

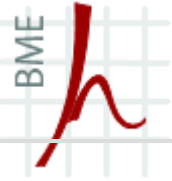
# HÁLÓZATI ALKALMAZÁSOK

Dr. Simon Vilmos  
docens

BME Hálózati Rendszerek és Szolgáltatások Tanszék

[svilmos@hit.bme.hu](mailto:svilmos@hit.bme.hu)

2014.Május 6.

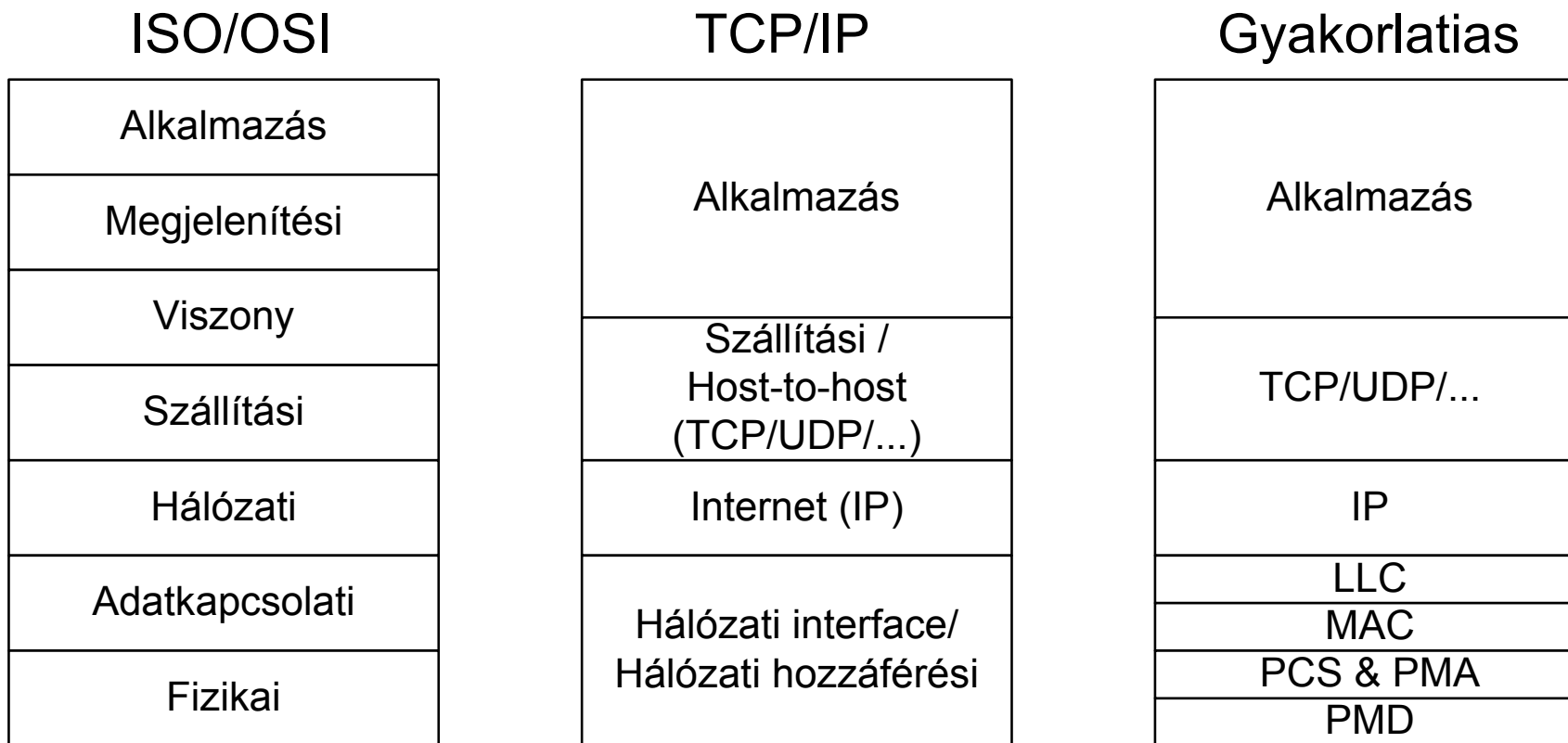


# A tartalomból

---

- Hálózati alkalmazások
- Alkalmazásprotokollok
- Infrastrukturális szolgáltatások
  - Névfeloldási szolgáltatás
  - Címkonfiguráció
- Levelezési rendszerek
- Webes rendszerek

