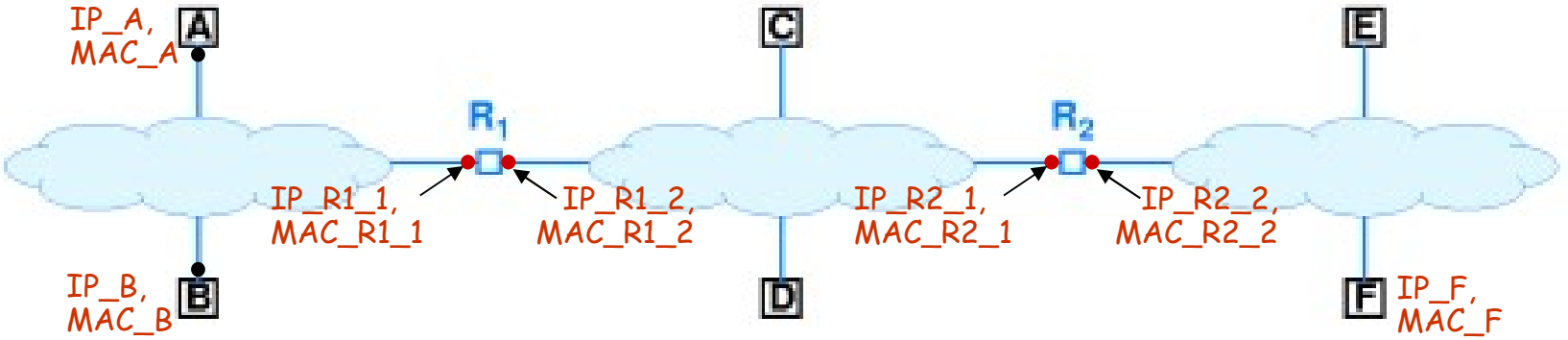
Beispiel:		
	Quell-Hardware- Adresse:	0x00C03D004A05
	Quell-IP-Adresse:	172.16.231.64
	Ziel-Hardware-Adresse:	0x00C03D002AD9
	Ziel-IP-Adresse:	172.16.231.12

Adress-Felder **in(!) der** (R)ARP-Nachricht

5.1 Address Resolution Protocol (ARP)

Kopplung physikalischer Netze durch Router

- *Das ARP-Protokoll funktioniert nicht über Router hinweg*
- *Beispiel:*
  - *Jeder Broadcast-Domäne wird ein IP-Netz zugeordnet*
  - *Die IP-Netze werden durch Router (R_1 und R_2) verbunden*
  - *Ein Sender versucht nie, die IP-Adresse eines Empfängers aufzulösen, der nicht zum gleichen IP-Netz gehört*



5.1 Address Resolution Protocol (ARP)

- *1. Fall: Kommunikation innerhalb eines physikalischen Netzes*
  - *Beispiel: Rechner A sendet an Rechner B*
    - *1. Schritt: A stellt fest, dass B zum gleichen IP-Netz gehört*
      → *Bitweises AND zwischen IP_B und Netzmaske von A*
    - *2. Schritt: Aus IP_B wird MAC_B abgeleitet*
      → *Über lokale ARP-Tabelle von Rechner A oder durch ARP-Request*
    - *3. Schritt: IP-Paket wird in Rahmen gekapselt und an MAC_B gesendet*
      → *Quell-MAC-Adresse des Rahmens ist MAC_A*
    - *4. Schritt: Rechner B extrahiert IP-Paket aus Rahmen und leitet es an die Vermittlungsschicht weiter*

5.1 Address Resolution Protocol (ARP)

- *2. Fall: Kommunikation über physikalische Netze hinweg*
  - *Rechner A sendet an Rechner F*
    - *1. Schritt: A stellt fest, dass F nicht zum IP-Netz_1 gehört*
    - *2. Schritt: A ermittelt aus IP-Adresse des Router-Interfaces R1_1 die zugehörige MAC-Adresse MAC_R1_1*
      - *→ A verpackt IP-Paket in Rahmen und sendet diesen an MAC_R1_1*
      - *→ Quell-MAC-Adresse des Rahmens ist MAC_A*
    - *3. Schritt: Router R1 entpackt IP-Paket und analysiert Ziel-IP-Adresse*
      - *→ Über Weiterleitungstabelle wird als Gateway IP_R2_1 ermittelt*
      - *→ Router R1 besorgt sich über einen ARP-Request MAC_R2_1*
      - *→ Das IP-Paket wird in einen Rahmen verpackt und an MAC_R2_1 verschickt*
      - *→ Quell-MAC-Adresse des Rahmens ist MAC_R1_2*
      - *→ Das übertragene IP-Paket wird nicht modifiziert*
    - *4. Schritt: Kommunikation zwischen R2 und F ähnlich wie Fall 1*

