Hálózattervezés alapjai
Címek, címkiosztás, routing
(IPv4, IPv6)

2007/2008. tanév, II. félév

Dr. Kovács Szilveszter

E-mail: szkovacs@iit.uni-miskolc.hu

Informatikai Intézet 106. sz. szoba

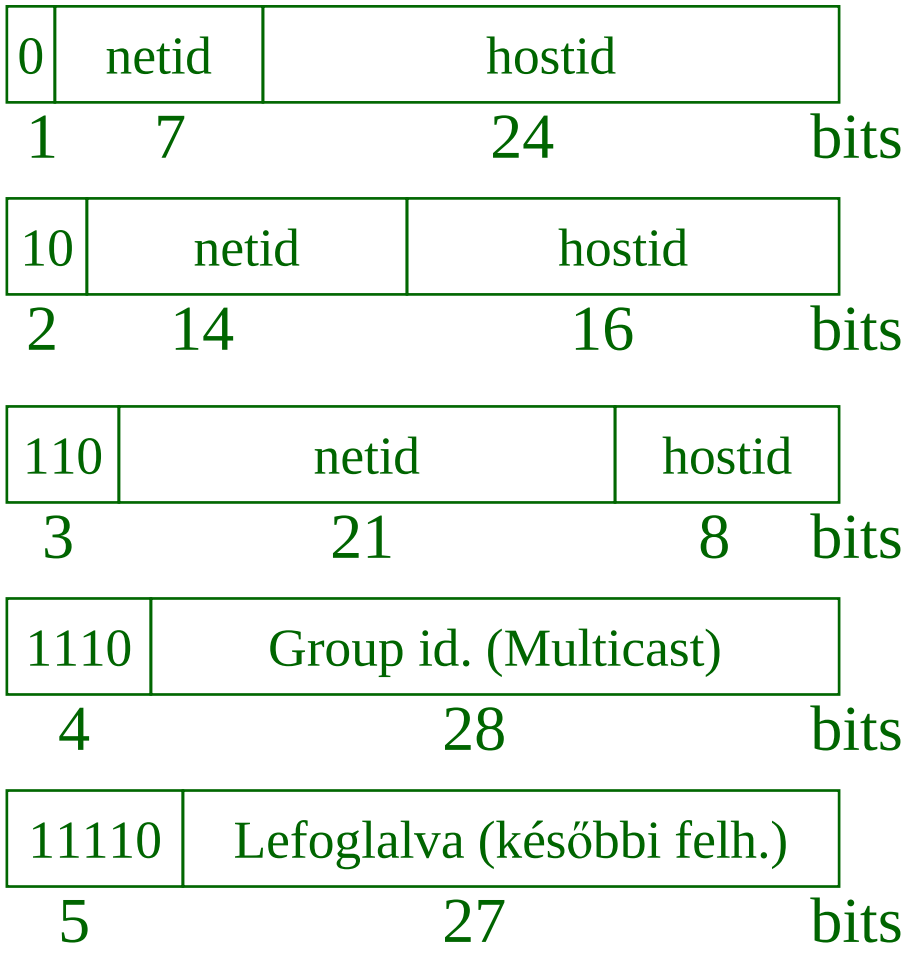
Tel: (46) 565-111 / 21-06

Internet címek

- **32 bit, 4 byte**
- **Pontok közötti decimális alak**
  (Dotted decimal notation - egészen jól olvasható)
  164.107.134.5
  10100100.01101011.10000110.00000101 (bin)
  A4:6B:86:05 (hexa)
- **Max címszám: 2^{32} = 4 milliárd csomópont**
- **Class A Networks = 15 millió csomópont**
- **Class B Networks = 64K csomópont**
- **Class C Networks = 250 csomópont.**

IP címosztályok

- **Hierarchia: hálózat cím + hoszt cím (netid+hostid)**
- **Class A**
  0.0.0.0 - 127.255.255.255
- **Class B**
  128.0.0.0 - 191.255.255.255
- **Class C**
  192.0.0.0 - 223.255.255.255
- **Class D** (többes címzés)
  224.0.0.0 - 239.255.255.255
- **Class E**
  240.0.0.0 - 247.255.255.255



multicast: többes címzés ⇒
az üzenet a multicast csoport minden tagjának szól (broadast → mindenkinek szól)

IP címtér

Osztály	Oszt.+hálózati bitek száma	Hálózatok száma	Gép bitek száma	Gépek száma	Címmező foglalás
A	1 + 7	$2^7 - 2 = 126$	24	$2^{24} - 2 = 16777214$	49,21%
B	2 + 14	$2^{14} = 16384$	16	$2^{16} - 2 = 65534$	24,99%
C	3 + 21	$2^{21} = 2097152$	8	$2^8 - 2 = 254$	12,40%
D Multicast	4 + 28	$2^{28} = 268435456$	-	-	6,25%
E Fenntartva	4	-	32 - 4	$2^{28} - 1 = 268435455$	6,25%

Speciális címek és jelentésük

- **Nem minden cím osztható ki állomáscímnek**

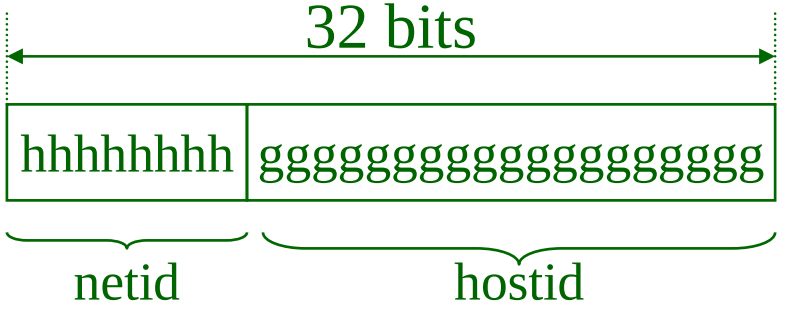
Hálózat bitek	Gép bitek	Jelentés
..0..	..0..	Ideiglenes forrás cím, amíg nem tanulja meg a gép a címét. Nem szabad célcímként használni. Default route: 0.0.0.0
..1..	..1..	Broadcast, mindenki ezen a helyi fizikai hálózaton. MAC broadcast keretben kell küldeni.
x	..0..	Ez a logikai hálózat. Korábban logikai broadcast.
x	..1..	Directed broadcast, mindenki ezen a távoli hálózaton. Távolról MAC unicast keretben kell
127.0.0	x	Loopback, a helyi TCP/IP stack pszeudó címe a hoszton belül. A hálózaton nem fordulhat elő.
224.0.0.2	-	Multicast, az összes router ezen a hálózaton van.

Klasszikus címzés összefoglaló

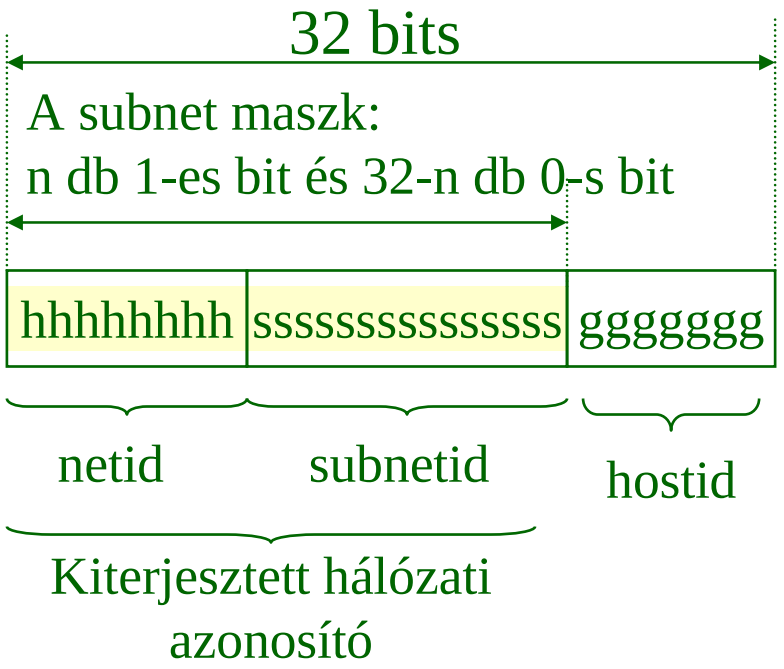
- A cím egyértelműen két részre bontható
  - az első bitek megmondják, hol a határ
  - ugyanakkor merev bit-határok
  - broadcast cím egyértelműen számítható (a host id. csupa 1-es)
- Igény a címzési hierarchia bővítésére
  - Intézményi hálózatok fejlődése
  - a pazarló A és B osztályok elfogytak
  - pont-pont kapcsolatokra teljes C osztály

Alhálózat (subnet) bevezetése

- **Az eredeti felosztás**



- **A subnet maszkkal az**
  - **értékes biteket kijelöljük**



Alhálózat címzések

- A (sub)net maszk (RFC 950)
- A kiterjesztett hálózati azonosító hosszabb lehet, mint a címosztály hálózati azonosítója!
- C osztályú címnél a default maszk: 255.255.255.0
- A prefix jelölés:
  - 193.6.5.0/24
    - 193.6.5.0
    - 255.255.255.0

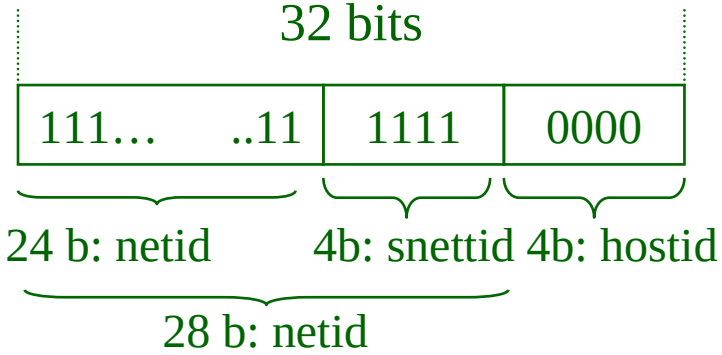
Osztály	Prefix	Netmask
A	/8	255.0.0.0
B	/16	255.255.0.0
C	/24	255.255.255.0

A subnetting eredménye

- **A címező jobb kihasználása**
  - **pont-pont kapcsolatok 2 biten elférnek**
  - **több LAN befér egy IP hálózatba**
- **A cím nem tartalmazza a hálózat-azonosítót**
  - **A maszkot is jól kell konfigurálni**
    - **a broadcast cím nem található ki az IP címből**
  - **A maszkot is kell továbbítani (plussz 4 byte az útvonalválasztási információkban)**
  - **De az útvonalválasztás egyszerűsödhet** (pl. hálózatok összefogása „szupernetting”)

A címfeldolgozás

- **Pl 193.6.5.1 IP címből a**
  - **/24 (255.255.255.0) maszk és az and logikai művelet leválasztja a hálózati címet**
  - **a /24 maszk negáltjának és az and logikai művelet leválasztja a gép címet**
- **Ha a szubnet maszk hosszabb**
  - **Pl. /28: 255.255.255.240, akkor**
    - **28 bits: net**
    - **4 bits: host**



- **Ha rövidebb, mint a címosztályé: „supernetting”**
  - **több hagyományos osztály összefogása**
  - **Pl. 16 C összefogása: /20: 255.255.240.0**

Alhálózat címkiosztási példa

- **Adott 193.6.5.0/24; és bontsuk öt egyforma méretű alhálózatra!**
  - **$2^2 < 5 < 2^3 \rightarrow$ 3 subnet bit kell $\rightarrow$ /27 a prefixes (255.255.255.224) jelölés $\rightarrow$ valójában 8 alhálózatra osztunk**

Bitminta	Címtartomány	Megjegyzés
11000001|00000110|00000101|000xxxxx	193.6.5.0/27	Subnet 0/All zeros *
11000001|00000110|00000101|001xxxxx	193.6.5.32/27	Subnet 1
11000001|00000110|00000101|010xxxxx	193.6.5.64/27	Subnet 2
11000001|00000110|00000101|011xxxxx	193.6.5.96/27	Subnet 3
11000001|00000110|00000101|100xxxxx	193.6.5.128/27	Subnet 4
11000001|00000110|00000101|101xxxxx	193.6.5.160/27	Subnet 5
11000001|00000110|00000101|110xxxxx	193.6.5.192/27	Subnet 6
11000001|00000110|00000101|111xxxxx	193.6.5.224/27	Subnet 7/All ones *

*** Ezeket régen nem volt szab használni!**

Alhálózat címkiosztási példa /2

- **A Subnet 4-et osszuk ki …**
  - **Csak 30 gépet tudunk azonosítani, mert**
  - **egyet elvisz a subnet azonosító,**
  - **egyet pedig a subnet broadcast cím ...**

Bitminta	IP cím	Megjegyzés
11000001|00000110|00000101|10000000	193.6.5.128	Subnet azonosító
11000001|00000110|00000101|10000001	193.6.5.129	Gép 1
11000001|00000110|00000101|10000010	193.6.5.130	Gép 2
11000001|00000110|00000101|10000011	193.6.5.131	Gép 3
…	…	…
11000001|00000110|00000101|10011101	193.6.5.157	Gép 29
11000001|00000110|00000101|10011110	193.6.5.158	Gép 30
11000001|00000110|00000101|10011111	193.6.5.159	Subnet broadcast

Változó alhálózat méretek

- **Variable Length Subnet Mask (VLSM) (RFC 1009)**
  - **Különböző alhálózatok létrehozása**
    - **hatékonyabb címfelhasználás**
  - **A routing-nak támogatnia kell (RIP-1 nem jó!)**
    - **a kiterjesztett prefixet (subnet maszkot) is át kell adni (terjeszteni kell)**
    - **Minden router a leghosszabb prefix egyezése elvén továbbítsa a csomagokat**
    - **Az aggregációhoz a címkiosztásnak követnie kell a topológiai feltételeket**
  - **A többszintű hierarchia előnye**
    - **alhálózatokat tovább tudunk bontani**
    - **az aggregáció miatt ez kívülről nem látszik**

Maszk (bin)	Maszk (dec)
10000000	.128
11000000	.192
11100000	.224
11110000	.240
11111000	.248
11111100	.252
11111110	.254

Longest prefix match

- **Tegyük fel, a 2.28.137.130 címre kell a csomagot továbbítani, az alábbi router tábla esetén: Forgatókönyv:**
  - **Kigyűjteni az összes bejegyzést, ahol cél IP és maszk a prefixet adja**
  - **Ezekből kiválasztani azt, amelyiknek a leghosszabb a maszkja. Legrosszabb esetben 0.0.0.0, azaz a default route a választás**

Route prefix	Interface	Next-hop	Target IP mask	
0.0.0.0./0	Serial 0	1.1.1.1	0.0.0.0	☺
2.28.0.0/16	Serial 1	2.2.1.1	2.28.0.0.	☺
2.28.137.0/24	Serial 2	2.3.1.1	2.28.137.0	☺
2.28.137.128/25	Ethernet 0	2.3.1.4	2.28.137.128	☺ ☺
3.10.0.0/16	Serial 1	2.2.1.1	2.28.0.0.	
3.10.11.0/24	Serial 2	2.3.1.1	2.28.137.0	

Az osztály nélküli címzés

- **Classless Inter-Domain Routing (CIDR) (RFC 1517-1520)**
  - **A maszk rövidebb is lehet, mint a hálózatazonosító (superneting)**
    pl:**193.6.0.0-193.6.15.0** 16db **C osztály**
    → 255.255.240.0 (/20) → 193.6.0.0 /20
  - **Több hagyományos A,B,C osztály összefogása**
  - **laza bithatárok: /4 … /30**
  - **szükségtelenné válik az osztályok használata**
  - **a routing nem az első bitek szerint dönt**
  - **a címtér sokkal jobban kihasználható**
- **A CIDR együtt élhet a klasszikus routinggal, de**
  - **a régebbi eszközök nem mindig kezelik**

A VLSM és a CIDR

- **Mindkettő támogatja egy A, B, C hálózaton**
  - **flexibilis alhálózat-rendszer kialakítását**
  - **belsejének elrejtését (aggregáció)**
- **A CIDR azonban lehetővé teszi**
  - **több bitszomszédos hálózatok összefogását** (supernetting)
    - **és ezen belül tetszőleges hierarchia kialakítását**
  - **több szomszédos A, B, C hálózat összevont útvonal-választási bejegyzését**

Címfoglalási szabályok

- **A globális Interneten minden IP cím egyedi**
  - **a globális IP címeket engedélyeztetni kell** (IANA, EU: RIPE)
- **Internettől elszigetelt magánhálózaton**
  - **tetszőleges kiosztást készíthetünk, de így a**
  - **későbbi esetleges csatlakozás gondot okozhat.**
  - **Lokális címtartományok (IANA) (RFC 1918)**
    - **10.0.0.0./8**
    - **172.16.0.0./12**
    - **192.168.0.0./16**

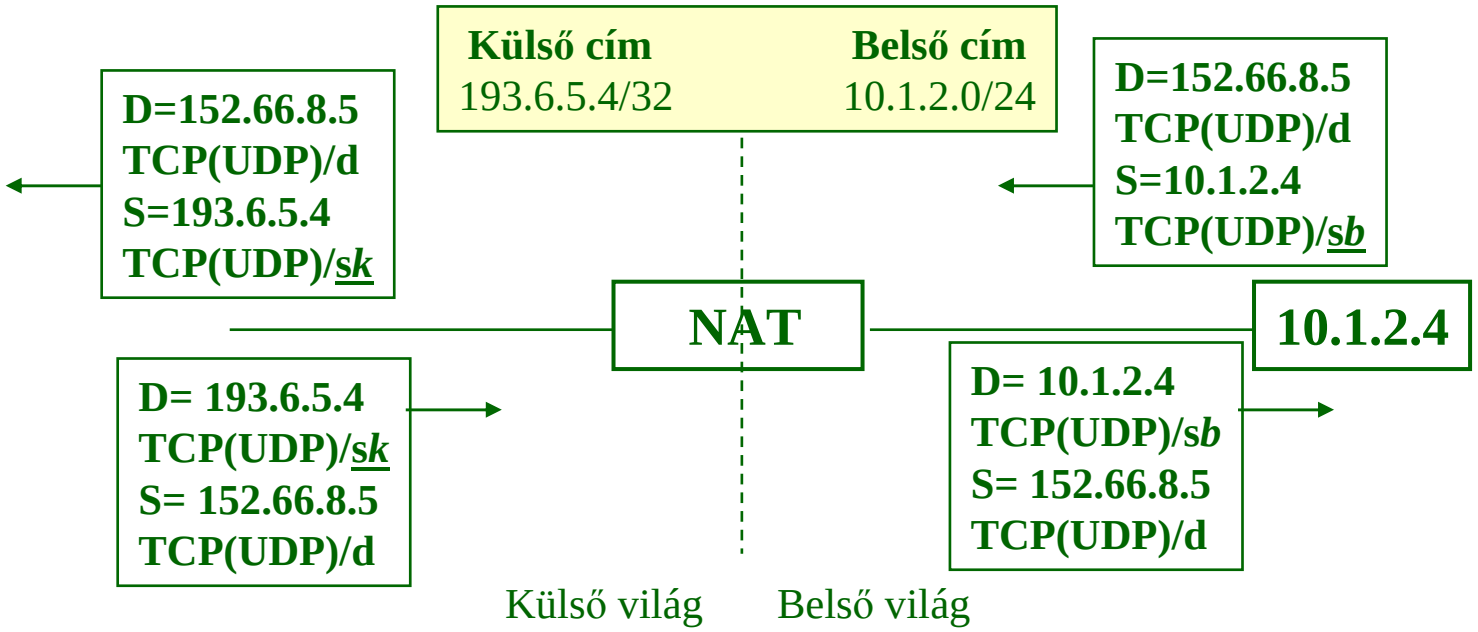
Magánhálózat csatlakoztatása az Internetre

- **Ha bejegyzett címtartományokat használtunk,**
  - **nincs gond.**
- **A lokális címtartományú magánhálózatot tűzfallal leválasztjuk (se ki, se be)**
  - **nincs gond, de nem használható az Internet közvetlenül**
- **Lokális címtartományú magánhálózatról bejegyzett címtartományra kívánunk áttérni**
  - **átszámozás (elég költséges),**
  - **címfordítás NAT (Network Address Translation) (RFC 1631) lehetséges.**

Címfordítás, NAT (IP masquerade)

- **A belső és a külső IP címek összerendelése**
  - **Címfordítási táblázat (ötlet ua. Protokoll – több port):**

Külső IP cím(ek)	⇔	**Belső IP cím(ek)**
Protokoll (TCP, UDP)		**Protokoll (TCP, UDP)**
Külső Port		**Belső Port**



Címfordítás, NAT

Egyetlen külső cím esetén:

- **Kicseréli a belső forrás címet a külső címre**
- **Megnézi, hogy az eredeti forrás port szabad-e a külső oldalon.**
- **Ha szabad, akkor azt választja.**
- **Ha foglalt, akkor a szabad (választható) portok közül választ egyet.**
- **Ha nincs szabad port, akkor eldobja a csomagot.**
- **Bejegyzi egy táblázatba a fordítást a visszafelé jövő, illetve a további csomagok érdekében.**

Több külső cím esetén:

- **Ha nincs szabad port, akkor veszi a következő külső címet és azon keres szabad portot.**
  (Ugyanúgy mint egy cím esetén.)