# TCP/IP architektúra és az ISO/OSI rétegmodell



IP: Internet Protocol

TCP: Transmission Control Protocol

UDP: User Datagram Protocol

LLC: Logical Link Control

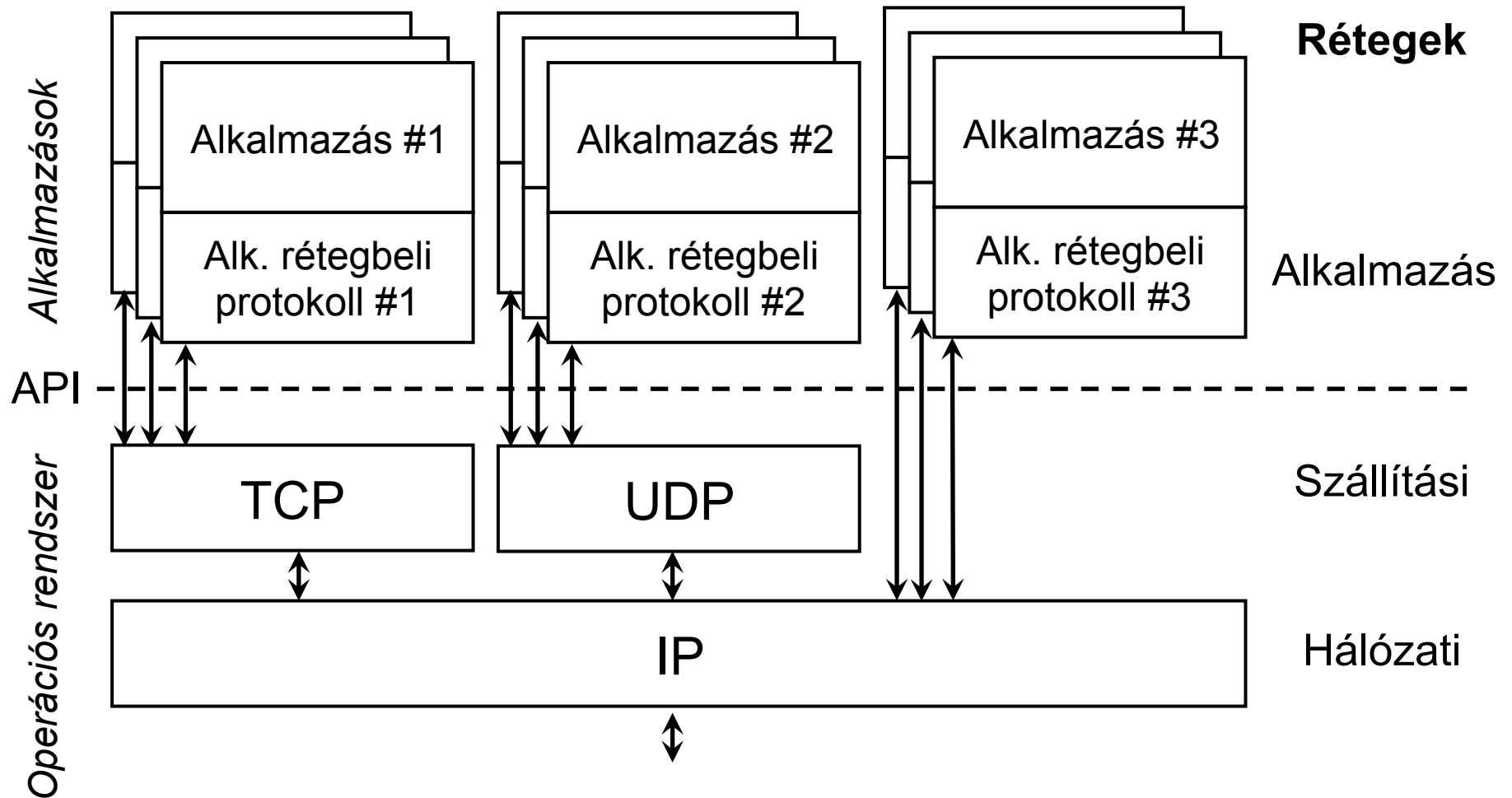
MAC: Medium Access Control

PCS: Physical Coding Sublayer

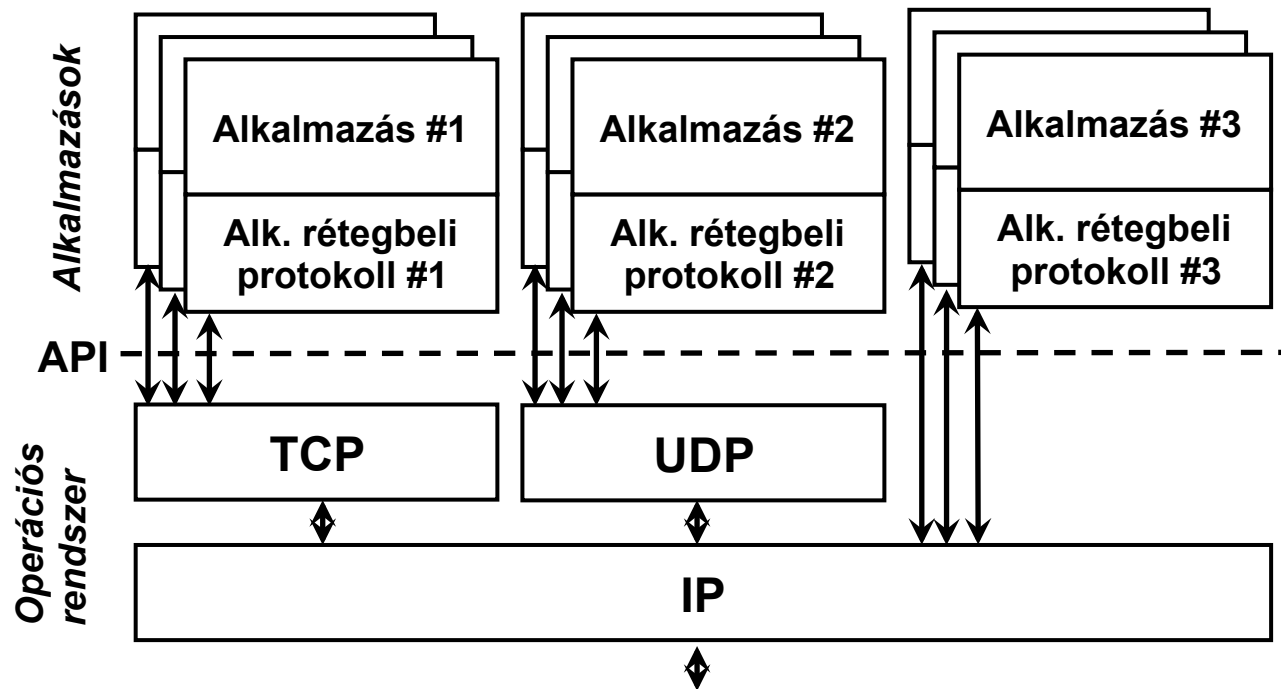
PMA: Physical Medium Attachment

PMD: Physical Medium Dependent

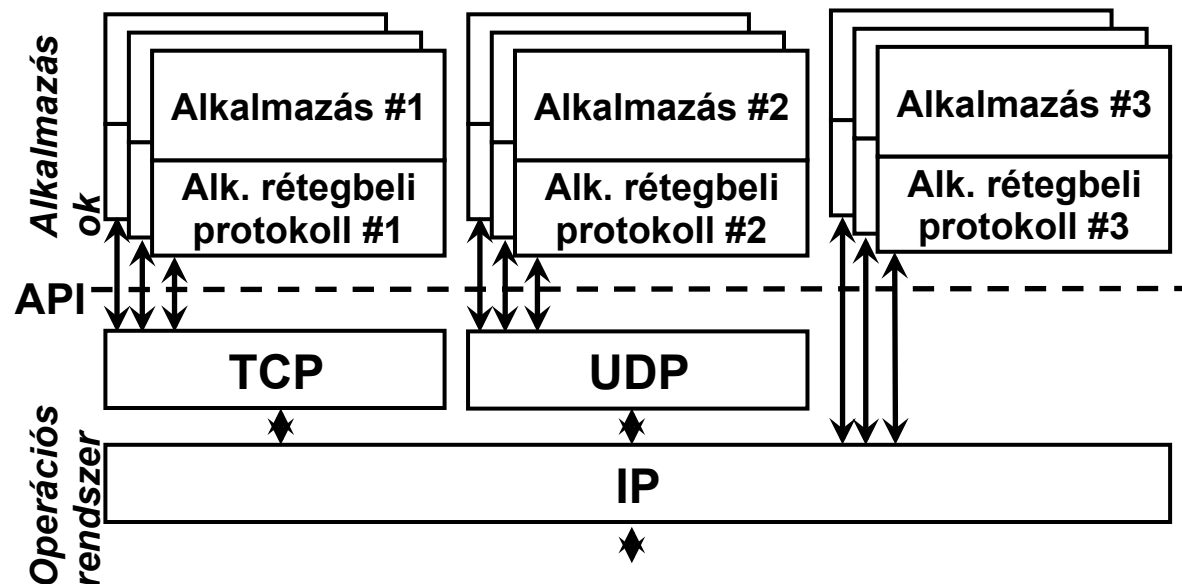
# Alkalmazások kapcsolata az alsóbb rétegekkel



- Legtöbbször az alkalmazásban kerül implementálásra
  - Alkalmazás logikájához szorosan kapcsolódik
  - Egy alkalmazás-rétegbeli protokollt tipikusan kevés alkalmazás használja
- Mégis szükséges szabványosítani (lásd RFC-k)
  - Alkalmazások együttműködése

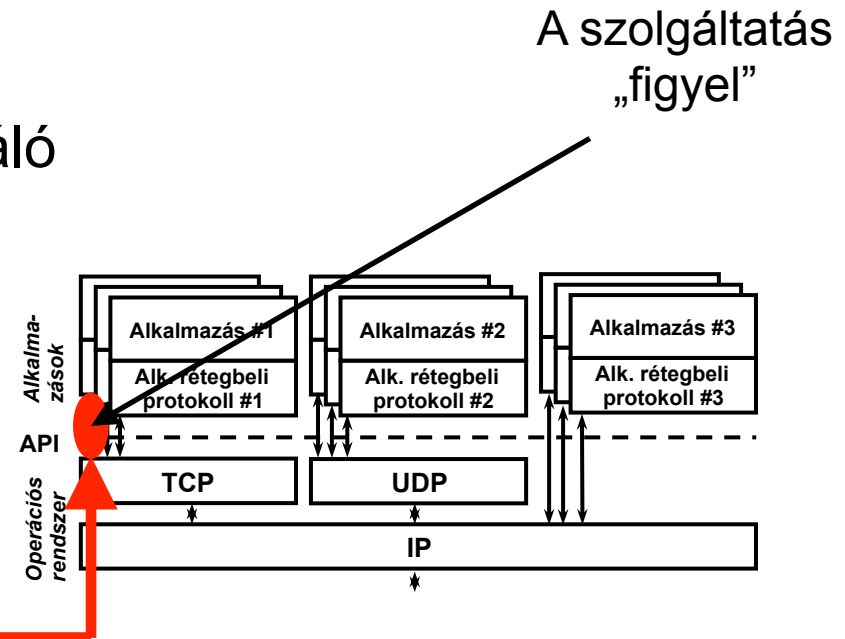
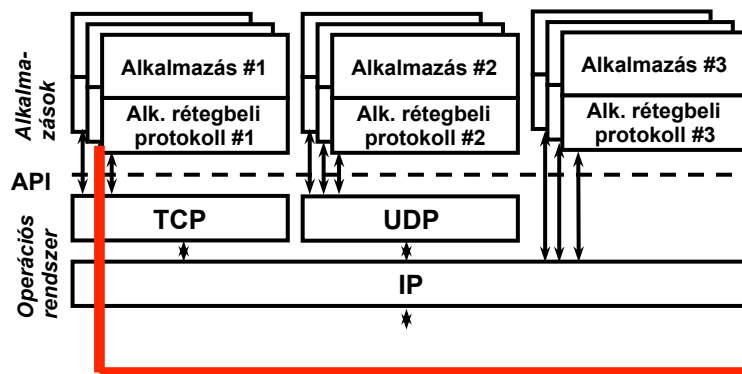


- **Alsóbb rétegeket** – mint szolgáltatásokat – az **operációs rendszer** biztosítja
- **Elfedi** a tényleges rétegeket
- Csak egy interfészt (API: Application Programming Interface) biztosít
  - SAP (Service Access Point)
- Ennek rendszerhívásait használva létrehozható a kívánt kommunikációs csatorna
  - az alkalmazás által használható végződése: **socket**



# Kliens-szerver architektúra

- Kliens
  - Kapcsolatot kezdeményező fél
- Szerver
  - Szolgáltatást nyújtó kiszolgáló



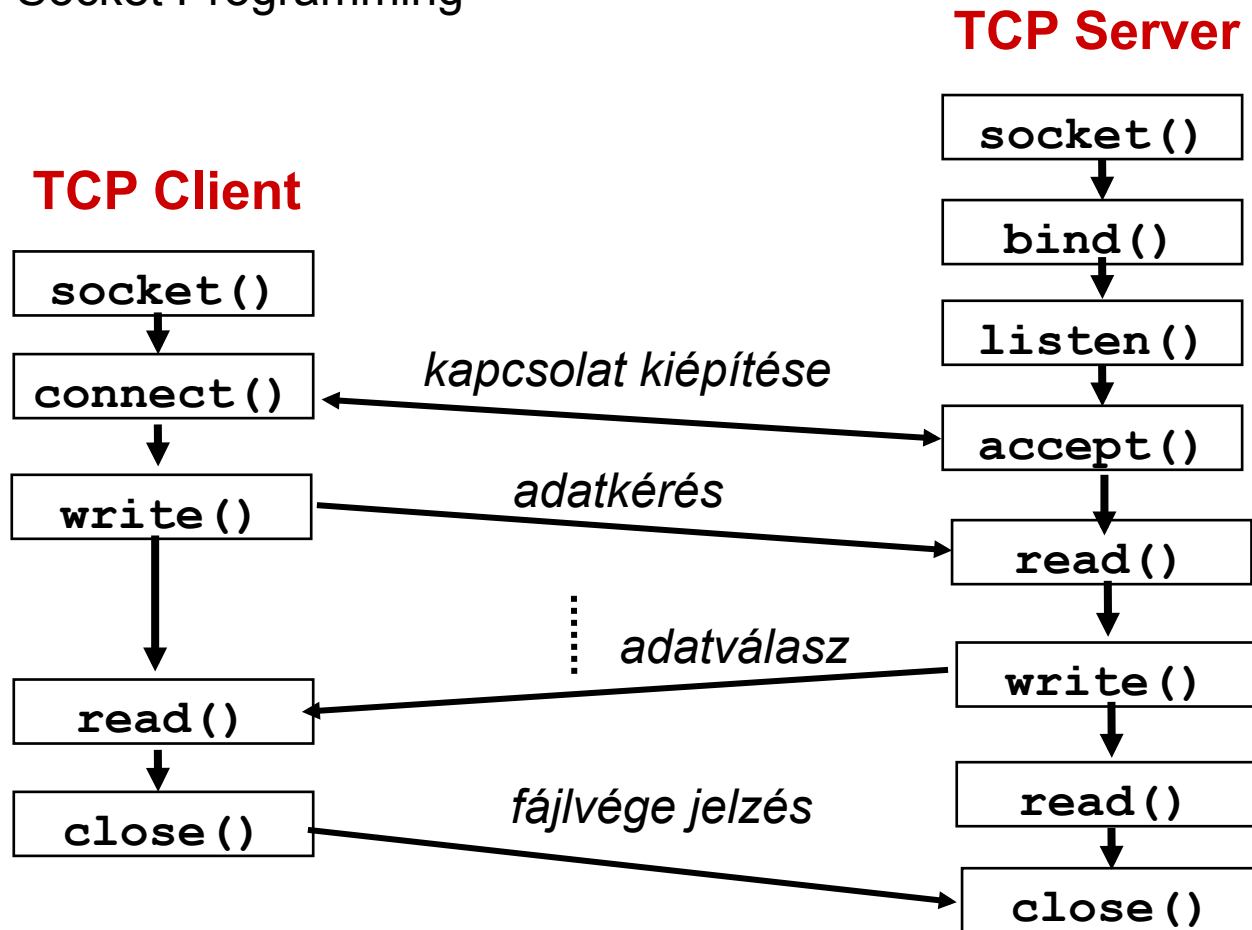
- Kliensnek a szolgáltatást meg kell címeznie
  - IP-cím (vagy DNS név) + szállítási protokoll + portszám