5.2 Dynamic Host Configuration Protocol (DHCP)

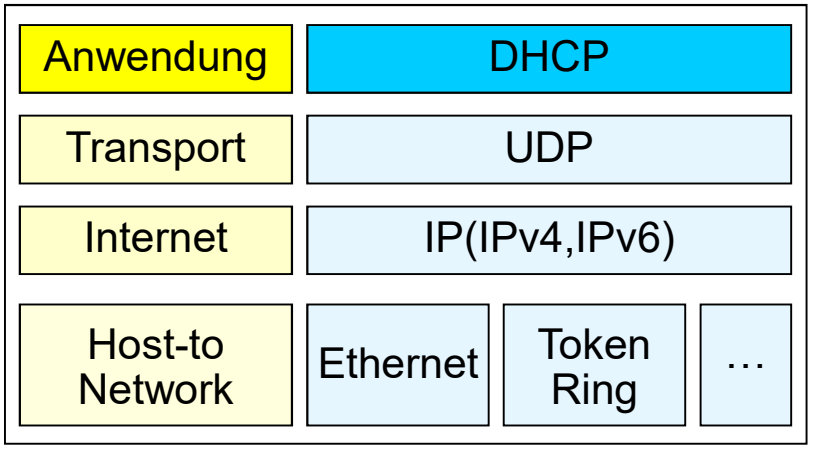
- *Funktion*

http://de.wikipedia.org/wiki/Dynamic_Host_Configuration_Protocol
[Ref 1] Kapitel 4, Seite 385-389 [Ref 2] Kapitel 5, Seite 536

  - *Client-Server-Prinzip: DHCP-Server/DHCP-Client*
  - *„Plug and Play“-Prinzip im Intranet*
- *Zentralisierte IP-Konfiguration von Netzwerk-Knoten*
  - *Statische Zuordnung*
    - *Client bekommt vom Server immer eine vorgegebene IP-Adresse und weitere IP-Konfig-Parameter zugeordnet*
  - *Dynamische Zuordnung*
    - *Client erhält von Server für bestimmte Zeit (Lease-Time) IP-Adresse aus bestimmten IP-Adressbereich zugeordnet*

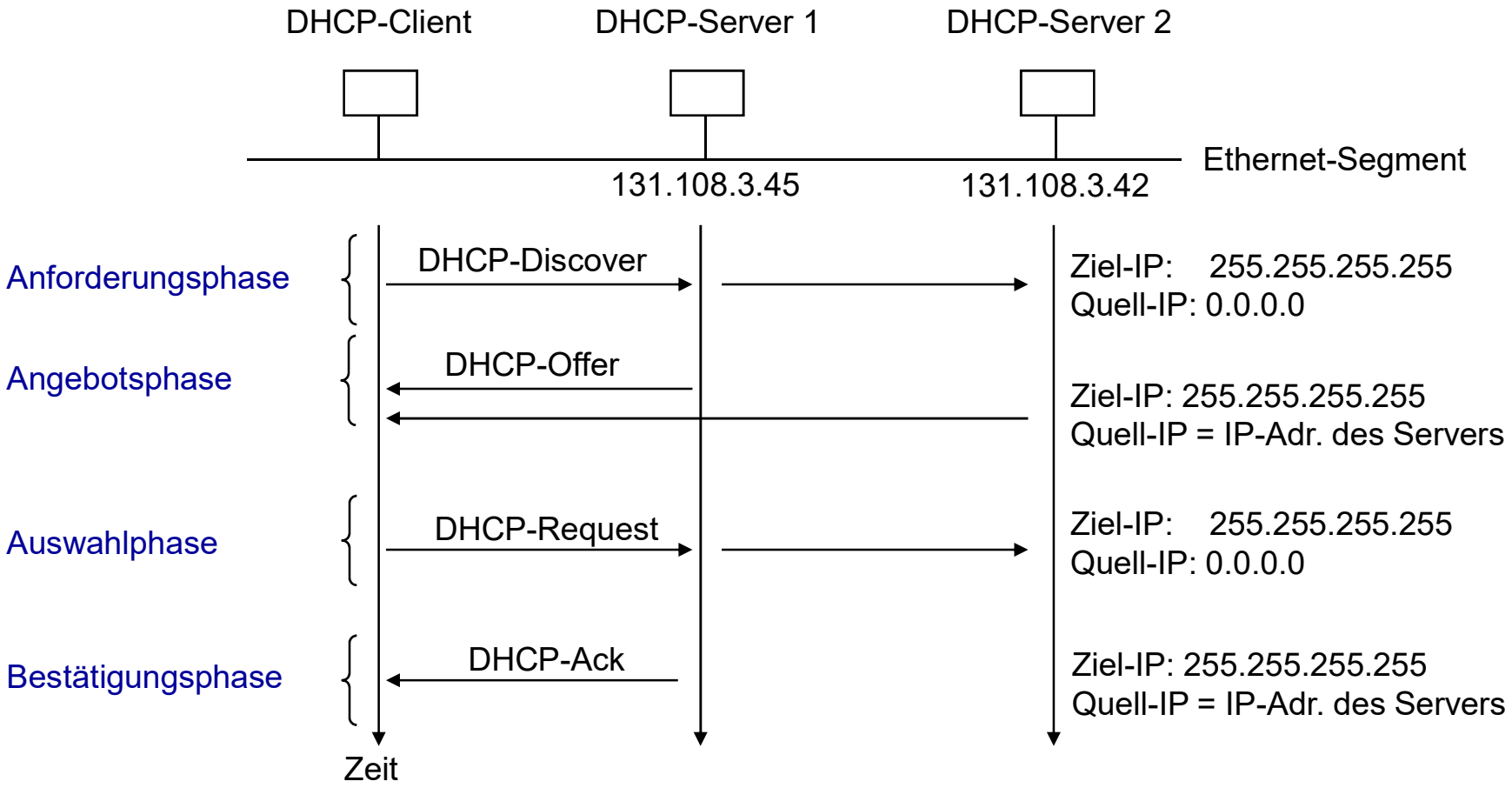
5.2 Dynamic Host Configuration Protocol (DHCP)

