



# Počítačové a komunikačné siete

## Smerovanie

Prednáška 10



# Obsah

- » Opakovanie
- » Čo je smerovač?
- » Smerovanie
- » Smerovacie algoritmy
- » Smerovacie protokoly

# Opakovanie

» Aké prístupové metódy k médiu poznáte?

# Opakovanie

- » Aké prístupové metódy k médiu poznáte?
  - súperiace vs. nesúperiace
  - deterministické vs. nedeterministické

# Opakovanie

» Akú prístupovú metódu používa Ethernet?

# Opakovanie

- » Akú prístupovú metódu používa Ethernet?
  - CSMA/CD - Carrier Sense Multiple Access with Collision Detection

# Opakovanie

» Čo je kolízna doména? Aké zariadenie ju ohraničuje?

# Opakovanie

- » Čo je kolízna doména? Aké zariadenie ju ohraničuje?
  - segment siete, v ktorom môže nastat' kolízia, keď sa 2 zariadenia pokúsia vysielat' naraz
  - prepínač (switch)



# Opakovanie

» Aké poznáte typy duplex a multiplex?

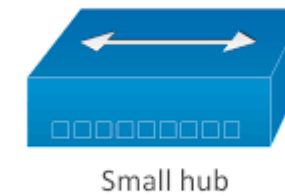
# Opakovanie

» Aké poznáte typy duplex a multiplex?

- (simplex) = jednosmerná komunikácia
- half-duplex = obojsmerná, ale nie naraz
- full-duplex = obojsmerná komunikácia
  
- FDM – frekvenčný
- TDM, STDM – časový
- WDM – vlnový
- CDM – kódový

# Smerovač

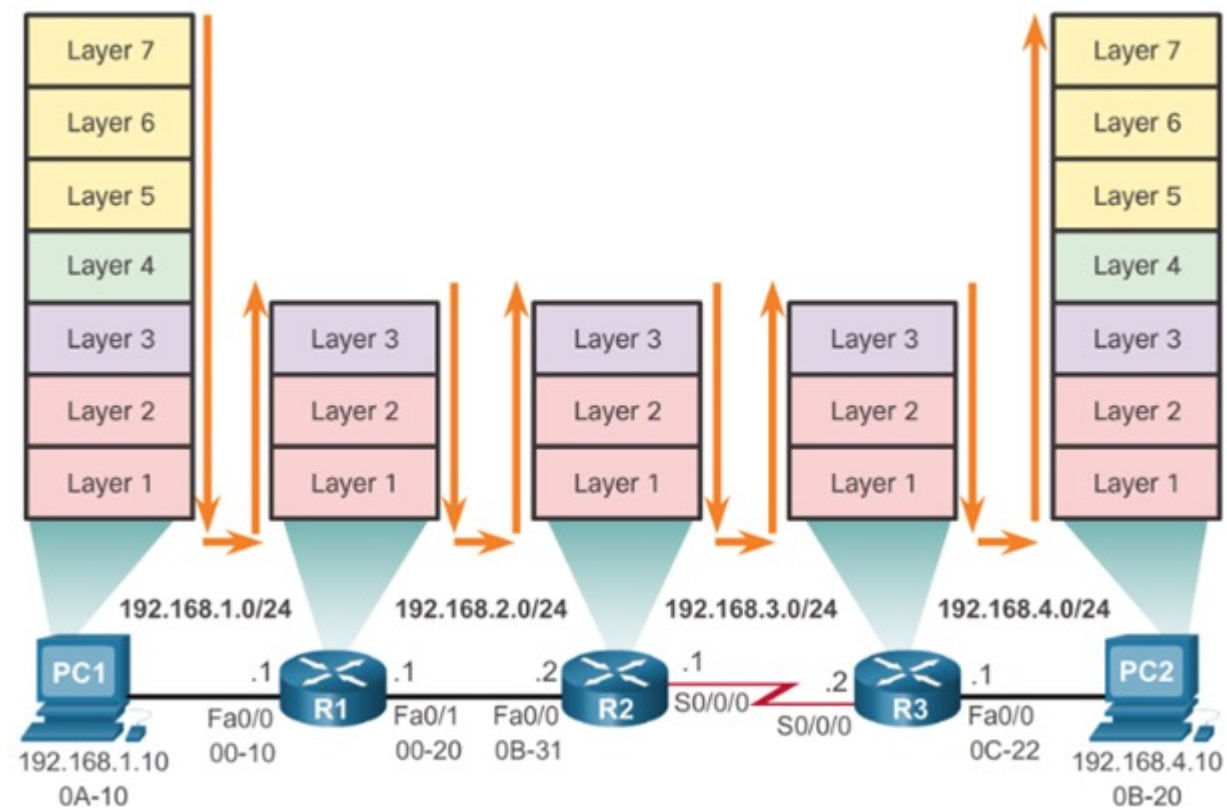
- » Zariadenia na prepájanie sietí
  - Rozbočovač (hub) L1
  - Prepínač (switch) L2
  - Smerovač (router) L3



# Smerovač

## » Smerovacia tabuľka (Routing Table)

- Priamo pripojené siete
- Statické cesty
- Dynamické cesty



## IPv4 Route Table

### Active Routes:

Network	Destination	Netmask	Gateway	Interface	Metric
	0.0.0.0	0.0.0.0	147.175.152.1	147.175.153.134	40
	127.0.0.0	255.0.0.0	on-link	127.0.0.1	331
	127.0.0.1	255.255.255.255	on-link	127.0.0.1	331
127.255.255.255	255.255.255.255	255.255.255.255	on-link	127.0.0.1	331
147.175.152.0	255.255.254.0	255.255.254.0	on-link	147.175.153.134	296
147.175.153.134	255.255.255.255	255.255.255.255	on-link	147.175.153.134	296
147.175.153.255	255.255.255.255	255.255.255.255	on-link	147.175.153.134	296
192.168.6.0	255.255.255.0	255.255.255.0	on-link	192.168.6.12	281
192.168.6.12	255.255.255.255	255.255.255.255	on-link	192.168.6.12	281
192.168.6.255	255.255.255.255	255.255.255.255	on-link	192.168.6.12	281
224.0.0.0	240.0.0.0	240.0.0.0	on-link	127.0.0.1	331
224.0.0.0	240.0.0.0	240.0.0.0	on-link	192.168.6.12	281
224.0.0.0	240.0.0.0	240.0.0.0	on-link	147.175.153.134	296
255.255.255.255	255.255.255.255	255.255.255.255	on-link	127.0.0.1	331
255.255.255.255	255.255.255.255	255.255.255.255	on-link	192.168.6.12	281
255.255.255.255	255.255.255.255	255.255.255.255	on-link	147.175.153.134	296

### Persistent Routes:

None

## IPv6 Route Table

### Active Routes:

If	Metric	Network	Destination	Gateway
1	331	::1/128		on-link
10	281	fe80::/64		on-link
2	296	fe80::/64		on-link
2	296	fe80::5d96:1ec5:3eee:46b5/128		on-link
10	281	fe80::dce9:eaec:ca65:1ad3/128		on-link
1	331	ff00::/8		on-link
10	281	ff00::/8		on-link
2	296	ff00::/8		on-link

### Persistent Routes:

None

R2#show ip route

Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP  
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area  
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2  
E1 - OSPF external type 1, E2 - OSPF external type 2  
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2  
ia - IS-IS inter area, \* - candidate default, U - per-user static route  
o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP  
a - application route  
+ - replicated route, % - next hop override, p - overrides from PfR

Gateway of last resort is not set

10.0.0.0/24 is subnetted, 2 subnets

O 10.1.1.0 [110/2] via 192.0.2.2, 00:00:35, GigabitEthernet0/1

O 10.2.2.0 [110/2] via 192.0.2.2, 00:00:35, GigabitEthernet0/1

192.0.2.0/24 is variably subnetted, 2 subnets, 2 masks

C 192.0.2.0/24 is directly connected, GigabitEthernet0/1

L 192.0.2.1/32 is directly connected, GigabitEthernet0/1

D 192.168.1.0/24 [90/3072] via 203.0.113.2, 00:00:22, GigabitEthernet0/2

D 192.168.2.0/24 [90/3072] via 203.0.113.2, 00:00:22, GigabitEthernet0/2

203.0.113.0/24 is variably subnetted, 2 subnets, 2 masks

C 203.0.113.0/24 is directly connected, GigabitEthernet0/2

L 203.0.113.1/32 is directly connected, GigabitEthernet0/2

R2#

# Smerovanie

- » Proces nájdenia cesty (route) zo zdrojového do cieľového uzla.
- » Z pohľadu uzla - ako spracovať / kam preposlať paket na 3. vrstve OSI modelu.

