

# Hálózati Technológiák és Alkalmazások

Vida Rolland  
BME TMIT

2020. november 17.



# Distance-Vector Protokollok

Bellman-Ford protokollok

# Klasszikus Bellman-Ford algoritmus

$d_{ij}$  :=  $i$ - $j$  link költsége (végtelen, ha nincs link)

- Tényleges ár, késleltetés, csomagvesztési ráta, stb.

Következmény: **additivitás**

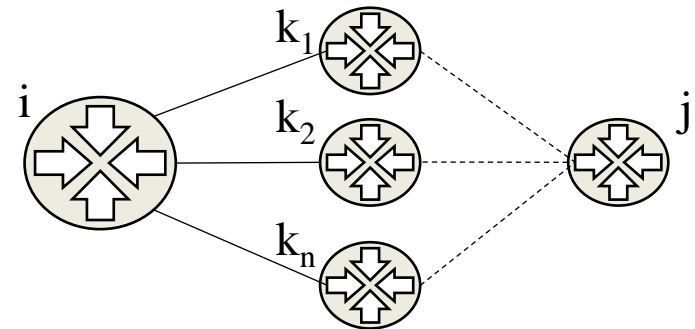
- egy útvonal költsége az azt alkotó linkek költségének összege

$D_{ij}$  := minimum költség  $i$  és  $j$  között

**Bellman egyenlet:**

$D_{ii} = 0$ , minden  $i$ -re

$D_{ij} = \min_k \{d_{ik} + D_{kj}\}$



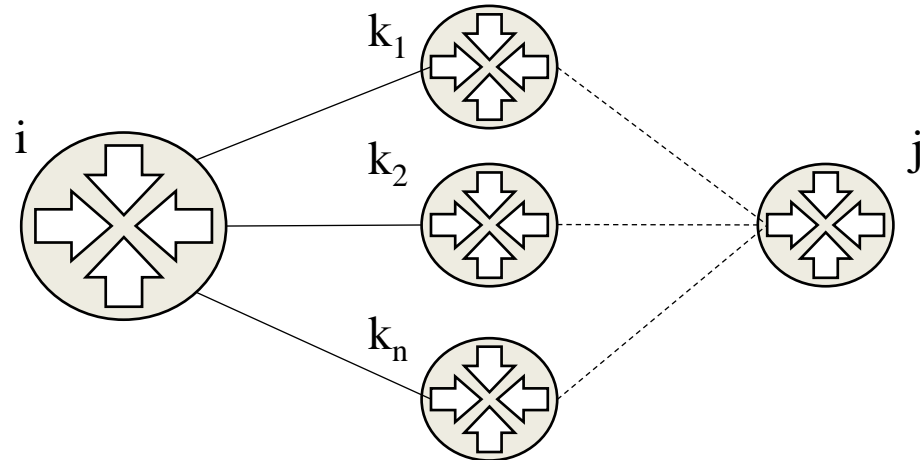
# Elosztott Bellman-Ford Algoritmus

$D_{kj}^i(t)$  = minimális távolság  $k$ -tól  $j$ -ig, melyet  $i$  router lát a  $t$  időpillanatban

$D_{ii} = 0$ , minden  $i$ -re

$D_{ij}(t) = \min_k \{ d_{ik} + D_{kj}^i(t) \}$

- önállóan működhet az algoritmus a routerekben



# Distance-vector protokollok

---

- RIPv1 (RFC 1058, '88)
  - Routing Information Protocol
    - Rest In Pieces 😊
- RIPv2 (RFC 2453, '98)
- RIPv6 (RFC 2080, '97)
  - IPv6-os verzió
  
- EIGRP
  - Enhanced Interior Gateway Routing Protocol
  - Cisco proprietary szabvány

# Distance Vector protokollok

---

- Távolságvektorokat tárol az útvonalakról
  - Adathármasok:
    - Cél (mi a célállomás)
    - Költség
    - Következő csomópont (merre küldje)
  - Rendszeresen frissítik adataikat a közvetlen szomszédok
    - Frissítő üzenet (2 részből áll):
      - Cél, költség
    - Ha a router ilyenkor jobb utat talál egy célhoz, frissíti tábláját 2 ok miatt:
      - Kisebb költségű utat talál
      - Szomszéd költsége megváltozik

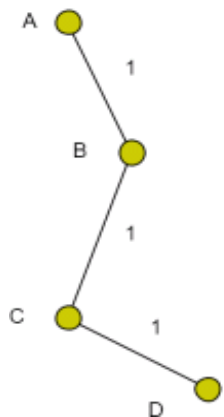
# Jellemzők

---

- Egyszerű, de nem tökéletes:
  - A kapcsolatok ára változhat
  - Kapcsolatok meg is szakadhatnak
    - Egy megszakadt kapcsolat ára végtelen
      - Egy olyan egész érték, mely nagyobb bármilyen lehetséges valós értéknél (RIP-nél jellemzően 16)
  - A routerek topológia változás esetén nem egyszerre frissítik táblázataikat
    - Periodikus időközönként (pl. 30 s) frissítő üzenet
      - Ha 6 frissítés elmarad, az ár végtelen
    - A szomszédok is frissítik a bejegyzéseiket
  - **Konvergál, de lassan**
    - Csak kis hálózatokban használható

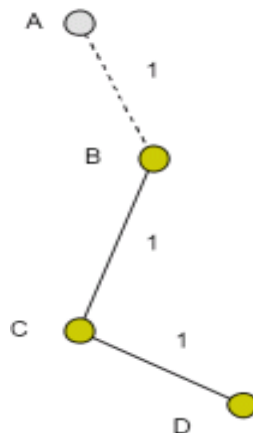
# Végtelenig számolás

- A routerek a célcím költségek hirdetésekor végtelenig inkrementálhatnak



B	C	D
1	2	3

Távolság A felé



B	C	D
3	2	3

B	C	D
3	4	3

B	C	D
5	4	5



# Megakadályozás

---

- **Split horizon módszer**
  - ha C B-től megismer egy utat, akkor az azt kiterjesztő utat B-vel már nem közli
- **Poisoned Reverse módszer**
  - Végtelen elérhetőség hirdetése az adott linken elérhető csomópontokhoz
    - ha C B-től megismer egy utat, akkor az azt kiterjesztő utat B-nek végtelen költségű útként jelzi