- *An Client zuweisbare IP-Konfigurationsparameter*
  - *IP-Adresse, Netzmaske, Default-Gateway-Adresse*
  - *DNS-/Time-/Wins-Server-Adresse*
  - *Name der lokalen DNS-Domain, …*
    - *Standardisierte Parameter-Optionen: http://www.iana.org/assignments/bootp-dhcp-parameters*
- *Einordnung in die Internet Protokoll Suite*
  - *Spezifiziert im RFC 2131*
    - *Arbeitet auf Anwendungs-Schicht*
    - *Nutzt das UDP-Protokoll*
  - *Verwendete UDP-Port-Nummern*
    - *DHCP-Server-Prozess: Port 67*
    - *DHCP-Client-Prozess: Port 68*



5.2 Dynamic Host Configuration Protocol (DHCP)

- *DHCP-Ablauf*
  - *Beispielkonfiguration*



5.2 Dynamic Host Configuration Protocol (DHCP)

- *Ablauf der Lease-Time*
  - *Der DHCP-Client muss nach Ablauf von 50% des Lease-Zeitraums weiteres Interesse an der IP-Adresse bei dem DHCP-Server anmelden, der die IP-Adresse vergeben hat*
    - *Erfordert einen DHCP-Request/DHCP-Ack-Austausch*
    - *Der angefragte DHCP-Server kann den Lease-Zeitraum erneuern*
    - *Wenn der gewünschte DHCP-Server nicht antwortet, kann der DHCP-Client die IP-Adresse weiterhin verwenden*
  - *Nach 87,5% des Lease-Zeitraums sendet der DHCP-Client einen erneuten DHCP-Request*
    - *Falls DHCP-Ack ausbleibt -> DHCP-Discover, um neue IP anzufordern*
  - *Falls der Lease-Zeitraum trotzdem abläuft, muss der DHCP-Client seine IP-Adresse freigeben und er kann damit nicht mehr kommunizieren*
    - *Problem: Was passiert, wenn Client sich nicht Protokoll konform verhält?*

5.3 Domain Name Service (DNS)

- *Hauptaufgabe*

[Ref 1] Kapitel 2, Seite 160-174
[Ref 2] Kapitel 7, Seite 695-707

  - *Verzeichnisdienst, der Hostnamen in IP-Adressen wandelt und umgekehrt*

- *Weitere Dienste*

Host-Aliasing

= regulärer Hostname

  - *Einem kanonischen Hostnamen werden mehrere Alias-Namen zugeordnet*
  - *Alias-Namen werden auch in die IP-Adresse des Hosts aufgelöst*

  - *Mail-Server-Aliasing*

=sprechend

    - *E-Mail-Adressen sollten mnemonisch sein (z.B. bob@gmail.com)*
    - *Der kanonische Hostname des Mail-Servers kann komplizierter sein (z.B. relay1.west-coast.gmail.com)*
    - *Mail-Anwendung nutzt Alias, um Adresse des Mailservers zu bestimmen*

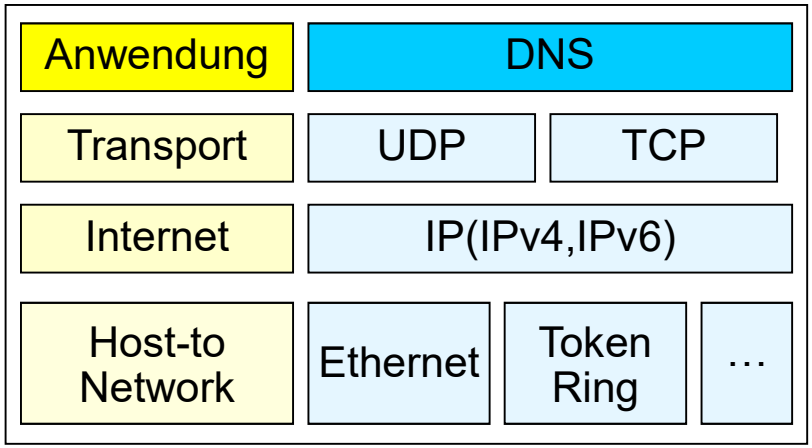
  - *Lastverteilung*
    - *Gruppe von IP-Adressen wird mit einem kanonischen Hostnamen assoziiert*
    - *Die DNS-Anfrage wird mit wechselnden IP-Adressen in einem "Round-Robin"-Verfahren beantwortet, um die Last zwischen replizierten Servern zu verteilen*

  - *Rufnummernabbildung*
    - *E.164-Telefonnummer wird in Internet-Adresse oder andere Ressourcen-Informationen übersetzt ; Konvergenz von traditionellem Telefondienst und Internet-Telefondienst (Voice over IP)*