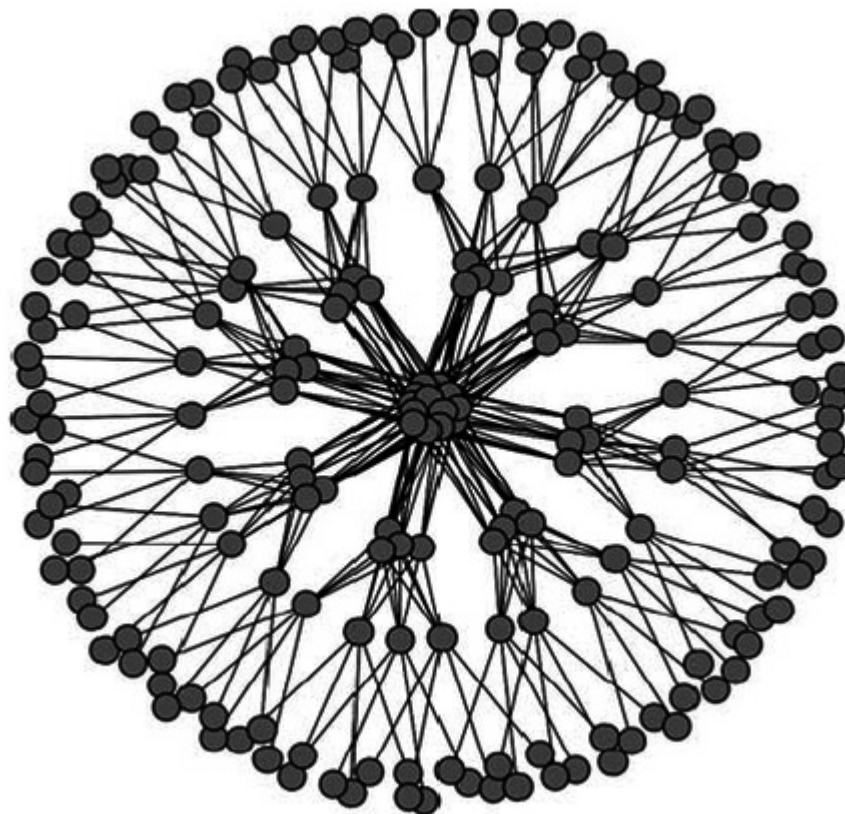
# Smerovanie

- » Výhody statického smerovania
  - Jednoduchá implementácia
  - Bezpečnosť – neprenášajú sa žiadne informácie o topológii siete
  - Stabilne rovnaká cesta
  - Žiadny výpočet algoritmu => nízke požiadavky na CPU a RAM



# Smerovanie

- » Nevýhody statického smerovania
  - Vhodné iba pre jednoduché topológie
  - Prudký nárast komplexnosti so zväčšovaním siete
  - Potreba manuálne vytvoriť / meniť konfiguráciu



# Smerovanie

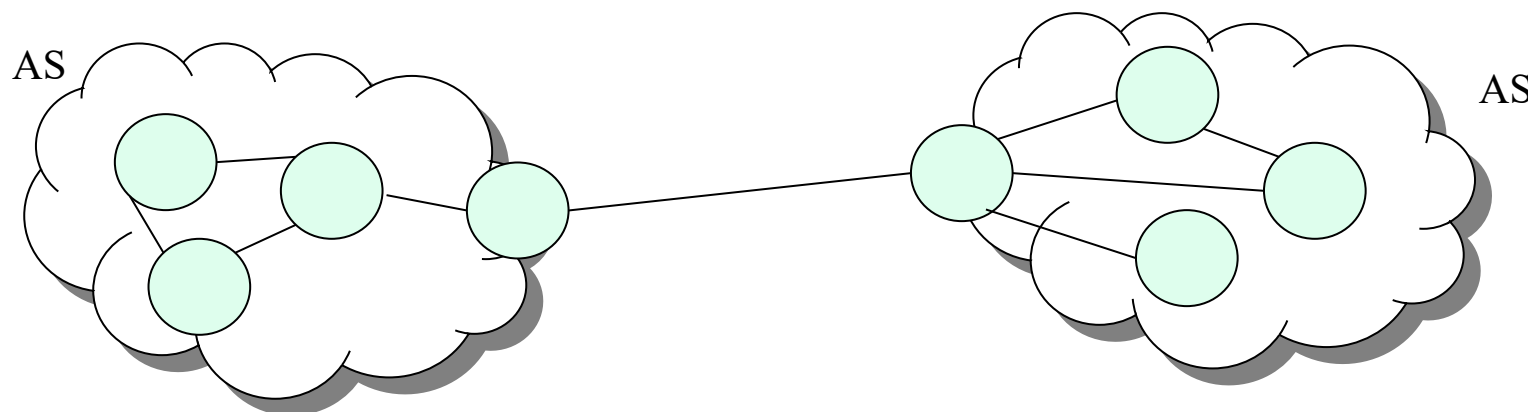
- » Statické smerovanie
- » Dynamické smerovanie
  - Automatické vytváranie smerovacej tabuľky pri zmenách v sieti
  - Smerovače komunikujú smerovacím protokolom

# Smerovanie

## Dve úrovne smerovania

- » **interné, vnútorné** (intranetwork, interior) smerovanie - smerovanie v rámci autonómnych systémov (AS) alebo administratívnych domén (AD)
- » **externé, vonkajšie** (internetwork, exterior) smerovanie - smerovanie medzi autonómnymi systémami alebo administratívnymi doménami (interdomain routing)

AS = autonomous system



# Smerovanie

- » IRP (IGP) – interné smerovacie protokoly (Interior Router/Gateway Protocols)
  - skupina interných smerovacích protokolov pre smerovanie vo vnútri autonómnych systémov alebo administratívnych domén (AS, AD)
    - otvorené (napr. RIP, OSPF, IS-IS)
    - proprietárne (napr. IGRP, EIGRP)
- » ERP (EGP) – externé smerovacie protokoly (Exterior Router/Gateway Protocols)
  - skupina externých smerovacích protokolov pre smerovanie medzi AS resp. AD
    - otvorené (napr. EGP, BGP)