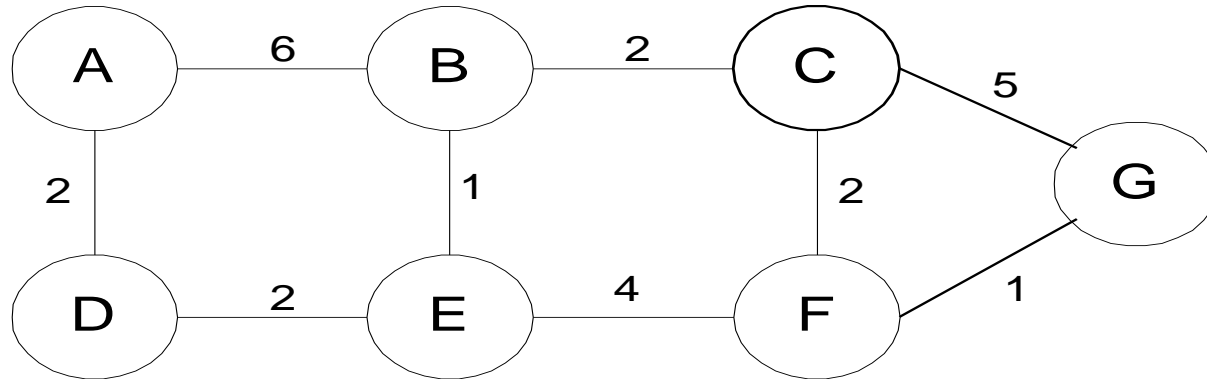
# Link-state protokollok működése

---

- A link-state protokollok működése 2 részből áll:
  1. minden állomás felderíti a hálózat topológiáját
    - Hálózati topológia leírása a link állapot leíró rekordokban található
      - Link állapot leíró rekordokat kell terjeszteni
  2. a kapott gráfban megkeresi a legrövidebb útvonalat és az ahhoz tartozó első állomást
- **Fontos!**
  - A routerekben lévő topológia megegyezzen
  - Az optimális út kiválasztása ugyanúgy történjen
    - ha A router B felé számolja az optimális utat, B meg A router felé – hurok!

# Link State Database

---

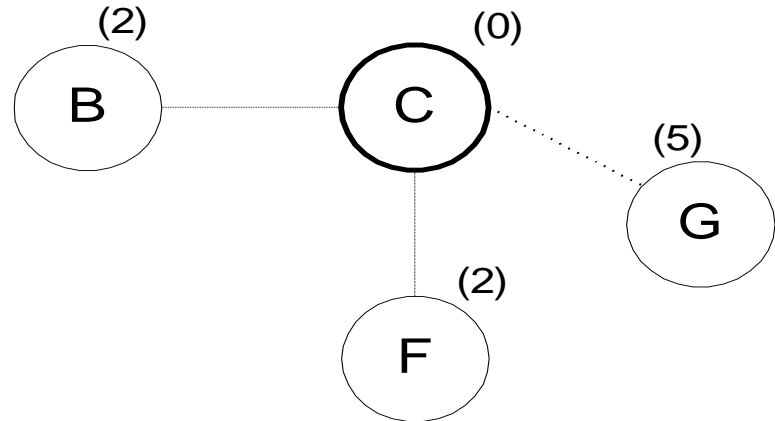
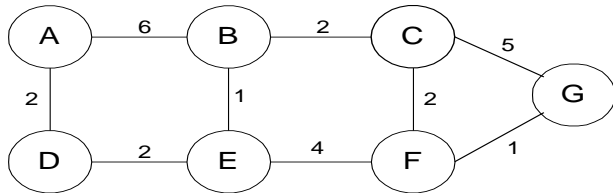


Link state Database						
A	B	C	D	E	F	G
B/6	A/6	B/2	A/2	B/1	C/2	C/5
D/2	C/2	F/2	E/2	D/2	E/4	F/1
	E/1	G/5		F/4	G/1	

# Dijkstra algoritmus

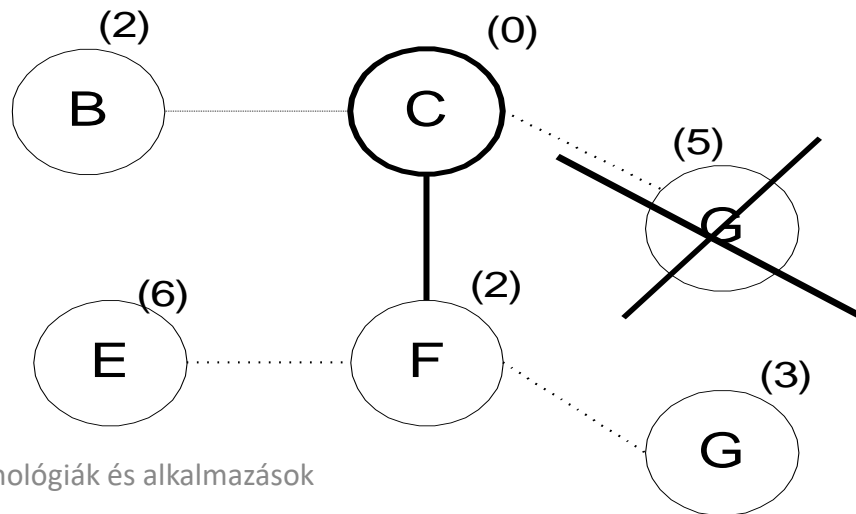
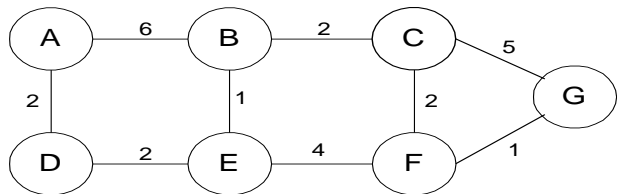
---

- Az útvonalválasztás Dijkstra algoritmus alapján
  - Legyen C a gyökér
  - Számoljuk ki a szomszédokhoz vezető utak költségét