5.3 Domain Name Service (DNS)

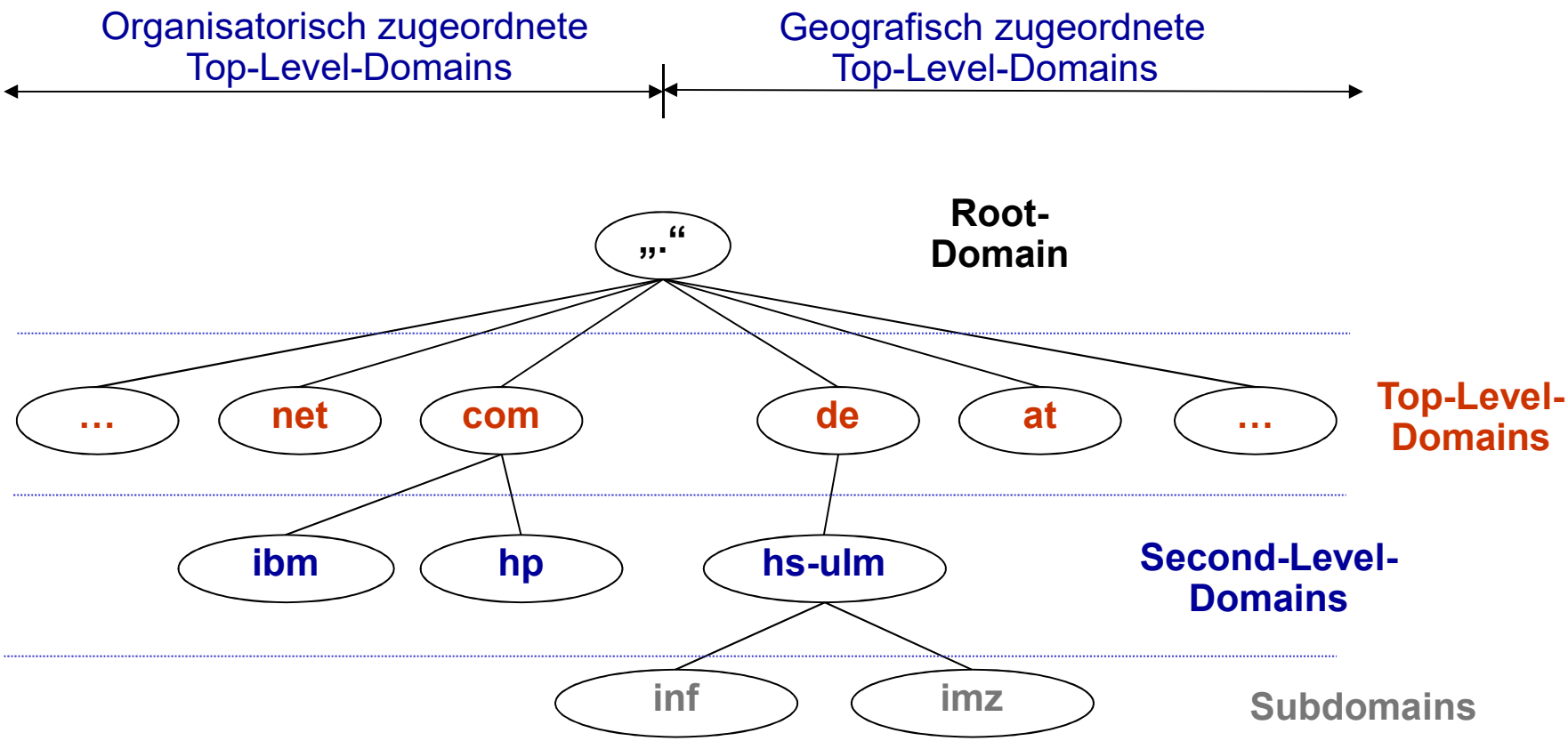
- *Einordnung in die Internet Protokoll Suite*
  - *Spezifiziert in RFCs 1034/1035 und ergänzenden RFCs*
    - *Arbeitet auf Anwendungs-Schicht*
    - *Nutzt standardmäßig das UDP-Protokoll*
      - → *Falls eine DNS-Nachricht nicht in ein UDP-Datagramm passt, wird auf TCP übergegangen!*
      - → *Bei Zonen-Transfers zwischen DNS-Servern kann TCP genutzt werden!*
  - *Verwendete Port-Nummer*
    - *DNS-Server: Port 53 (tcp/udp)*
      - → *Client sendet DNS-Request mit einem Quell-Port>1023*
      - → *Server antwortet mit Quell-Port 53*
      - → *Bei Server-Server-Kommunikation (Zonen-Transfer) ist Quell- und Source-Port=53*