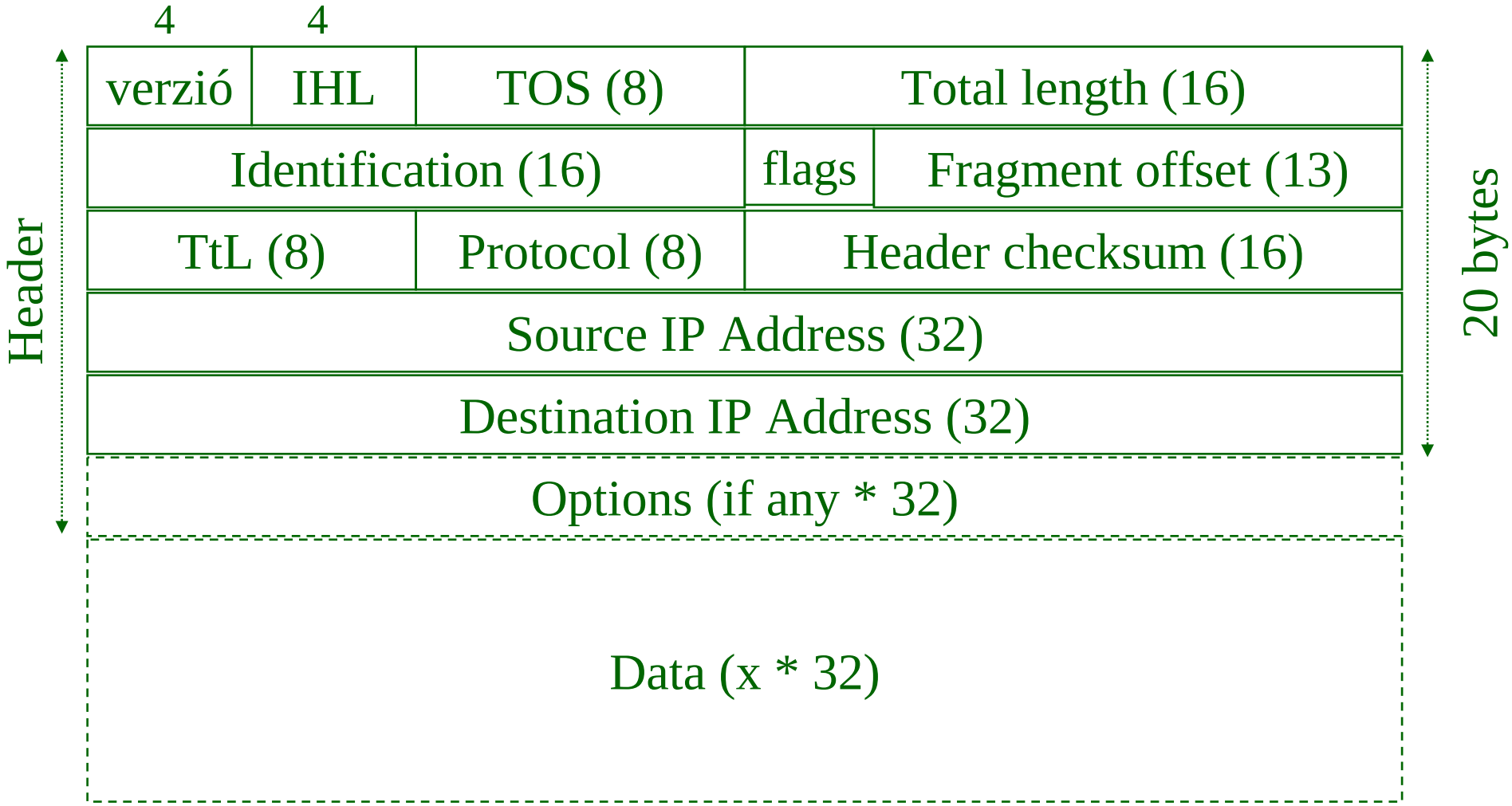
Címfordítás, NAT

- **A NAT transzparens mindazon protokollokra melyek**
  - **nem használnak IP címeket a csomag belsejében,**
  - **nem használnak előre megbeszélt, vagy magasabb szinten egyeztetett címet.**
- **A NAT amennyiben felismeri (és ismeri) a magasabb szintű protokollokat, úgy a csomag belsejében is elvégezheti a címváltoztatást.**
  Pl:
  - **FTP** (a behívó kliens mondja meg a szervernek, hogy hova hívjon vissza a szerver – aktív FTP, (passzívnál a behívó hív újra)).
  - **embedded IP Addresses in DNS "A and PTR" records.**
    NAT will translate the address(es) which appear in DNS responses to name lookups (A queries) and inverse lookups (PTR queries).

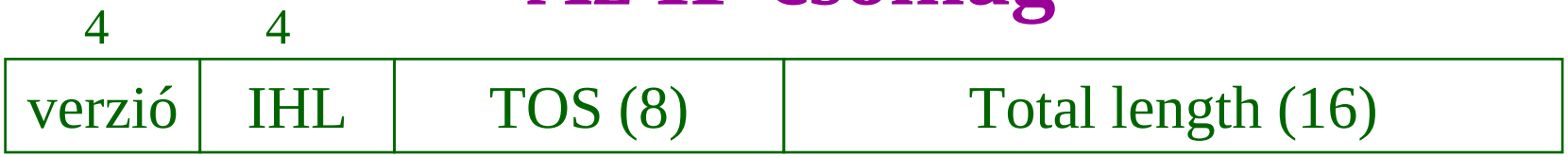
IPv4 címzés fejlődése

- **Klasszikus címosztályok: 1981**
  - **a címzési rendszer alapelvei**
- **Alhálózatok: 1985**
  - **kétszintű hierarchia**
- **Változó méretű alhálózatok: 1987**
  - **többszintű hierarchia, aggregáció**
- **Osztálymentes címzés: 1993**
  - **tetszőleges hálózatméret, hálózatok közti aggregáció**
- **Címfordítás: 1994**
  - **a címtér többszörös lefedése**

Az IP csomag


4
4
verzió
IHL
TOS (8)
Total length (16)
Identification (16)
flags
Fragment offset (13)
TtL (8)
Protocol (8)
Header checksum (16)
Source IP Address (32)
Destination IP Address (32)
Options (if any * 32)
Data (x * 32)
Header
20 bytes

Az IP csomag



- **Verzió: 4 (IPv4)**
- **IHL: Header length** (a header hossza az opciókkal együtt)
  32 bites szavakban 4 bit ⇒ a header max 60 byte hosszú lehet
- **TOS: Type of Services, csak 3+4 bitet használ:**
  - **3 bit a prioritásra (7 a magas, 0 az alacsony) + 4 bit:**
  - **D bit: Minimize delay (Pl. telnet)**
  - **T: Maximize throughput (Pl. Ftp data)**
  - **R: Maximize reability (pl SNMP)**
  - **Minimize monetary cost**

  egyszerre csak egy bit lehet 1

  (Nem minden implementáció használja (pl. OSPF dönthet ez alapján))
- **Total length: az IP datagram teljes mérete bájtokban**
  16 bit ⇒ IP datagram max. 65535 byte

Az IP csomag

Identification (16)	Flags	Fragment offset (13)

- **Identification: a datagram egyedi azonosítója, amit a küldő hoszt állít be (pl fregmentáció esetén azonosítja az egyes darabokat)**
- **Flags (3 bit):**
  - **1 bit nem használt**
  - **1 bit (DF): "don't fragment" bit: ha 1, a csomag nem fregmentálható.**
    - **Ha mégis kellene: ICMP error**

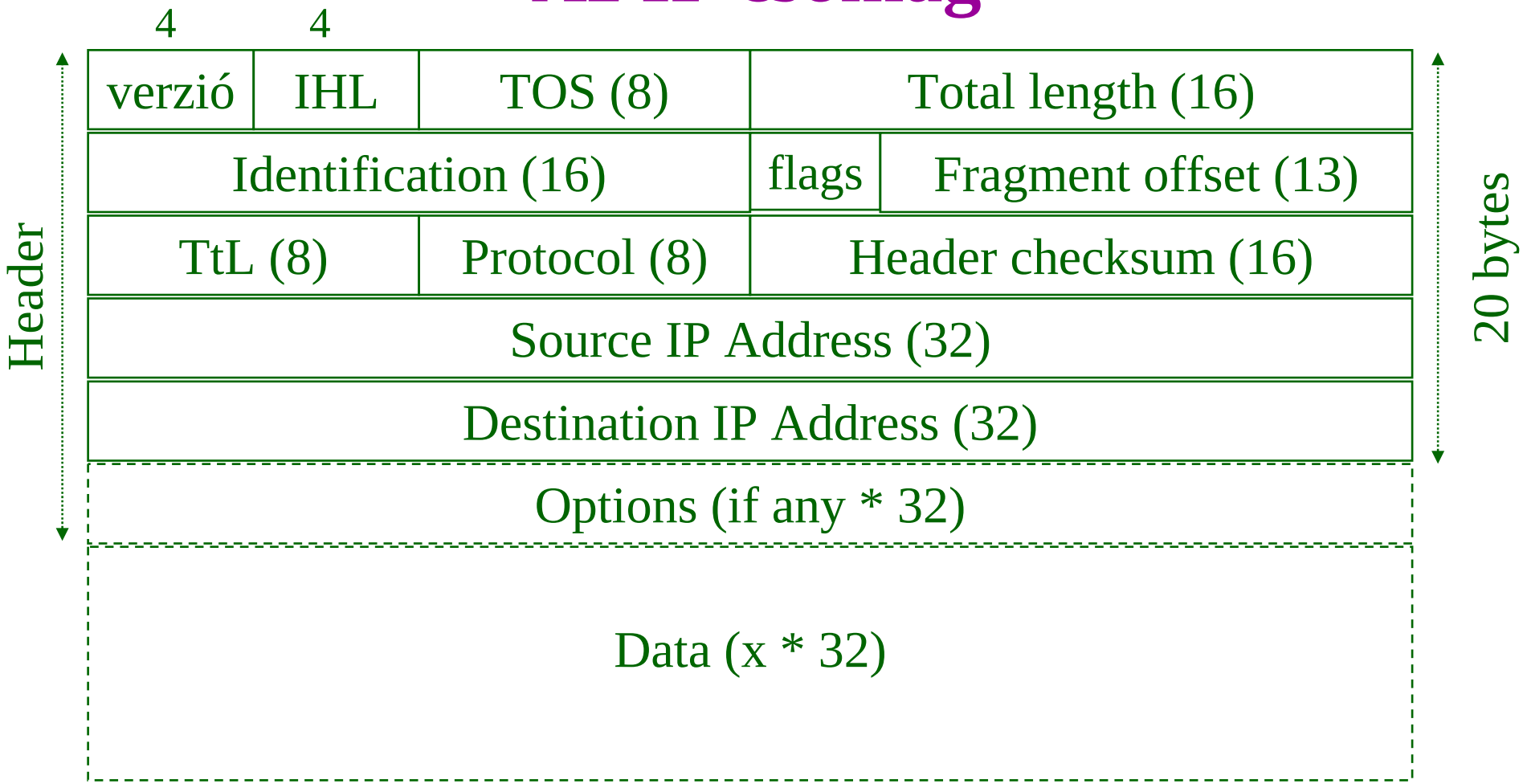
    "fragmantation needed but don't fragment bit is set"
  - **1 bit (MF): fregmentálás esetén 1, ha van még további darab; 0, ha ez az utolsó**
- **Fragment offset (13 bit): fregmentáció esetén a data melyik része (milyen az eltolás 8 byte-okban számolva). Az első darab esetén = 0. Az összes darab hossza csak 8 egész többszöröse byte lehet (kivéve az utolsó darabot).**

Az IP csomag

TTL (8)	Protocol (8)	Header checksum (16)

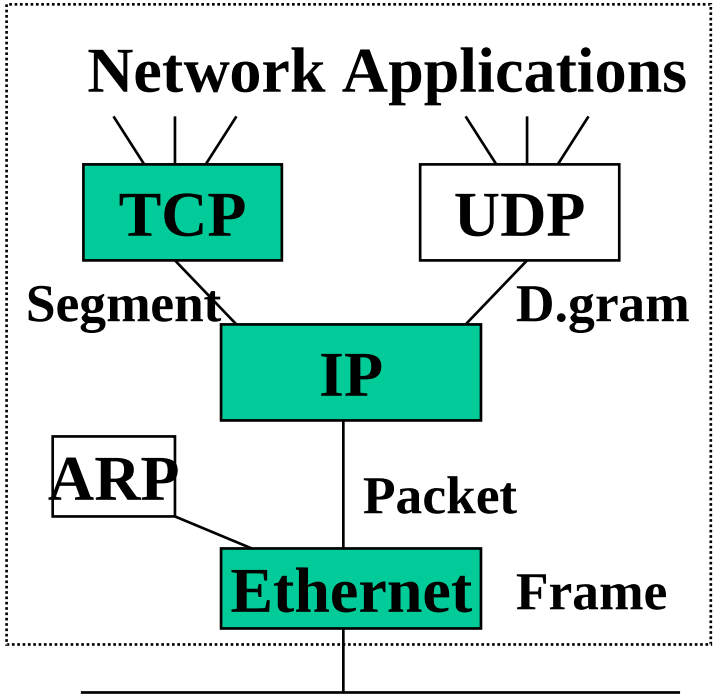
- **Time to Live (TTL, 8 bit):**
  - **Minden ugrás esetén a router annyival csökkenti, ahány sec-ot állt nála (de legalább 1-gyel).**
  - **Régebben 32 v. 64, manapság 128 kezdeti értékkel**
  - **Ha eléri a 0-át,**
    - **a router eldobja és**
    - **ICMP "time exceeded" error a feladónak.**
- **Protocol: az IP csomagot előállító protokollt (pl TCP, UDP, ICMP, IGMP) azonosítja**
- **Header cheksum: az IP fejrészre vonatkozó 1 komplemens 16 bites összeg. Mivel a TTL változik, mindig újraszámítandó (Hop). Hiba esetén eldobják a csomagot.**
  (A vevő az egészre számol 1 komplemens összeget → ha jó, csupa 1)

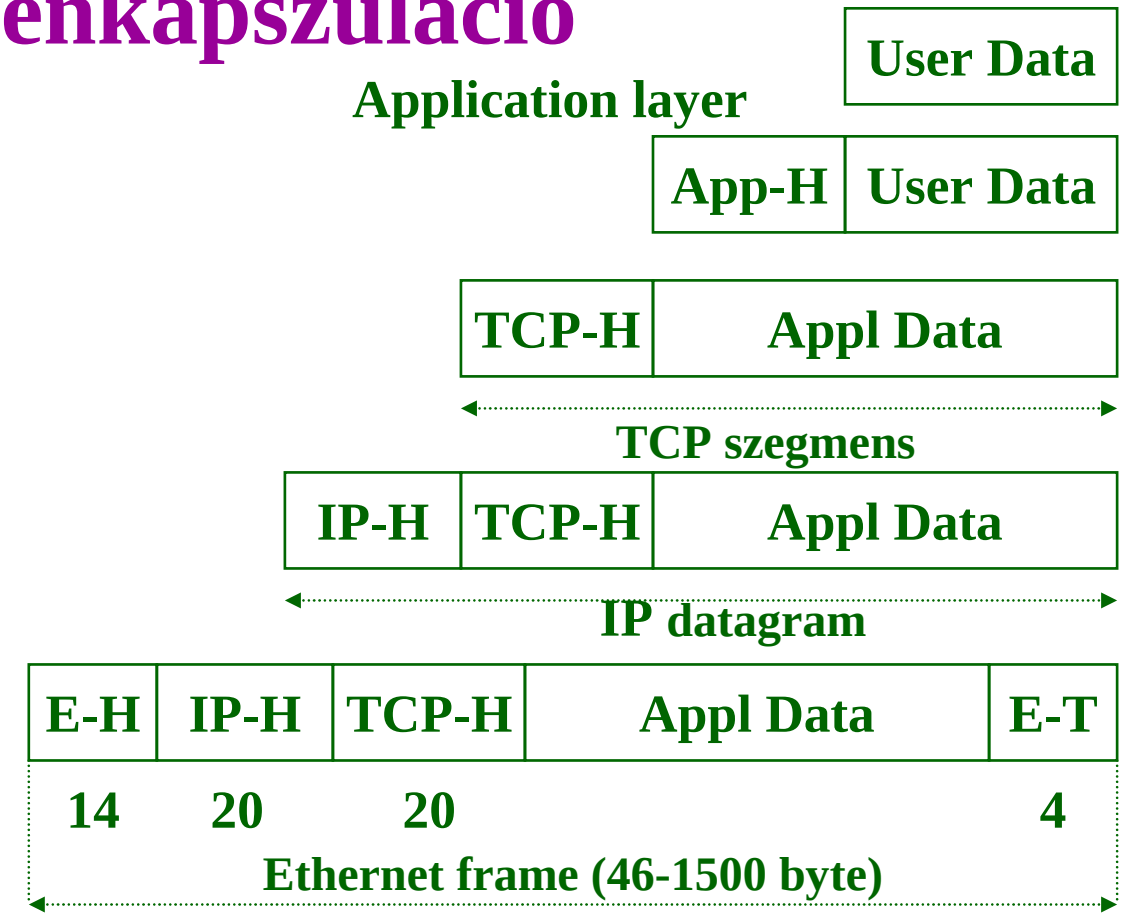
Az IP csomag



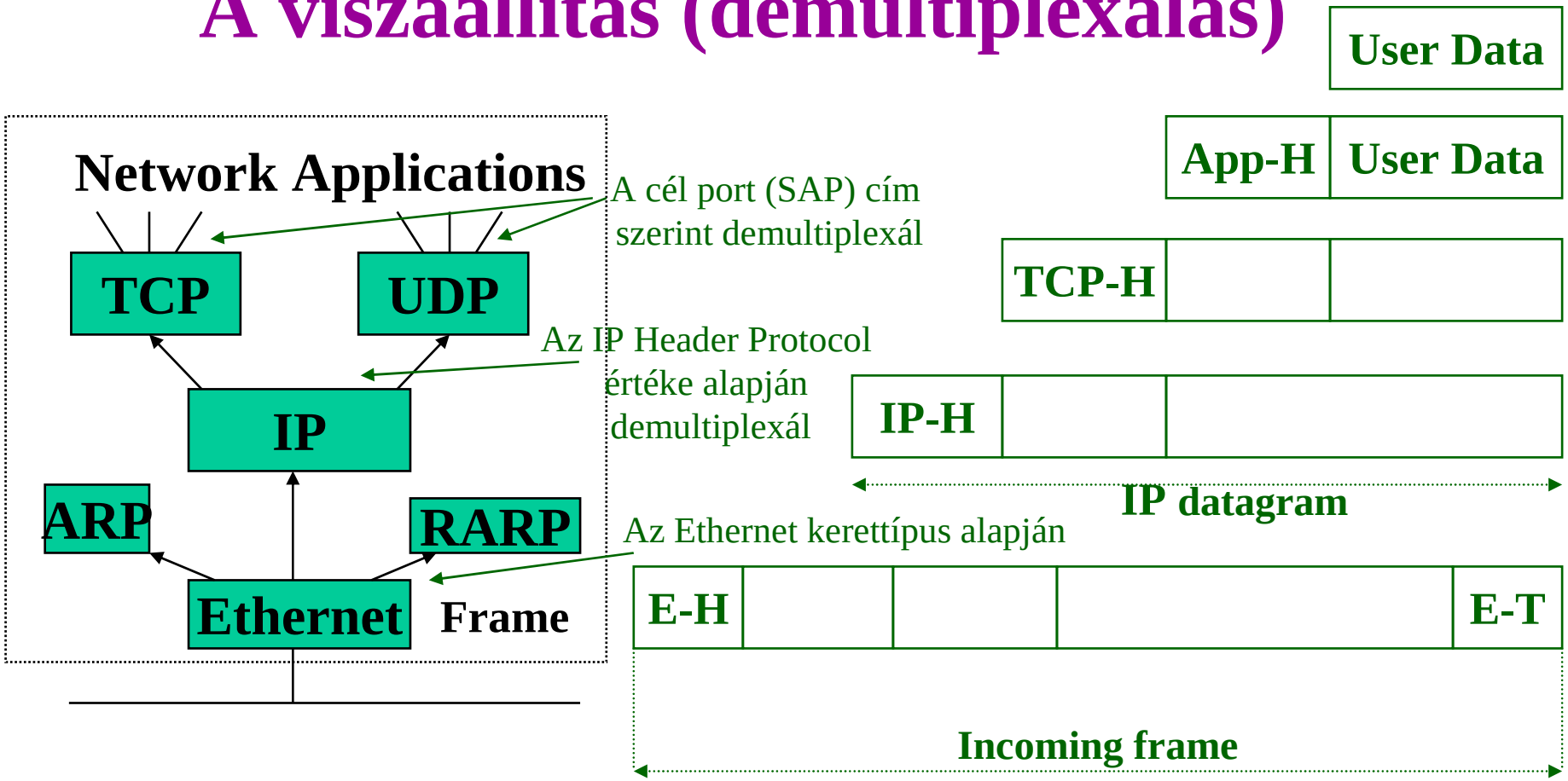
- **SA, DA (IP címek)**
- **Opciók és adatok**