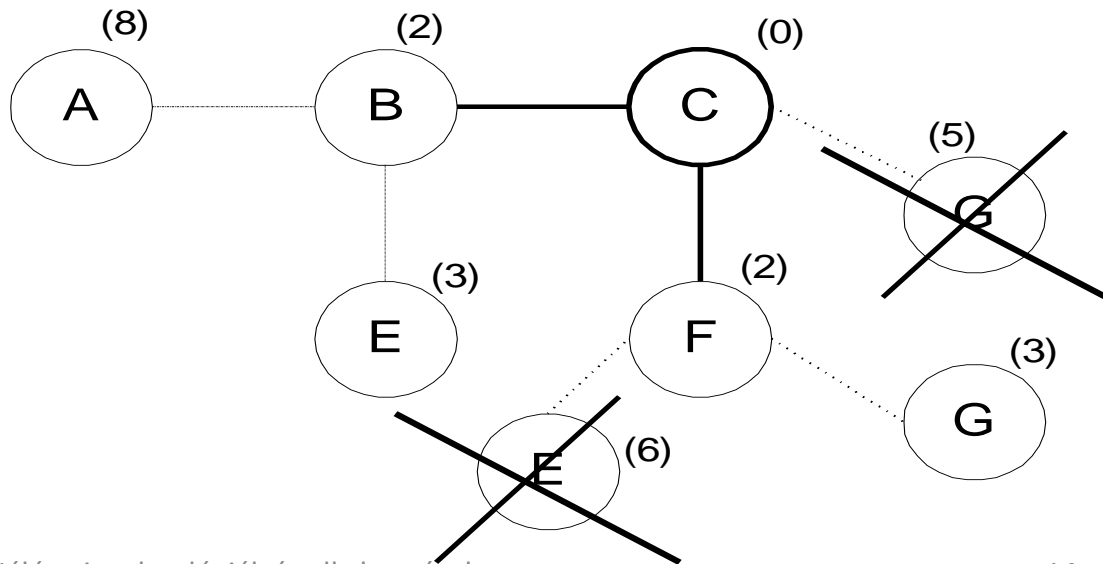
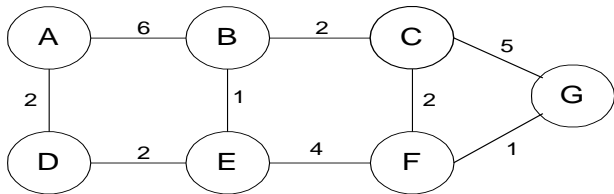
# Dijkstra algoritmus 2

- Vegyük be F-et (legkisebb költségű, még nem elemzett csomópont) és számoljuk ki F szomszédjaihoz vezető utak költségét.
- Rövidebb út G-hez F-en keresztül, E megjelenik



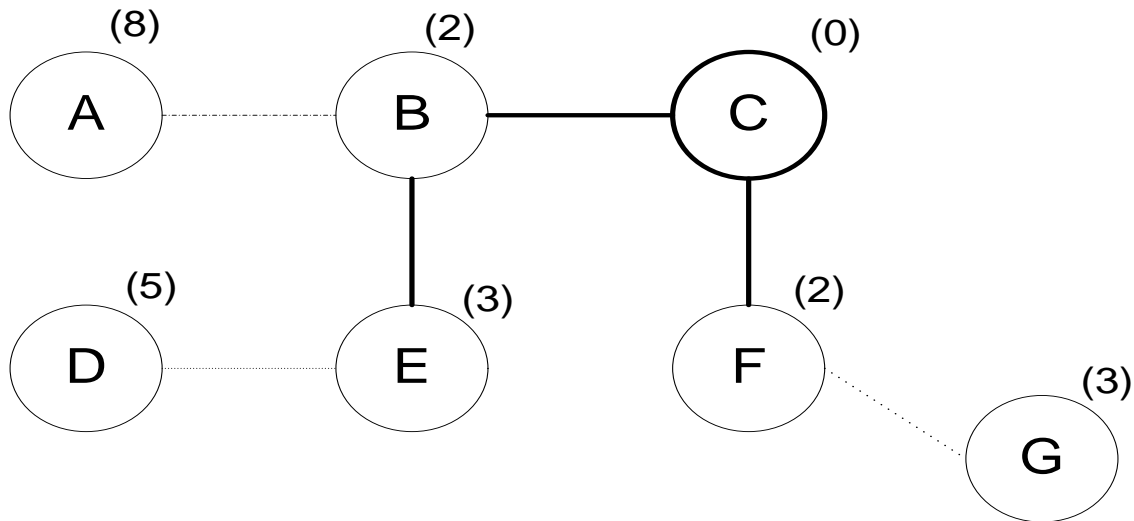
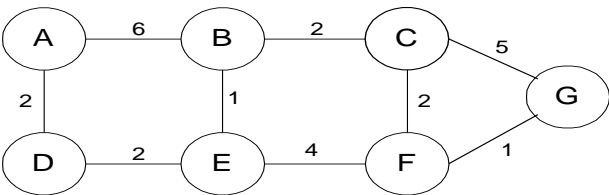
# Dijkstra algoritmus 3

- Vegyük be B-t és számoljuk ki B szomszédainak költségét
- Rövidebb út E-hez B-n keresztül, A megjelenik



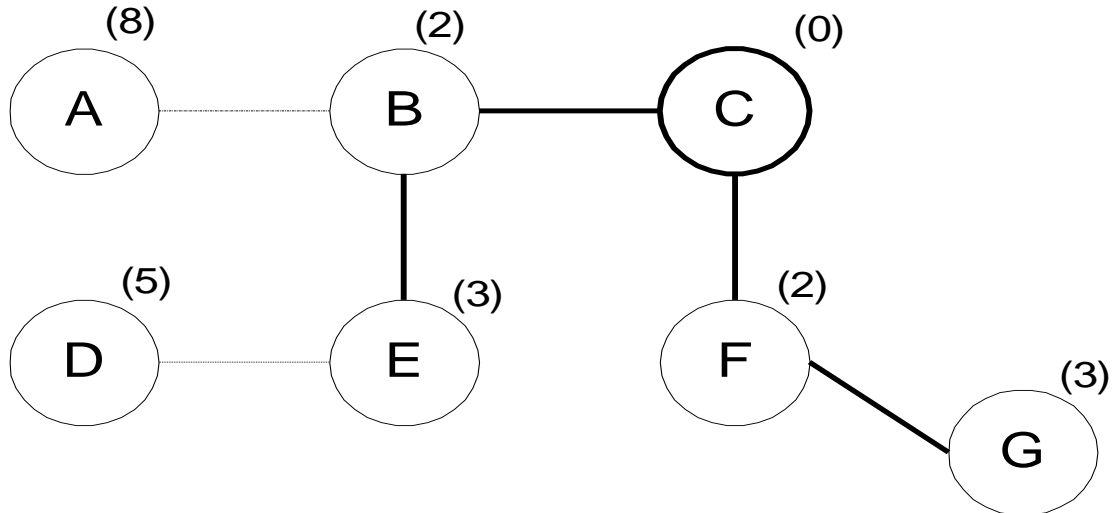
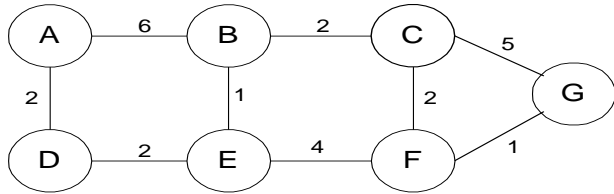
# Dijkstra algoritmus 4