Über Port 53 kann Service mit telnet getestet werden



5.3 Domain Name Service (DNS)

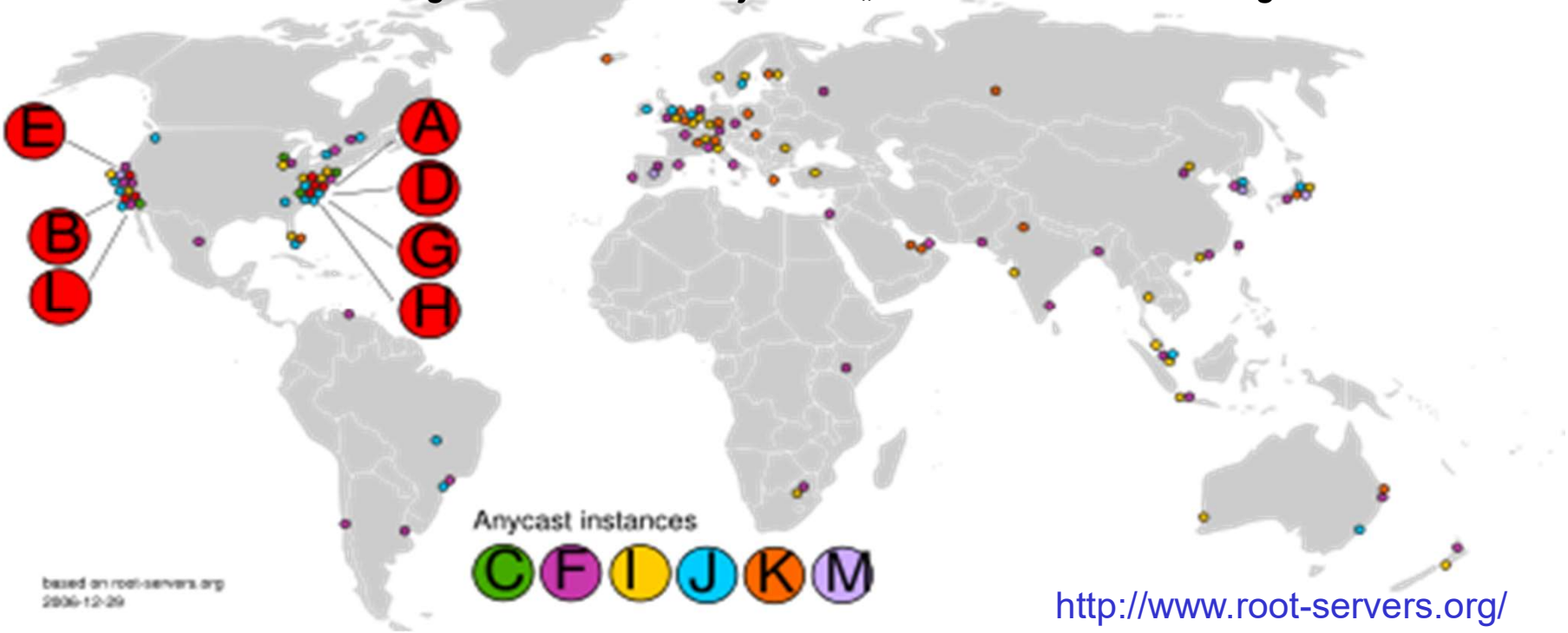
- *Aufbau des DNS-Namensraums*



- *DNS-Root-Server*

http://www.heise.de/newsticker/meldung/32451

  - *Gegenwärtig existieren 13 logische Root-Server*
    - *Alle Server bestehen eigentlich aus mehreren Anycast-Instanzen*
      - → *Identische Kopien eines Servers mit gleichen IP-Nummern*
      - → *DNS-Anfragen werden an die jeweils „nächste“ Instanz weitergeleitet*



http://www.root-servers.org/

- *DNS-Zonen*
  - *DNS-Zonen sind die Verwaltungseinheiten des DNS-Namensraums*
    - *Eine Zone ist Untermenge einer Domain*
    - *z.B. kann „thu“-Domain die Subdomains „inf“ und „imz“ besitzen*

      → *Zwei Zonen definieren die Einträge von „inf.thu.de“ und „imz.thu.de“*

      → *Eine dritte Zone beinhaltet die restlichen Einträge für „thu.de“*
  - *Eine Zone wird durch einen autoritativen „Primary Server“ verwaltet*
    - *Veränderungen der Einträge einer Zone sind nur über Primary Server möglich*
    - *Die Einträge für Zonen werden in „Zonendateien“ gespeichert*
    - *Ein Eintrag einer Zonendatei heißt „Resource Record“*

5.3 Domain Name Service (DNS)

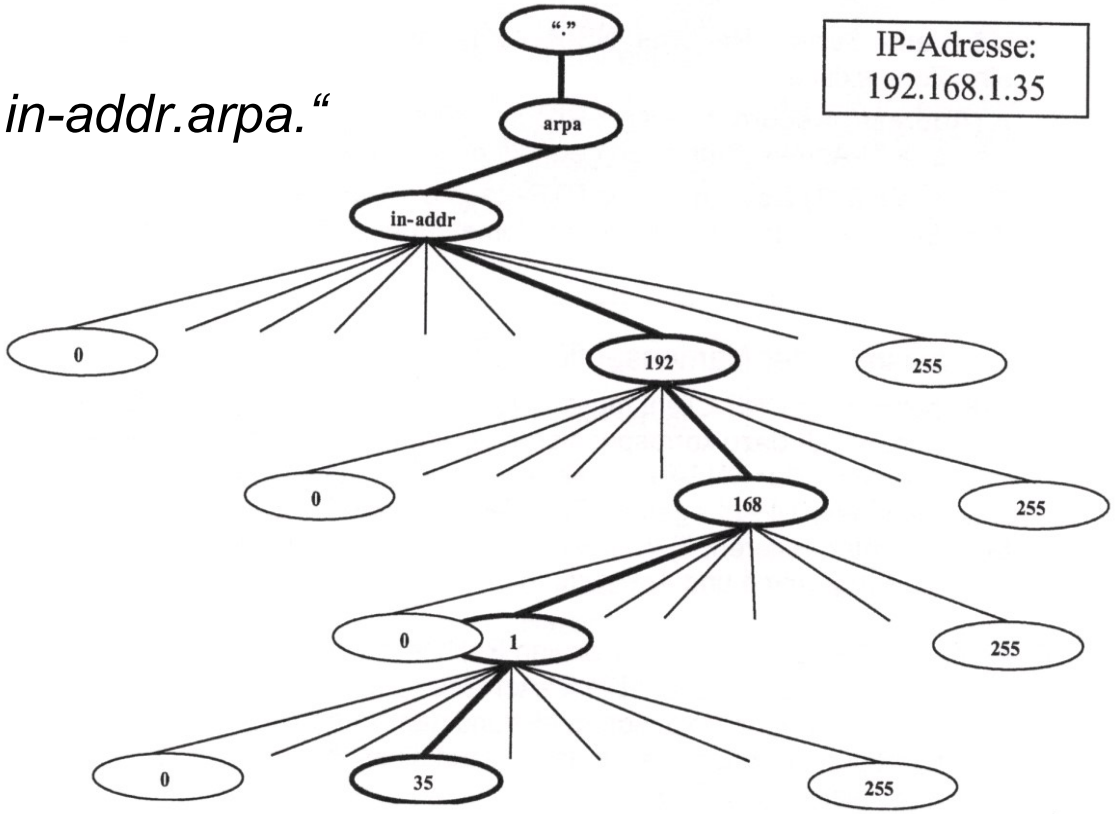
- *Ressource Records*
  - *Die Einträge in Zonen-Dateien heißen Resource Records (RR)*
    - *Es gibt verschiedene Typen von Records:*