Az enkapszuláció





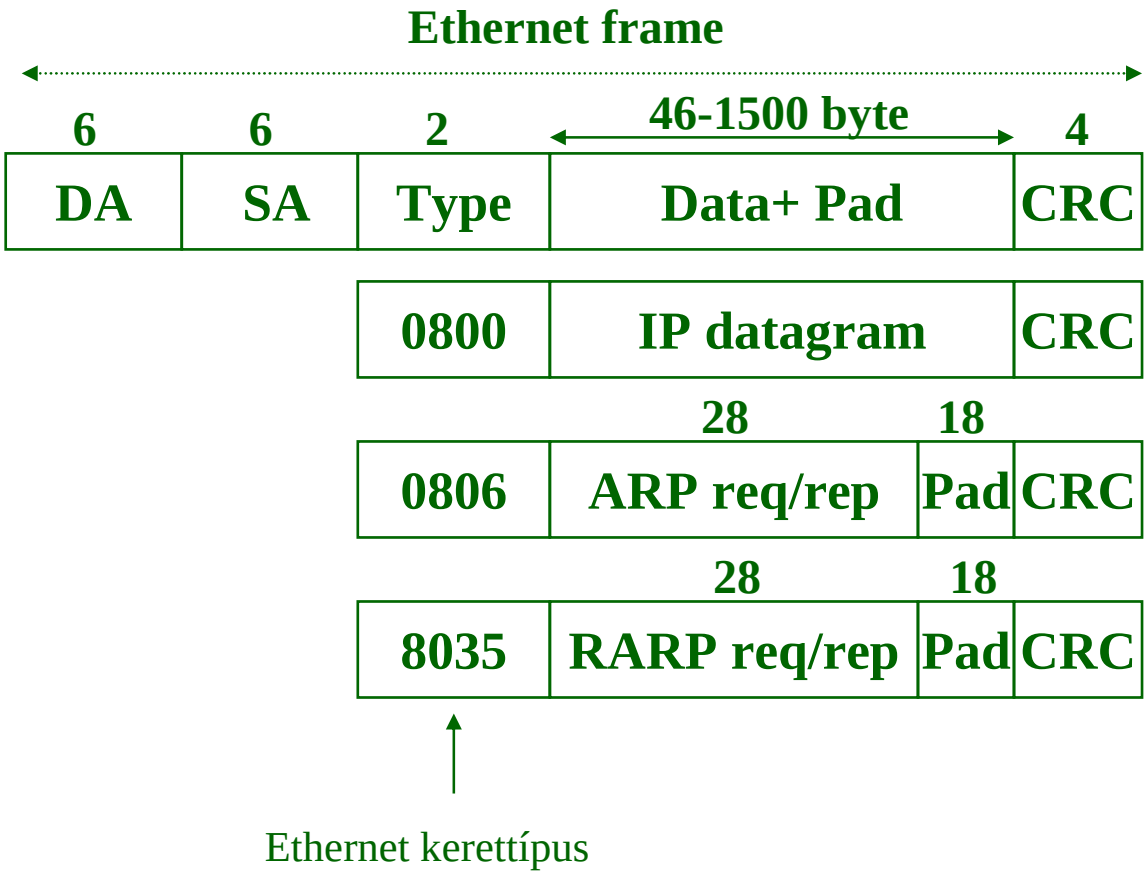
A viszaállítás (demultiplexálás)



A TCP SAP, UDP SAP azonosítás: 16 bites port szám alapján (port number)

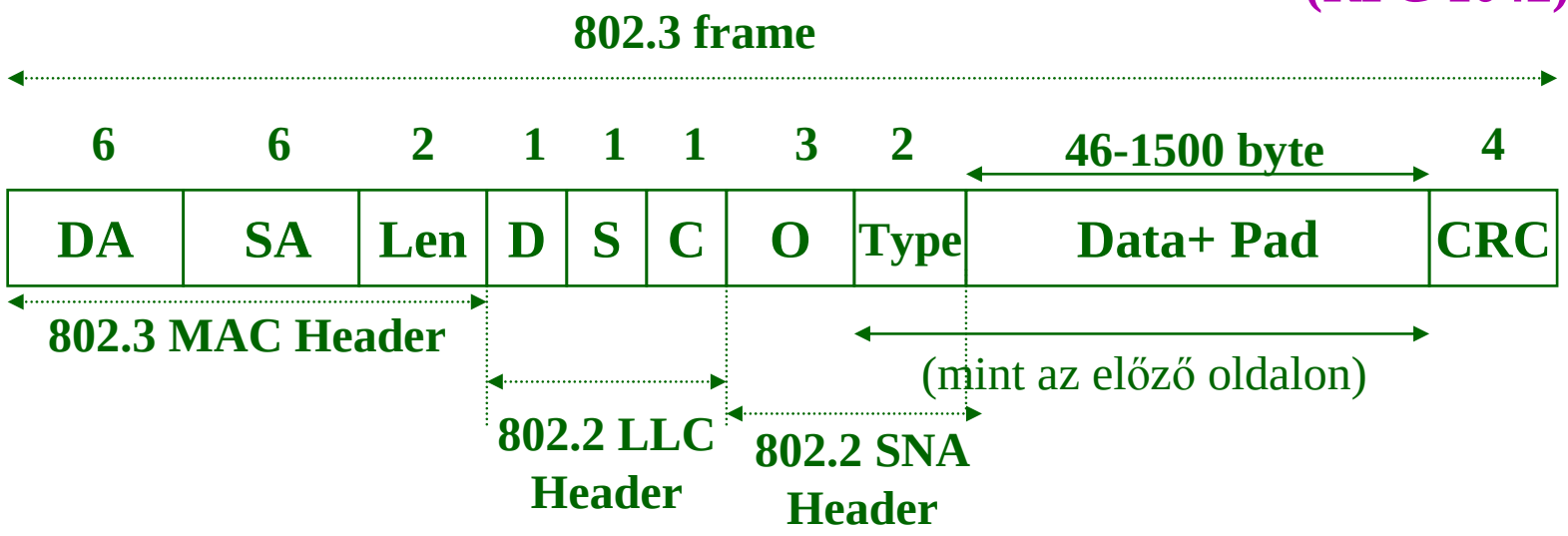
Link Layer: Ethernet enkapszuláció

(RFC 894)



Link Layer: 802.3 enkapszuláció

(RFC 1042)



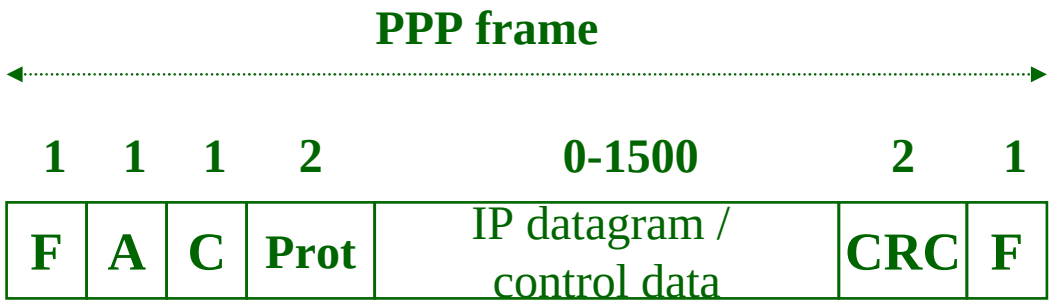
D: Destination SAP (AA)
S: Source SAP (AA)
C: Control (vezérlés) (03)

} LLC: Logical Link Control Header

O: Organisation Code (00)
Type: Lásd Ethernet

} SNA: Sub Network Access Protocol Headre

Link Layer: PPP enkapszuláció



F: Flag

szinkron:7E (bitbeszúrás: 01111110)

aszinkron: 7E, de karakter orientált

transzparens: adatok között

7E: 0x7D=escape: 0x7E→7D,5E;

adatok közt 0x7D→7D,5D;

20h-nál kisebb →7D,20H+d

A: Address (FF)

C: Control (03)

Prot: 0021: IP datagram

C021: Link Control Data

8021: Network Control Data

Nincs ARP ← ez pont-pont kapcsolat !

IP soros vonalon (nem csak IP)

Point to Point Protocol

- aszinkron, 8 bites adatok,
- szinkron, bit orientált.

A protokoll lehet:

LCP (Link Control P) (RFC 1548):

Data link kapcsolat létrehozása, tesztje, konfigurációja

NCP (Network Cont. Prot.) (RFC 1332):

különböző hálózati protokollok (pl. IPX) átvitele PPP-n

Az ARP (RFC 826)

- **Feladat: host vagy router IP címének leképzése MAC címmé**
- **Fogalmak, alapok:**
  - **IP cím: hálózat + host cím, a subnet maszk segít a szétválasztásban**
  - **Default router: egy hálózathoz tartozó router és annak IP címe**
  - **Helyi kommunikáció: egy hálózaton belüli**
    - **Ua a hálózati cím (ua a subnet-mask)**
  - **Távoli kommunikáció: hálózaton kívüli**
    - **más a hálózati cím**

Az ARP (RFC 826)

- **Fogalmak:**
  - **Címzési szabályok:**
    - **minden hosztnak (legalább egy) egyedi IP címe van**
    - **az egy hálózaton lévőknek közös a hálózati címe (netid) és a szubnet maszkja**
  - **A hálózat itt azonos a „Broadcast Domain”-nel!**
    - **A hálózat azon része, melyről „Local Broadcast Packet” használatával információt nyerhetek**
      - **ismétlők, hidak továbbítják a Local Broadcast Packet-et,**
      - **routerek nem!**
  - **A szegmensen belül helyi kommunikáció, „Direct Delivery” (közvetlen kézbesítés) van.**

Az ARP (RFC 826)

- **A MAC címek nyerhetők:**
  - **Local Broadcast ARP_REQUEST küldése után a válaszokból ARP_REPLY** (amiket azonnal cache-elni lehet)
  - **Majd a későbbiekben a cache-ből (IP - MAC párok)**
- **A továbbiakhoz tegyük fel, hogy megvan a cél IP címe (Pl. DNS-ből)**

Az RARP

- **Saját IP cím lekérdezése (pl. boot) a saját MAC cím alapján**
  - **RARP_REQUEST Broadcast-al**
  - **A szerver táblázat alapján válaszol RARP_REPLY**

További protokollok

- **Boot Protocol (RFC 1542)**
  - **MAC és IP cím statikus összerendelése**
  - **Kliens-szerver-relay_agent konfiguráció**
  - **UDP csomagokban request-reply**
- **Dynamic Host Configuration Protocol (RFC 1541)**
  - **MAC és IP cím dinamikus összerendelése,**
  - **címtartományok kijelölhetők,**
  - **címhasználat időben korlátozódhat,**
  - **kérheti a korábbi címét,**
  - **hasznos erőforrások (pl. DNS) jelezhető,**
  - **BOOTP-vel felülről kompatibilis.**

Az IP csomagok továbbítása

- **Megvizsgálja a cél IP címet, az „helyi”, vagy „távoli”**
  - **A saját subnet maszkkal leválasztja a hálózati címrészt, és összeveti a sajátjával: ha egyezik: helyi, ha nem: távoli.**
- **Ha helyi, akkor (Direct Delivery)**
  - **Nézi a cache-ében, van-e hozzá MAC cím. Igen: a MAC szinten elküldi a címzettnek.**
  - **Nincs: Local Broadcast kezdeményezéssel választ kér, és így megkapja a cél MAC címet. Mindjárt cache-eli, és MAC szinten elküldi a címzettnek.**

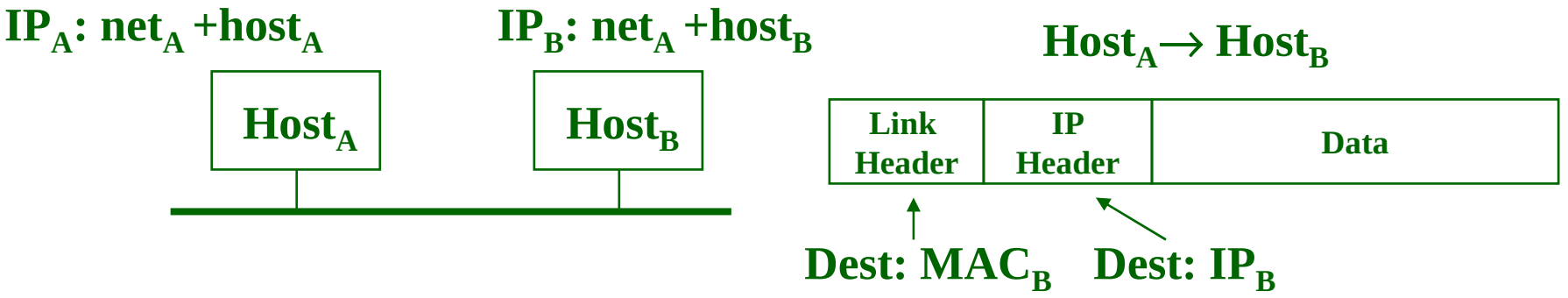
Az IP csomagok továbbítása

Ha a cél cím „távoli”, akkor (Indirect Delivery)

- **Nézi saját forgalomirányító tábláját (route table), van-e speciális út a célhoz, ha van,**
  - **keresi a cache-ében, ismeri-e az úthoz rendelt router MAC cím. Igen: MAC elküldi annak.**
  - **Nincs: Local Broadcast kezdeményezéssel választ kér, és így megkapja a router MAC címet. Mindjárt cache-eli, és MAC szinten elküldi neki**
- **Ha nincs speciális út (esetleg nincs is forgalomirányító tábla) – a default router-nek küldi:**
  - **Nézi a cache-ében, van-e a default router-hez MAC cím. Igen: MAC elküldi annak.**
  - **Nincs: Local Broadcast kezdeményezéssel választ kér, és így megkapja a default router MAC címet. Mindjárt cache-eli, és MAC szinten elküldi neki**

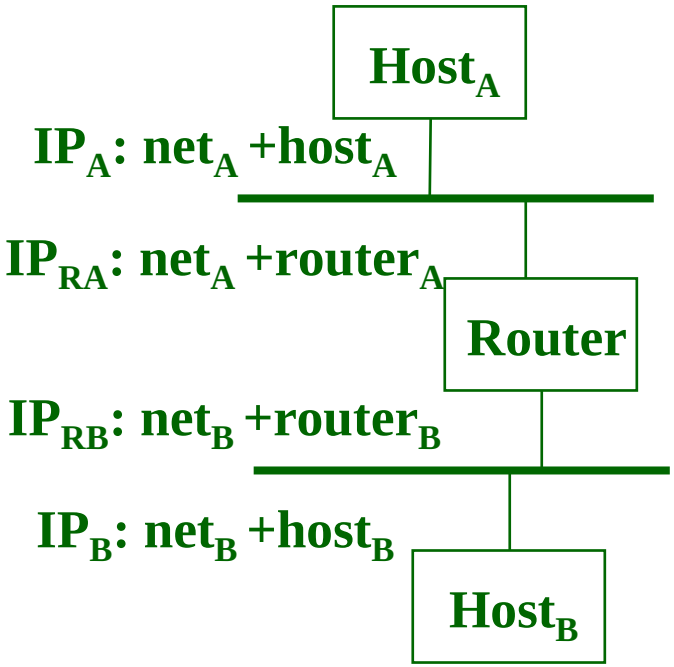
Az IP csomagok továbbítása

- **Két host közös hálózaton (netid), közös adatkapcsolati réteg**



Az IP csomagok továbbítása

- **Két host különböző hálózaton**



1:

Host$_A$ → Router$_A$

Dest: MAC$_{RA}$ Dest: IP$_B$

2:

Router$_B$ → Host$_B$

Dest: MAC$_B$ Dest: IP$_B$

Az IP csomag nem (alig (TTL+checksum)) változik, csak a Link Layer Header más.

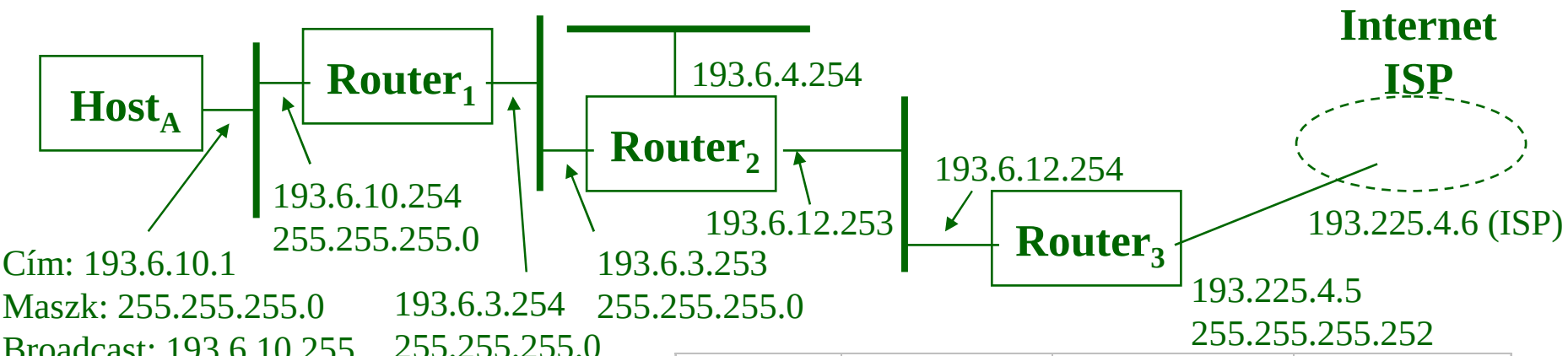
Az IP routing

- **Az útvonalválasztó az eredeti datagram-on a következőket változtatja meg:**
  - **Dekrementálja a Time-to-Live mezőt (amiből eldönthető, hány sec-ig, vagy ugrásig maradhat meg a datagram).**
  - **Újraszámítja a checksum-ot.**

IP routing tábla

- **Egy router a routing tábláját nézi végig, hogy melyik portjára (melyik interfészére) küldje a datagramot.**
  - **A keresési kulcs a cél IP hálózati címe.**
  - **A kereséshez kell a szubnet maszk is.**
- **A csomagtovábbítás**
  - **a leghosszabb illeszkedő prefix (longest prefix match),**
  - **hop-by-hop (azaz minden router maga dönt),**
  - **nem megfelelő router választása esetén (a router ugyanazon interfészén visszaküldi a csomagot) ICMP Redirect a küldőnek.**

Routerek konfigurációja



A gateway (router) címe mindig olyan, amit a saját hálóján közvetlenül elér.

Ez a statikus kitöltési módja a routing tábláknak (static routing)

Router1	net id	mask	gateway
IP Route	193.6.12.0	255.255.255.0	193.6.3.253
	193.6.4.0	255.255.255.0	193.6.3.253
Default	0.0.0.0	0.0.0.0	193.6.3.253
Router2	net id	mask	gateway
IP Route	193.6.10.0	255.255.255.0	193.6.3.254
Default	0.0.0.0	0.0.0.0	193.6.12.254
Router3	net id	mask	gateway
IP Route	193.6.3.0	255.255.255.0	193.6.12.253
	193.6.4.0	255.255.255.0	193.6.12.253
	193.6.10.0	255.255.255.0	193.6.12.253
Default	0.0.0.0	0.0.0.0	193.225.4.6

Az ICMP

- **Internet Control Message Protocol**
  - **Az alapvetően a hálózati réteggel kapcsolatos üzenetek továbbítására**
- **Az ICMP enkapszuláció**
- **ICMP message**

Type (8)	Code (8)	Checksum (16)
Content (type és code függő tartalom)		

Checksum: a teljes ICMP üzenet ellenőrzése

- **Típusok:**
  - **hibaüzenetek,**
  - **információk,**
  - **diagnosztikai üzenetek.**

ICMP példák

Type	Code	Üzenet (RFC792)
0	0	echo reply (ping)
3		Destination unreachable
	0	Network unreachable
	1	Host unreachable
	3	Port unreachable
	4	Fregmentation is needed but don't fragment bit set
4	0	Source quench: fojtócsomag (flow control)
5		Redirect
	0	Redirect for network
	1	Redirect for host
8	0	echo request (ping)
11		Time exceed
	0	Time to live = 0 during transmit (traceroute)
	1	Time to live = 0 during reassembly

stb.