- Szerveren
  - Szolgáltatást azonosítja
    - egy port maximum egy szolgáltatáshoz lehet hozzárendelve
  - **Statikus**
  - 1-65536 tartományból tipikusan 1-1023-ig
    - *well-known ports* (“jól ismert” portok)
    - privilegizált szerver programok kapcsolódhatnak csak
  
- Kliensen
  - Dinamikusan kerül kiosztásra a még nem használtak közül
  - 1-65536 tartományból 1024-65535-ig
    - **Regisztrált** (1024-49151) illetve **dinamikus** portok (49152-65535)



# TCP kommunikáció sockethívásokkal

Socket Programming



UDP: egyszerűbb, kevesebb hívás

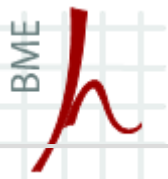
# Milyen alkalmazás milyen szállítási rétegbeli protokollon?

	Natív IP	UDP	TCP
Kapcsolatorientált	x	x	✓
Megbízható	x	x	✓
Üzenetméret (tipikus)	rövid	rövid	hosszú
Adattovábbítás jellege	datagram	datagram	bitstream pipe
Portkezelés	x	✓	✓
Overhead	minimális	kicsi	nagy
Alkalmazások	vezérlési és menedzsment <ul style="list-style-type: none"> <li>•ICMP, IGMP</li> <li>•Routing</li> </ul>	multimédia -átvitel, névfeloldás	fájltávitel, web, levelezés

- Natív IP szolgáltatást igénybevevő protokollok és az IP Protocol mező értéke\*:
  - Vezérlés:
    - 1: ICMP (Internet Control Message Protocol)
    - 2: IGMP (Internet Group Management Protocol)
  - Routing:
    - 8: EGP (Exterior Gateway Protocol)
    - 89: OSPF (Open Shortest Path First)
  - Egyéb:
    - 132: SCTP (Stream Control Transmission Protocol)
  - Valódi szállítási rétegbeli protokollok:
    - 6: TCP (Transmission Control Protocol)
    - 17: UDP (User Datagram Protocol)

\* kezdeti lista az RFC 790-ben

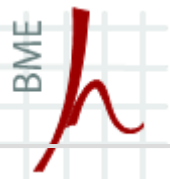
\* a mező értékeit **az IANA** (Internet Assigned Numbers Authority) felügyeli



# Alkalmazások, protokollok és portszámok – UDP felett

- UDP szolgáltatásait igénybevevő protokollok és a tipikus **szerveroldali** UDP portszám értéke\*:
  - 53 DNS (Domain Name System)**
    - névfeloldás
  - 67 BOOTP (Bootstrap Protocol) (Server)**  
**DHCP (Dynamic Host Configuration Protocol)**
    - konfigurálás
  - 68 BOOTP (Bootstrap Protocol) (Client)**  
**DHCP (Dynamic Host Configuration Protocol)**
    - konfigurálás
  - 69 TFTP (Trivial File Transfer Protocol)**
    - fájlátvitel
  - 123 NTP (Network Time Protocol)**
    - időszinkronizáció
  - 161 SNMP (Simple Network Management Protocol)**
    - hálózatmenedzsment
  - 520 RIP (Routing Information Protocol)**
    - routing

\* a portszámokat az IANA (Internet Assigned Numbers Authority) felügyeli



# Alkalmazások, protokollok és portszámok – TCP felett

- TCP szolgáltatásait igénybevevő protokollok és a tipikus szerveroldali TCP portszám értéke\*:
  - 20 és 21 FTP (File Transfer Protocol)**
    - Ugyanahhoz a protokollhoz két port is!
  - 22 SSH (Secure Shell)**
  - 23 Telnet**
  - 25 SMTP (Simple Mail Transfer Protocol)**
  - 53 DNS (Domain Name System)**
    - Ugyanaz a protokoll UDP-n és TCP-n is!
  - 80 HTTP (HyperText Transfer Protocol)**
  - 110 POP3 (Post Office Protocol version 3)**
  - 143 IMAP4 (Internet Message Protocol version 4)**
  - 443 HTTPS (HTTP Secure)**
  - 465 SMTPS (SMTP Secure)**
  - 993 IMAP4S (IMAP4 Secure)**
  - 995 POP3S (POP3 Secure)**

\* a portszámokat az IANA (Internet Assigned Numbers Authority) felügyeli

DNS – Domain Name System

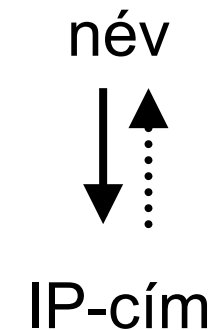
DHCP – Dynamic Host Configuration Protocol

# INFRASTRUKTURÁLIS SZOLGÁLTATÁSOK

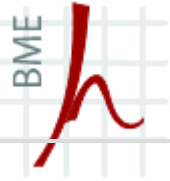
DNS – Domain Name System

# NÉVFELOLDÁSI RENDSZER

- Eltérő reprezentációk közötti megfeleltetés
  - Humán: könnyen megjegyezhető, beszédes nevek
  - Gépi: IP-címek→ névfeloldás
- Követelmények:
  - Jó skálázhatóság
  - Hibatűrés
  - Aktuális (friss) információk



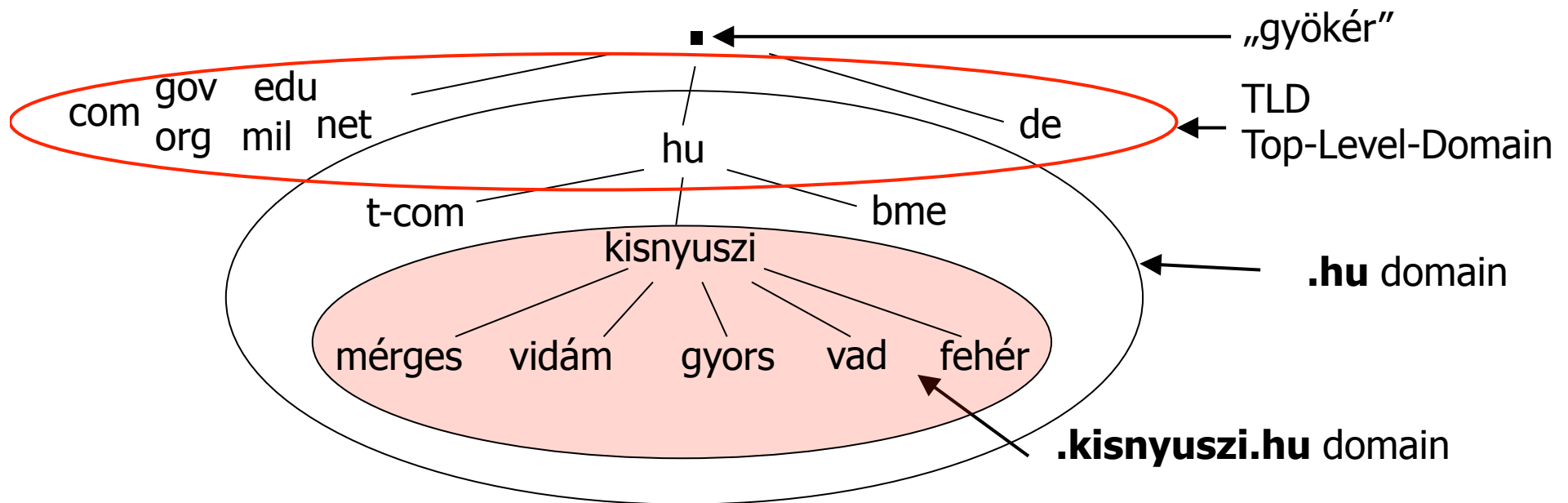




# A DNS (Domain Name System) névtere

- Hierarchikus
- Állomások azonosítása: FQDN (Fully Qualified Domain Name)

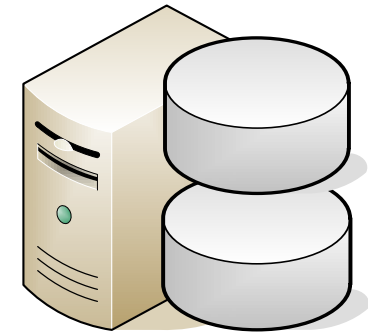
vad.kisnyuszi.hu.  
host neve tartomány neve  
(FQDN suffix)