- Vegyük be E-t, és számoljuk ki E szomszédainak költségét
- Nincs változás, D megjelenik



# Dijkstra algoritmus 5

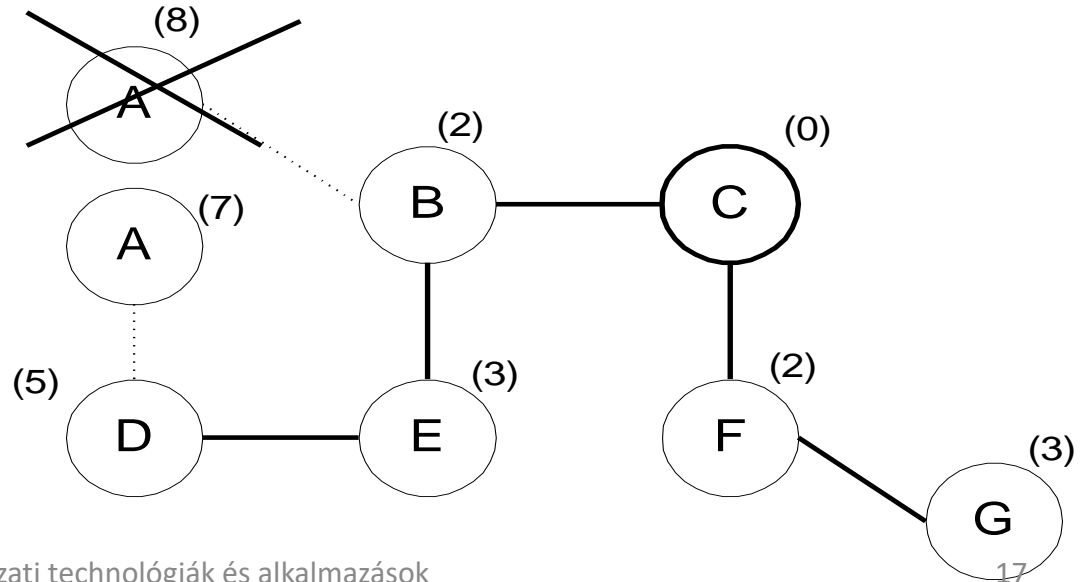
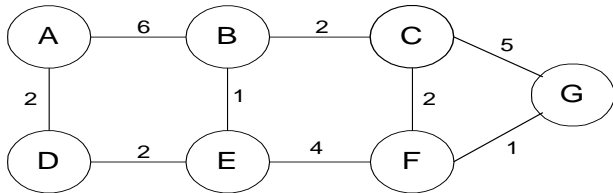
- Vegyük be G-t, és számoljuk ki G szomszédainak költségét
- Nincs változás





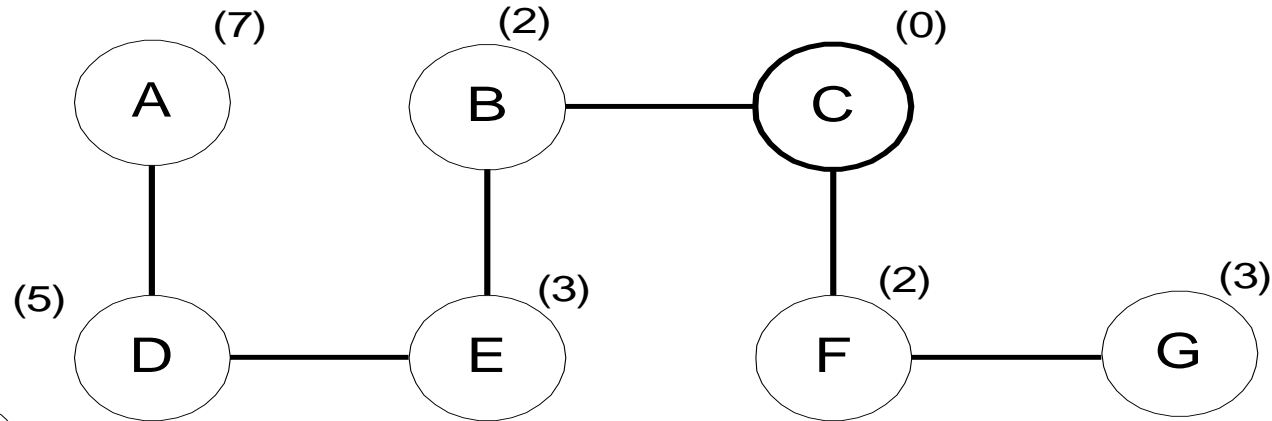
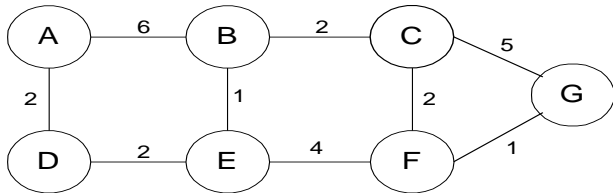
# Dijkstra algoritmus 6

- Vegyük be D-t, és számoljuk ki D szomszédainak költségét
- Rövidebb út A-hoz!



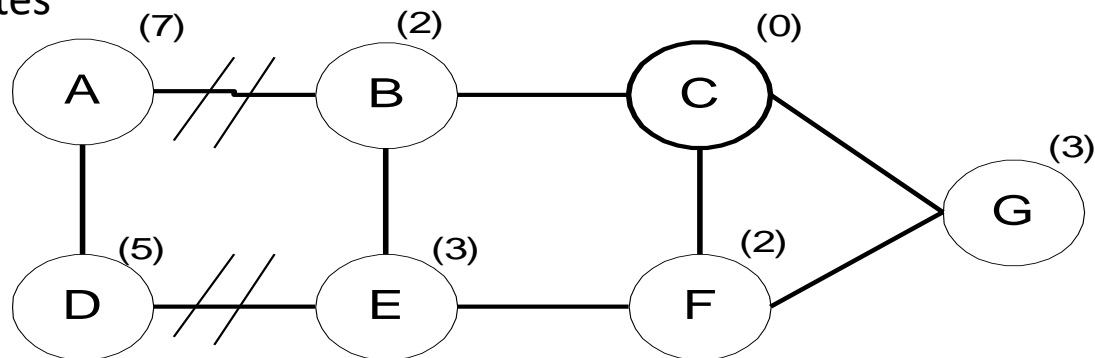
# Dijkstra algoritmus 7

- Vegyük be A-t és számold ki A szomszédjainak költségét
- Nincs több szomszéd
- Befejezés