ICMP

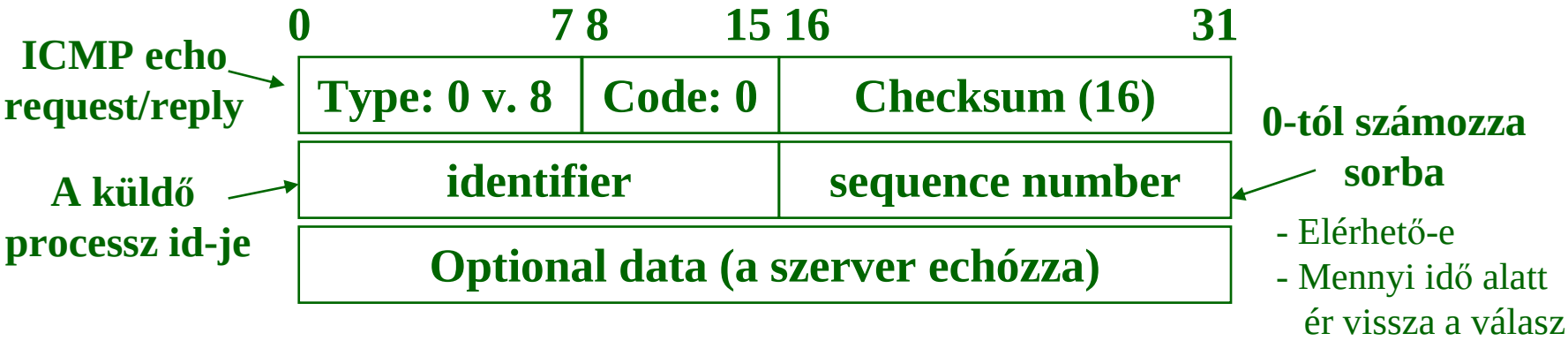
- **Az ICMP hibaüzenetek mindig tartalmazzák annak az IP datagram-nak a fejrészét (20 byte) és első 8 bájtját, ami a hibát okozta.**
- **Így a fogadó ICMP modul meghatározhatja a protokollt és a user processzt, amihez a hiba tartozik.**

IP hálózatok vizsgálata

- A vizsgálatok szükségessége
  - Hálózat beüzemelése, tesztelése
    - a végpontok látják-e egymást? (connectivity)
    - a csomagszűrés jól van-e beállítva?
  - Üzemelő hálózat teljesítményének fokozása
    - hatékony a működés (perfomance)
    - torlódások vannak-e?
    - Erőforrások kihasználtsága?
- Van néhány egyszerű alkalmazás
  - ping: az elérhetőség ellenőrzésére
  - traceroute: állomás elérési útvonalának vizsgálata

A ping

- **Állomás elérhetőségének ellenőrzésére**
  - **ping kliens:**
    aki kezdeményez egy ICMP echo request-tel;
  - **ping szerver: aki válaszol egy ICMP echo reply-vel.**
  - **A csomagok sorszámot és időbélyeget kapnak**
    - **csomagvesztés detektálható,**
    - **duplikáció detektálható,**
    - **sorrendcsere detektálható,**
    - **késletetési viszonyok változása (torlódás) detektálható.**



A traceroute

- **Állomás elérési útvonalának vizsgálata**

  Ötlet:

  – **Ha router TTL = 1 vagy 0 IP datagramot kap, azt nem továbbítja, hanem ICMP time exceed üzenetet küld.**

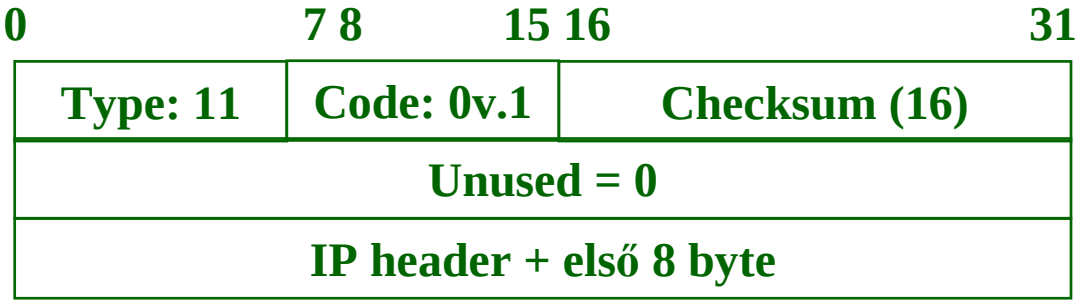
  – **Ha a router IP datagramot továbbít, a TTL értéket annyival csökkenti, ahány sec-ig nála volt a csomag, de legalább 1-el → gyak. a router 1 sec.-nél rövidebb ideig tart egy csomagot, így a TLL olyan mint egy ugrásszámláló**

  – **Küldjünk csomagokat rendre TTL=1,2,3 … értékekkel olyan UDP portra, amihez nem tartozik szerver alkalmazás (pl. 30000 feletti portszám)**

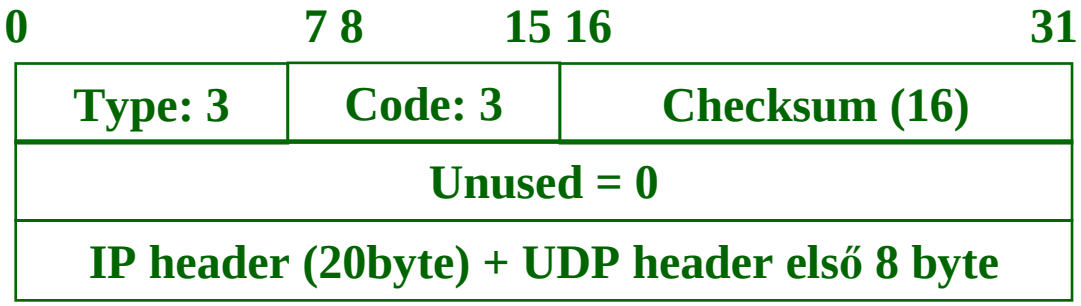
    - **A soron következő első, második stb. router eldobja és ICMP üzenetet küld a saját címével mint feladóval → megtudható a köztes routere-ek címe;**
    - **Amikor eljut a célállomásra → ICMP port unreachable üzenet jön vissza (a célállomás címével), ebből tudható, hogy elérte a célállomást.**

A traceroute

- **ICMP time exceeded:**



- **ICMP UDP port unreachable:**



IP Version 4 - IP Version 6

- **Az IPv4 címtartomány kimerülőfélben van (még a subnet maszkokkal + NAT is)**
- The current IPv4 Internet routing infrastructure is a **combination of both flat and hierarchical routing** ⇒ there are routinely **over 85,000 routes in the routing tables of Internet backbone routers .**

IP Version 4

IPv4 address allocation history:

- **1981 - IPv4 protocol published**
- **1985 ~ 1/16 total space**
- **1990 ~ 1/8 total space**
- **1995 ~ 1/4 total space**
- **2000 ~ 1/2 total space**
- 2005 ~ 1 ??

Despite increasingly intense conservation efforts since 1994

- **CIDR (classless inter-domain routing)**
- **NAT (network address translation)**

Theoretical limit of 32-bit space: ~4 billion devices;

- **practical limit of 32-bit space: ~250 million devices**

IP Version 6

- **Az IPv6 címek 128 bitesek (16 byte), $2^{128} = 3.4 * 10^{38}$ cím**
  - **$665 * 10^{21}$ cím per négyzetméter a földön!**
  - **Ha 10^6/μs sebességgel osztanánk ki a címeket, 20 év alatt tölthenénk be a címteret.**
  - **Könnyű subnet-eket kialakítani**
  - **Nem kell NAT**
- **Az IPv6 címek nem hoszt/node címek** (mint IPv4)**, hanem „interfész” címek**
- **Egy hosztnak lehet több interfésze (címe)**
- **IPv6 includes support for addresses of different “scope” (többes címzések – link local, site local)**
- **Unicast, multicast, anycast is lehet**
- **Ugyanakkor nincs „broadcast”**

További IPv6 előnyök

- New header format
- Large address space
- Efficient and hierarchical addressing and routing infrastructure
- Stateless and stateful address configuration
- Built-in security
- Better support for QoS
- New protocol for neighboring node interaction
- Extensibility

IPv6 előnyök - New header format

- **Keeping header overhead to a minimum.**
- **By moving both non-essential fields and optional fields to extension headers.**
- **IPv4 headers and IPv6 headers are not interoperable. IPv6 is not a superset of functionality that is backward compatible with IPv4.**
- **A host or router must use an implementation of both IPv4 and IPv6 in order to recognize and process both header formats.**
- **The new IPv6 header is only twice as large as the IPv4 header, even though IPv6 addresses are four times as large as IPv4 addresses.**

IPv6 előnyök – Hierarchical Addressing and Routing

- **Efficient, hierarchical, and summarizable routing infrastructure.**
- **Smaller routing tables.**
- **“Aggregatable Global Unicast Addresses.”**

IPv6 előnyök – Stateless and Stateful Address Configuration

- **Supports both stateful address configuration**
- **Stateful: e.g. DHCP server**
- **Stateless: hosts on a link automatically configure themselves with IPv6 addresses for the link (called link-local addresses) and with addresses derived from prefixes advertised by local routers.**
- **Even in the absence of a router, hosts on the same link can automatically configure themselves with link-local addresses and communicate without manual configuration.**

IPv6 előnyök – Built-in Security

- **Support for IPsec is an IPv6 protocol suite requirement.**
- **This requirement provides a standards-based solution for network security needs and promotes interoperability between different IPv6 implementations.**

IPv6 előnyök – Better Support for QoS

- New fields in the IPv6 header define how traffic is handled and identified.
- Traffic identification using a Flow Label field in the IPv6 header allows routers to identify and provide special handling for packets belonging to a flow, a series of packets between a source and destination.
- Because the traffic is identified in the IPv6 header, support for QoS can be achieved even when the packet payload is encrypted through IPsec.

IPv6 előnyök –
New Protocol for Neighboring Node Interaction

- **The Neighbor Discovery protocol for IPv6 is a series of Internet Control Message Protocol for IPv6 (ICMPv6) messages that manage the interaction of neighboring nodes (nodes on the same link).**
- **Neighbor Discovery replaces the broadcast-based Address Resolution Protocol (ARP), ICMPv4 Router Discovery, and ICMPv4 Redirect messages with efficient multicast and unicast Neighbor Discovery messages.**

IPv6 előnyök – Extensibility

- **IPv6 can easily be extended for new features by adding extension headers after the IPv6 header.**
- **Unlike options in the IPv4 header, which can only support 40 bytes of options, the size of IPv6 extension headers is only constrained by the size of the IPv6 packet.**

IPv4 - IPv6 összevetés

IPv4	IPv6
Source and destination addresses are 32 bits (4 bytes) in length.	Source and destination addresses are 128 bits (16 bytes) in length.
IPsec support is optional.	IPsec support is required.
No identification of packet flow for QoS handling by routers is present within the IPv4 header.	Packet flow identification for QoS handling by routers is included in the IPv6 header using the Flow Label field.
Fragmentation is done by both routers and the sending host.	Fragmentation is not done by routers, only by the sending host.
Header includes a checksum.	Header does not include a checksum.

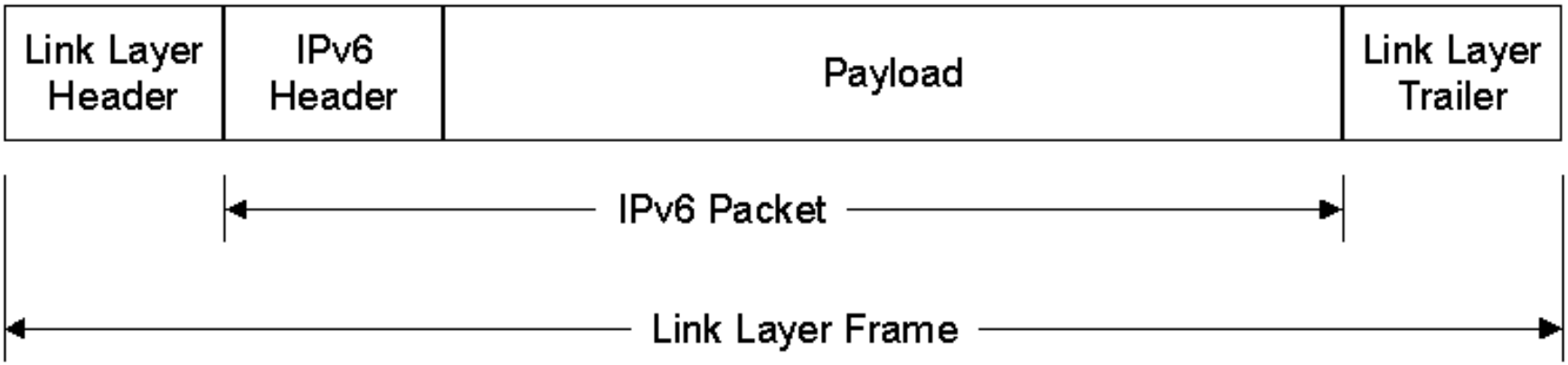
IPv4 - IPv6 összevetés

IPv4	IPv6
Header includes options.	All optional data is moved to IPv6 extension headers.
Address Resolution Protocol (ARP) uses broadcast ARP Request frames to resolve an IPv4 address to a link layer address.	ARP Request frames are replaced with multicast Neighbor Solicitation messages. “Neighbor Discovery.”
Internet Group Management Protocol (IGMP) is used to manage local subnet group membership.	IGMP is replaced with Multicast Listener Discovery (MLD) messages. “Multicast Listener Discovery.”
ICMP Router Discovery is used to determine the IPv4 address of the best default gateway and is optional.	ICMP Router Discovery is replaced with ICMPv6 Router Solicitation and Router Advertisement messages and is required. “Neighbor Discovery.”

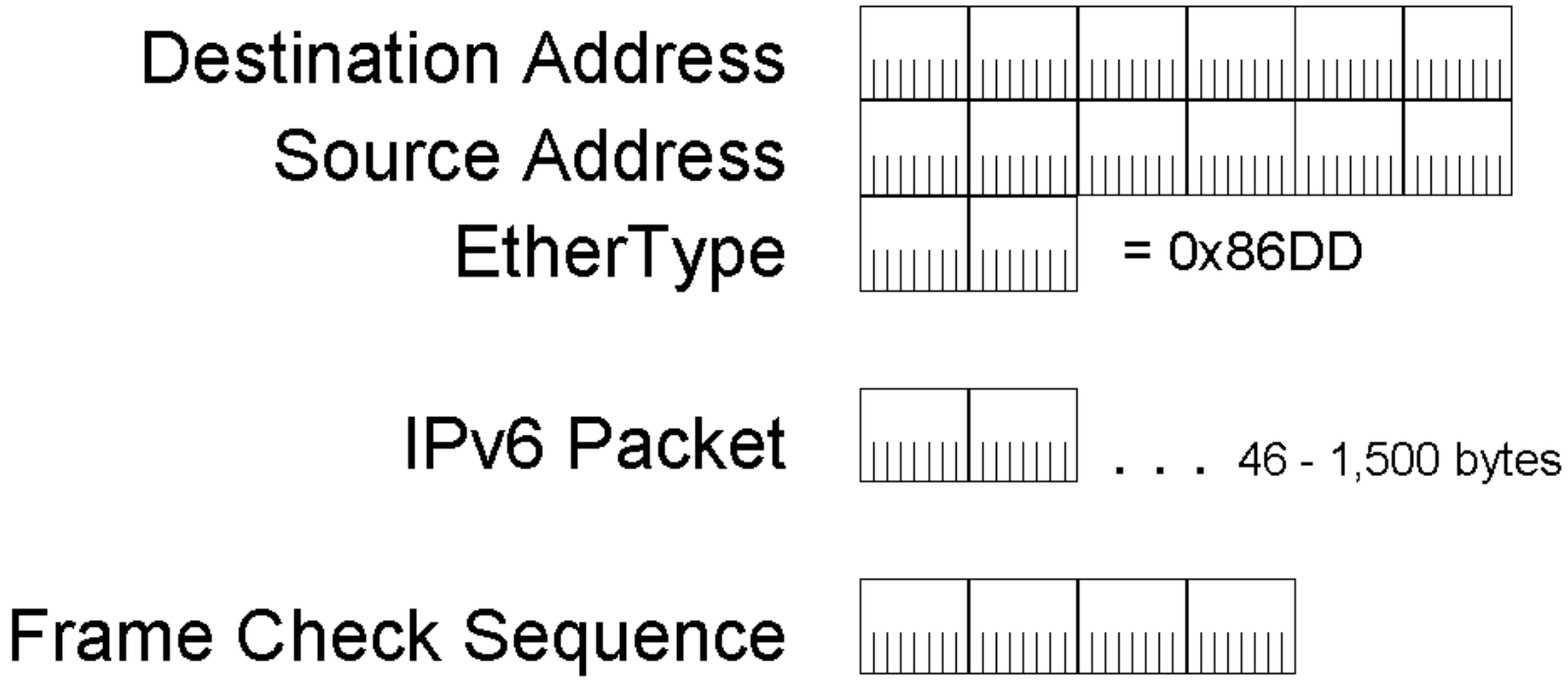
IPv4 - IPv6 összevetés

IPv4	IPv6
Broadcast addresses are used to send traffic to all nodes on a subnet.	There are no IPv6 broadcast addresses. Instead, a link-local scope all-nodes multicast address is used.
Must be configured either manually or through DHCP.	Does not require manual configuration or DHCP. “Address Autoconfiguration.”
Uses host address (A) resource records in the Domain Name System to map host names to IPv4 addresses.	Uses host address (AAAA) resource records in the Domain Name System to map host names to IPv6 addresses.
Uses pointer (PTR) resource records in the IN-ADDR.ARPA DNS domain to map IPv4 addresses to host names.	Uses pointer (PTR) resource records in the IP6.INT DNS domain to map IPv6 addresses to host names.
Must support a 576-byte packet size (possibly fragmented).	Must support a 1280-byte packet size (without fragmentation). “IPv6 MTU.”

IPv6 – Link Layer Enkapszuláció

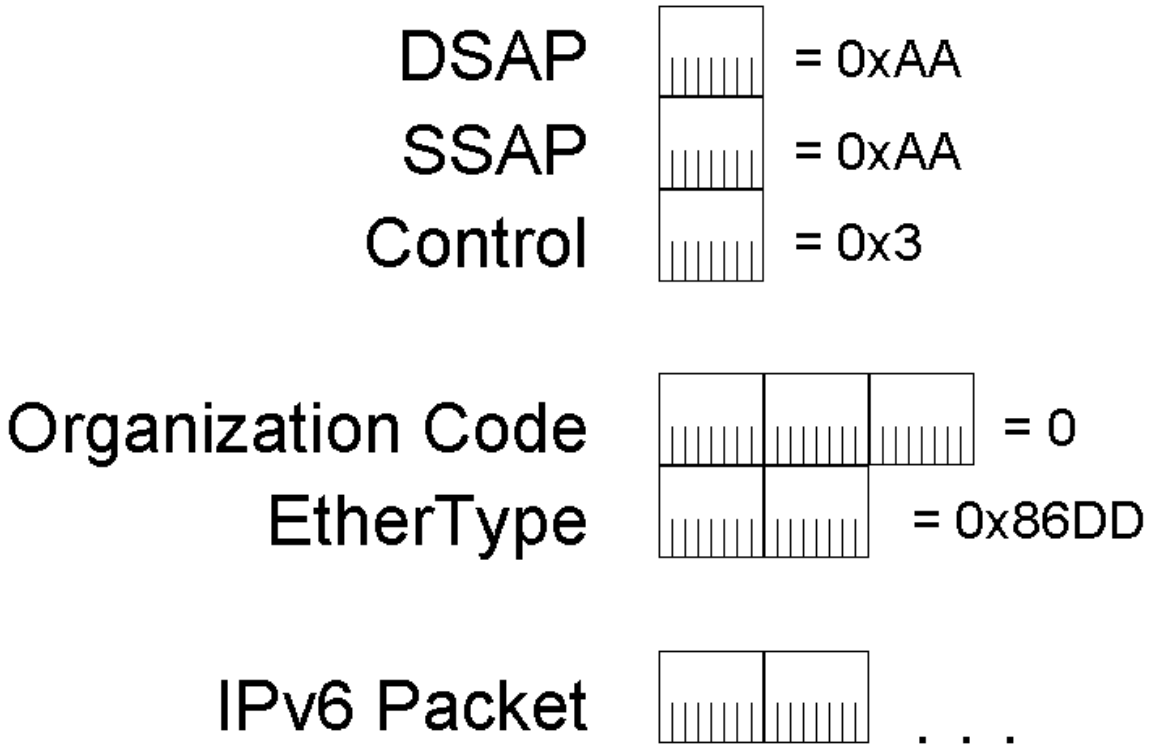


IPv6 – Ethernet II Enkapszuláció



- **EtherType field: 0x86DD (IPv4 esetén 0x800).**
- **Minimális IPv6 csomagméret: 46 byte**
- **Maximális IPv6 csomagméret: 1,500 byte**

IPv6 – IEEE 802.3 Enkapszuláció



- **Sub-Network Access Protocol (SNAP) header**
- **EtherType field: 0x86DD**
- **Minimális IPv6 csomagméret: 38 byte**
- **Maximális IPv6 csomagméret: 1,492 byte**

IPv6 – Címzés: Format Prefix (FP)

Allocation	Prefix (Binary)	Fraction of Address Space	
Reserved	0000 0000	1/256	
Unassigned	0000 0001	1/256	
Reserved for NSAP Allocation	0000 001	1/128	
Reserved for IPX Allocation	0000 010	1/128	
Unassigned	0000 011	1/128	
Unassigned	0000 1	1/32	
Unassigned	0001	1/16	
Aggregatable Global Unicast Addresses	001	1/8	
Unassigned	010	1/8	
Unassigned	011	1/8	
Unassigned	100	1/8	
Unassigned	101	1/8	
Unassigned	110	1/8	
Unassigned	1110	1/16	
Unassigned	1111 0	1/32	
Unassigned	1111 10	1/64	
Unassigned	1111 110	1/128	
Unassigned	1111 1110 0	1/512	
Link-Local Unicast Addresses	1111 1110 10	1/1024	FE80::/10
Site-Local Unicast Addresses	1111 1110 11	1/1024	FEC0::/10
Multicast Addresses	1111 1111	1/256	FF00::/8