WICHTIGE DATENTYPEN IM DNS

Abkürzung	Erklärung
A	IP-Adresse. Ein Hostname darf auf mehrere IP-Adressen zeigen!
CNAME	Alias: Ein Hostname zeigt auf einen anderen Hostnamen, erst dieser zeigt auf eine IP-Adresse.
PTR	Pointer: Eine IP-Adresse zeigt auf einen Hostnamen.
MX	Mail Exchange: Welcher Server nimmt E-Mail für die betreffende Domain entgegen?
NS	Gibt einen Name-Server an, der für eine bestimmte Adresse oder einen ganzen Adressbereich autorita-
tive Daten vorhält.	
SOA	„Start Of Authority": Die hier gezeigten Daten gelten für alle folgenden Adressen. Derartige Einträge enthalten Gültigkeitszeiten und verweisen über eine E-Mail-Adresse auf den verantwortlichen Host- master, auf den Menschen also, der die Daten pflegt.
ANY	Ein Pseudo-Datentyp, den Sie *dig* übergeben, wenn Sie einfach alle verfügbaren Daten auflisten wollen.

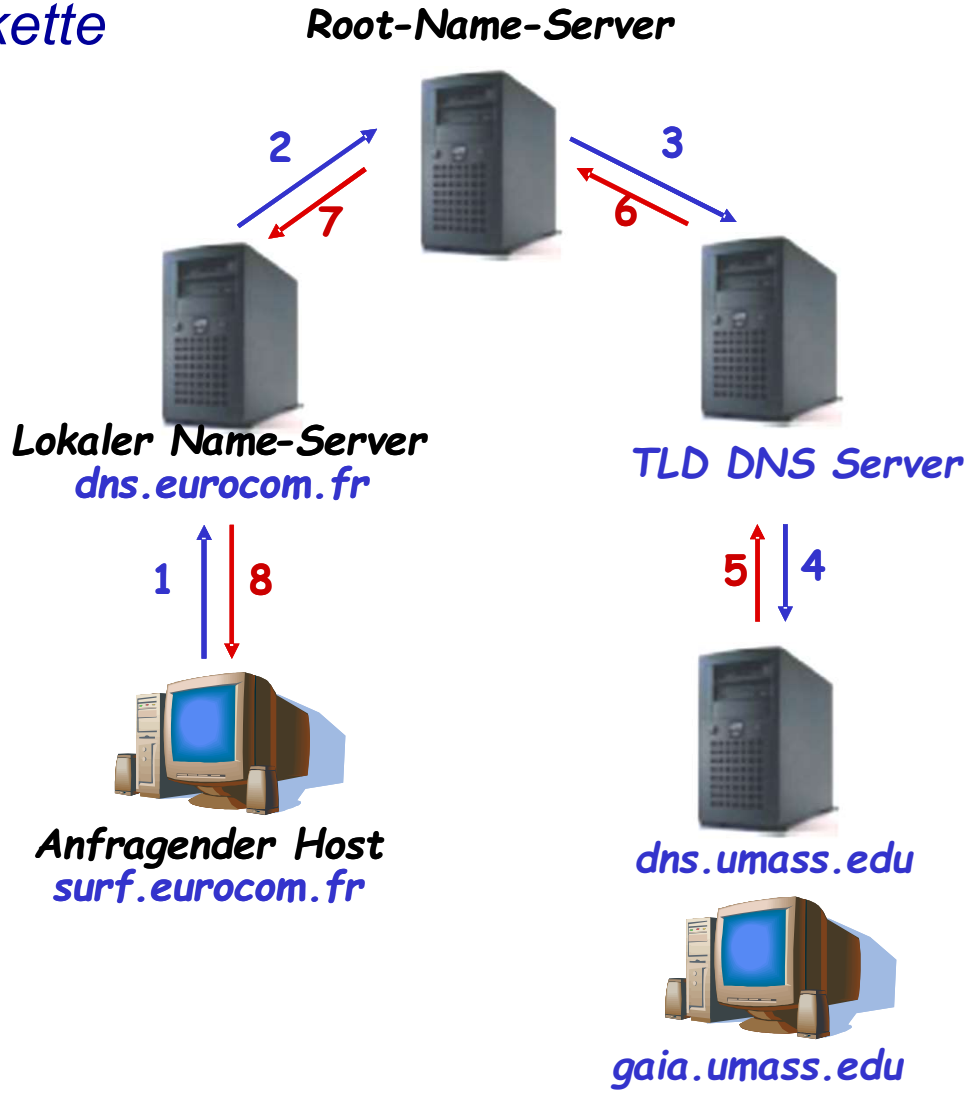
- *DNS-Server-Konfigurationen*
  - *Primary (Master)-Server*
    - *Einzige Instanz, über die Zonen-Info modifiziert werden kann*
    - *Verbindliche Quelle (authoritative source) für alle Infos einer Zone; kann also Anfragen zu Zonen-Info „authoritativ“ beantworten*
  - *Secondary (Slave)-Server*
    - *Erhält Kopie des kompletten Satzes der Zonen-Infos vom Primary Server*
    - *Ist daher, wie Primary Server, eine authoritative Quelle*
    - *Es kann mehrere Secondary Server pro Zone geben*
  - *Caching-only (Forwarder)*
    - *Hält keine eigene Zonen-Info vor*
    - *Kann Anfragen an andere DNS-Server weiterleiten*
    - *Kann erfolgreiche Namensauflösungen zwischenspeichern und später direkt beantworten; er liefert dann „nicht-authoritative“ Antworten*