- Root (gyökér):
  - „.” – ponttal jelölik
  - Elméletileg minden FQDN emiatt ponttal zárul
  - **Gyökérszóna**: legmagasabb szintű DNS zóna
    - Valójában világszerte 13 DNS szerver klaszter szolgáltatja
    - *a.root-servers.net*-től *m.root-servers.net*-ig terjednek
  - **Anycast címzés** előnye!
- Top level domains: mindegyik a root domain-ben van
  - Az IANA adminisztrálja  
(Internet Assigned Numbers Authority)
  - country code top-level domains (ccTLD): pl. .hu (284 db)
  - generic top-level domains (gTLD, 20db):  
pl. .org, .edu, .net, .com, .gov, .mil
  - infrastructure top-level domains: egy van, az .arpa

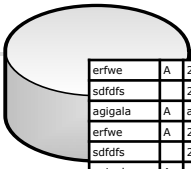
# A DNS zóna és névszerverek

- Zóna
  - Minden tartomány csúcsának vagy egészének adatait tároló adatbázis
  - A DNS nevével azonosítjuk (pl. kisnyuszi.hu.)
- DNS szerver
  - Egy vagy több zónát tárol, szolgál ki
- **Elsődleges/másodlagos DNS szerver**
  - Egy adott zónára vonatkozóan
    - Elsődleges: írható és olvasható
    - Másodlagos: csak olvasható
  - Minden esetben pontosan 1 elsődleges és legalább 1 másodlagos kell
    - → hibatűrés
    - → terheléelosztás és skálázhatóság
  - Szinkronizálás monoton növekvő verziószám alapján
    - Egyre elfogadottabb a verziószám: YYYYMMDDnn



# A zóna elemei: rekordok

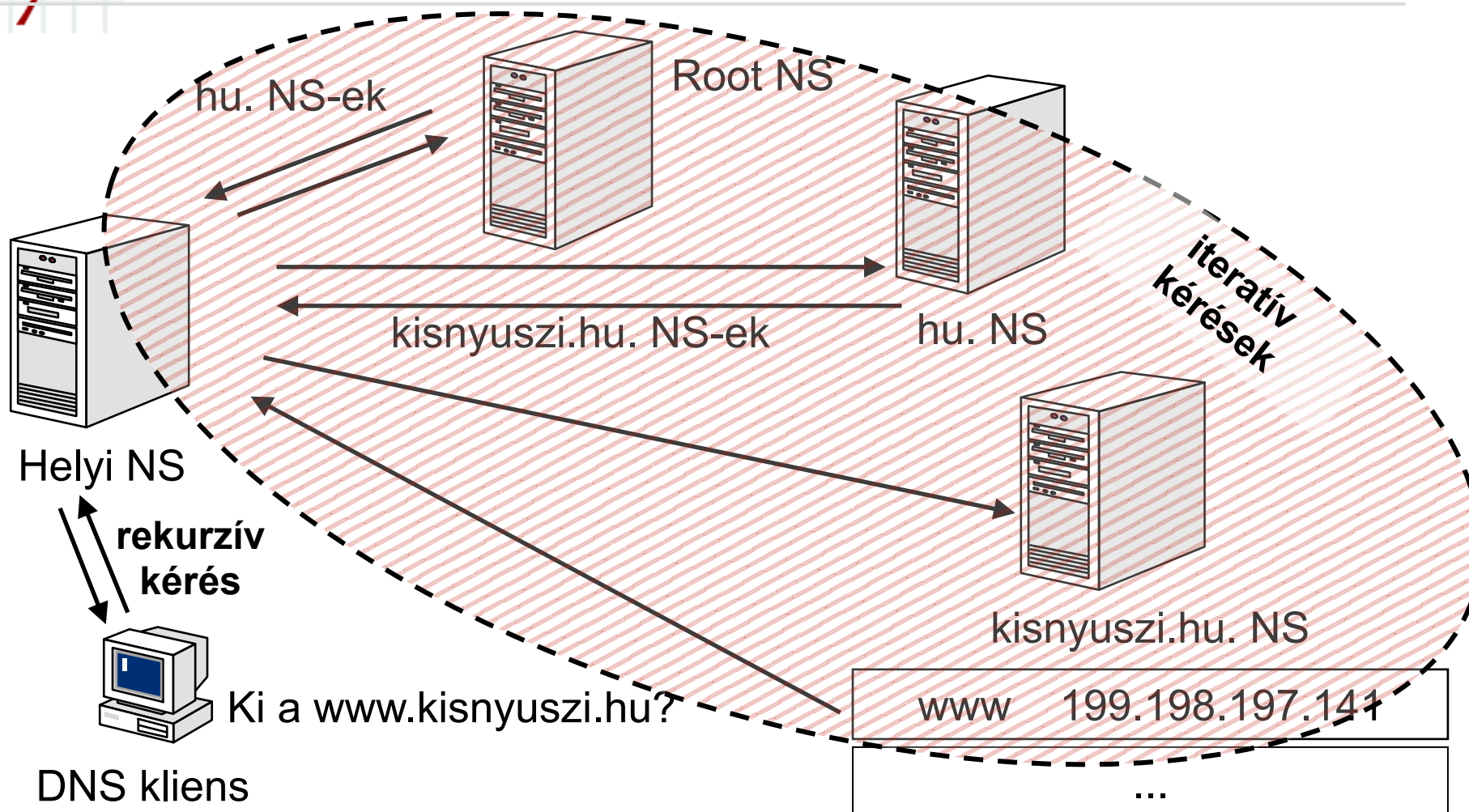
Az erőforrásrekordok (**Resource Records - RR**) gyakori típusai:



erfwe	A	23231
sdfdfs		2314434
agigala	A	ahfhkahrk
erfwe	A	23231
sdfdfs		2314434
agigala	A	ahfhkahrk

- **SOA (Start of Authority)**
    - adminisztratív adatok
      - az elsődleges DNS szerver neve
      - zóna verziószáma (ez alapján a szinkronizálás)
      - kapcsolattartó e-mail címe
  - **A (Address)**
    - név – IP-cím
    - a legtipikusabb felhasználás
  - **CNAME (Canonical Name)**
    - más néven „alias”
    - név – név összerendelés
  - **PTR (Pointer)**
    - IP-cím – név
    - ún. reverse zónában
  - **NS (Name Server)**
    - az adott zónát kiszolgáló DNS szerverek
    - legalább kettő kell
  - **MX (Mail Exchange)**
    - SMTP kiszolgálót azonosít
    - Több is megadható **preferenciával** (prioritással)
  - **SRV (Service Locator)**
    - MX általánosítása
    - tetszőleges szolgáltatásra (pl. SIP)
- Altartományok (subdomain)
- delegálhatóság

# Névfeloldás menete

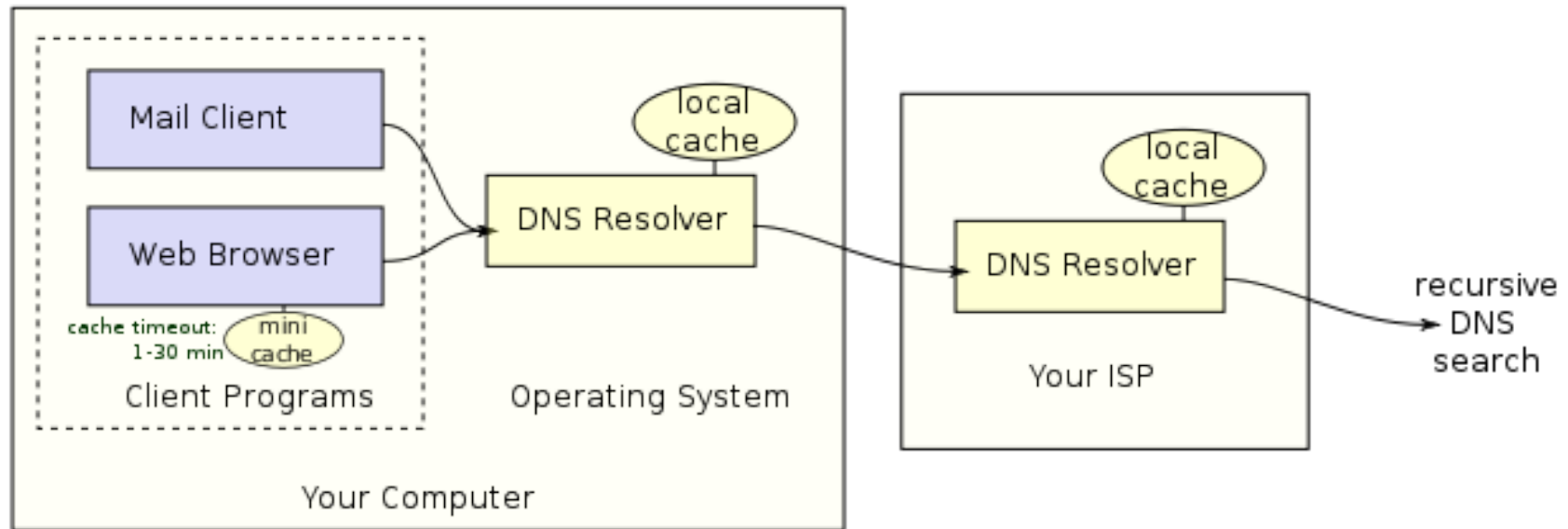


DNS kliens

- **Rekurzív kérés:** a konkrét (vagy a negatív) választ várja, a címzettnek a feladata a névfeloldás
- **Iteratív kérés:** a minél közelebbi felelős megtalálása, arra egy hivatkozást ad vissza: **referral válasz**

# Névfeloldás gyorsítótárral

- A DNS-kéréseket a helyi gépen az operációs rendszer oldja fel egységesen
- **DNS gyorsítótár (cache)** a helyi gépen és a DNS szerveren



- Minden rekordnak **TTL-je (Time To Live)** valódi másodpercben megadva → elévülés
- **Autoritív válasz:** ha a rekordért felelős szerverek valamelyikétől származik a válaszol
- **Nem autoritív:** ha gyorsítótárból származik