IPv6 – Címzés: Jelölés

- **Preferred form (16 byte):**
  - **FEDC:BA98:7654:3210:FEDC:BA98:7654:3210**
  - **1080:0:0:0:0:8:800:200C:417A**
- **Compressed form:**
  - **1080::8:800:200C:417A**
  - **0:0:0:0:0:0:0:1 ==> ::1 (Unicast Loopback address)**
  - **FF01:0:0:0:0:0:0:42 ==> FF01::42 (Multicast address)**
  - **0:0:0:0:0:0:0:0 ==> :: (The unspecified address)**

IPv6 – Címzés: Kompatibilis címek

- **IPv4-kompatibilis cím**
  - **0:0:0:0:0:0:193.6.5.73 ==> ::193.6.5.73**
  - **0:0:0:0:0:0:w.x.y.z ==> ::w.x.y.z**
  - **Csak akkor, ha IPv4/IPv6 dual stack.**
  - **Ha IPv4-kompatibilis címet adnak meg úgy, mint egy IPv6 cél címet, akkor az IPv6 forgalom automatikusan IPv4 fejrészt kap és az IPv4 hálózaton küldik a cél felé.**
- **IPv4-mapped address**
  - **0:0:0:0:0:FFFF:193.6.5.73 ==> ::FFFF:193.6.5.73**
  - **Csak belső reprezentáció, senki sem küld ilyet.**
  - **Az IPv6 node jelöli így a csak IPv4 node-ot**

IPv6 – Címzés: Kompatibilis címek

- **6to4 cím**
  - **2002::/16 cím 32 bites publikus IPv4 node címmel, egy 48-bites prefixet alkot.**
  - **Pl: 193.6.5.73 esetén (Hexában: C1.6.5.49) 2002:C106:0549::/48**
  - **Két IPv4 és IPv6 dual stack node között használják, ha azok IPv4 routing infrastruktúra felett kommunikálnak.**
  - **A 6to4 egy RFC 3056 szerinti tunnel technika.**
- **Az IPv6 nem használ maszkot, csak prefixet.**

IPv6 – Címzés: Cím típusok

Unicast:

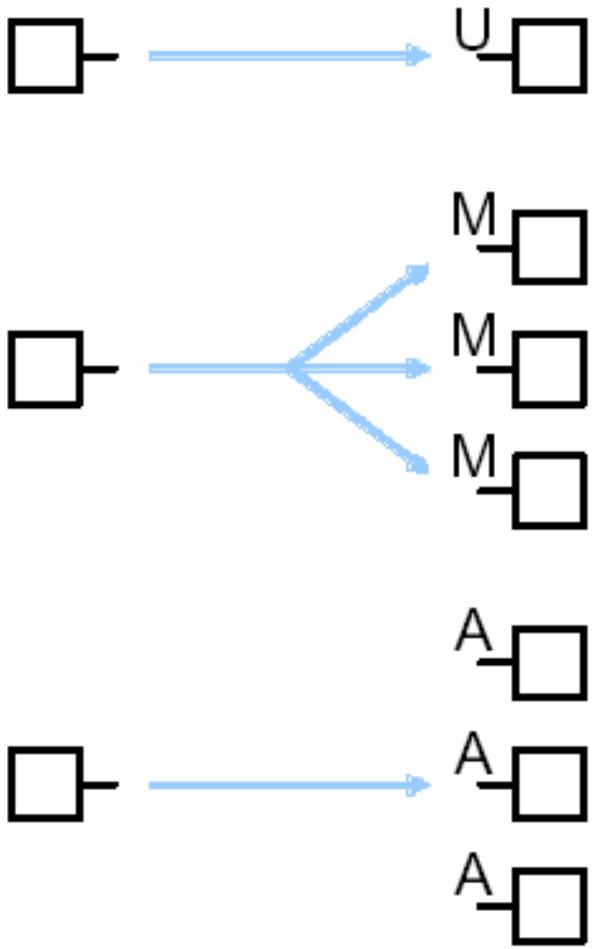
for one-to-one communication

Multicast:

for one-to-many communication

Anycast:

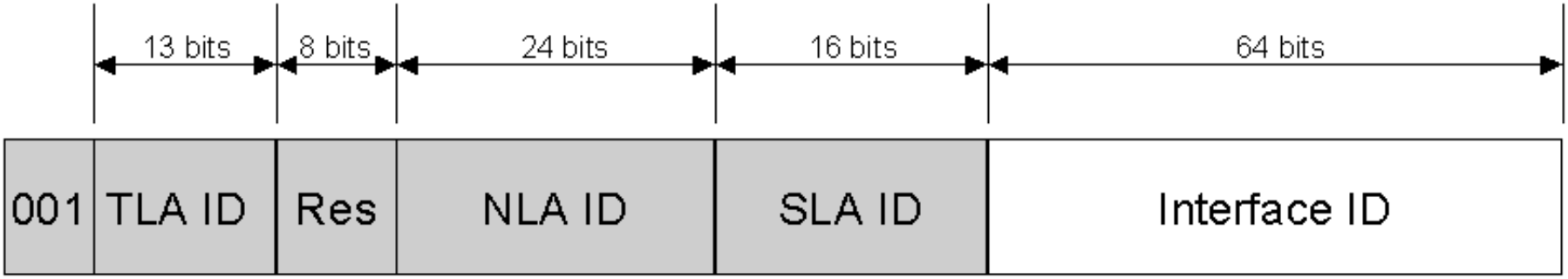
for one-to-nearest communication



IPv6 – Címzés: Unicast címek

- **Aggregatable global unicast addresses**
- **Link-local addresses**
- **Site-local addresses**
- **Special addresses**
  (loopback, unspecified compatible)

IPv6 – Aggregatable global unicast addresses



- **TLA ID – Top-Level Aggregation Identifier**
  - **Highest level in the routing hierarchy (called default-free routers)**
  - **Administered by IANA for long haul Internet service providers assigned to the routing region**
- **Res – Bits that are reserved for future use**
- **NLA ID – Next-Level Aggregation Identifier.**
  - **Az intézmény azonosítására szolgál.**
- **SLA ID – Site-Level Aggregation Identifier.**
  - **Az SLA ID az intézményen belüli alhálózatok azonosítására szolgál.**
- **Interface ID – Egy alhálózaton belül az interface-t azonosítja.**

IPv6 – Miskolci Egyetem Hbone 6Net

- **2001:0738:6001::/48**

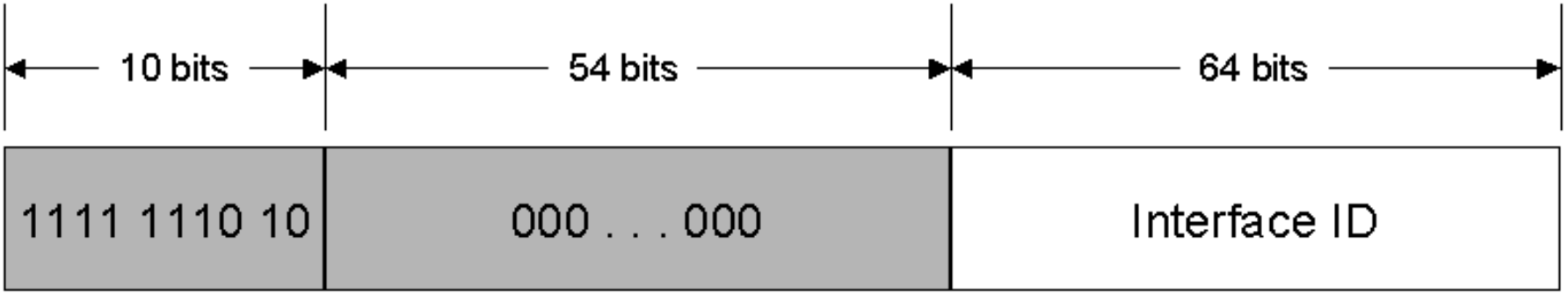
inet6num: 2001:0738:6001::/48
netname: UNI-MISKOLC
descr: University of Miskolc
descr: Miskolci Egyetem
descr: H-3515 Miskolc Egyetemvaros
country: HU
admin-c: LB18-RIPE
tech-c: SK38-RIPE
tech-c: NB12-RIPE
status: ASSIGNED
remarks: ourid=445
mnt-by: NIIF6-MNT
source: RIPE
changed: hostmaster6@iif.hu 20030103

- **person: Laszlo Balla**
  address: University of Miskolc
  address: Miskolci Egyetem
  address: Miskolc Egyetemvaros
  address: H-3515 MISKOLC-Egytemvaros
  address: Hungary
  phone: +36 46 565111 ext. 1012
  fax-no: +36 46 363450
  e-mail: szkballa@uni-miskolc.hu
  nic-hdl: LB18-RIPE
  changed: hostmaster@iif.hu 19920705
  changed: hostmaster@iif.hu 20020103
  source: RIPE

  person: Szilveszter Kovacs
  address: University of Miskolc
  address: Miskolci Egyetem
  address: H-3515 Miskolc-Egyetemvaros
  address: Hungary
  phone: +36 46 565111 ext. 2108
  fax-no: +36 46 563450
  e-mail: szkszilv@uni-miskolc.hu
  nic-hdl: SK38-RIPE
  notify: hostmaster@iif.hu
  changed: szkszilv@uni-miskolc.hu 19960502
  changed: hostmaster@iif.hu 20020103
  source: RIPE

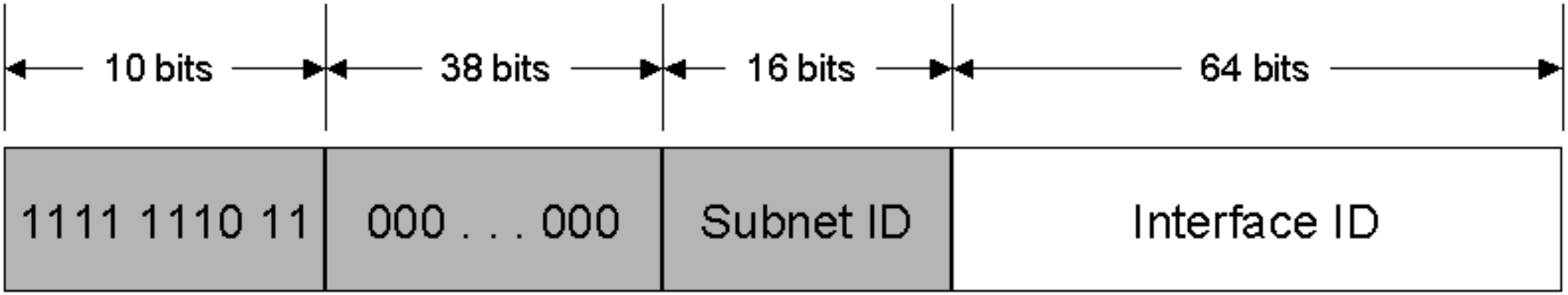
  person: Norbert Burmeister
  address: University of Miskolc
  address: Miskolci Egyetem
  address: H-3515 Miskolc-Egyetemvaros
  address: Hungary
  phone: +36 46 565111 ext. 1070
  fax-no: +36 46 563450
  e-mail: szkburma@uni-miskolc.hu
  nic-hdl: NB12-RIPE
  notify: hostmaster@iif.hu
  changed: hostmaster@iif.hu 19920705
  changed: hostmaster@iif.hu 20020103
  source: RIPE

IPv6 – Link-Local unicast addresses



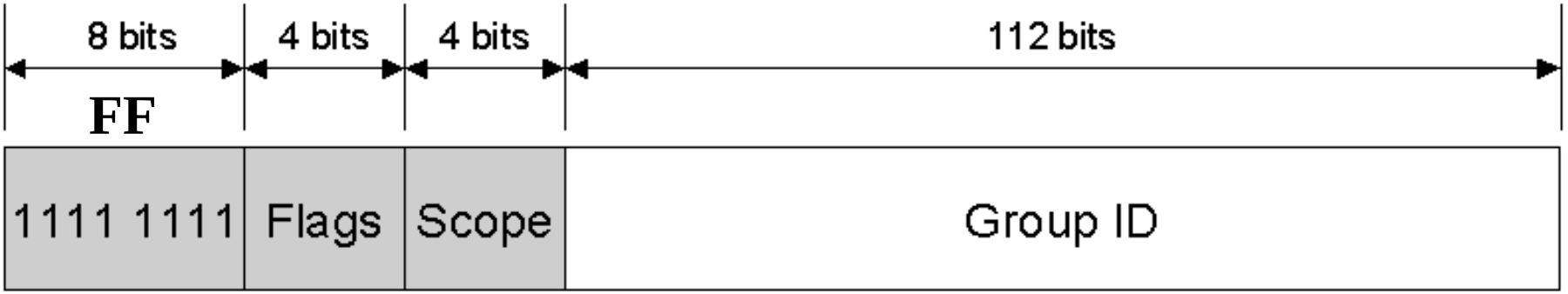
- **A link-local címek a Neighbor Discovery eljáráshoz kellenek és mindig automatikusan konfigurálódnak, még akkor is, ha semmilyen más unicast cím sem létezik.**
- **A link-local címek prefix-e mindig FE80::/64**

IPv6 – Site-Local unicast addresses



- **A link-local címekkel ellentétben, a site-local címek nem automatikusan konfigurálódnak, hanem vagy állapotmentes (stateless), vagy állapot alapú (stateful) cím konfigurációval kell megadni azokat.**
- **A site-local címek esetén az első 48-bit mindig ugyanazzal a FEC0::/48 címmel kezdődik.**
- **A fix 48 bitet követi a 16-bites subnet identifier (Subnet ID field).**

IPv6 – Multicast címek

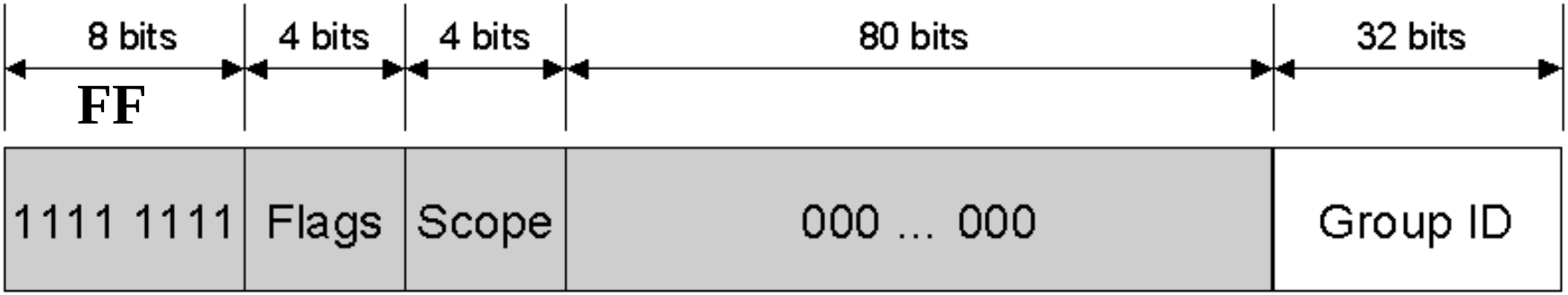


- **Flags – RFC 2373, jelenleg csak: Transient (T) flag. (legalsó bit)**
  - **0: multicast address is a permanently assigned (well-known) multicast address allocated by the Internet Assigned Numbers Authority (IANA). FF01:: - FF0F:: reserved, well-known addresses**
  - **1: transient (non-permanently-assigned) multicast address.**
- **Scope:**
  - **0: Reserved, 1: Node-local scope, 2: Link-local scope, 5: Site-local scope,**
  - **8: Organization-local scope, E: Global scope, F: Reserved**
  - **Pl. FF02::2 link-local scope. (Az IPv6 router-ek nem továbbítját)**
- **Group ID: A multicast csoport egyedi azonosítója**

IPv6 – Multicast címek

- Speciális node multicast címek:
  - FF01::1 (node-local scope all-nodes multicast address)
  - FF02::1 (link-local scope all-nodes multicast address)
- Speciális router multicast címek:
  - FF01::2 (node-local scope all-routers multicast address)
  - FF02::2 (link-local scope all-routers multicast address)
  - FF05::2 (site-local scope all-routers multicast address)

IPv6 – Multicast címek (módosítás)

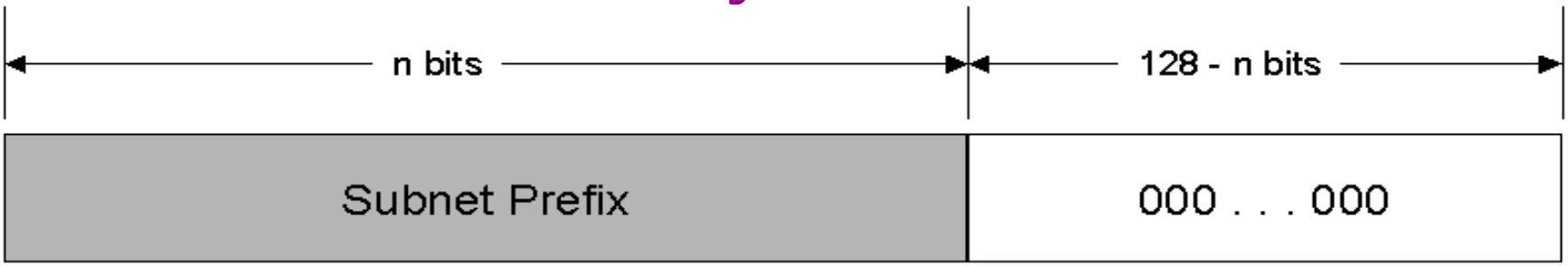


- **However, because of the way in which IPv6 multicast addresses are mapped to Ethernet multicast MAC addresses, RFC 2373 recommends assigning the Group ID from the low order 32 bits of the IPv6 multicast address and setting the remaining original group ID bits to 0.**
- **By using only the low-order 32 bits, each group ID maps to a unique Ethernet multicast MAC address.**

IPv6 – Solicited-Node Multicast Address

- A cím felfejtés (address resolution) során elősegíti a hatékony hálózati címhez tartozó adatkapcsolati cím lekérdezést.
- Az IPv6 Neighbor Solicitation üzenetet használ a cím felfejtéshez (address resolution).
- Local-link scope *all-nodes* multicast címzés helyett *solicited-node* multicast címzést használ.
- A solicited-node multicast címet az FF02::1:FF00:0/104 prefixből és a felfejtendő IPv6 cím utolsó 24-bitjéből képzi. Pl:
  - Node A link-local címe FE80::2AA:FF:FE28:9C5A valamint hallgat az ehhez tartozó solicited-node multicast címre: FF02::1:FF28:9C5A
  - Ha Node B keresi a Node A link-local címéhez FE80::2AA:FF:FE28:9C5A tartozó link-layer címet, akkor Neighbor Solicitation üzenetet küld a FF02::1:FF28:9C5A solicited node multicast címre.
  - Mivel Node A hallgat erre a címre, megválaszolja azt B-nek az unicast Neighbor Advertisement üzenettel

IPv6 – Anycast Address



- **Az anycast címeket több interfészhez is hozzá lehet rendelni.**
- **Packets addressed to an anycast address are forwarded by the routing infrastructure to the nearest interface to which the anycast address is assigned.**
- **Anycast addresses are assigned out of the unicast address space and the scope of an anycast address is the scope of the type of unicast address from which the anycast address is assigned.**
- **All router interfaces attached to a subnet are assigned the Subnet-Router anycast address for that subnet.**
- **The Subnet-Router anycast address is used for communication with one of multiple routers attached to a remote subnet**

IPv6 – Addresses for a Host

- **Egy IPv6 host-hoz az alábbi unicast címek vannak hozzárendelve:**
  - **A link-local address for each interface**
  - **Unicast addresses for each interface (which could be a site-local address and one or multiple aggregatable global unicast addresses)**
  - **The loopback address (::1) for the loopback interface**
- **Valamennyi host hallgat az alábbi multicast címekre:**
  - **The node-local scope all-nodes multicast address (FF01::1)**
  - **The link-local scope all-nodes multicast address (FF02::1)**
  - **The solicited-node address for each unicast address on each interface**
  - **The multicast addresses of joined groups on each interface.**