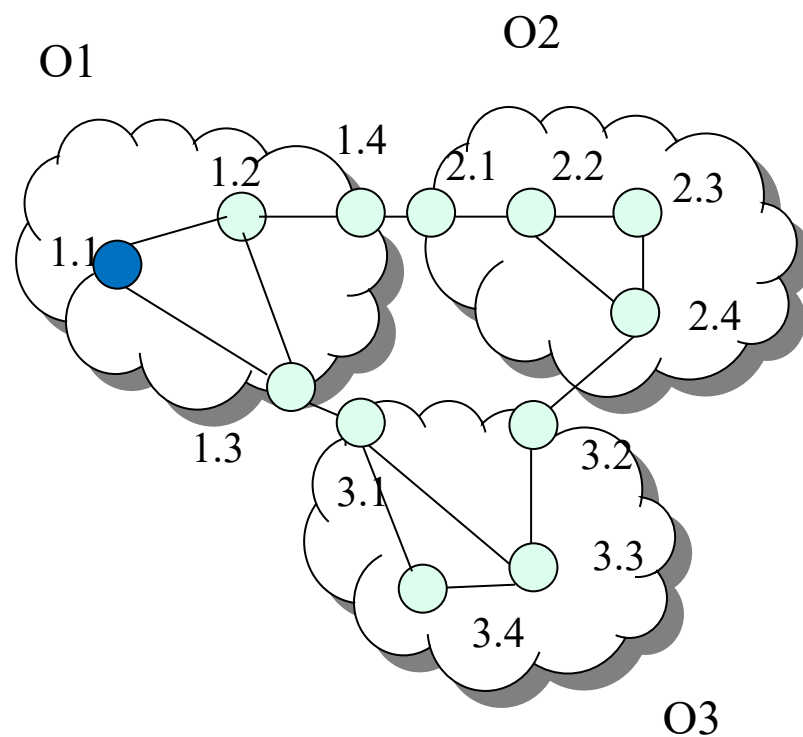
# Smerovanie

- » smerovanie práve jednému (unicast routing)
- » smerovanie viacerým (multicast routing)
- » smerovanie všetkým (broadcast routing)
- » hierarchické smerovanie (hierarchical routing)

# Smerovanie

## » hierarchické

— hierachical



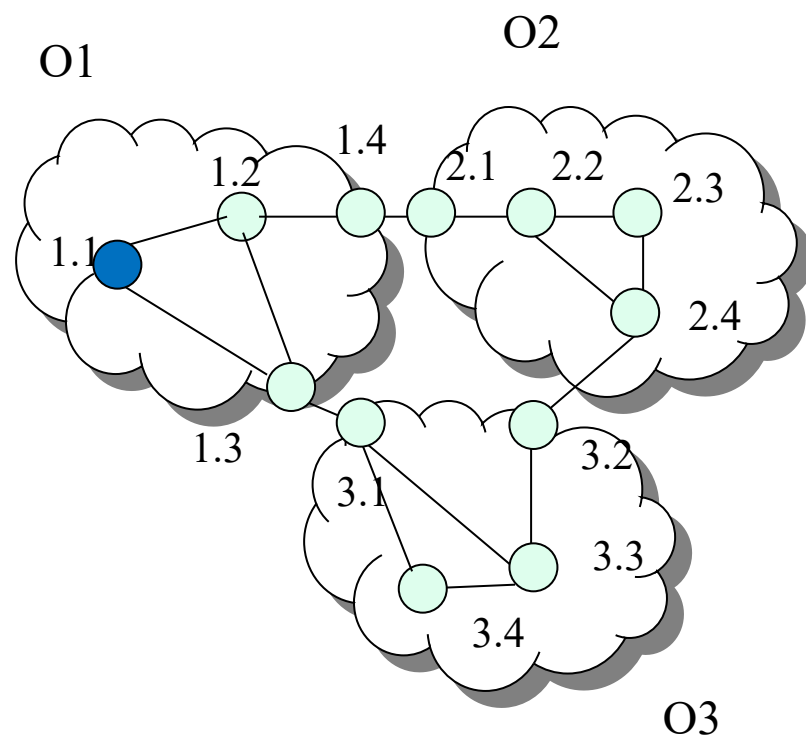
Úplná smerovacia  
tabuľka pre  
smerovač 1.1

cieľ	nasled.uzol
1.1	--
1.2	1.2
1.3	1.3
1.4	1.2
2.1	1.2
2.2	1.2
2.3	1.2
2.4	1.2
3.1	1.3
3.2	1.3
3.3	1.3
3.4	1.3

# Smerovanie

## » hierarchické

— hierachical



Úplná smerovacia  
tabuľka pre  
smerovač 1.1

cieľ	nasled.uzol
1.1	--
1.2	1.2
1.3	1.3
1.4	1.2
2.1	1.2
2.2	1.2
2.3	1.2
2.4	1.2
3.1	1.3
3.2	1.3
3.3	1.3
3.4	1.3

Hierarchická  
smerovacia tabuľka  
pre smerovač 1.1

cieľ	nasled.uzol
1.1	---
1.2	1.2
1.3	1.3
1.4	1.2
O2	1.2
O3	1.3

# Smerovanie

## Klasifikácia smerovania

- » neadaptívne  $\Leftrightarrow$  adaptívne (statické  $\Leftrightarrow$  dynamické)
  - nonadaptive vs. adaptive (static vs. dynamic) routing
- » jednoduché  $\Leftrightarrow$  viacnásobné
  - single vs. multiple path routing
- » na báze zdroja  $\Leftrightarrow$  „hop-by-hop“
  - source-based or loose source routing vs. hop-by-hop
- » centralizované  $\Leftrightarrow$  distribuované  $\Leftrightarrow$  izolované
  - centralized vs. distributed vs. isolated routing



# Smerovanie - stratégie a algoritmy

- metrika ( rôzne kritéria )
  - počet “hops”
  - oneskorenie
  - priepustnosť
  - cena
- kedy sa vykonáva smerovanie
  - pakety, datagramy
  - virtuálne kanály

# Smerovanie - stratégie a algoritmy

- kde sa získavajú smerovacie informácie
  - každý uzol
  - centrálny uzol
  - zdroj
- na základe akých poznatkov sa získavajú smerovacie informácie
  - žiadne
  - lokálne
  - susedné
  - všetky uzly

# Smerovanie - stratégie a algoritmy

- kedy sa obnovuje smerovacia informácia
  - periodicky
  - po zmene záťaže
  - po zmene topológie

# Smerovacie algoritmy

- » Požiadavky na smerovacie algoritmy
  - jednoduchosť (simplicity)
    - vnesenie čo najmensej rézie
  - korektnosť (correctness)
    - poskytnutie správnych a použitel'nych výsledkov
  - robustnosť (robustness)
    - zotavenie z rôznych porúch a neštandardných situácií
  - optimálnosť (optimality)
    - nájdienie optimálnych prepojení z hľadiska niektorých sieťových parametrov, napr. „hops“, oneskorenie, priepustnosť.....

# Smerovacie algoritmy

- » Požiadavky na smerovacie algoritmy
  - stabilita (stability)
    - konvergovanie do stabilného stavu pri obnove smerovacích tabuliek
  - „férovosť“ (fairness)
    - každý smerovač musí mať možnosť smerovať dátovú jednotku

# Smerovacie algoritmy

## » Výsledok smerovacieho algoritmu

- cesta (cesty) zo zdrojového uzla do cieľového uzla, ktorá je zaznamenaná do smerovacej tabuľky
- smerovacia tabuľka obsahuje aspoň dva stĺpce
  - adresa cieľového uzla (cieľovej siete)
  - adresa smerovača, ktorý reprezentuje „next hop“ na optimálnej ceste do cieľového uzla (príp. adresa výstupu)

# Smerovacie algoritmy

- » „Shortest path routing“ algoritmus
  - statický algoritmus
  - reprezentácia siete grafom
  - nájdenie najkratšej cesty medzi uzlami grafu (medzi smerovačmi)
  - ohodnotenie hrán grafu (korešpondencia so sieťovými parametrami - vzdialenosť, cena, oneskorenie ....
  - Dijkstra-ov alg., Floyd-ov alg., ....