- *Reverse Adressauflösung*
  - *Ableitung des Rechnernamens aus der IP-Adresse*
    - *Gelöst durch Integration der Domain „in-addr.arpa.“*
  - *Beispiel:*
    „35.1.168.192.in-addr.arpa.“



5.3 Domain Name Service (DNS)

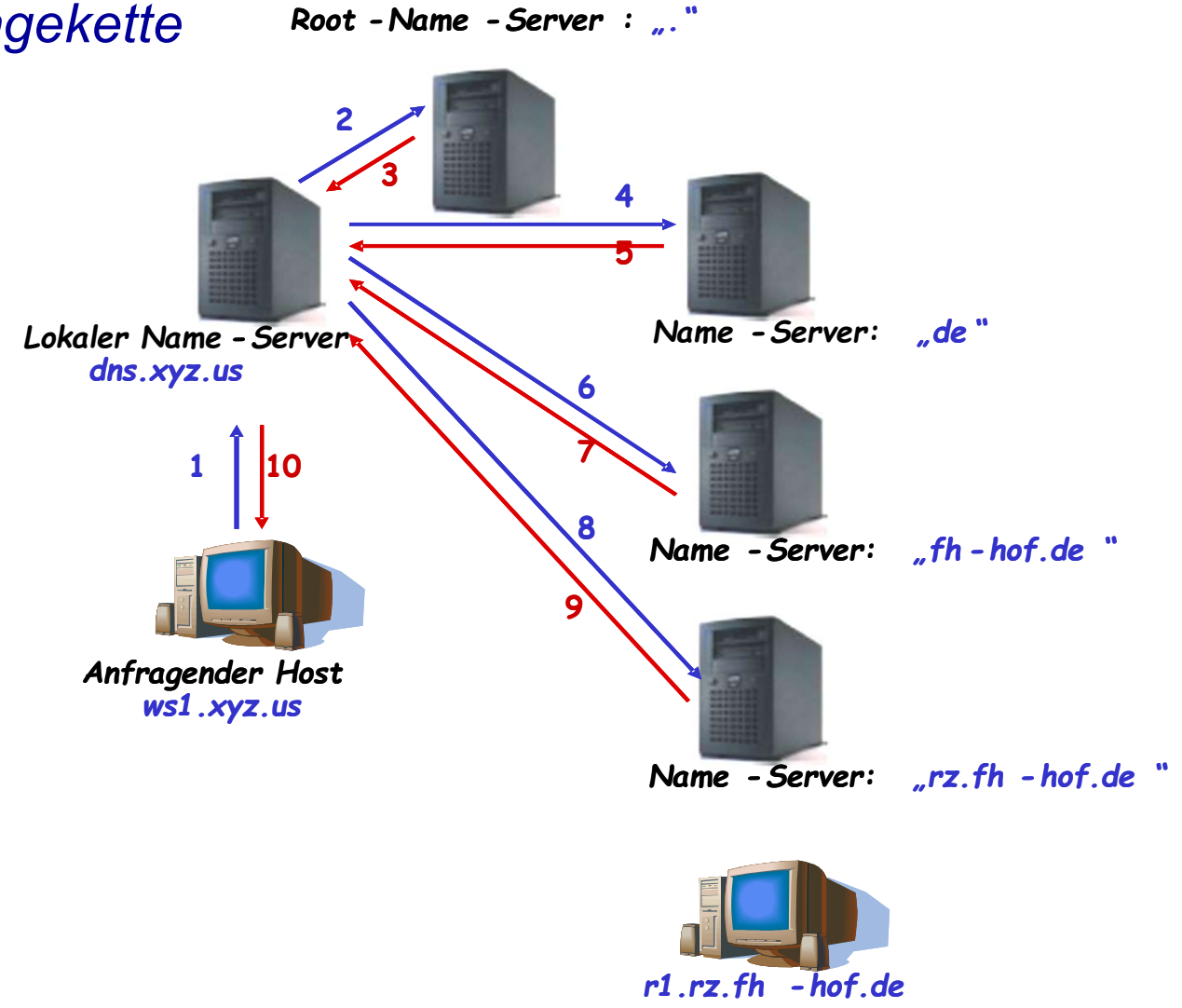
- *Rekursive DNS-Abfragekette*
  - *Beispiel: IP-Adresse zu „gaia.cs.umass.edu“ ermitteln:*