IPv6 – Addresses for a Router

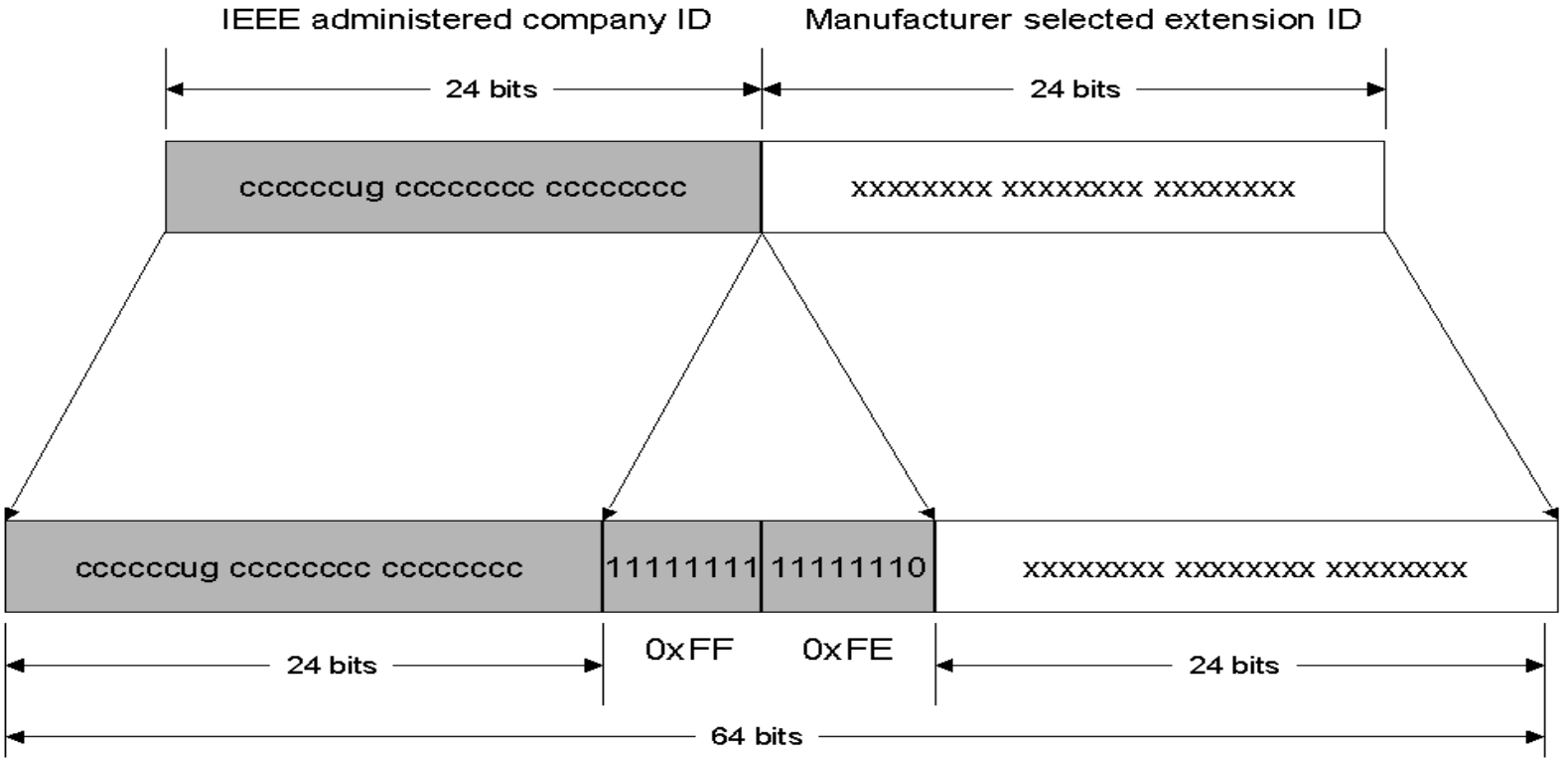
- **Egy IPv6 router az alábbi unicast címek vannak hozzárendelve:**
  - **A link-local address for each interface**
  - **Unicast addresses for each interface (which could be a site-local address and one or multiple aggregatable global unicast addresses)**
  - **A Subnet-Router anycast address**
  - **Additional anycast addresses (optional)**
  - **The loopback address (::1) for the loopback interface**
- **Valamennyi router hallgat az alábbi multicast címekre:**
  - **The node-local scope all-nodes multicast address (FF01::1)**
  - **The node-local scope all-routers multicast address (FF01::2)**
  - **The link-local scope all-nodes multicast address (FF02::1)**
  - **The link-local scope all-routers multicast address (FF02::2)**
  - **The site-local scope all-routers multicast address (FF05::2)**
  - **The solicited-node address for each unicast address on each interface**
  - **The multicast addresses of joined groups on each interface**

IPv6 – Interface ID

- **A 64 bit prefix alatti egyedi 64 bites IF cím**
- **RFC 2373: valamennyi 001-111 prefixű unicast címnek EUI-64 (IEEE) kompatibilis IF ID címének kell lennie.**

EUI-48:
(MAC)

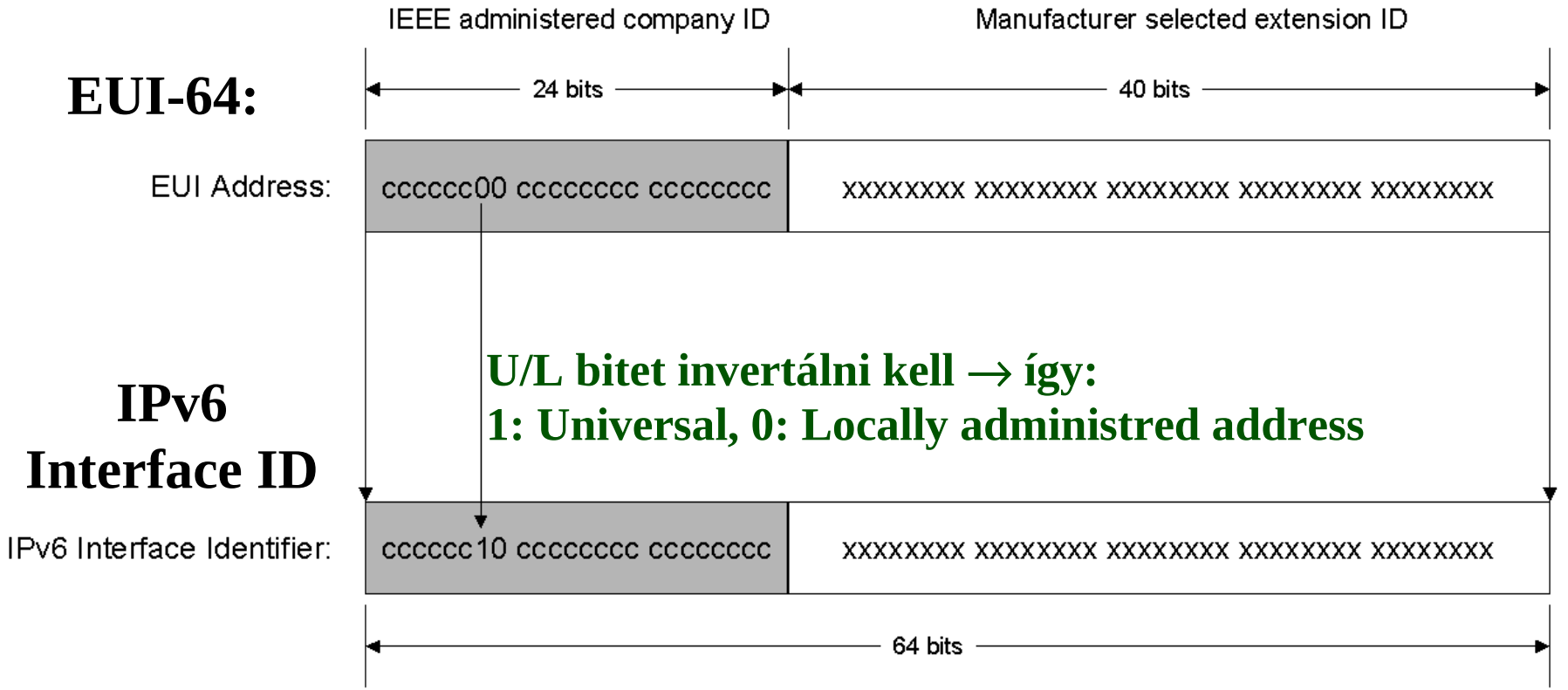
EUI-64:



- **U/L bit – 0: Universal, 1: Locally administred address**
- **I/G bit – 0: Individual (unicast), 1: Group (multicast) address**

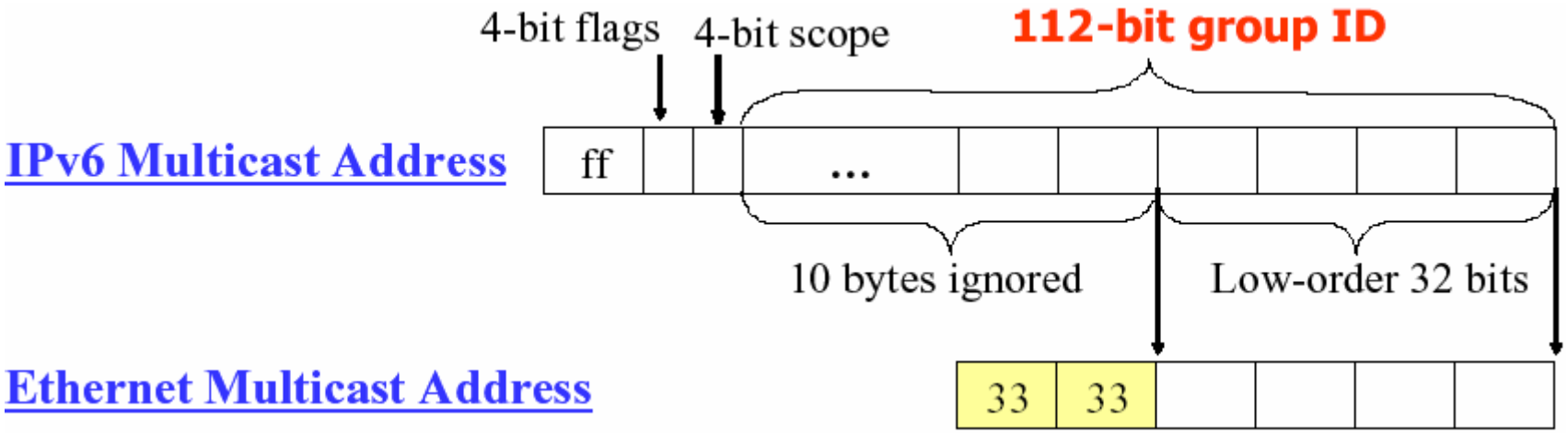
IPv6 – Interface ID

- **Az Interface ID képzése az EUI-64 alapján:**



Lokálisan adminisztrált Interface ID-k esetén tehát a 7. bitnek 0-nak kell lennie ⇒

Mapping IPv6 Multicast Addresses to Ethernet Addresses



Pl:

- **Link-local scope all-nodes multicast address of FF02::1 ⇒ 33-33-00-00-00-01**
- **Solicited-node address of FF02::1:FF3F:2A1C ⇒ 33-33-FF-3F-2A-1C**
  - **Remember that the solicited-node address is the prefix FF02::1:FF00:0/104 and the last 24-bits of the unicast IPv6 address**

IPv6 – Header tervezési megfontolások

- **Felismerhető, egyszerűsített fejrész formátum.**
- **Csökkentse a gyakori esetek csomag-feldolgozási költségeit.**
- **A címmező méretének növekedése ellenére a fejrész overhead maradjon alacsony.**
- **Támogassa a rugalmasan bővíthető fejrész opciók használatát.**
- **A 64-bites feldolgozási architektúrára legyen optimalizálva** (Headers are 64-bit aligned)

IPv6 – Header – forma

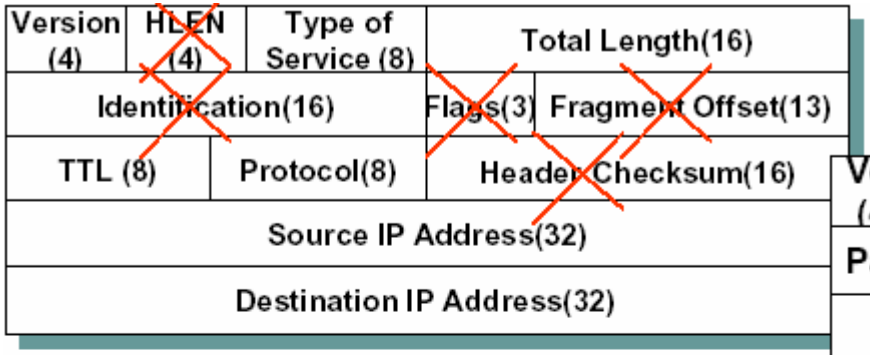
- **Fix méretű IPv6 Header**
  - **Az IPv4-el ellentétben – az opciók nincsenek 40 byte-ra korlátozva**
- **Az alap fejrészben kevesebb mező van**
  - **Faster processing of basic packets**
- **64-bitre illesztett fejrész/opciók**
- **Hatékony opció feldolgozás**
  - **Az opció mezőket csak akkor kell feldolgozni, ha léteznek.**
  - **Processing of most options limited performed only at destination**

IPv6 – Header – feldolgozási sebesség

- **A Network Layer-ből eltűnik az ellenőrző összeg**
  - **Az adatkapcsolati réteg megbízhatóbbá vált**
  - **A felsőbb rétegekben kötelező a hibaellenőrzés**
    Pl: TCP, UDP, ICMPv6
- **A hálózatban nincs további csomag fregmentáció**
  - **Csökkenti a routerek terhelését**
  - **Egyszerűbb hardver implementáció**
  - **Könnyű Layer 3 switching of IP**
- **Minimum link MTU is 1280 bytes**
  - **Az IPv4-ben ez 68 byte volt.**

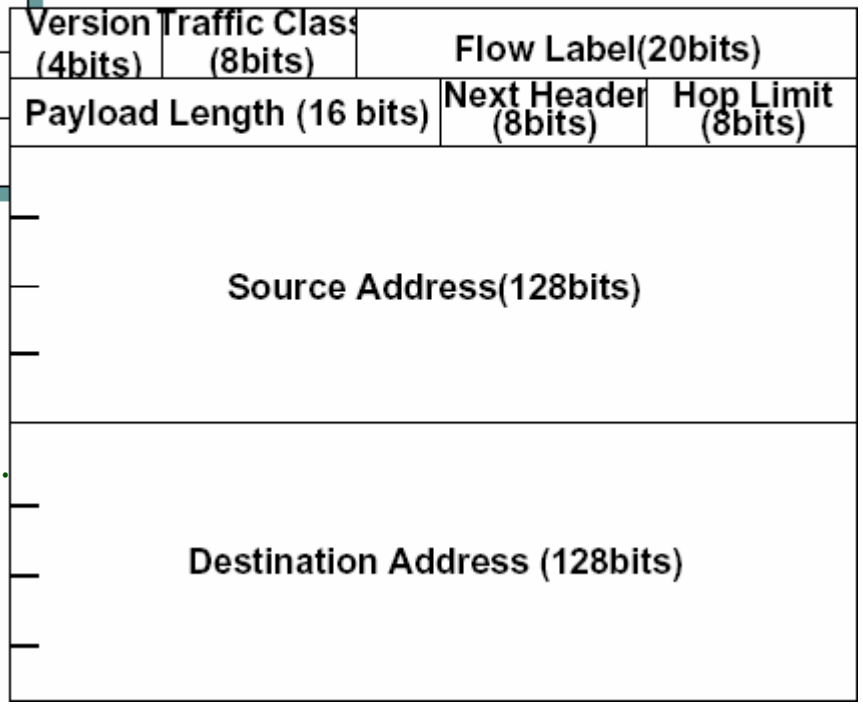
IPv6 – Basic Header (RFC 2460)

IPv4



IPv6

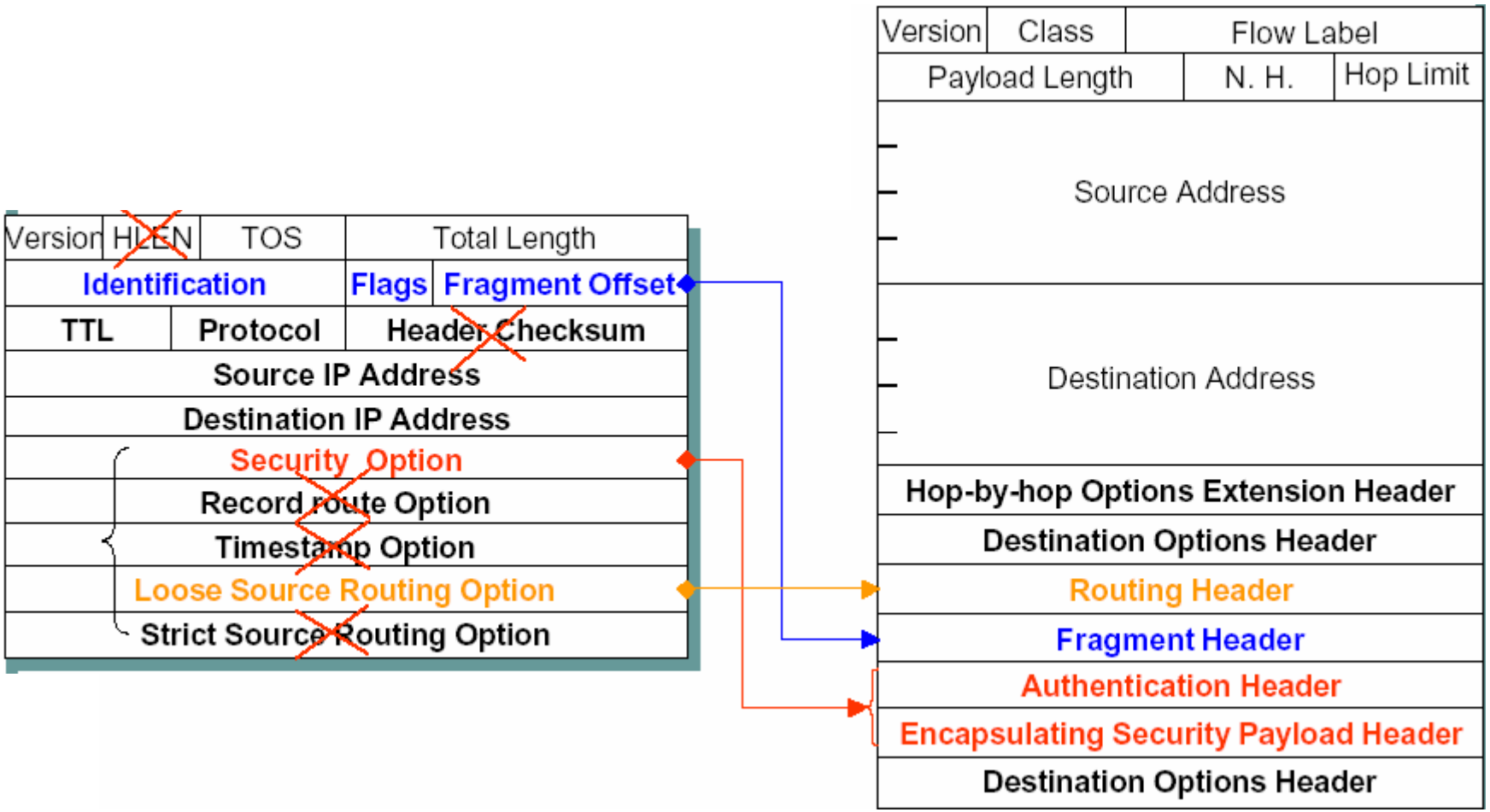
20 bytes ⇒ 40 bytes



- **Version: 0110, azaz 6**
- **Traffic Class: IPv6 Class of Priority**
  (mint IPv4 TOS), még nincs definiálva
- **Flow label: adatfolyam azonosító**
  For non-default quality of service connections.
  Default: Flow Label = 0.
- **Payload Length: a csomag hasznos mérete, max. 65535 byte**
  – ha hosszabb: Payload Length = 0 és
  Jumbo Payload option in the
  Hop-by-Hop Options Extension Header.
- **Next Header: a következő extension header típusa**
- **Hop limit: ugrásszámláló (csökken, 0 eldob)**

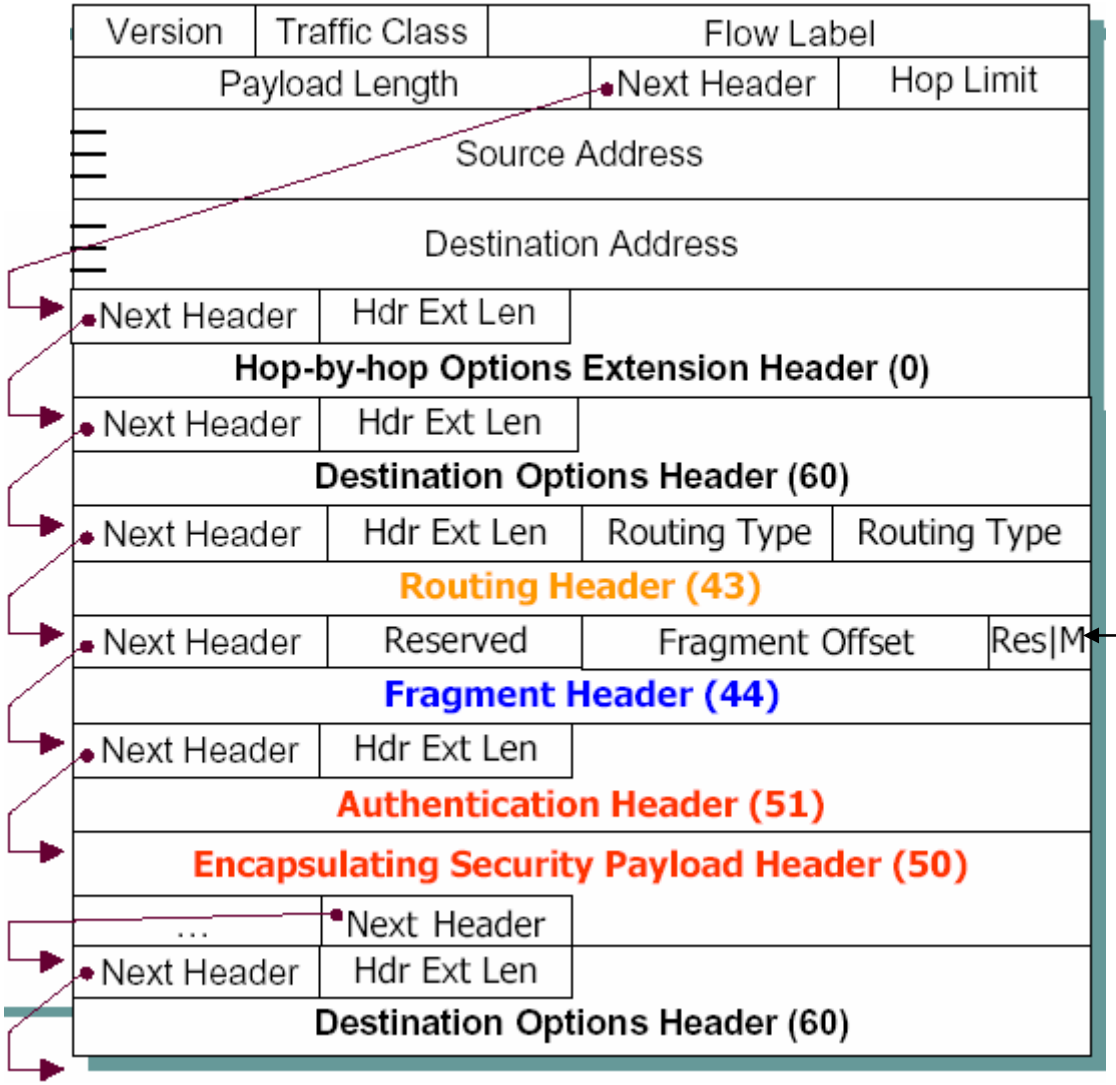
Next Header – 0:Hop-by-Hop Options Header, 6:TCP, 17:UDP, 43:Routing Header, 44:Fragment Header, 58:ICMPv6, 59:No next header, 60:Destination Options Header