# Hibás link következménye

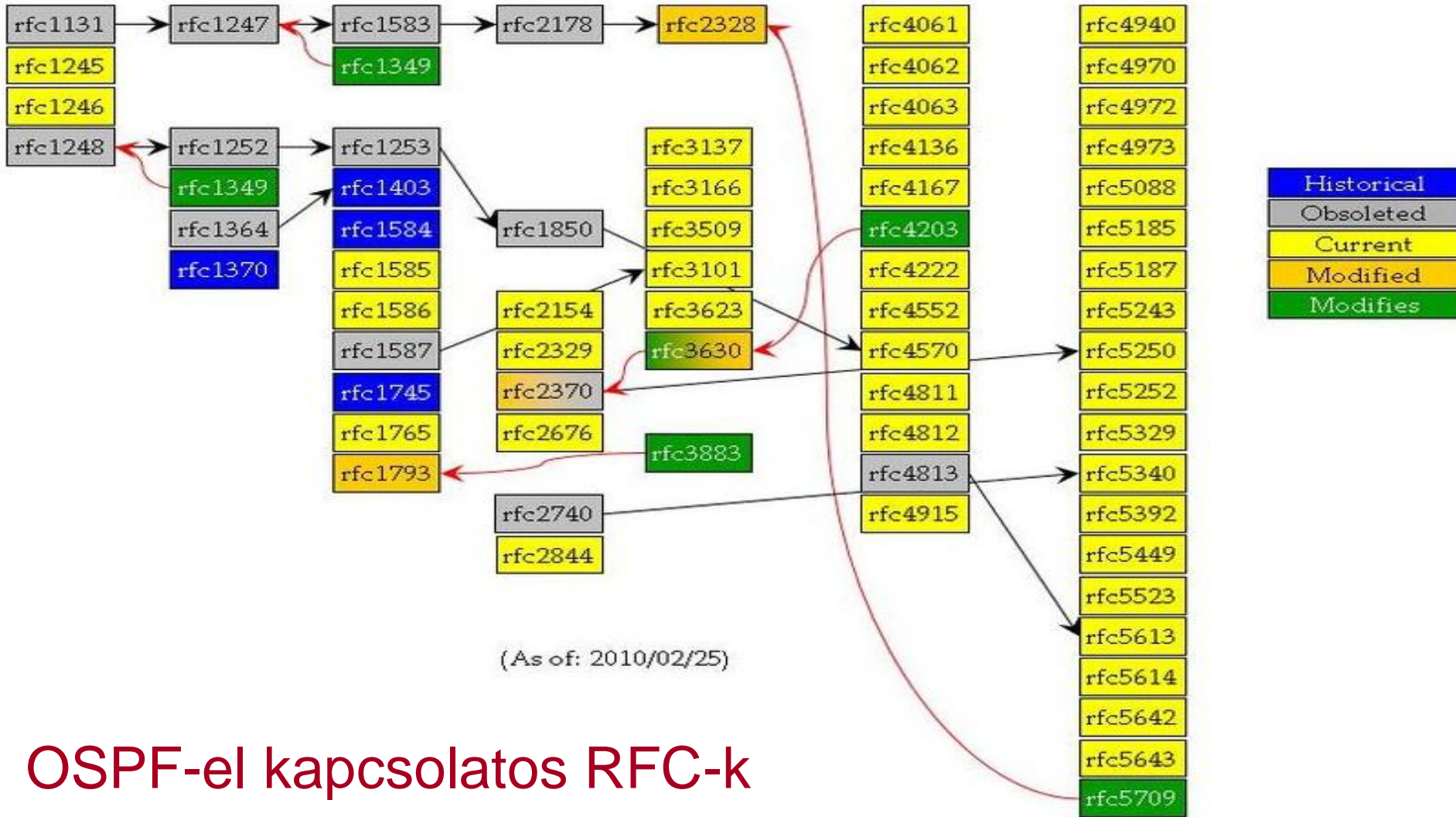
- A-B és D-E linkek megsérülnek
  - hálózat kettészakad
  - két rész képtelen értesíteni egymást a változásokról
- A és D elérhetetlennek nyilvánítja a hálózat többi részét
- Hibás link megjavulása után a routerek szinkronizálják adatbázisukat
  - Topológia frissítés



# Link-state protokollok

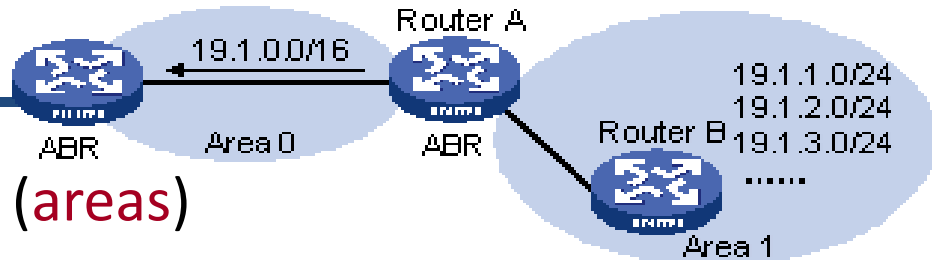
---

- OSPF – Open Shortest Path First
  - Első szabvány – RFC 1131 ('89)
  - OSPFv2 – RFC 2178 ('97)
  - OSPFv3 – RFC 2740 ('99)
    - IPv6-os verzió