# Smerovacie algoritmy

- » „Flooding“ algoritmus (záplava)
  - jednoduchý statický algoritmus
  - znásobovanie dátových jednotiek
  - smerovač vysiela vstupné dáta na každý port okrem portu, z ktorého dáta prijal
  - podporný algoritmus



# Smerovacie algoritmy

- » „Distance vector routing“ algoritmus (DVA) a „Link state routing“ algoritmus (LSA)
  - najznámejšie dynamické distribuované algoritmy
  - predpokladá sa, že smerovač pozná
    - identitu každého suseda
    - vzdialenosť k susedovi (rôzna metrika)
  - algoritmus umožní smerovaču
    - nájsť globálnu smerovaciu informáciu („next hop“ pre každý cieľ) výmenou smerovacích informácií len s jeho susedmi

# Smerovacie algoritmy

- zostaviť správu so smerovacou informáciou

Distance vector routing algorithm (Bellman-Ford, Ford-Fulkerson):

správa je „dištančný“ vektor, ktorý udáva vzdialenosť ku každému inému uzlu v sieti:

$\{ \langle \text{destination } 1, \text{distance } 1 \rangle, \dots, \langle \text{destination } n, \text{distance } n \rangle \}$

Link state routing algorithm (Dijkstra)

správa je „stavový“ vektor, ktorý udáva „stav“ linky ku každému susedovi:

$\{ \langle \text{neighbor } 1, \text{link state } 1 \rangle, \dots, \langle \text{neighbor } k, \text{link state } k \rangle \}$

# Smerovacie algoritmy

- poslať správu

Distance vector routing algorithm:

k jeho priamym susedom

Link state routing algorithm:

k všetkým smerovačom

# Smerovacie algoritmy

## » Distance Vector (DVA)

- Posielajú sa informácie o všetkých naučených sieťach
- Update iba priamym susedom
- Každý smerovač vidí iba cesty dostupné cez priamych susedov
- Bellman-Ford

# Smerovacie algoritmy

## » Link-State (LSA)

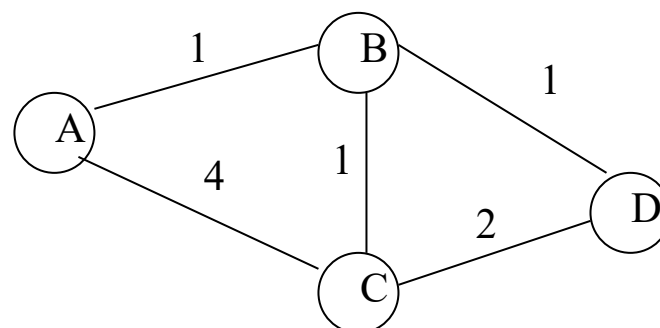
- Posielajú sa iba informácie o stave pripojených liniek
- Update každému smerovaču
- Každý smerovač si vytvorí databázu s celou topológiou
- Dijkstra

# Porovnanie DVA a LSA

	DVA	LSA
správa	vektor vzdialeností ku každému smerovaču	vektor stavov priamo pripojených liniek
správa sa posiela	susedným smerovačom	všetkým smerovačom
algoritmus nájdenia cesty	Bellman-Ford	Dijkstra
obnova smerovacej informácie	periodicky ( <i>aj aktualizácia po zmene topológie</i> )	po zmene
škálovateľnosť	malá	dobrá
konvergencia	veľmi pomalá	rýchla
problémy	slučky	synchronizácia správ
robustnosť	malá	vyššia

# Príklad DVA

## DVA algoritmus



(A)

do uzla	metrika
A	0
B	1
C	4
D	$\infty$

(B)

do uzla	metrika
A	1
B	0
C	1
D	1

(C)

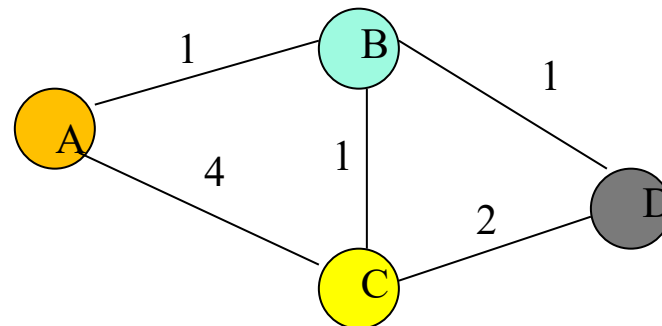
do uzla	metrika
A	4
B	1
C	0
D	2

(D)

do uzla	metrika
A	$\infty$
B	1
C	2
D	0

# Príklad DVA

DVA algoritmus



do uzla	metrika	nasl. uzol
A	0	A
B	1	B
C	4	C
D	$\infty$	-



do uzla	metrika
A	1
B	0
C	1
D	1



do uzla	metrika
A	4
B	1
C	0
D	2

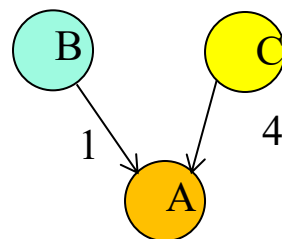
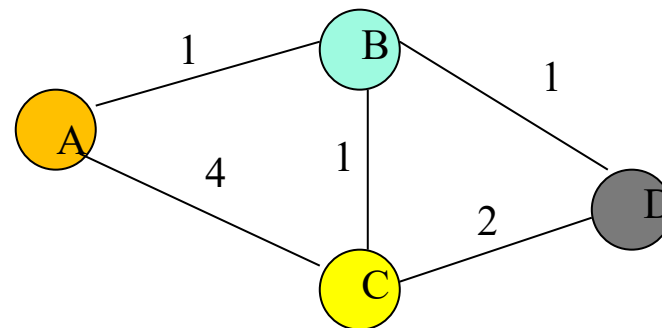


do uzla	metrika
A	$\infty$
B	1
C	2
D	0

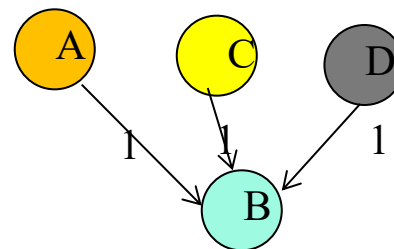


# Príklad DVA

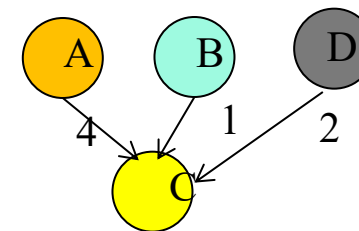
## DVA algoritmus



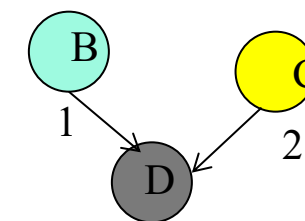
do	uzla	metrika	nasl. uzol
A	0	A	
B	1	B	
C	4	C	
D	$\infty$	-	