# Névfeloldás menete

(0. Böngésző gyorsítótára)

1. Helyi gép gyorsítótára

2. Helyi gépen „hosts” fájl

3. Lekérdezés DNS szerverektől

- Ha van DNS szerver megadva
  - Lekérdezés az elsődleges DNS szervertől, ha elérhető (rekurzív)
    - Az a cache-ből kiszolgál vagy névfeloldást végez (iteratív)
  - Lekérdezés a másodlagos DNS szervertől, ha meg van adva és az elsődleges nem érhető el (rekurzív)
- Ha nincs DNS szerver megadva vagy nem elérhető, akkor lekérdezés valamely root NS-től, majd a hivatkozott NS-ektől (iteratív)

# DNS mint protokoll: kapcsolatok

- A DNS protokoll felhasználási területei:
  - Lekérdezés:
    - DNS kliens  $\leftrightarrow$  DNS szerver
    - UDP 53  $\leftarrow$  rövid, gyors üzenetváltás  
(TCP 53)
  - Zónaletöltés
    - Elsődleges DNS szerver  $\Rightarrow$  másodlagos DNS szerver
    - TCP 53  $\leftarrow$  hosszabb, megbízhatóbb



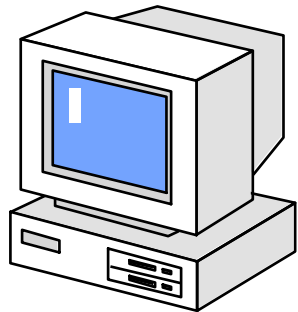
- **Kérés** elemei
  - Kért rekord típusa(i)
  - Feloldandó név vagy IP-cím
  - **Rekurzív kérés esetén** RD (Recursion Desired) bit beállítva
- **Válasz**
  - Pozitív válasz: egy vagy több elemű lista
    - Ebből „véletlenszerűen” (round-robin) választ
  - **Authoritatív válasz esetén** az AA (Authoritative Answer) bit beállítva
  - **Referral** válasz (egy vagy több elemű lista)
    - Kiegészítő hivatkozásokat tartalmaz, mely közelebb visz a feloldáshoz
      - pl.: illetékesebb NS; A rekord helyett azonos nevű CNAME rekord
  - Negatív válasz: nem található bejegyzés, nem oldható fel
    - Nincs válasz  $\neq$  negatív válasz

Dynamic Host Configuration Protocol

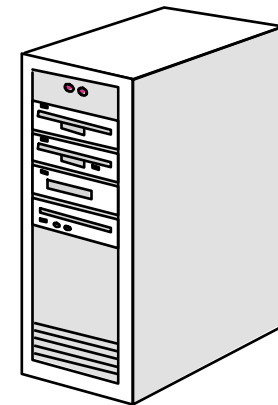
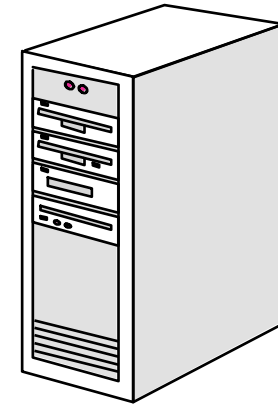
# DHCP

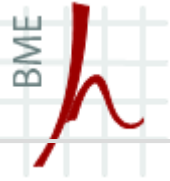
- Mi a DHCP?
  - IP-beállításokat oszthatunk ki vele dinamikusan
  
- Előnyei:
  - Kliensek egyszerű beállítása
  - Módosítások központilag
  - Mobilitás hálózatok között (eltérő beállítások)

Igénylő



DHCP szerver



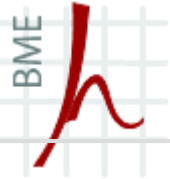


# DISCOVER és OFFER

## Bérlet kérése és felajánlása

---

- Kérés:  
0.0.0.0-tól → 255.255.255.255-nek  
(még nincs IP-címe) (bárkinek)
  
- Ajánlat:  
255.255.255.255-nek ← 192.168.1.1-től  
(bárkinek; nem címezhető) (egy DHCP szerver címe)
  
- Ajánlat tartalma
  - IP-cím
  - Alhálózati maszk
  - Bérleti idő
  - DHCP szerver IP címe
  - **Igénylő MAC címével!**



# REQUEST és ACK

## Bérlet kiválasztása és nyugtázása

---

- Kiválasztás
  - az első ajánlatot (pl. ha több DHCP szerver)
  - Kiválasztási üzenet:
    - 0.0.0.0-tól 255.255.255.255-nek
    - Az üzenet tartalma
      - **kért IP-cím**
      - **DHCP szerver IP-címe**
- Nyugta
  - 192.168.1.1-től, 255.255.255.255-nek
  - Nyugtázó üzenet tartalma
    - **kiosztott IP-cím**
    - **alhálózati maszk**
    - **bérleti idő**



# Bérleti idő (TTL)

- Bérlet időbeli kezelése
  - Félidőben hosszabbítási kérés
  - 7/8 TTL-nél új igénylése
  
- Hosszú vagy rövid TTL?
  - Rövid mellett
    - Ha a kliens szabálytalanul távozik a hálózathoz
      - a bérletét nem adja vissza
    - Ha a kliens szabálytalanul újraindul
      - nem adja vissza a bérletét, és még újat is igényel
    - A beállításváltozások gyorsan életbe lépjenek
  - Hosszú mellett
    - Ne legyen nagy hálózati forgalom



# Ami DHCP-vel beállítható...

## DHCP opciók, paraméterek

---

- **0x01 Subnet Mask (alhálózati maszk)**
- **0x0F Domain Name (FQDN suffix)**
- **0x03 Router (alapértelmezett átjáró(k))**
- **0x06 DNS (DNS szerver(ek))**
- **0x0C Host Name (gép neve is kiosztható)**
- 0x1F Router Discovery
- 0x21 Static Route
- 0x2B Vendor Specific (gyártófüggő beállítások)
- 0x2C WINS
- 0x2E NBT
- 0x2F Node Type
- **0x32 Requested Address (igényelt IP-cím)**
- **0x33 Lease Time (TTL)**
- **0x36 DHCP Server (DHCP szerver IP-címe)**
- **0x37 Parameter Request List (igényelt paraméterek listája)**
- **0x3A Renewal Time (megújítási idő)**
- 0x3B Rebinding Time
- 0x3C Client Class Information
- 0x4D User Class Information
- 0xF9 Static Route CIDR

**Stb...**

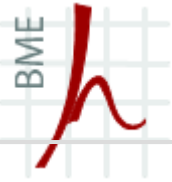


- DHCP hibatűrés
  - Több DHCP használata egy hálózatban, de **diszjunkt IP-címtartományok osztása!**
- DHCP kiterjesztése több hálózati szegmensre
  - A routerek nem engedik át a DHCP üzeneteket
  - A routerekre ún. „**DHCP Relay Agent**”-et telepítve az továbbítja a DHCP forgalmat a DHCP szerverek és kliensek között
- IPv6-ban minden router egyben DHCP szerver is

Telnet

FTP – File Transfer Protocol

# **SZÖVEG- ÉS FÁJLÁTVITEL**



- Egyik legrégebbi alkalmazás
  
- Távoli parancssor
  - Parancsok elküldése
  - Visszajelzések megjelenítése
  
- Még ma is alkalmazzák főként hálózati eszközök egyszerű hálózati adminisztrációjára
  
- Nem biztonságos (jelszavak védelem nélkül)
  - SSH (Secure Shell) helyette

# FTP – File Transfer Protocol

---

- Az egyik legelső fájlátvitelre tervezett protokoll
  - RFC 959
  
- TCP 21-es port
  - Ha a TCP 20-as portot használjuk adatcsatornaként, akkor ez csak vezérlés
  
- Parancsok
  - open – kapcsolat létrehozása
  - ls – aktuális könyvtár listázása
  - put – feltöltés
  - get – letöltés
  - delete – törlés
  - bye – kapcsolat lebontása
  - ...