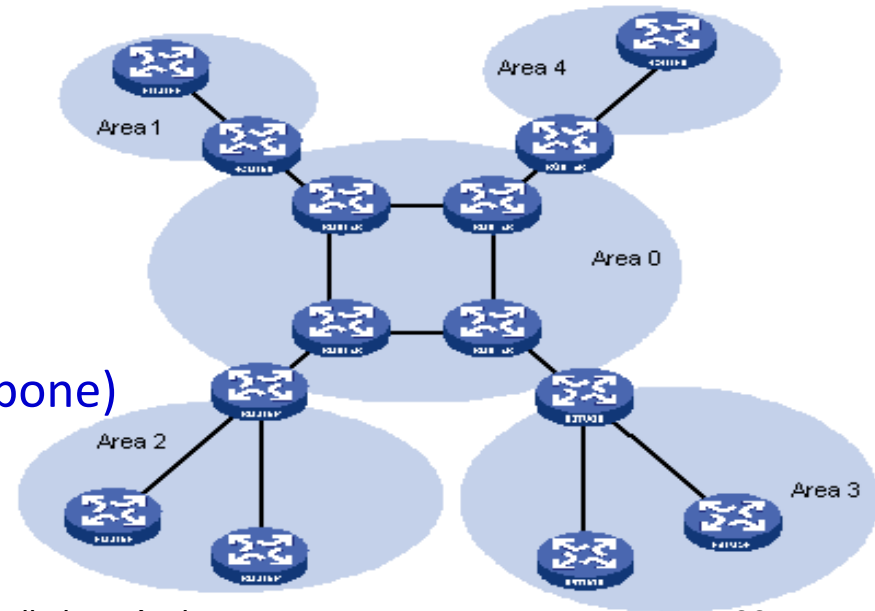


# OSPF-el kapcsolatos RFC-k

# 2 szintű hierarchia



- Egy OSPF tartomány területekre (**areas**) oszlik
  - Skálázhatósági szempontok miatt
- **LSA (Link State Advertisement)** terjesztés a területeken belül
- Területek között aggregáció
  - A változások egy területen belül nem látszanak kívülre
  - Speciális terület - **Gerinchálózat (backbone)** területe (AreaID=0)



# OSPF protokoll összetevők

---

- Szomszéd felismerés
  - Hello protokollal
- Designated Router (DR), Backup Designated Router (BDR) kiválasztása
  - Prioritás alapú
    - 0-től 254-ig
    - Ha 0 prioritás, akkor soha nem lehet DR vagy BDR
  - Egyenlőség esetén a nagyobb Router ID nyer
    - RID = a legnagyobb konfigurált loopback cím a routeren (127.x.x.x)
    - Ha nincs loopback cím konfigurálva, RID = a legnagyobb aktív interfész cím
  - Ha a DR választás után egy nagyobb prioritású router megjelenik (bekapcsolják), nem veszi át a DR szerepét amíg a DR és a BDR jól működik
  - Ha a DR „meghal”, a BDR átveszi a szerepet
    - Új BDR választás

# OSPF protokoll összetevők

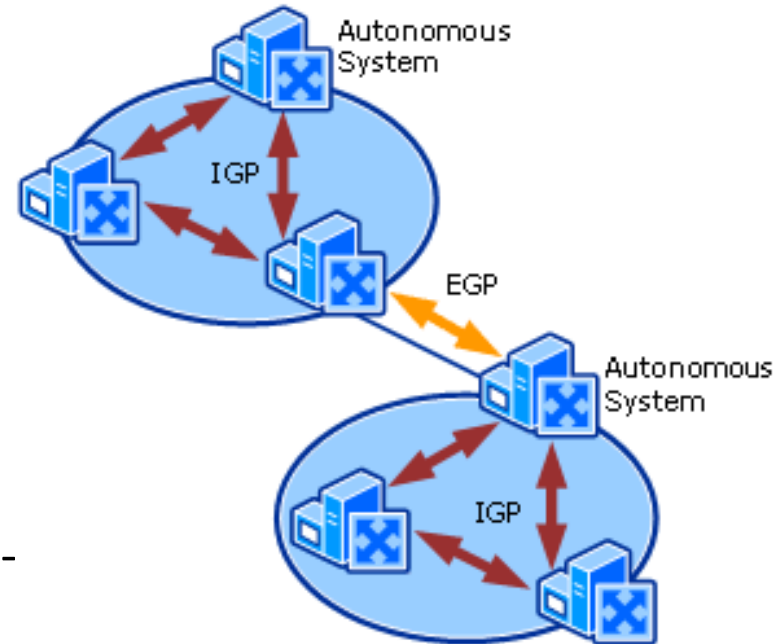
---

- Szomszédosság meghatározása (Forming adjacencies)
  - Adatbázis szinkronizálás és LSA terjesztés csak a szomszédok között
  - A DR csökkenti a hálózati forgalmat egy üzenetszórásos hálózaton
    - A DR karbantart egy táblázatot a teljes hálózati topológiáról
    - Minden router egy területen belül master-slave kapcsolatban a DR-rel
    - A routerek a 224.0.0.6 multicast címre küldik a változásokat
      - All OSPF DR and BDR routers
    - A DR a 224.0.0.5 címre küldi az új táblázatot
      - All OSPF routers



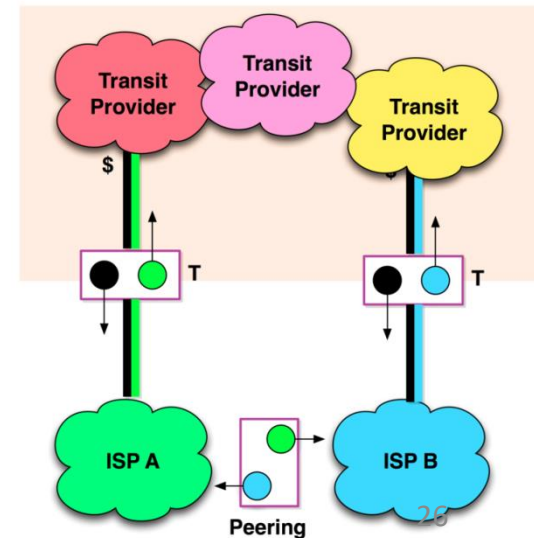
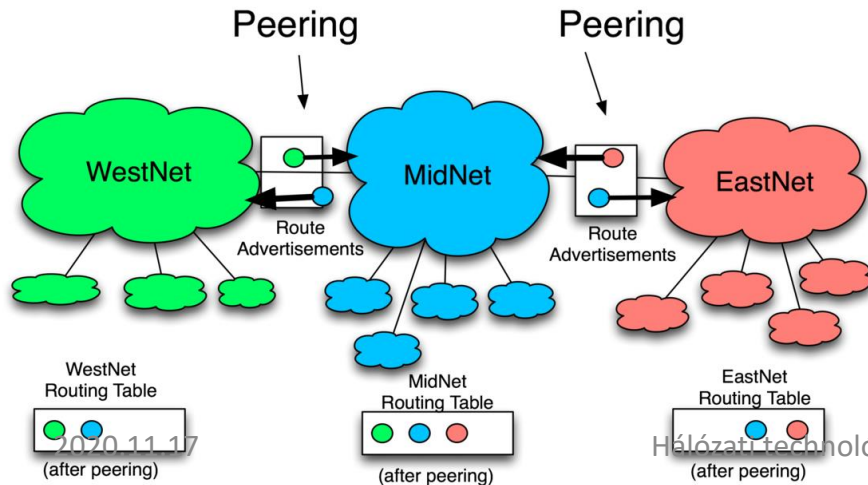
# Autonóm rendszer

- Útvásztási tartomány = autonóm rendszer (**AS – autonomous system**)
  - Útvásztók összessége egy technikailag összetartozó területen
    - Egy szolgáltató, egy adminisztráció
  - Valamilyen IGP (Interior Gateway Protocol) protokollt használ (belül)
    - pl. RIP, OSPF
  - Exterior Gateway Protokollok (EGP) az AS-ek közötti útvásztáshoz
    - pl. BGP-4



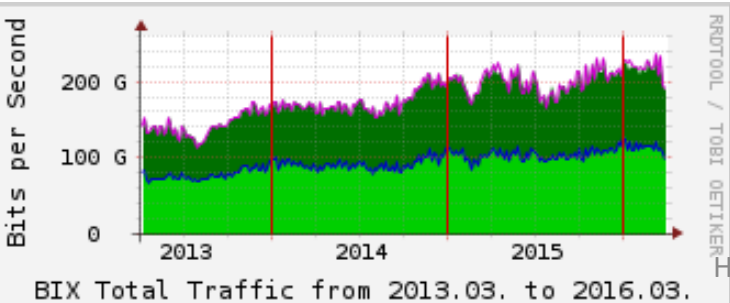
# Internet topológia

- Autonóm rendszerek hálózata
  - Vevő – szolgáltató (customer-provider) kapcsolat
    - **Tranzit kapcsolat** – kapcsolódás a globális hálózathoz
  - **Peering kapcsolat** - két egyenrangú AS, két egyenrangú szolgáltató között
    - Nem tranzitív



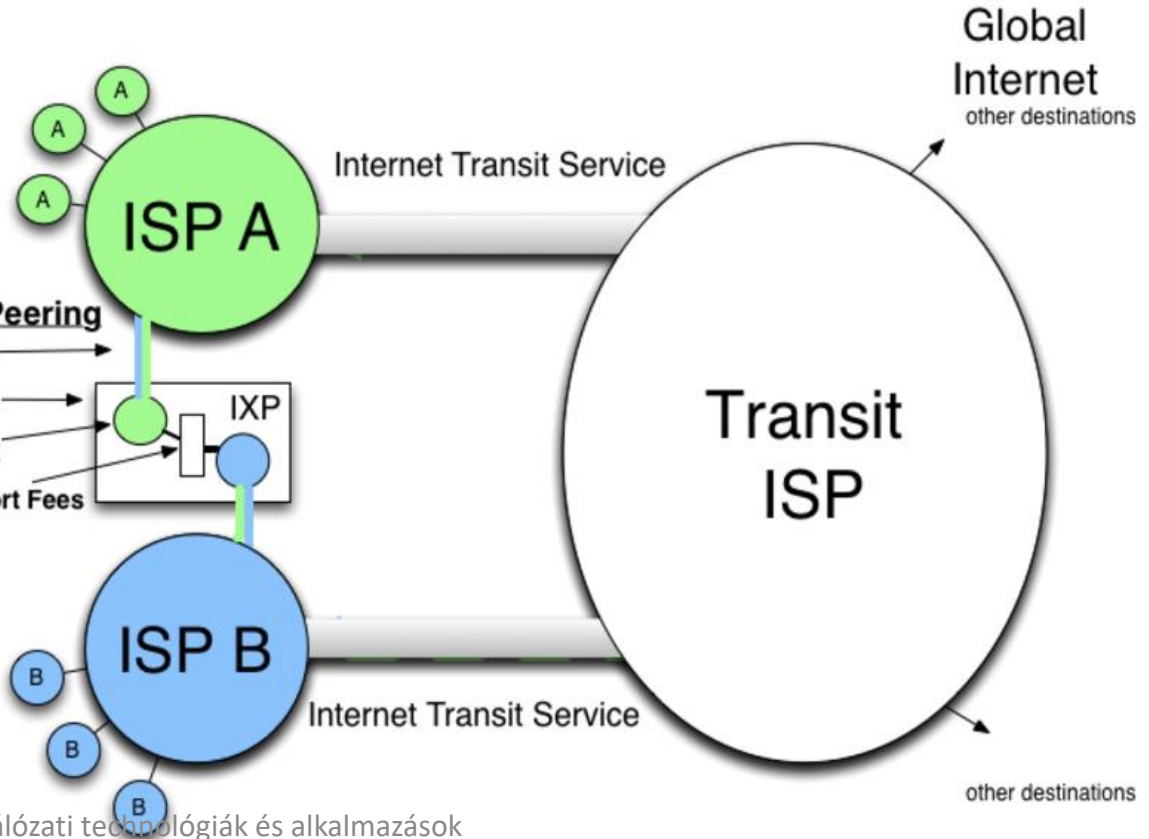
# Tranzit vs. Peering

BIX2  
(Budapest Internet Exchange)  
210 Gbit/s (2014)



## Costs of Peering

- 1) Transport
- 2) Colocation
- 3) Equipment
- 4) Peering Port Fees



# Internet topológia

---

- IGP-EGP hierarchia előnyei
  - Skálázhatóság nagy hálózatokra
    - Kevesebb prefix terjesztése
    - Gyorsabb konvergencia
  - Hibák terjedését korlátozza
  - Adminisztratív autonómia
    - Minden AS-en belül igény szerinti IGP protokoll

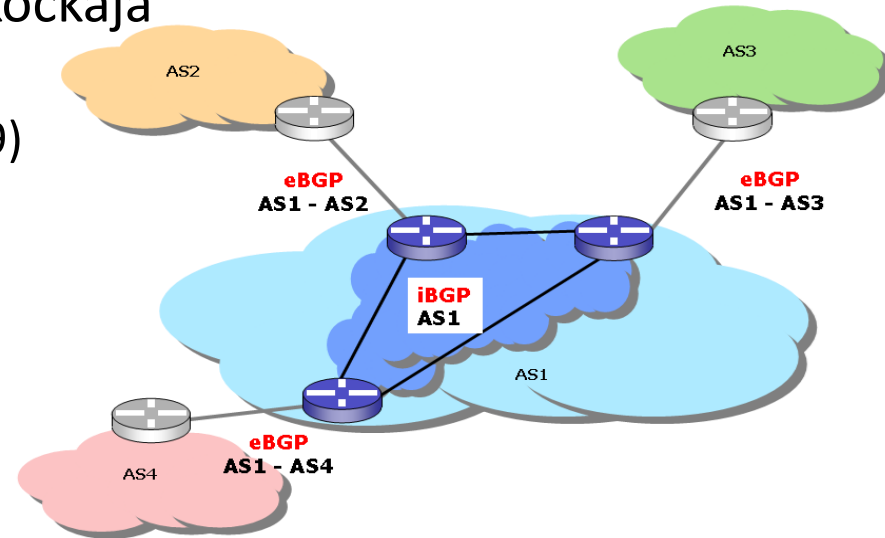
# IGP vs. EGP

---

- IGP-ben automatikus szomszéd felderítés
- EGP-ben specifikusan konfigurált peerek
  
- IGP esetén bizalom az útválasztókban
- EGP esetén (viszonylag) megbízhatatlan kapcsolatok más hálózatokkal
  
- IGP-ben a prefixek terjesztése a teljes hálózatban
- EGP-ben a prefixek terjesztése adminisztratíván korlátozott
  
- Egy IGP protokoll egy AS routereit köti össze
- Egy EGP protokoll AS-eket köt össze

# Border Gateway Protocol

- Az Internet egyik legfontosabb építőköve
- BGP kronológia
  - Eredeti szabvány – BGP – RFC 1105 ('89)
  - BGP-3 – RFC 1267 ('91)
  - BGP-4 – RFC 1771 ('95)
  - Legújabb verzió – RFC 4271 ('06)
- **External BGP (eBGP)**
  - BGP viszony egy szomszédos útvásztóval más AS-ből
- **Internal BGP (iBGP)**
  - BGP viszony egy szomszédos útvásztóval ugyanabból az AS-ből



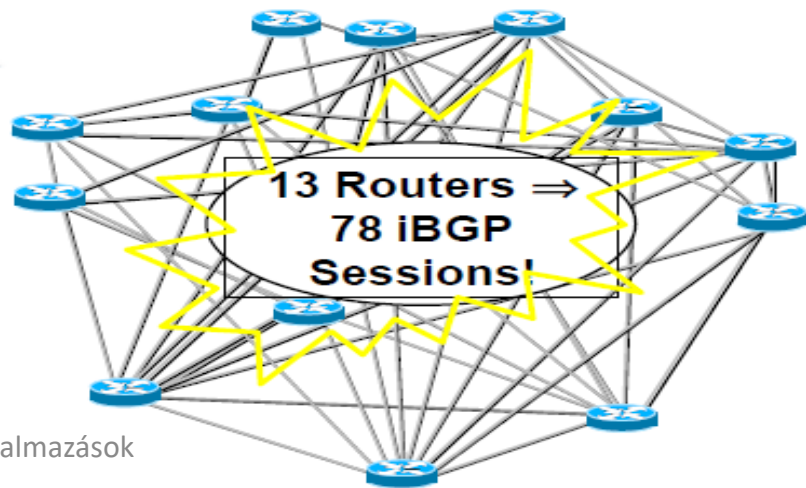
# BGP jellemzők

---

- **CIDR (Classless Inter-Domain Routing)** támogatás
  - Változó hosszúságú hálózati prefixek
  - Hatékony címtartomány aggregáció
- Manuális szomszéd beállítás
  - Nincs automatikus felfedezés
- Nincsenek periodikus frissítések
  - UPDATE üzenetek – NLRI bejegyzések
    - Network Layer Reachability Information
      - (Destination prefix, AS útvonal, next hop)
    - Hurkok elkerülése az AS-ek felsorolásával
  - A nem elérhető útvonalakat is explicit módon jelzik

# iBGP

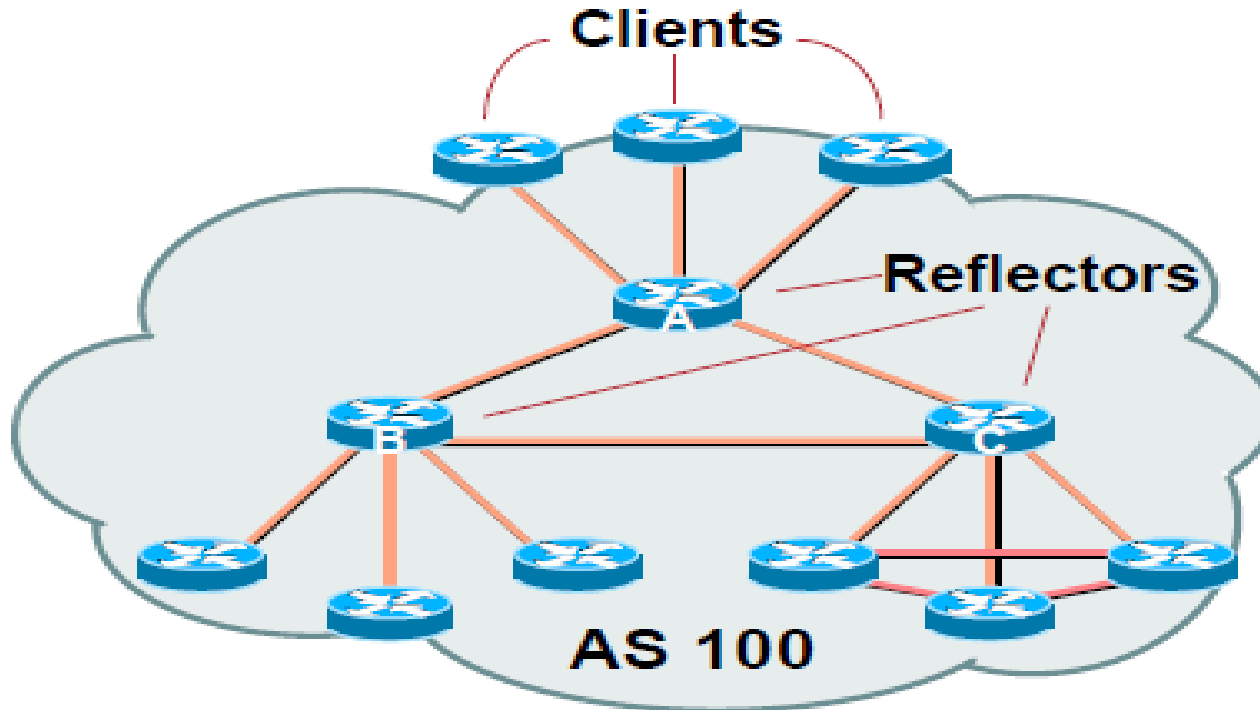
- eBGP szomszédoktól való címeket terjeszti
- iBGP csomópontok - teljes összeköttetés
  - iBGP nem szükséges útválasztás
- Hátrány – egy teljes mesh nem skálázódik
  - pl. ha  $n=1000$ ,  
 $n(n-1)/2 = 499.500$   
iBGP kapcsolat





# Route reflector

---



# Route reflector redundancia

---