- *DNS-Caching*
  - *Vorteil der rekursiven DNS-Abfragekette*
    - *Alle involvierten DNS-Server erfahren Ergebnis der Namensauflösung*
    - *Kann zur Beschleunigung von DNS-Anfragen verwendet werden*
    - *DNS Caching-Mechanismen:*
      - → *Mitgehörte Information werden in lokalem Speicher gepuffert*
      - → *Wird noch einmal die gleiche Anfrage gestellt, kann diese direkt beantwortet werden*
    - *Problematik:*
      - → *Antworten stammen nicht von autoritativem Nameserver*
      - → *Antworten können eventuell veraltete Daten enthalten*
      - → *Cache-Einträge müssen nach gewisser Zeit gelöscht werden; Einträge werden meist nach einigen Stunden verworfen.*

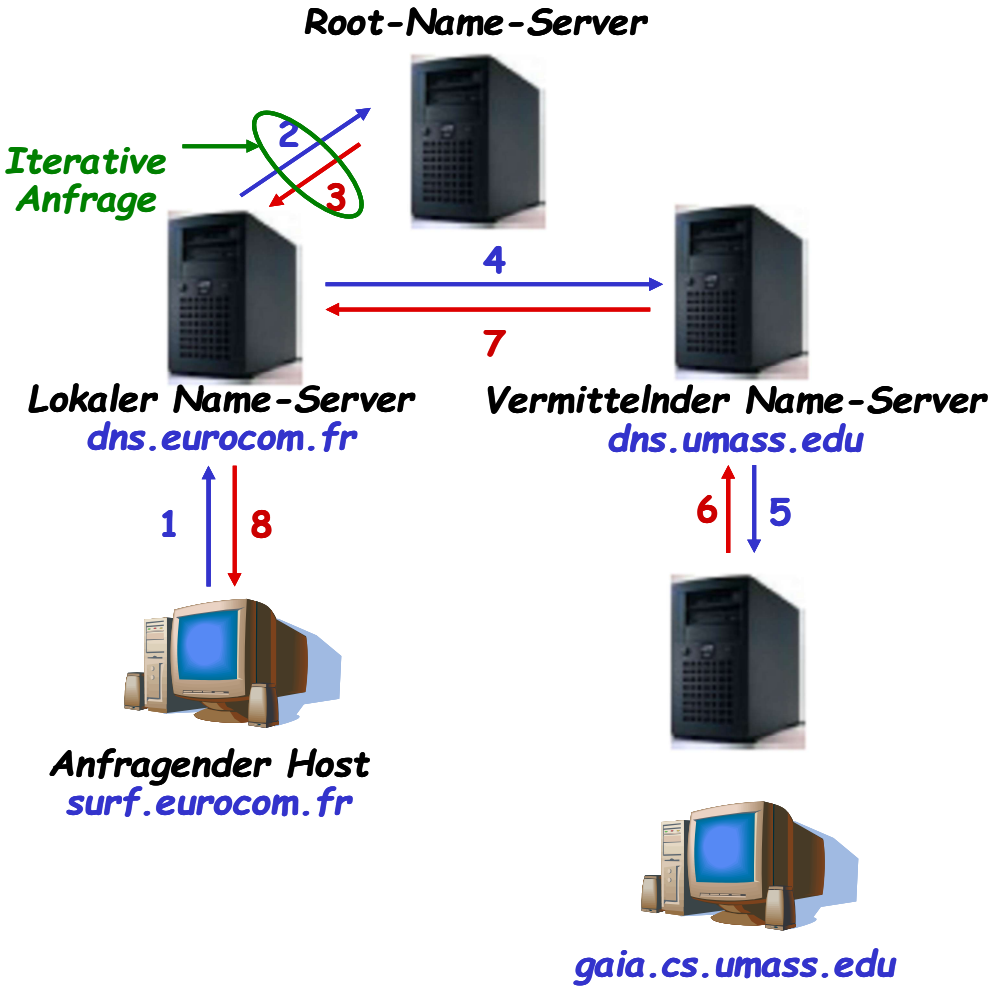
5.3 Domain Name Service (DNS)

- *Iterative DNS-Abfragekette*
  - *Beispiel: IP-Adresse zu „r1.rz.fh-hof.de“ ermitteln:*



5.3 Domain Name Service (DNS)

- *Kombinierte Abfrageketten*



Root-Server werden ***iterativ*** angefragt
→rekursiv würde bedeuten, dass Anfragestatus zwischen gespeichert werden muss!

Die ***rekursive*** Anfragekette ist vorteilhaft für Cache-Management der involvierten, vermittelnden DNS-Server.