SMTP – Simple Mail Transfer Protocol

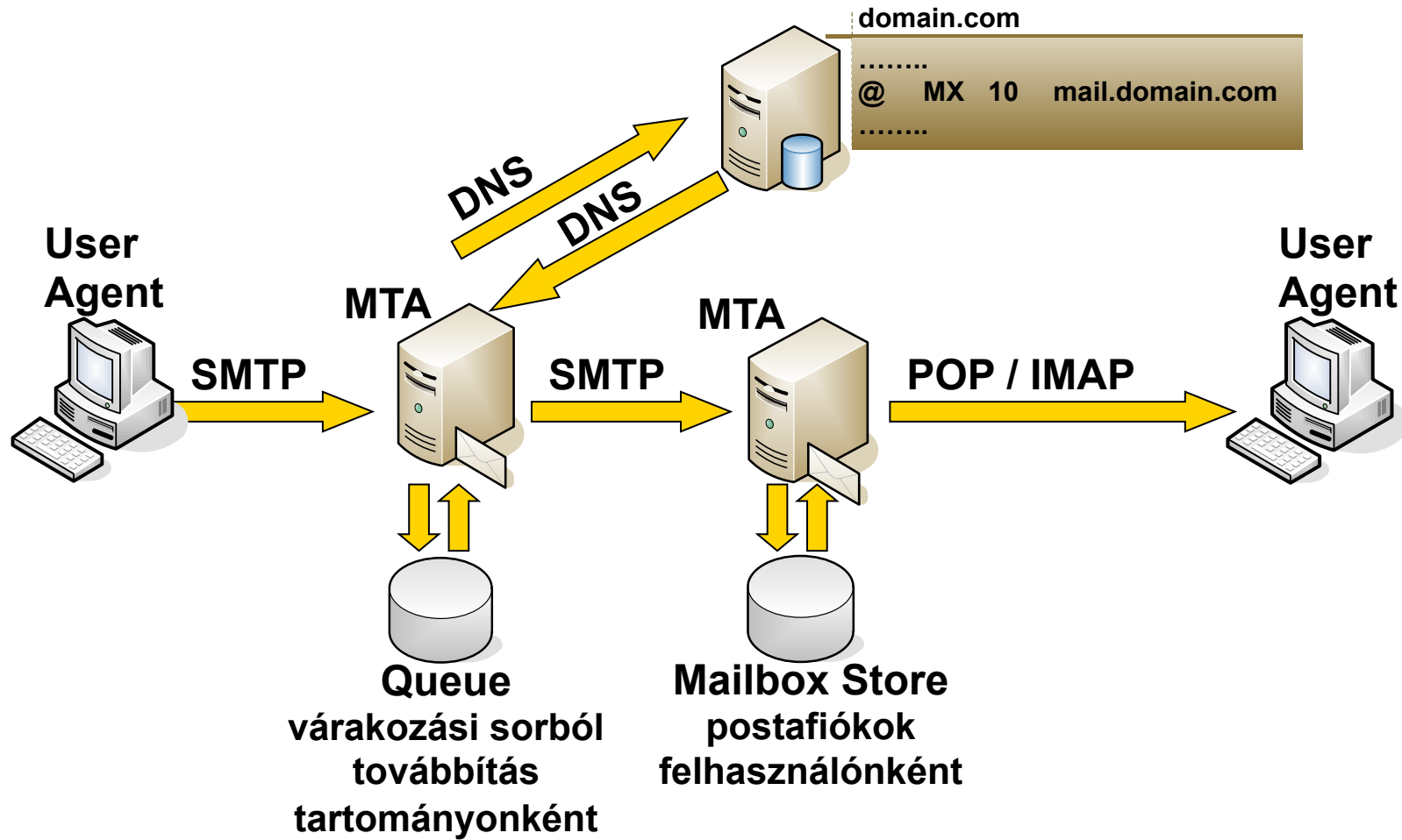
POP3 – Post Office Protocol v3

IMAP4 – Internet Message Protocol v4

## **LEVELEZŐ RENDSZEREK**

- **Komponensek**
  - (Mail) User Agent ((M)UA) (levelező kliens)
  - Mail Transfer Agent (MTA) (SMTP szerver)
  
- **Használt protokollok**
  - SMTP: levél továbbításra
  - POP3: levelek lekérdezése
  - IMAP4: levelek lekérdezése
  
- **Címzett meghatározása**
  - DNS segítségével (MX rekord)

# Levelező rendszerek



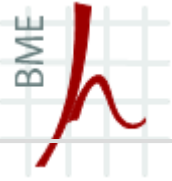
## POP3

- Post Office Protocol version 3
- RFC 1939, 1957, 1725
- Parancsorientált
- TCP 110-es port
- Levelek lekérdezésére
- POP3S
  - POP3 TLS titkosítással
  - TCP 995

## IMAP4

- Internet Message Protocol version 4
- RFC 2060, 1731, 1730
- Parancsorientált
- TCP 143-as port
- Levelek lekérdezésére
- IMAP4S
  - IMAP4 TLS titkosítással
  - TCP 993
- Intelligensebb a POP3-nál:
  - Könyvtárstruktúra támogatása
  - Keresés támogatása
  - Nem törli automatikusan a szerveren tárolt leveleket





# SMTP – Simple Mail Transfer Protocol

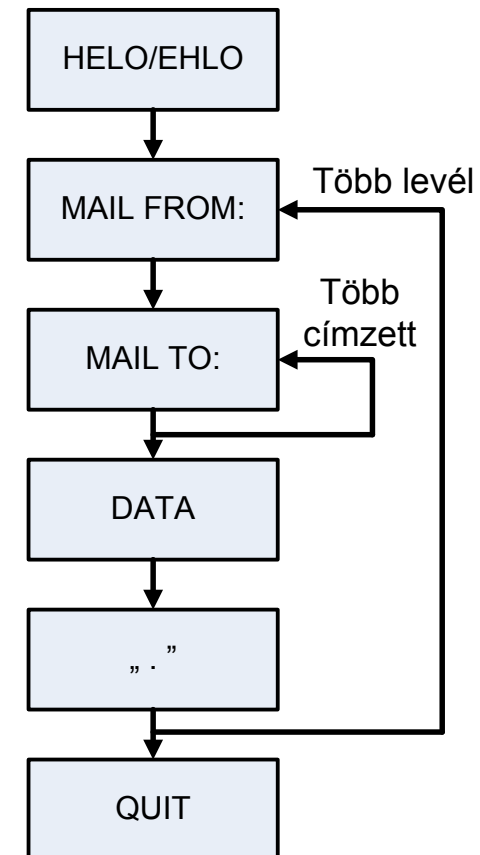
---

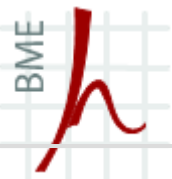
- RFC 821, 822
- Levelek továbbítására (első „killer application”)
- Parancsorientált állapotkódokkal
- TCP 25-ös port
- SMTP relay
  - Nem közvetlen továbbítás
  - Egy (vagy több) SMTP (relay) szerver közbeiktatásával
- SMTPS (SMTP Secure)
  - SMTP TLS csatornában
  - TCP 465

# A leggyakoribb SMTP parancsok

- HELO
  - Üdvözlés
  - ESMTP esetén EHLO
- MAIL FROM:<feladó e-mail címe>
- RCPT TO:<címzett e-mail címe>
- DATA
  - Adat következik
- <CR><LF>.<CR><LF>
  - Adat vége
- QUIT
  - SMTP kapcsolat bontása
- VRFY <e-mail cím>
  - Létezik-e az adott e-mail cím
- HELP
- NOOP
  - Kapcsolat ellenőrzése, fenntartása

Tipikus kapcsolat  
folyamatábrája





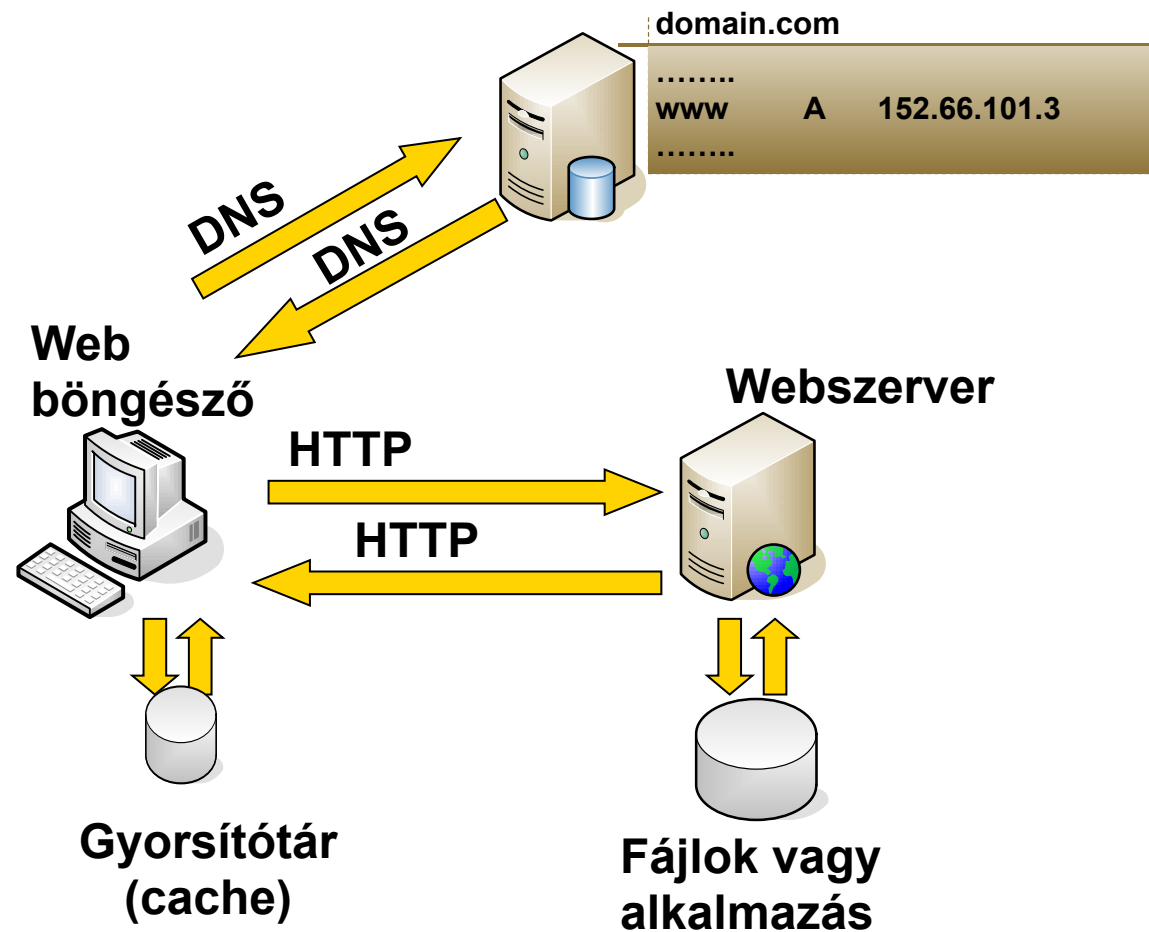
# Példa egy SMTP kommunikációra

```
S: 220 lappfold.fi
C: HELO bme.hu
S: 250 Hello bme.hu, pleased to meet you
C: MAIL FROM: <jogyerek@bme.hu>
S: 250 jogyerek@bme.hu... Sender ok
C: RCPT TO: <mikulas@lappfold.fi>
S: 250 mikulas@lappfold.fi ... Recipient ok
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: Kedves Mikulás!
C: Jó gyerek voltam. Hozzál sok csokit!
C: Köszönöm,
C: Jógyerek Jóska
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 lappfold.fi closing connection
```