IPv6 – Extension Headers (RFC 2460)



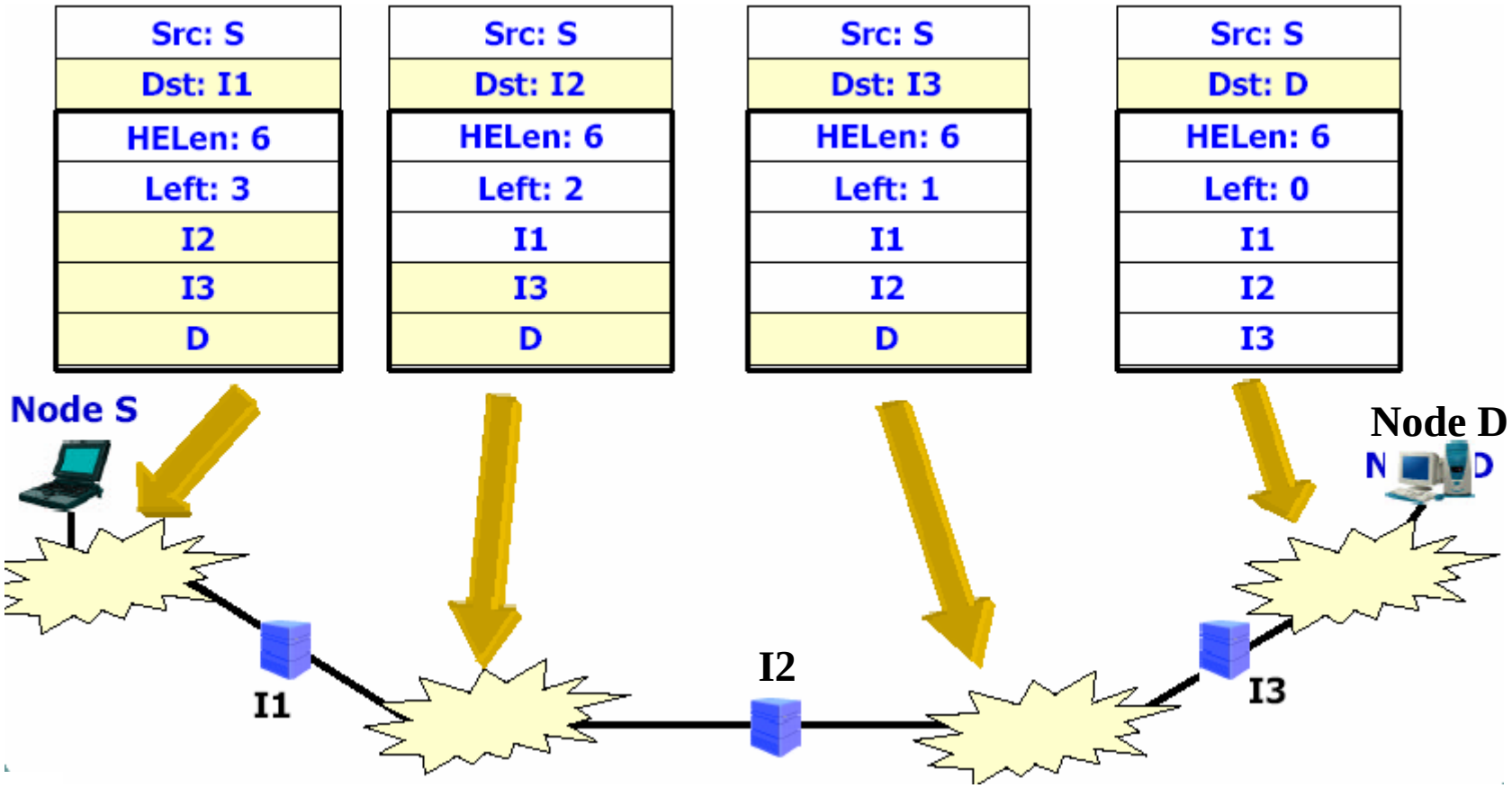
- **Delivery and forwarding options are moved to extension headers.**
- **The only extension header that must be processed at each intermediate router is the Hop-by-Hop Options extension header**

IPv6 – Extension Headers (RFC 2460)



- **Minden Extension Header-ben van egy Next Header, ami a következő típusa.**

IPv6 – Extension Headers (pl. Routing Header)



- **A Routing extension header használható a loose source route meghatározására (az útvonal listája a célig).**

IPv6 – Extension Headers (RFC 2460)

- **Hop-by-hop options header (type:0)**
  - **jumbo payload (csomagméret > 65535)**
    → az eredeti Payload Lenght:0 – helyette a kiegészítő fejlécben 32 bit hossz (max 4 terabyte) hasznos csomagméret
  - **router alert: routernek szóló információ**
- **Routing header**
  - **loose source routing (mezők a kívánt út IP címeinek)**
- **Fragment header**
  - **Az IPv6-ban, csak a forrás darabolhatja a küldendő adatokat (payload). If the payload submitted by the upper layer protocol is larger than the link or path MTU, then IPv6 fragments the payload at the source and uses the Fragment extension header to provide reassembly information.**

IPv6 – ICMPv6

- IPv6 does not provide facilities for reporting errors.
- Instead, IPv6 uses Internet Control Message Protocol version 6 (ICMPv6).
- Error Messages:
  - Destination Unreachable
  - Packet Too Big
  - Time Exceeded
  - Parameter Problem
- Informational Messages:
  - Echo Request/Reply
  - De nincs forrásfolytás (Source Quench)
- Multicast Listener Discovery (MLD)
- Neighbor Discovery (ND)

ICMPv6 - Path MTU Discovery

- The sending node assumes that the path MTU is the link MTU of the interface on which the traffic is being forwarded.
- The sending node sends IPv6 packets at the path MTU size.
- If a router on the path is unable to forward the packet over a link with a link MTU that is smaller than the size of the packet, it discards the IPv6 packet and sends an ICMPV6 Packet Too Big message back to the sending node. The ICMPV6 Packet Too Big message contains the link MTU of the link on which the forwarding failed.
- The sending node sets the path MTU for packets being sent to the destination to the value of the MTU field in the ICMPv6 Packet Too Big message.

ICMPv6 - Multicast Listener Discovery

- MLD is a set of messages exchanged by routers and nodes, enabling routers to discover the set of multicast addresses for which there are listening nodes for each attached interface.

Multicast Listener Query:

- Used by a router to query a link for multicast listeners.
  - The General Query is used to query for multicast listeners of all multicast addresses.
  - Multicast-Address-Specific Query is used to query for multicast listeners of a specific multicast address.

Multicast Listener Report:

- Used by a multicast listener to either report interest in receiving multicast traffic for a specific multicast address or to respond to a Multicast Listener Query.

Multicast Listener Done:

- Used by a multicast listener to report that it is no longer interested in receiving multicast traffic for a specific multicast address.

ICMPv6 - Neighbor Discovery (ND)

ND is used by hosts to:

- **Discover neighboring routers.**
- **Discover addresses, address prefixes, and other configuration parameters.**

ND is used by routers to:

- **Advertise their presence, host configuration parameters, and on-link prefixes.**
- **Inform hosts of a better next-hop address to forward packets for a specific destination.**

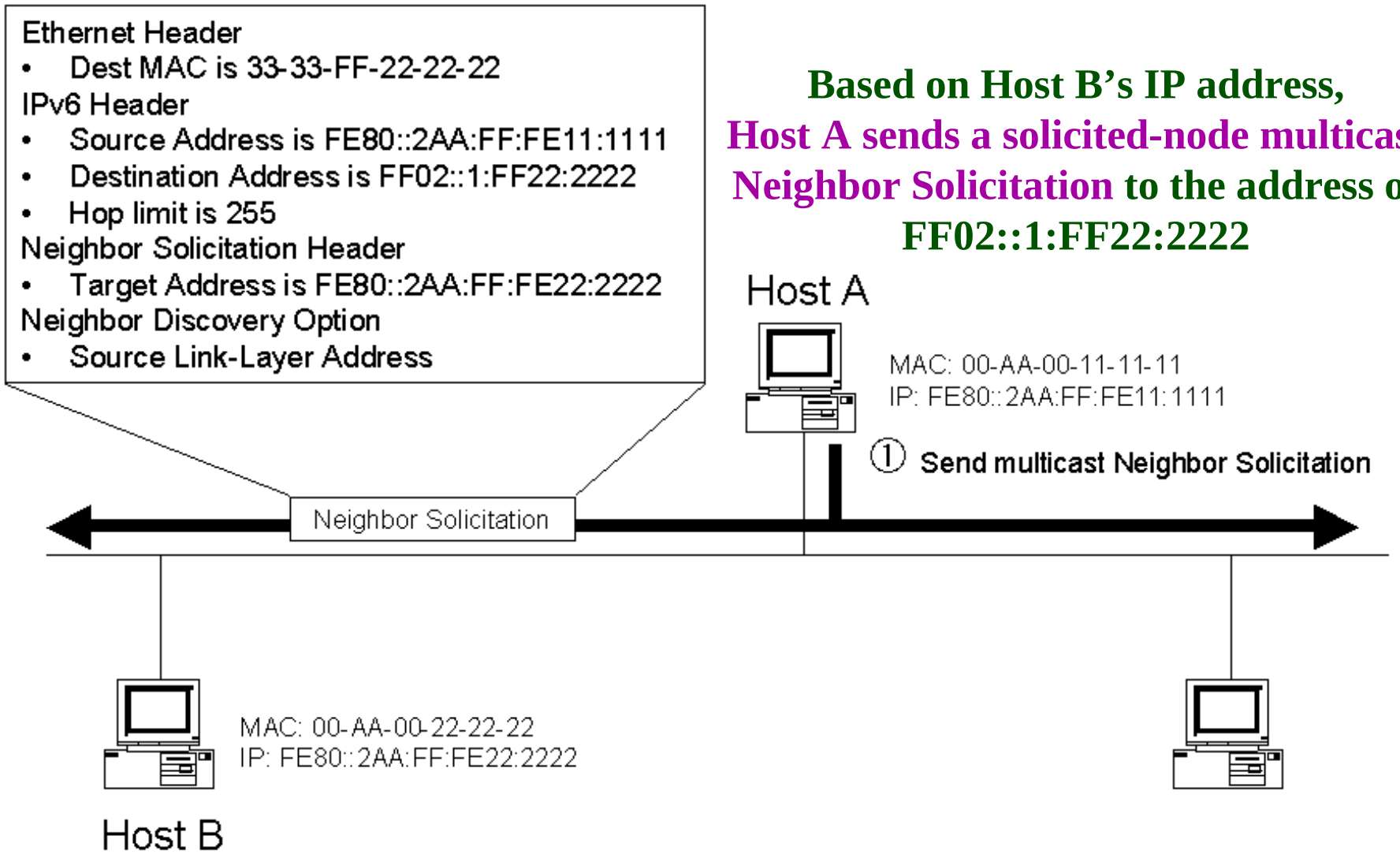
ND is used by nodes to:

- **Resolve the link-layer address of a neighboring node to which an IPv6 packet is being forwarded and determine when the link-layer address of a neighboring node has changed.**
- **Determine whether a neighbor is still reachable.**

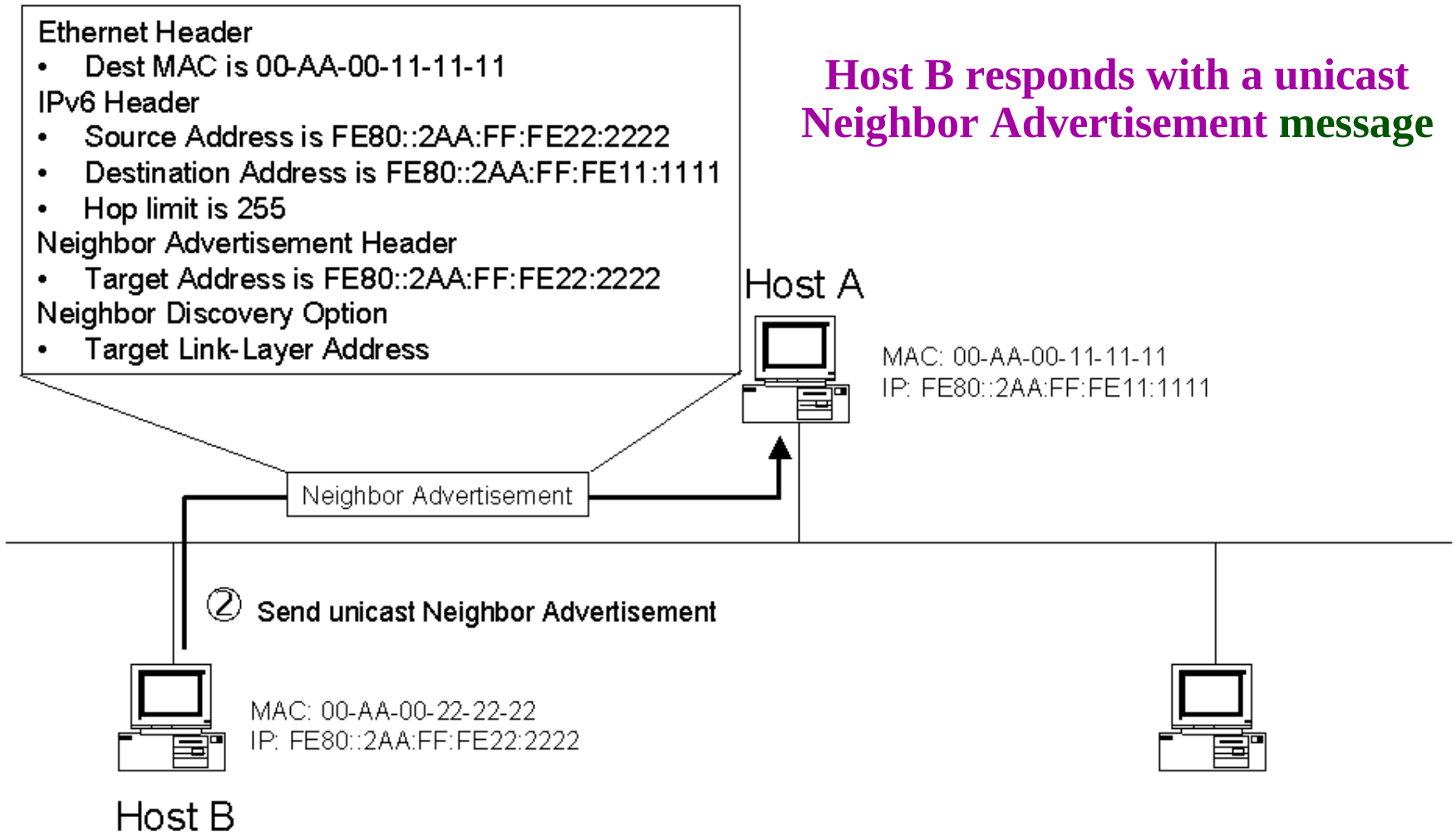
ICMPv6 - Address Resolution Example

- **Host A has an Ethernet MAC address of 00-AA-00-11-11-11 and a corresponding link-local address of FE80::2AA:FF:FE11:1111.**
- **Host B has an Ethernet MAC address of 00-AA-00-22-22-22 and a corresponding link-local address of FE80::2AA:FF:FE22:2222.**
- **To send a packet to Host B, Host A must use address resolution to resolve Host B's link-layer address.**
- **Based on Host B's IP address, Host A sends a solicited-node multicast Neighbor Solicitation to the address of FF02::1:FF22:2222**