do uzla	metrika
A	1
B	0
C	1
D	1



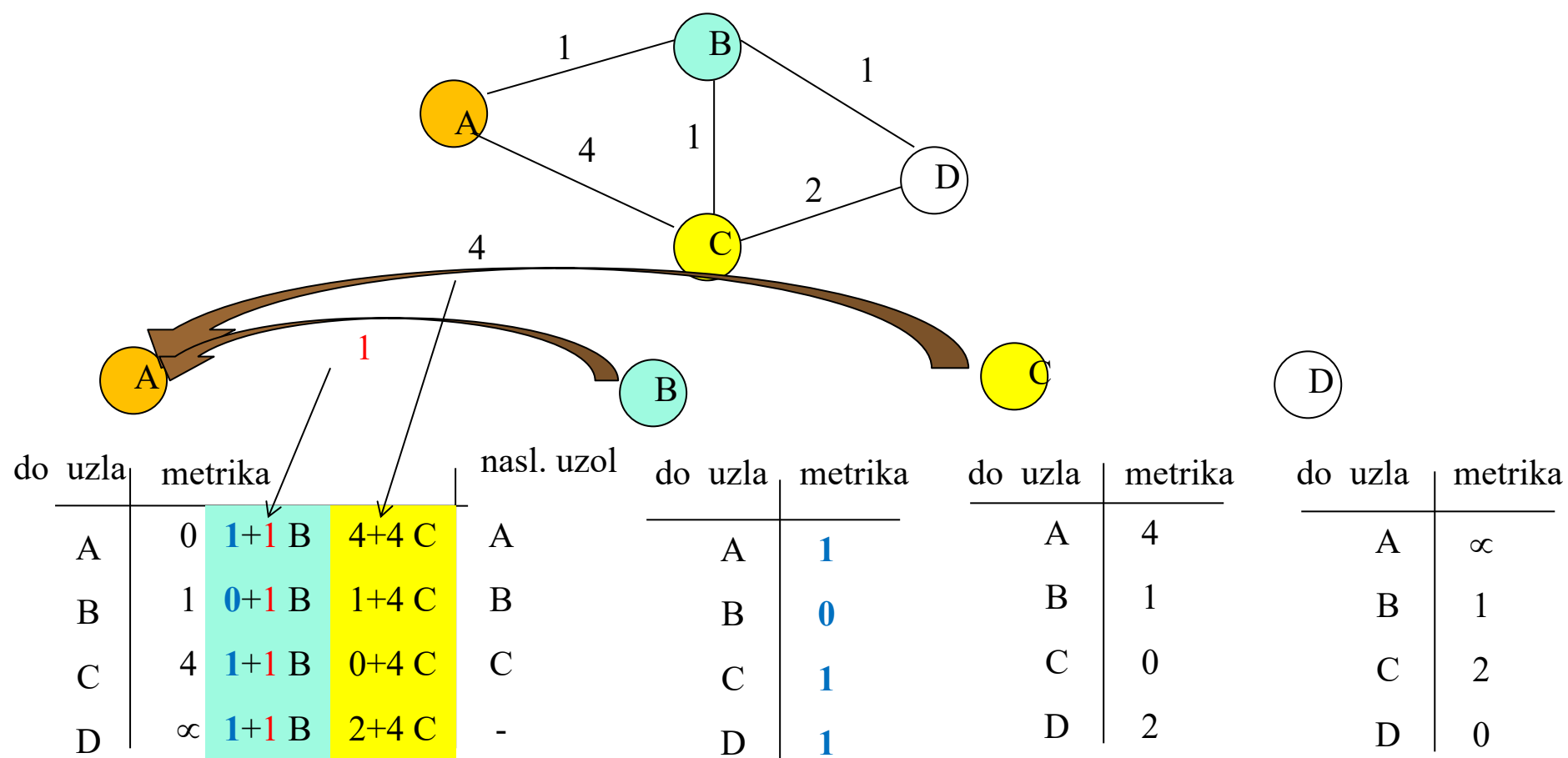
do uzla	metrika
A	4
B	1
C	0
D	2



do	uzla	metrika
A		$\infty$
B		1
C		2
D		0

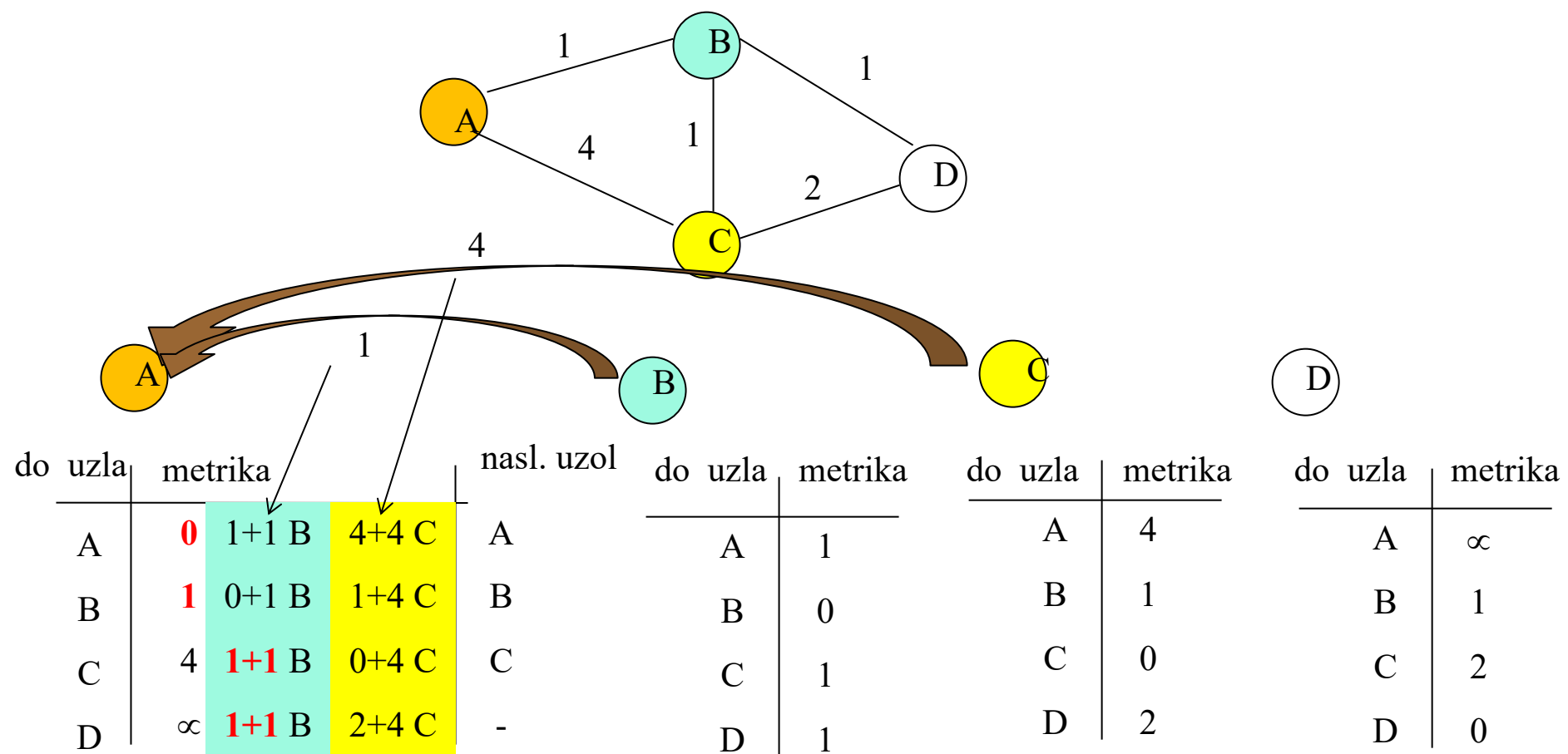
# Príklad DVA

## DVA algoritmus



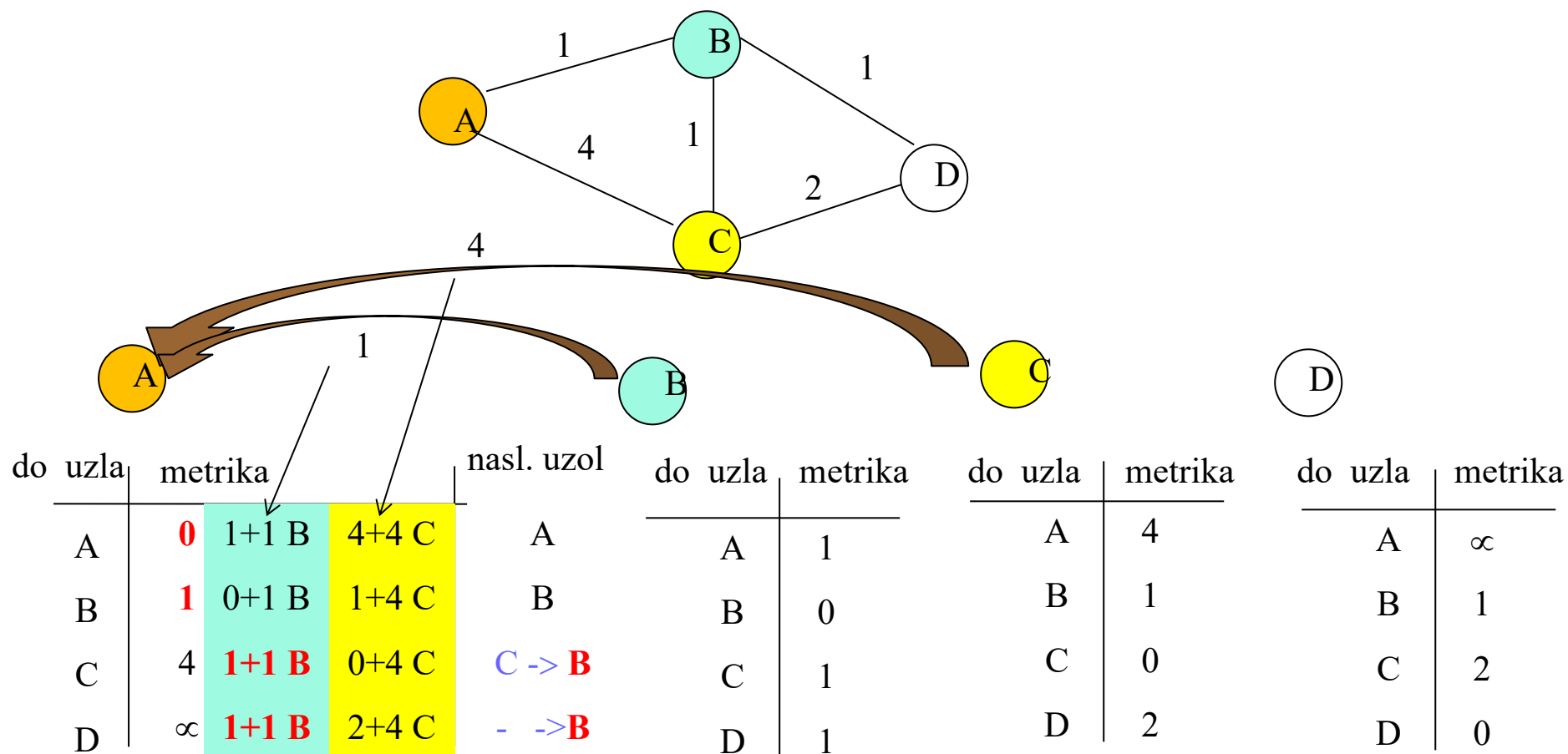
# Príklad DVA

## DVA algoritmus



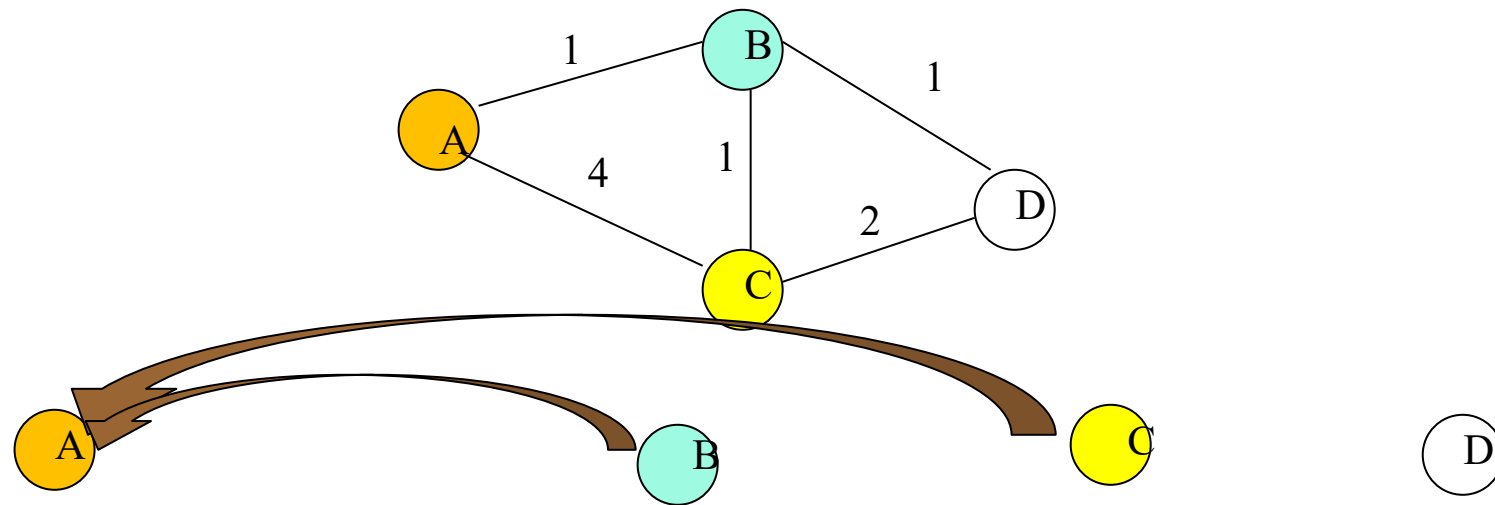
# Príklad DVA

## DVA algoritmus



# Príklad DVA

## DVA algoritmus



do uzla	metrika	nasl. uzol
A	<b>0</b>	<b>A</b>
B	<b>1</b>	<b>B</b>
C	<b>2</b>	<b>B</b>
D	<b>2</b>	<b>B</b>

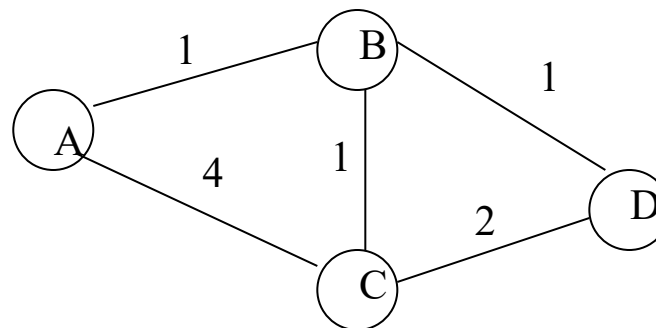
do uzla	metrika
A	1
B	0
C	1
D	1

do uzla	metrika
A	4
B	1
C	0
D	2

do uzla	metrika
A	$\infty$
B	1
C	2
D	0

# Príklad DVA

## DVA algoritmus



A

do uzla	metrika	nasl. uzol
A	<b>0</b>	<b>A</b>
B	<b>1</b>	<b>B</b>
C	<b>2</b>	<b>B</b>
D	<b>2</b>	<b>B</b>

B

do uzla	met.	n.u.
A	<b>1</b>	<b>A</b>
B	<b>0</b>	<b>B</b>
C	<b>1</b>	<b>C</b>
D	<b>1</b>	<b>D</b>

C

do uzla	met.	n.u.
A	<b>2</b>	<b>B</b>
B	<b>1</b>	<b>B</b>
C	<b>0</b>	<b>C</b>
D	<b>2</b>	<b>D</b>

D

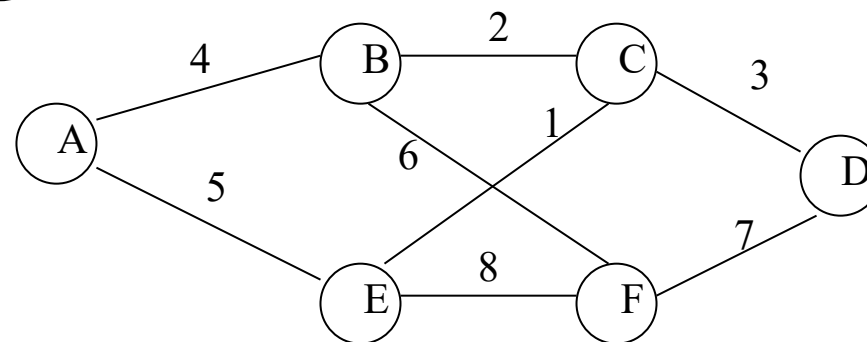
do uzla	met.	n.u.
A	<b>2</b>	<b>B</b>
B	<b>1</b>	<b>B</b>
C	<b>2</b>	<b>C</b>
D	<b>0</b>	<b>D</b>

# Príklad DVA

- » Problémy pri DVA – slučky (count-to-infinity problem)
- » Možné riešenia:
  - zápis cesty
  - limit pre dĺžku cesty
  - rozložený horizont (split horizon)
  - nekorektné spätné informácie (poison reverse)
  - časové pozdržanie (hold-down timer)
  - atď.

# Príklad LSA

## LSA algoritmus



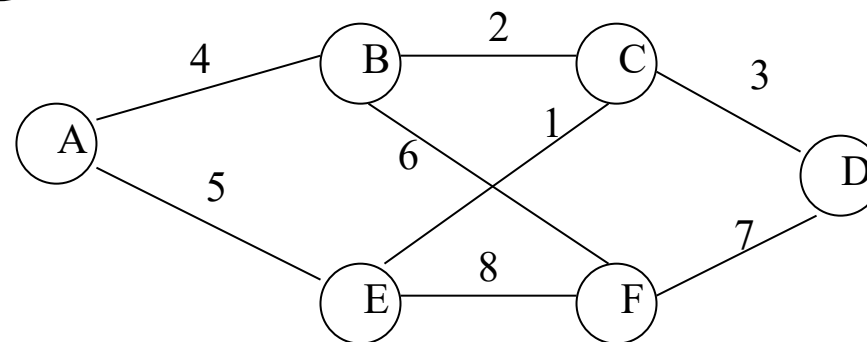
LSP:

A		B		C		D		E		F	
B	4	A	4	B	2	C	3	A	5	B	6
E	5	C	2	D	3	F	7	C	1	D	7
		F	6	E	1			F	8	E	8



# Príklad LSA

## LSA algoritmus

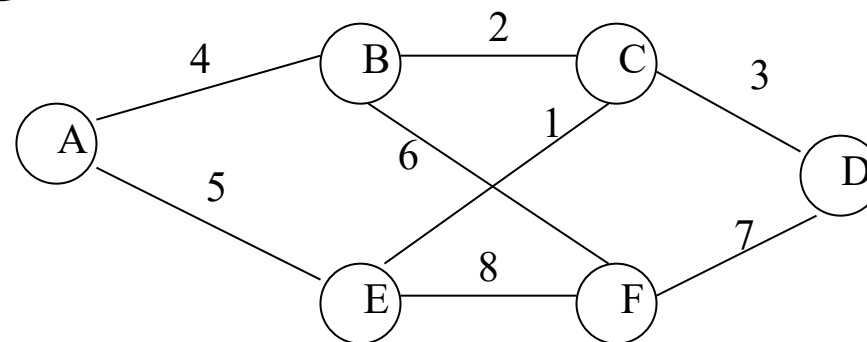


LSP:

A	B	C	D	E	F
SEQ	SEQ	SEQ	SEQ	SEQ	SEQ
B   4	A   4	B   2	C   3	A   5	B   6
E   5	C   2	D   3	F   7	C   1	D   7
	F   6	E   1		F   8	E   8

# Príklad LSA

## LSA algoritmus



LSP:

A	B	C	D	E	F
SEQ	SEQ	SEQ	SEQ	SEQ	SEQ
AGE	AGE	AGE	AGE	AGE	AGE
B   4	A   4	B   2	C   3	A   5	B   6
E   5	C   2	D   3	F   7	C   1	D   7
	F   6	E   1		F   8	E   8

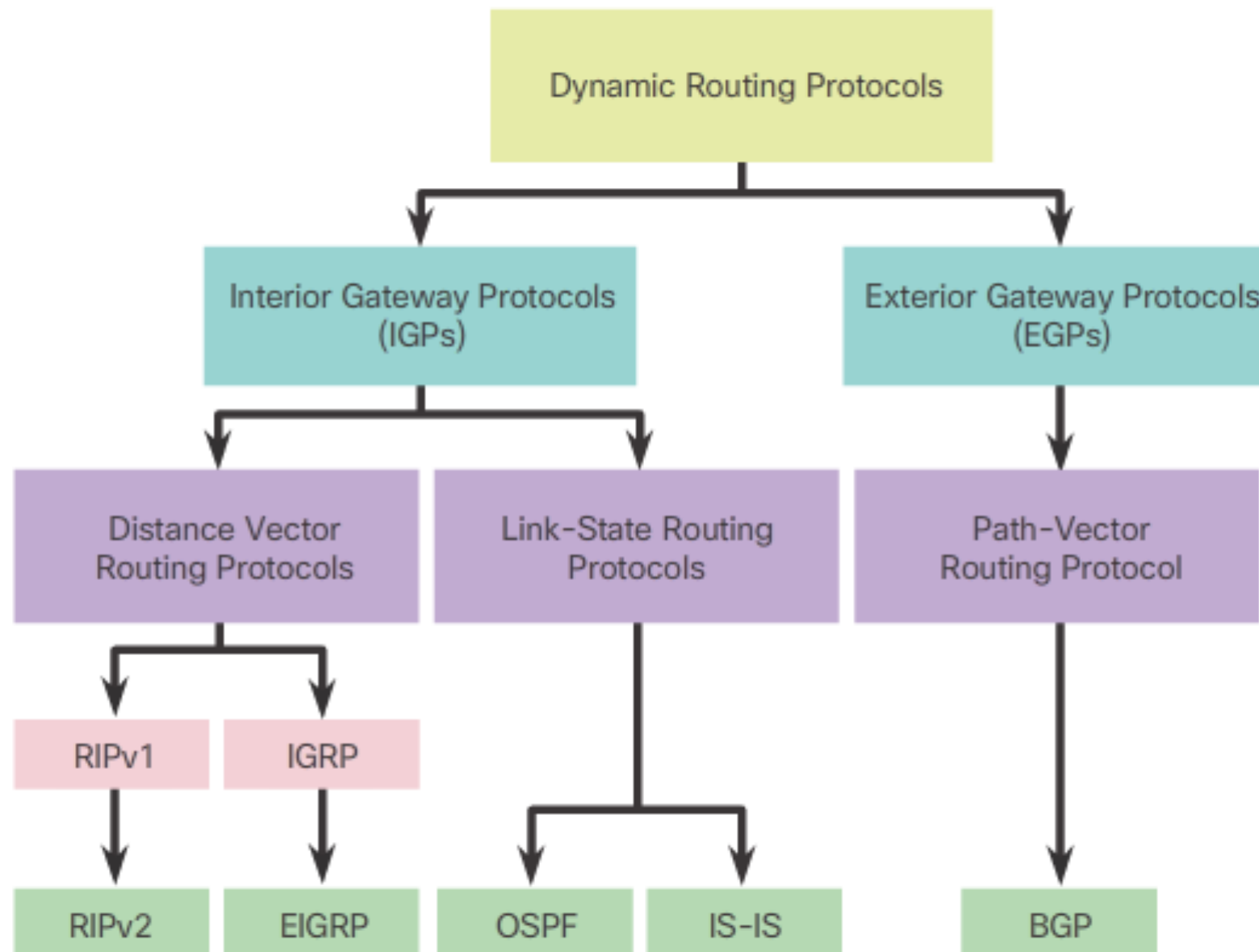
# Smerovacie protokoly

- » implementujú smerovacie stratégie a smerovacie algoritmy
- » prenášajú sieťové parametre ku každému smerovaču
- » obnovujú smerovacie tabuľky (databázy) v každom smerovači

# Smerovacie protokoly

- » Požiadavky na smerovacie protokoly
  - minimalizovať smerovacie tabuľky -> menšia pamäť a réžia
  - minimalizovať riadiace správy -> menšia réžia

# Smerovacie protokoly



# Smerovacie protokoly

Protokol	Typ	Vnorenie
RIP	IGP / DVA	RIP -> UDP -> IP
OSPF	IGP / LSA	OSPF -> IP
BGP	EGP / DVA	BGP -> TCP -> IP
IGRP	IGP / DVA	IGRP -> IP
Integrated IS-IS	IGP / LSA	do spojovej vrstvy

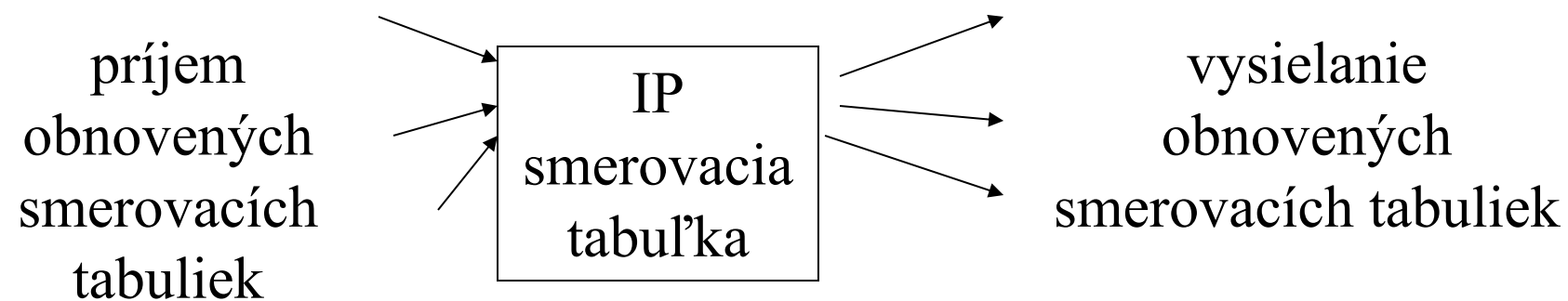
## Smerovací algoritmus

Skupina	DVA	LSA
IRP (IGP)	RIP IGRP	OSPF Integrated IS-IS
ERP (EGP)	BGP*	(IDRP)

\*BGP používá Path Vector Algoritmus

# Príklady smerovacích protokolov v Internete

## » RIP – Routing Information Protocol



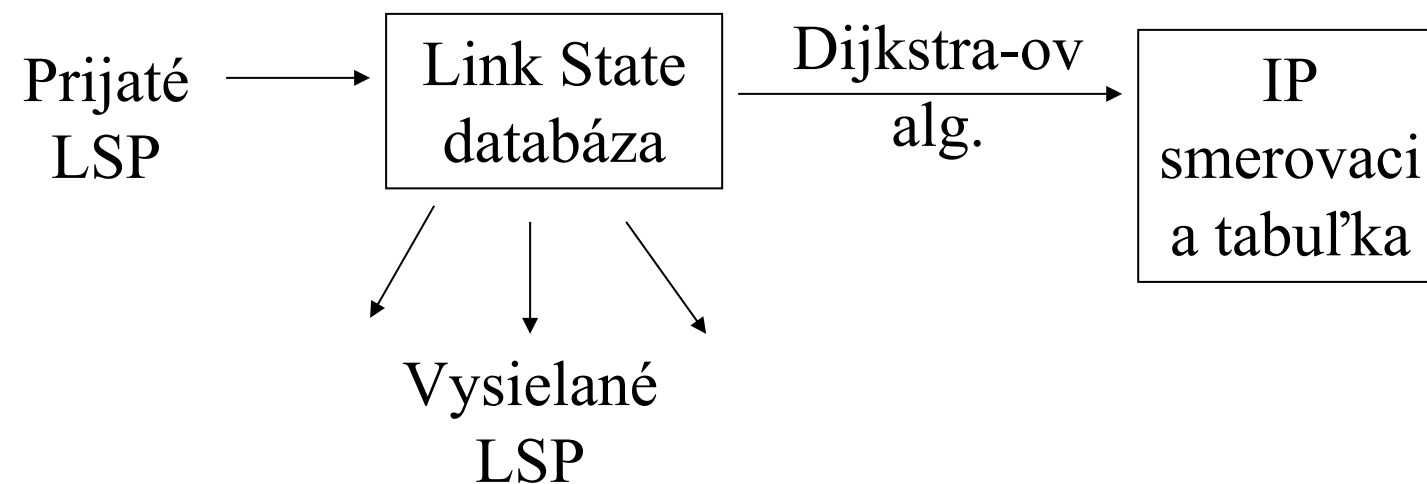
RIPv1 - IP adresy A,B,C

RIPv2 - podpora CIDR



# Príklady smerovacích protokolov v Internete

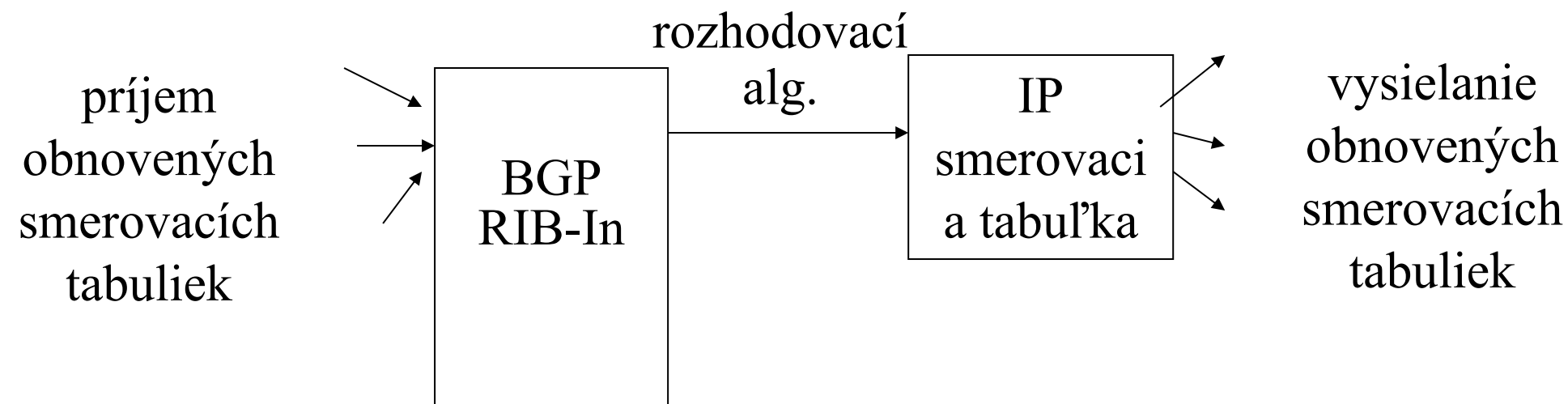
## » OSPF – Open Shortest Path First



OSPF - IP adresy A,B,C; podpora  
CIDR

# Príklady smerovacích protokolov v Internete

## » BGP – Border Gateway Protocol



BGPv4 - podpora CIDR