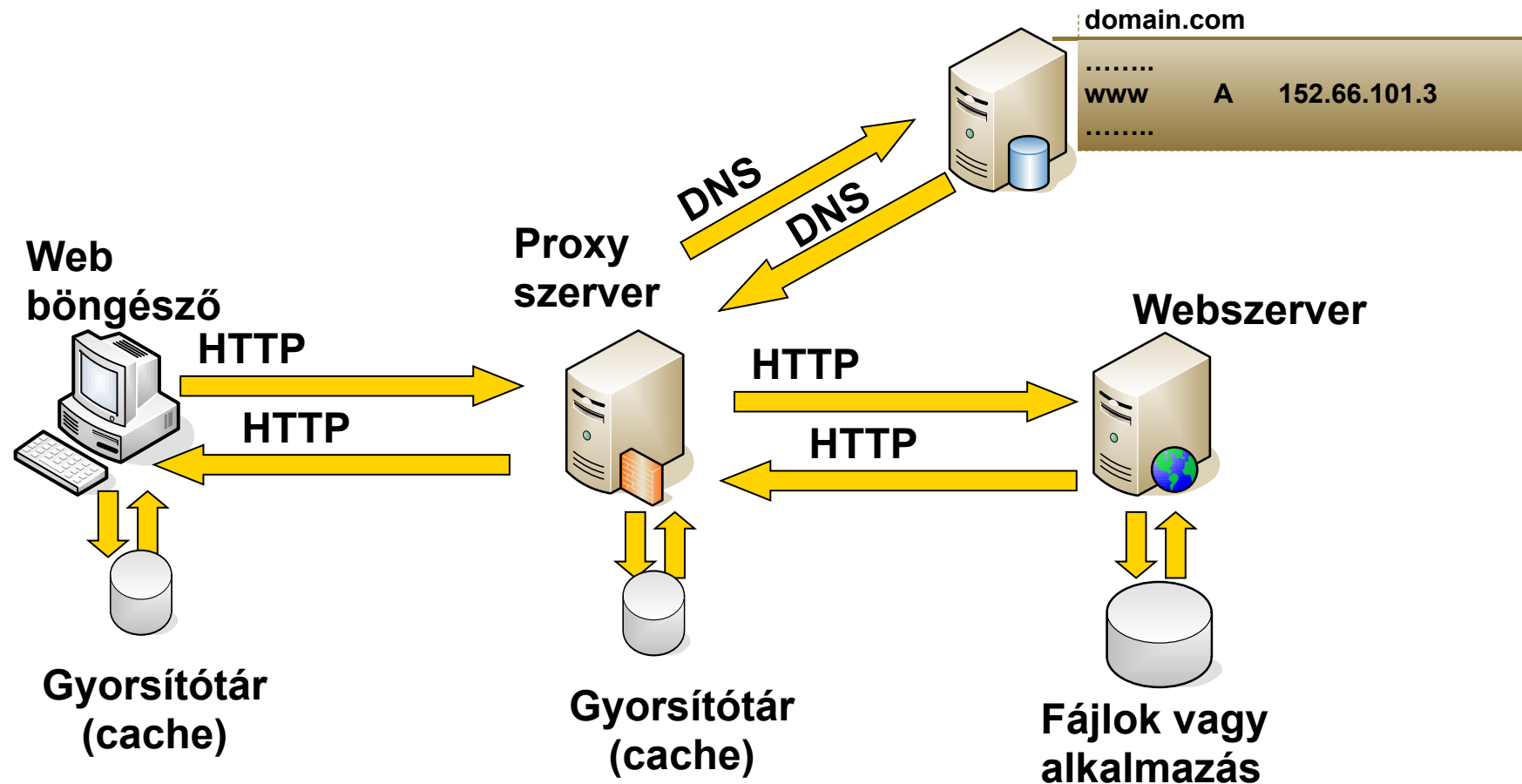
HTTP – HyperText Transfer Protocol

# WEBES RENDSZEREK

# Webes rendszerek



# Webes rendszerek proxyval



# HTTP – HyperText Transfer Protocol

- RFC 2068
- Web
  - Tim Berners-Lee, CERN
  - a második „killer application”
- Parancsorientált állapotkódokkal
- Speciális fejlécek
- Állapotmentes protokoll
- TCP 80
- Proxy
  - Kliens nevében jár el
  - Főként a hatékony gyorsítótárzás miatt





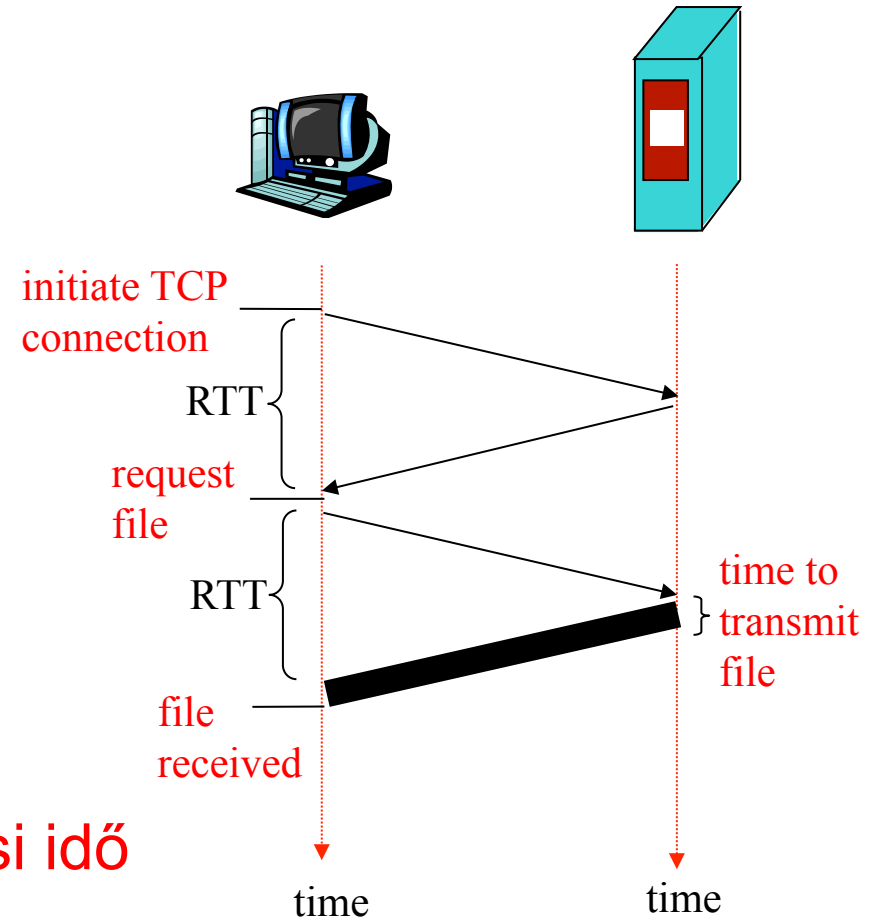
# Perzisztens HTTP

---

- *Nem-perzisztens:*
  - minden kérés/válasz pár egy **külön** TCP kapcsolaton keresztül
  - Legfeljebb egy objektumot visz át egy TCP kapcsolaton
  - HTTP/1.0
- *Perzisztens:*
  - **egy TCP kapcsolaton belül** több kérés/válasz pár
  - Alapbeállítás HTTP1.1-nél



# Késleltetés nem perzisztens kapcsolatnál



Teljes késleltetés =  $2RTT + \text{adási idő}$

## Nonperzisztens HTTP:

- 2 RTT/objektum
- + TCP kapcsolat overhead
- Böngészők használhatnak **párhuzamos TCP kapcsolatokat** a különböző objektumok letöltésére

## Perzisztens HTTP

- Szerver **nyitva hagyja a kapcsolatot** a válasz elküldése után
- HTTP üzenetek sorozata ugyanazon kliens és szerver között **egy kapcsolaton belül**

## Perzisztens pipelining nélkül:

- Kliens csak akkor küld új kérést, ha előzőre **megjött a válasz**
- 1 RTT/objektum

## Perzisztens + pipelining:

- Alapbeállítás HTTP/1.1-nél
- Kliens elküldhet több kérést is egymás után **anélkül, hogy megvárná a választ**
- 1 RTT minden objektumra

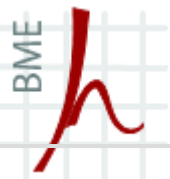
# Példa egy HTTP kérésre

Kérés (request line)  
(GET, POST,  
HEAD parancs)

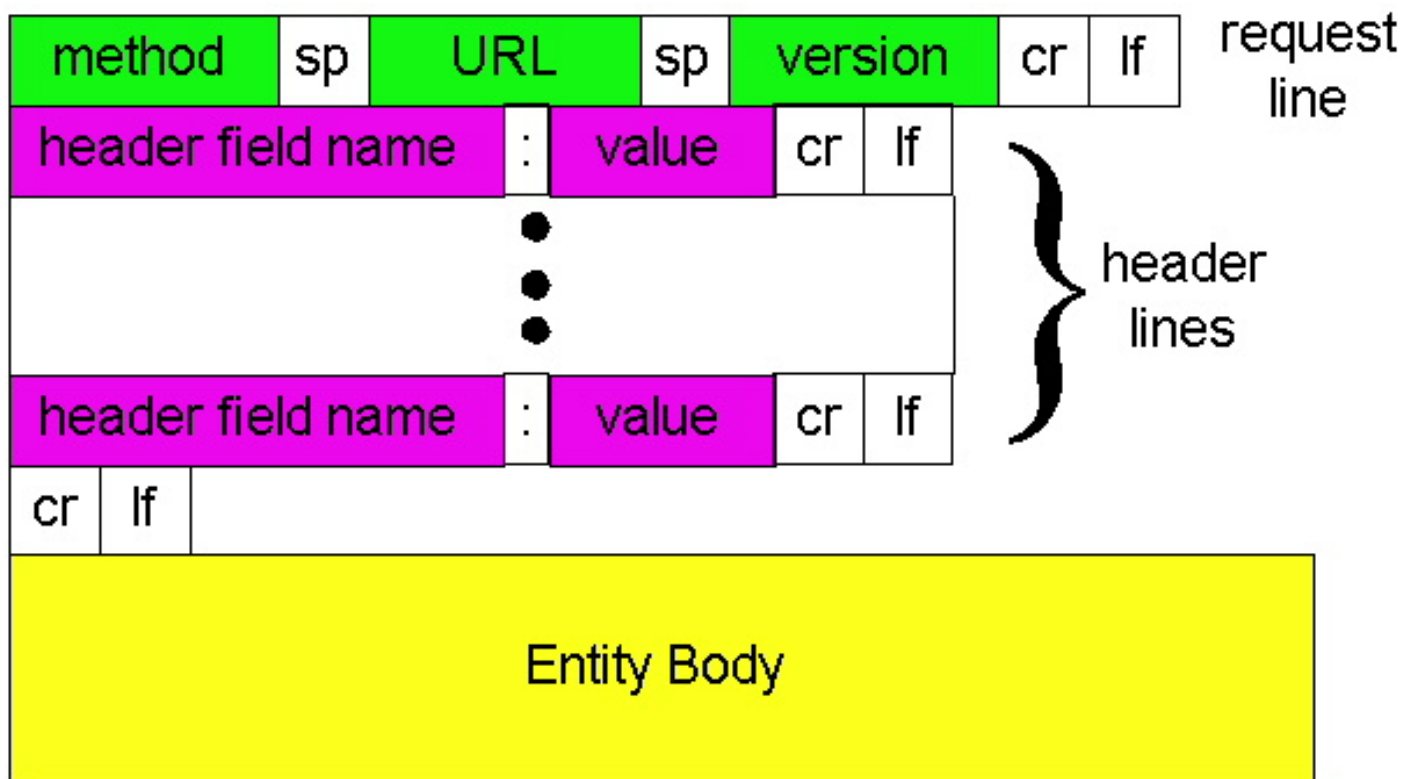
fejlécek  
(header lines)

üzenet végét  
jelző soremelés

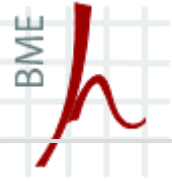
```
GET /somedir/page.html HTTP/1.1
Host: www.someschool.edu
User-agent: Mozilla/4.0
Connection: close
Accept-language: hu
<CR><LF>
```



# Általános HTTP kérés üzenet formátum



**Entity Body: POST-nál van értelme!**



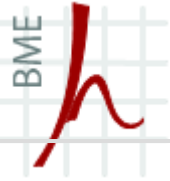
# Gyakori HTTP parancsok

---

- GET <URL> HTTP/1.1
  - adott URL tartalmának lekérése
- HEAD
  - mint a GET, de csak a metaadatokat adja vissza
- POST
  - a kliens ezzel tud adatokat küldeni a szervernek
- PUT
  - a POST-hoz hasonló, fájlfeltöltésre alkalmas
- DELETE
  - adott URL tartalmának törlése

# Gyakori HTTP fejlécek

- Accept: elfogadható MIME típus
- Accept-Charset: elfogadható karakterkészlet
- Allow: szerver által támogatott parancsok
- Authorization: támogatott hitelesítési módok
- Content-Encoding: tömörítés típusa
- Content-Length: tartalom mérete
- Content-Type: MIME típus
- Date: lekérés dátuma és ideje
- From: a látogató e-mail címe (nem autentikációra)
- Pragma: nem meghatározott paraméter (pl. „no-cache”)
- Referer: a hivatkozó oldal URL-je
- Retry-After: 503 utáni újrapróbálkozási idő
- Server: szerver neve és verziója
- User-Agent: böngésző neve és verziója
- WWW-Authenticate: Hitelesítési információk (credentials)



# Példa egy HTTP válaszüzenetre

állapotkód

HTTP/1.1 200 OK

fejlécek

Connection close

Date: Thu, 06 May 2014 12:00:15 GMT

Server: Apache/1.3.0 (Unix)

Last-Modified: Mon, 22 Jun 2013 .....

Content-Length: 6821

Content-Type: text/html

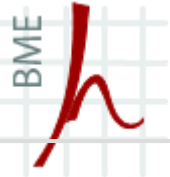
adat  
(pl. a kért  
HTML fájl)

data data data data data ...

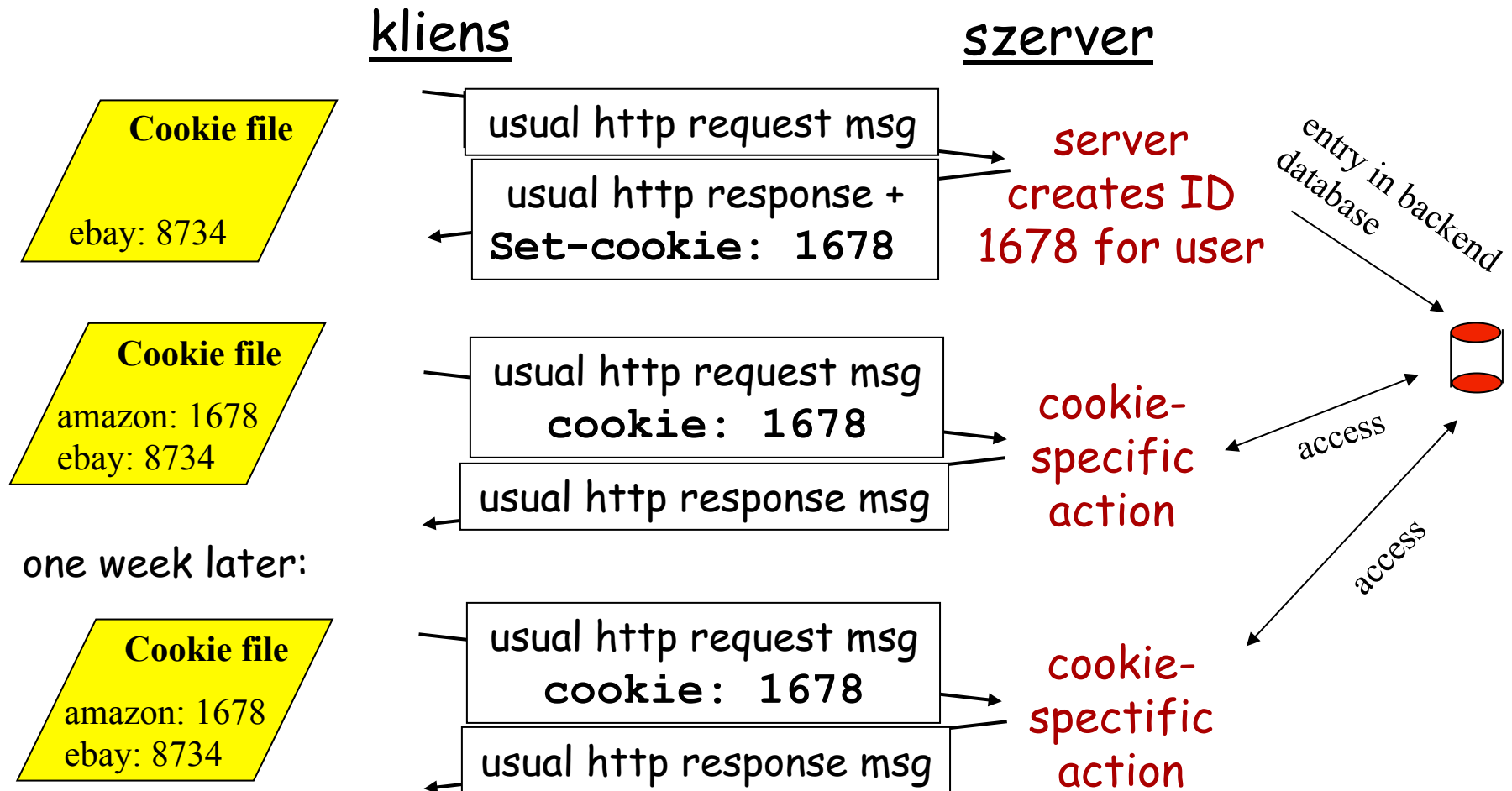
# Gyakori HTTP állapotkódok

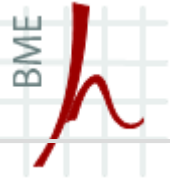
Kód	Jelentés	Leírás
200	OK	
201	Created	POST sikeres
202	Accepted	Kérés elfogadva
204	No content	Nincs semmi a kliensnek
400	Bad request	Hibás kérés
401	Unauthorized	Hitelesítés szükséges
403	Forbidden	Hozzáférés megtagadva
404	Not found	Nem található
500	Internal Server Error	Belső szerver hiba
503	Service Unavailable	Pillanatnyilag nem szolgálható ki





# Cookie-k használata





# HTTP jövője

---

- Következő verzió: HTTP 2.0, IETF fejlesztésben
- A Google SPDY protokolljának lényeges elemeit használja
  - De facto szabvány?
- Csökkenti a web oldalak betöltési idejét:
  - multiplexált stream-ek egy TCP kapcsolaton belül
  - kérések prioritizálása
  - kérés és válasz fejrészek tömörítése
    - a statikusok nincsenek újraküldve
  - szerver push és hint funkció

- Alkalmazások hálózati kapcsolata
  - Portok és socketek használata
- Infrastrukturális szolgáltatások
  - DNS – Névfeloldási szolgáltatás
  - DHCP
- Szöveg- és fájlátvitel
  - Telnet, FTP
- Levelezési rendszerek
  - SMTP, POP3, IMAP4
- Webes rendszerek
  - HTTP

# Kérdések?

# **KÖSZÖNÖM A FIGYELMET!**



Dr. Simon Vilmos  
BME Hálózati Rendszerek és Szolgáltatások Tanszék  
svilmos@hit.bme.hu