ICMPv6 - Address Resolution Example



ICMPv6 - Address Resolution Example

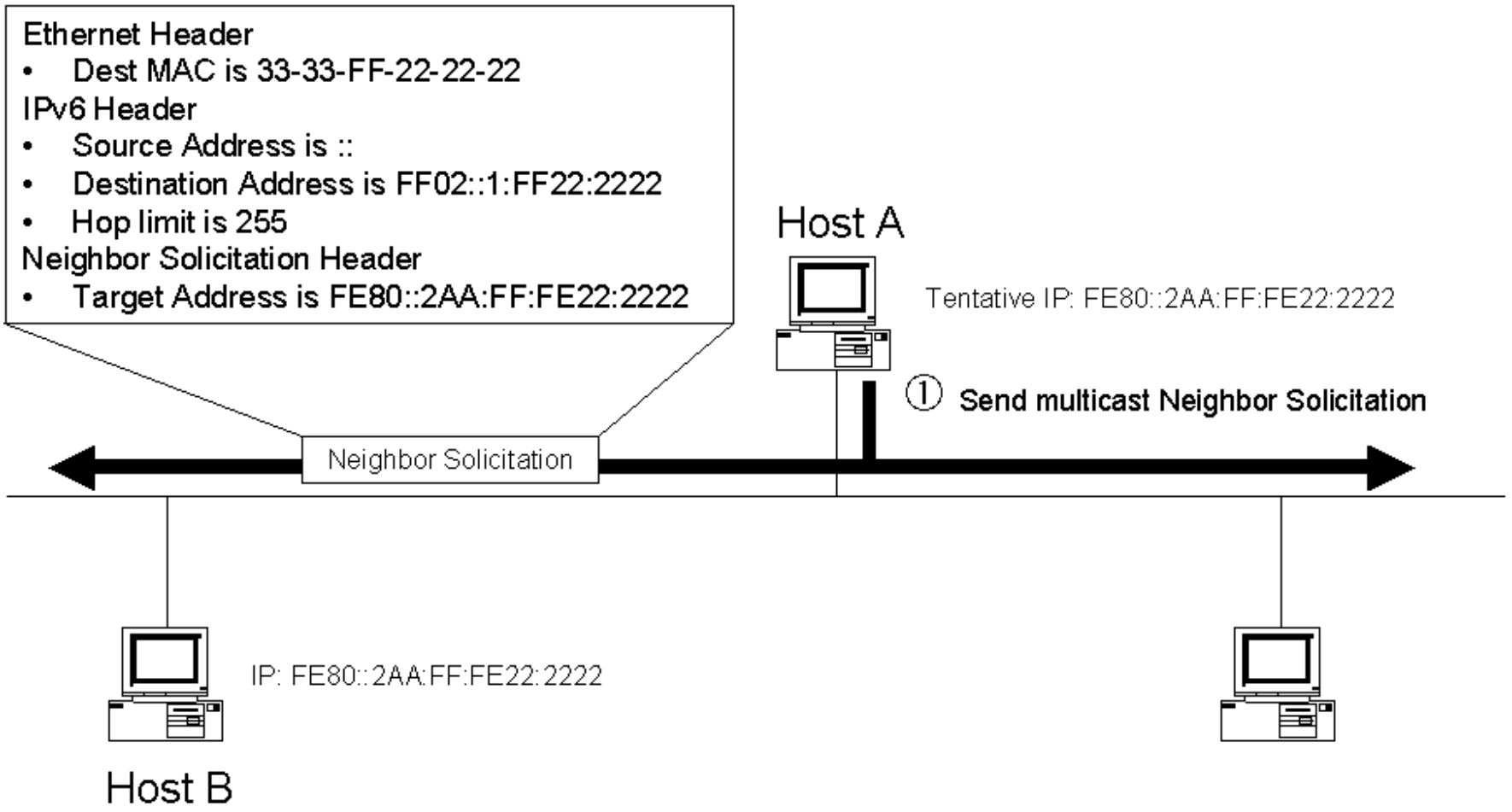


Host B responds with a unicast Neighbor Advertisement message

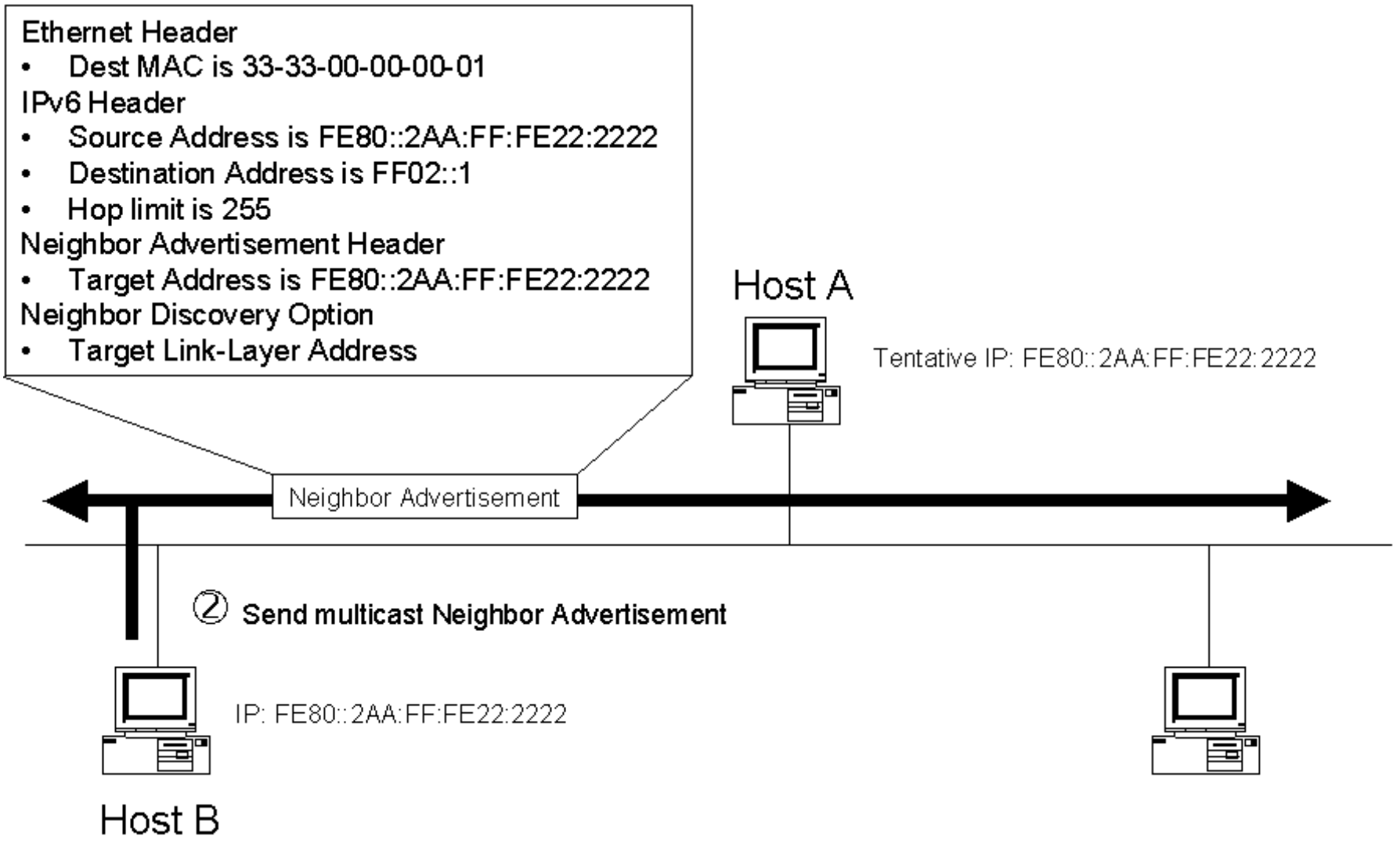
ICMPv6 - Duplicate Address Detection

- IPv6 nodes use the Neighbor Solicitation message to detect duplicate address use on the local link.
- In the duplicate address detection Neighbor Solicitation message, the Source Address field in the IPv6 header is set to the unspecified address (::).
  - The address being queried for duplication cannot be used until it is determined that there are no duplicates.
- In the Neighbor Advertisement reply to a duplicate address detection Neighbor Solicitation message, the Destination Address in the IP header is set to the link-local scope all-nodes multicast address (FF02::1).
  - The Solicited flag in the Neighbor Advertisement message is set to 0.
  - Because the sender of the duplicate address detection Neighbor Solicitation message is not using the desired IP address, it cannot receive unicast Neighbor Advertisements.
  - Therefore, the Neighbor Advertisement is multicast.
- Duplikáció esetén a Node nem használja a duplikált címet

ICMPv6 - Duplicate Address Detection Pl:



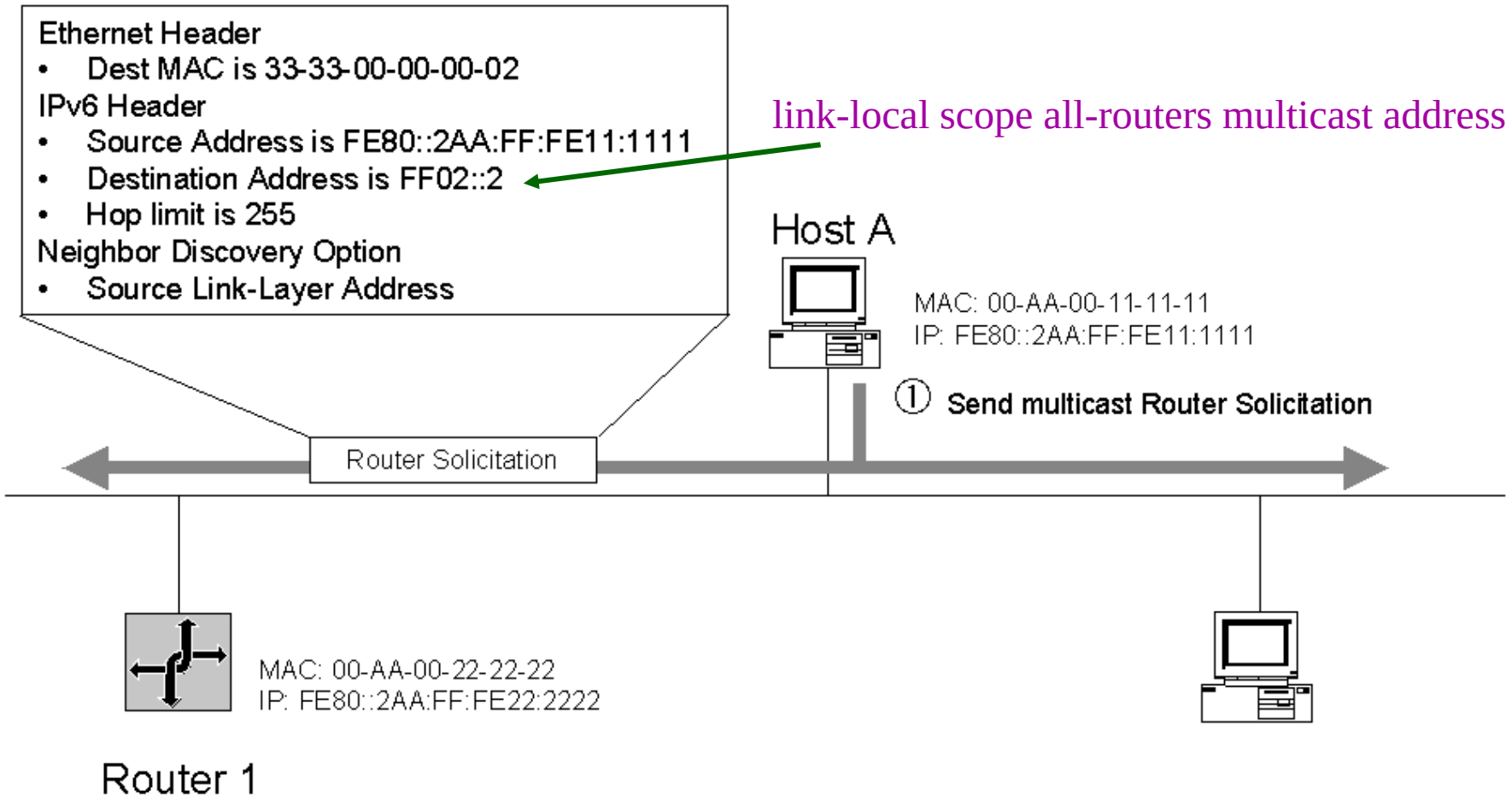
ICMPv6 - Duplicate Address Detection Pl:



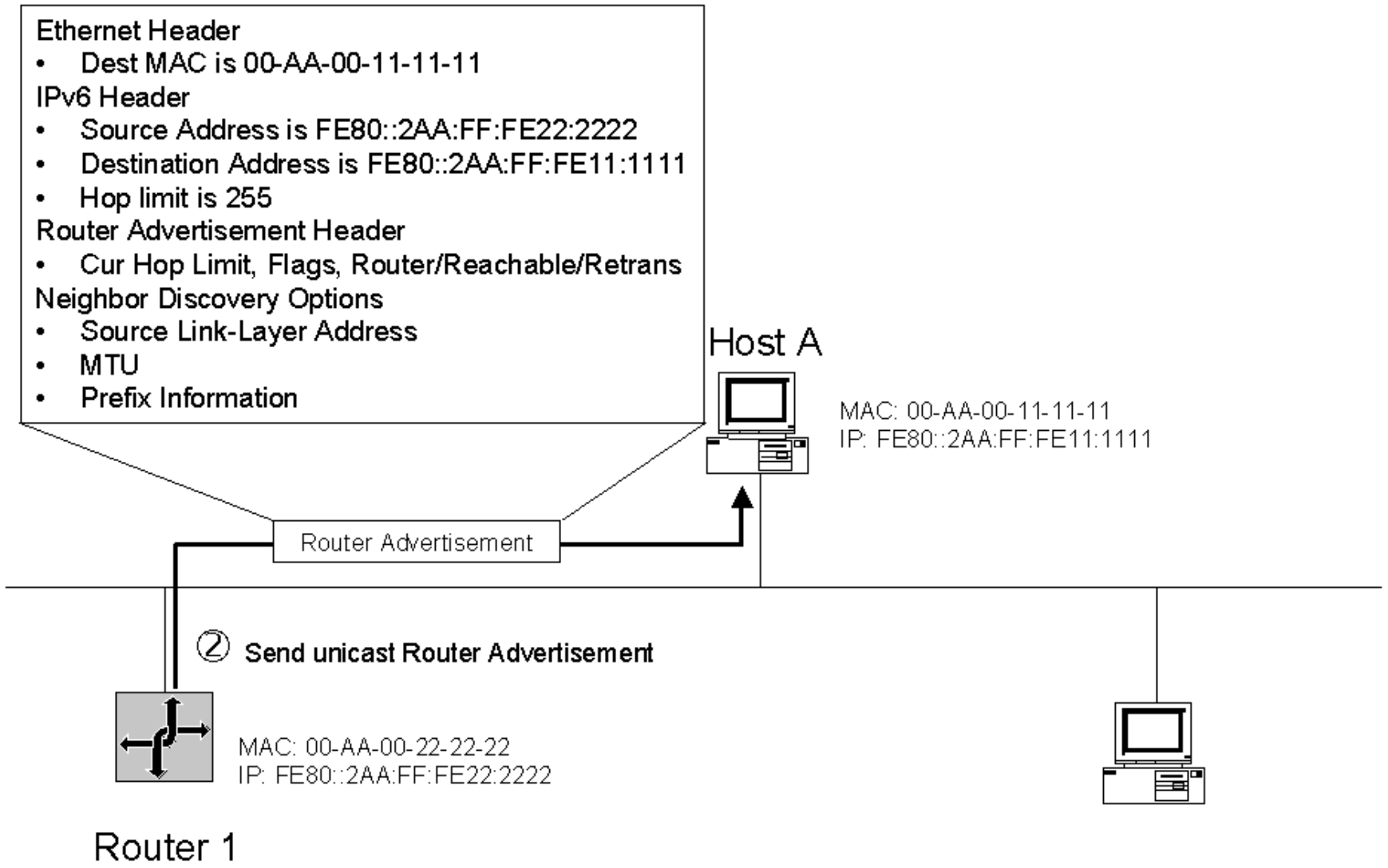
ICMPv6 - Router Discovery

- Router discovery is the process through which nodes attempt to discover the set of routers on the local link.
- IPv6 routers periodically send a Router Advertisement message on the local link advertising their existence as routers.
  - They also provide configuration parameters such as default hop limit, MTU, and prefixes.
- Active IPv6 hosts on the local link receive the Router Advertisement messages and use the contents to maintain the default router list, the prefix list, and other configuration parameters.
- A host that is starting up sends a Router Solicitation message to the link-local scope all-routers multicast address (FF02::2).
- Upon receipt of a Router Solicitation message, all routers on the local link send a unicast Router Advertisement message to the node that sent the Router Solicitation.
- The node receives the Router Advertisement messages and uses their contents to build the default router and prefix lists and set other configuration parameters.

ICMPv6 - Router Discovery Pl:



ICMPv6 - Router Discovery Pl:



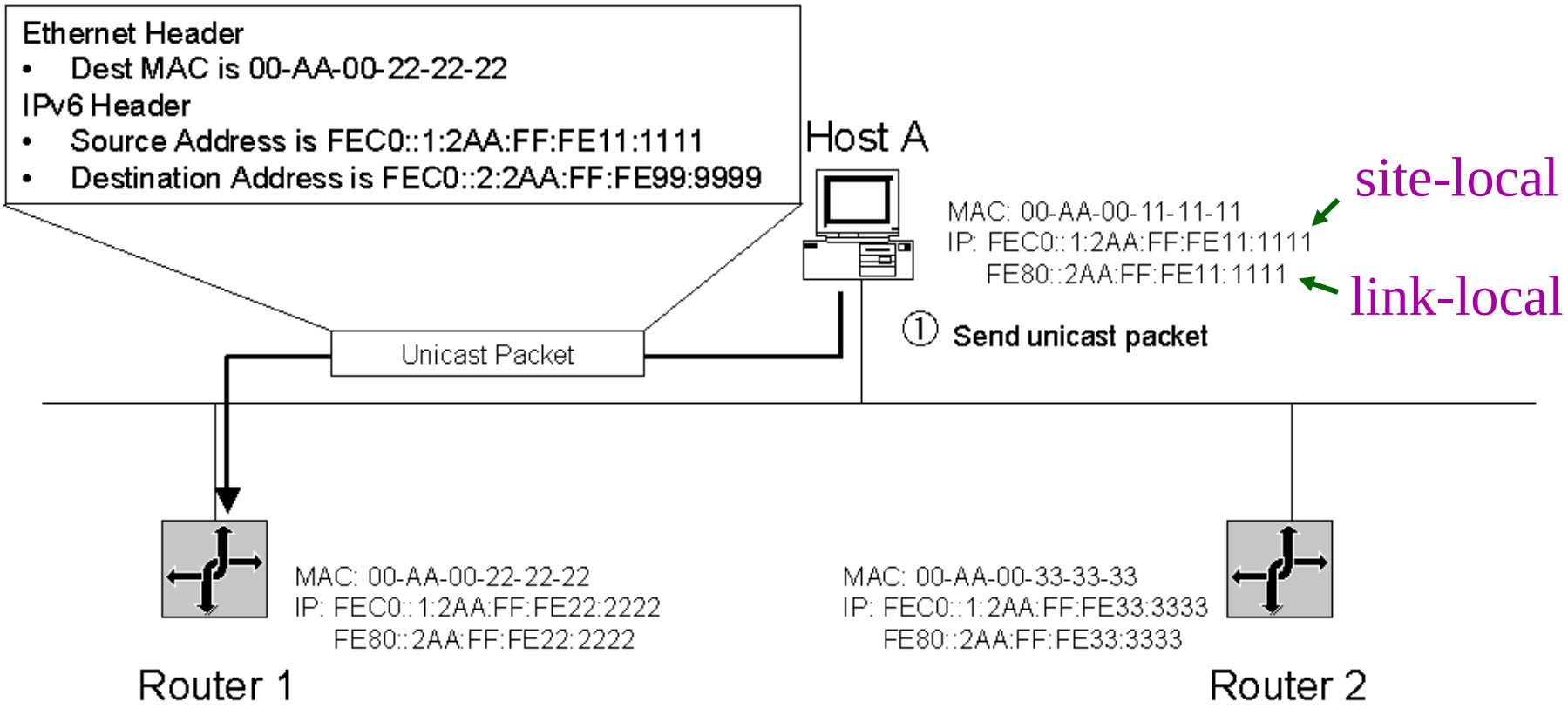
ICMPv6 – Redirect Function

- Routers use the redirect function to inform originating hosts of a better first-hop neighbor to which traffic should be forwarded for a specific destination.
- Redirect messages are only sent by the first router in the path between the originating host and the destination and like ICMPv6 error messages are rate limited.
- Hosts never send Redirect messages and routers never update routing tables based on the receipt of a Redirect message

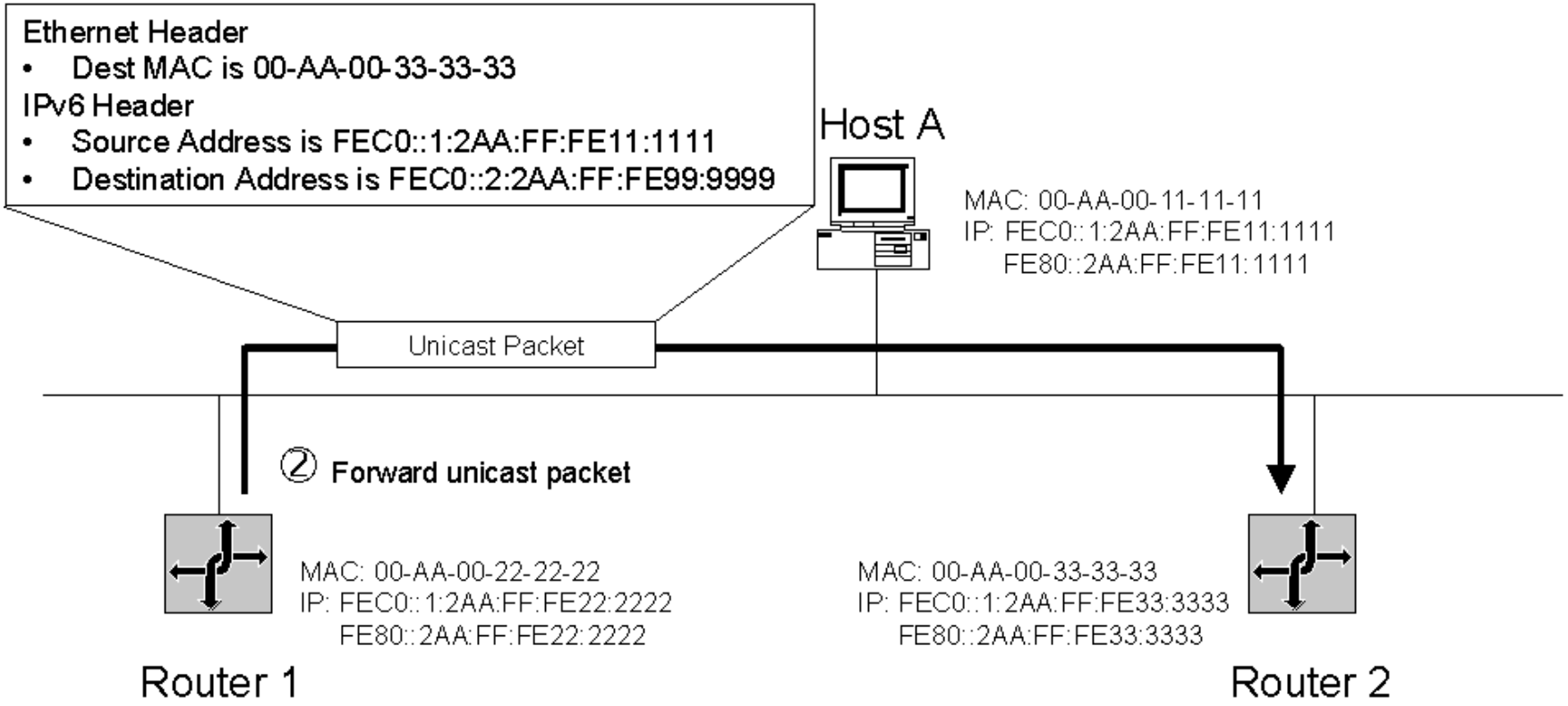
ICMPv6 – Redirect Function

- The originating host forwards a unicast packet to its default router.
- The router processes the packet and notes that the address of the originating host is a neighbor. Additionally, it notes that the addresses of both the originating host and the next-hop are on the same link.
- The router forwards the packet to the appropriate next-hop address.
- The router sends the originating host a Redirect message. In the Target Address field of the Redirect message is the next-hop address of the node to which the originating host should send packets addressed to the destination.
- For packets redirected to a router, the Target Address field is set to the link-local address of the router. For packets redirected to a host, the Target Address field is set to the destination address of the packet originally sent.
- The Redirect message includes the Redirected Header option. It might also include the Target Link-Layer Address option.
- Upon receipt of the Redirect message, the originating host updates the destination address entry in the destination cache with the address in the Target Address field. If the Target Link-Layer Address option is included in the Redirect message, its contents are used to create or update the corresponding neighbor cache entry.

ICMPv6 – Redirect Function Pl:



ICMPv6 – Redirect Function Pl:



ICMPv6 – Redirect Function Pl: