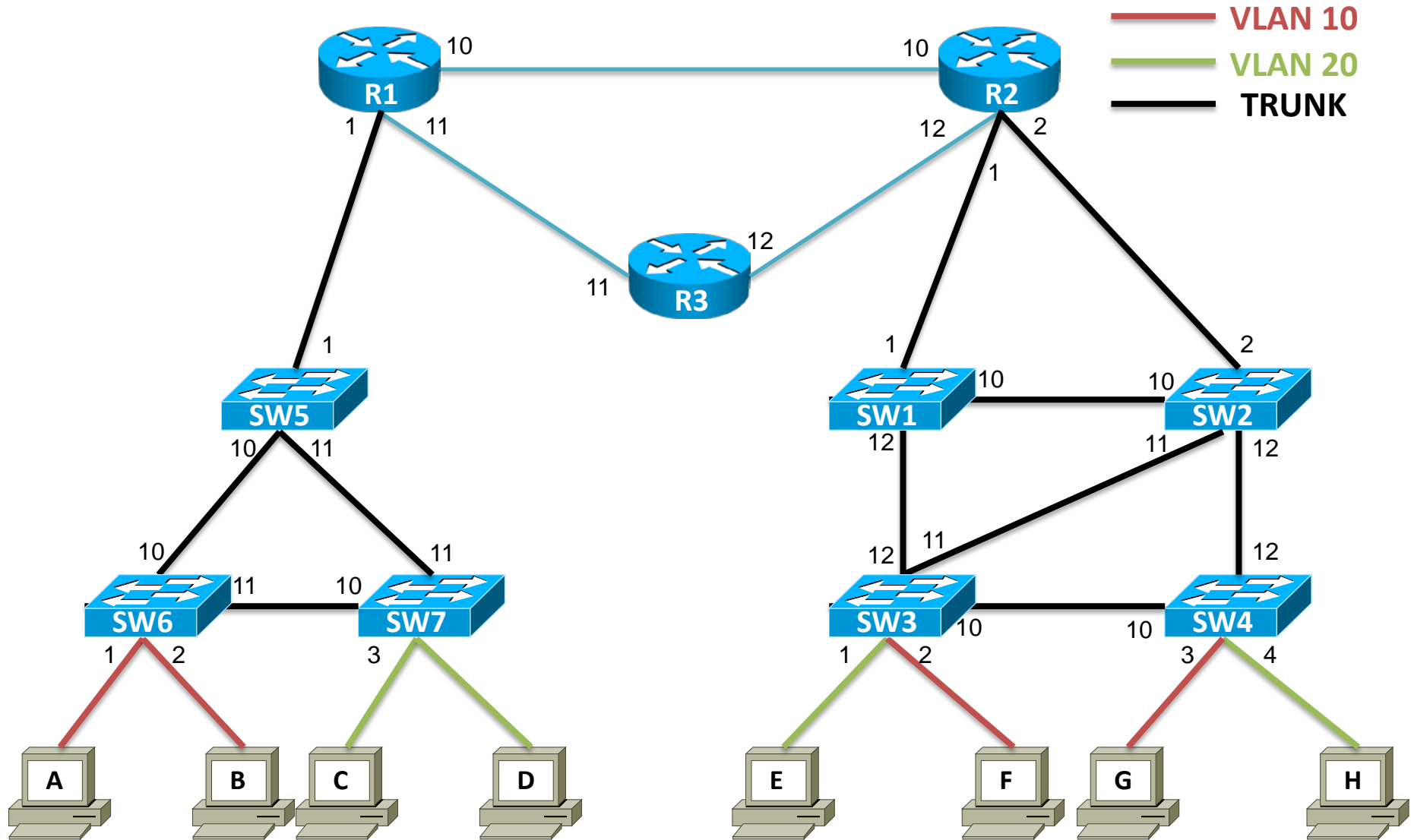


Recapitulare



NETWORKZ 8

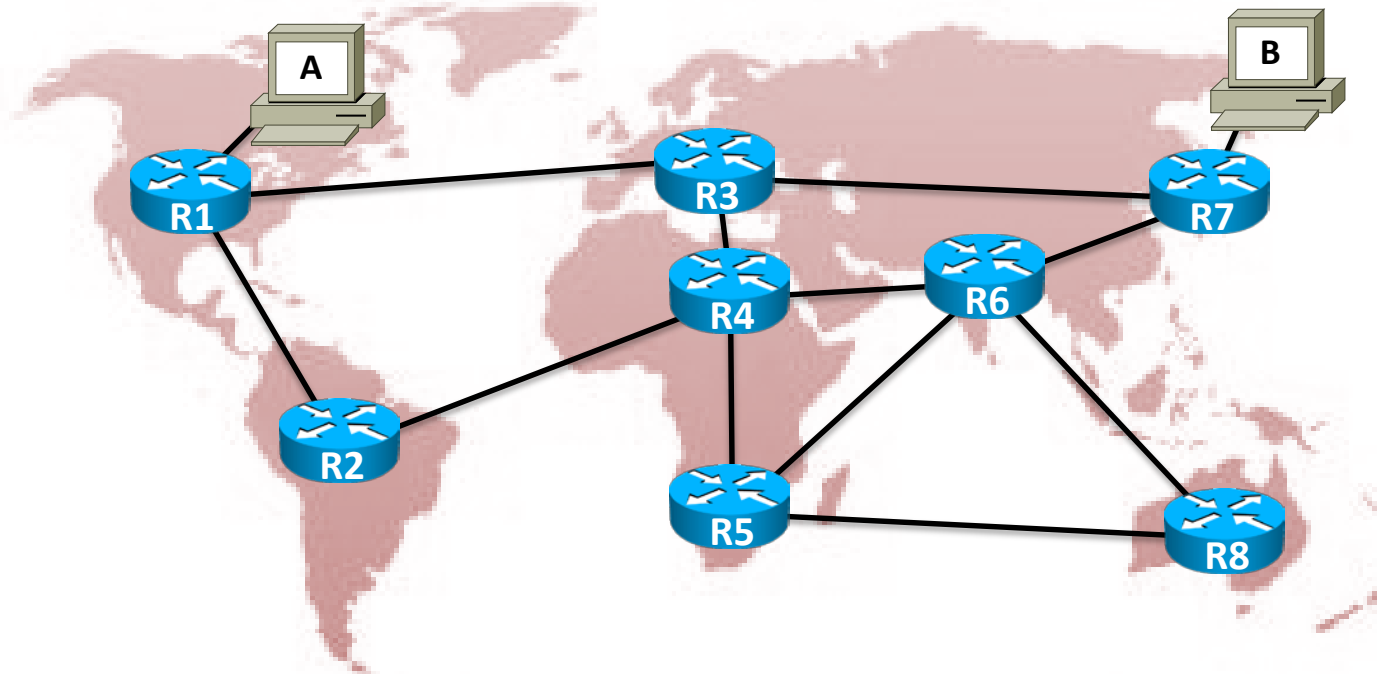
Routing & Routers

Obiective

- Rolul unui ruter
- Procesul de rutare
- Distanța administrativă și metrică
- Configurarea rutelor statice
- Protocoale dinamice de rutare

Ce este un ruter

- Comunicația în Internet este formată din **pachete**
- Când destinația se află la distanțe mari (de exemplu pe un alt continent) trebuie decisă calea ce va fi luată de pachete



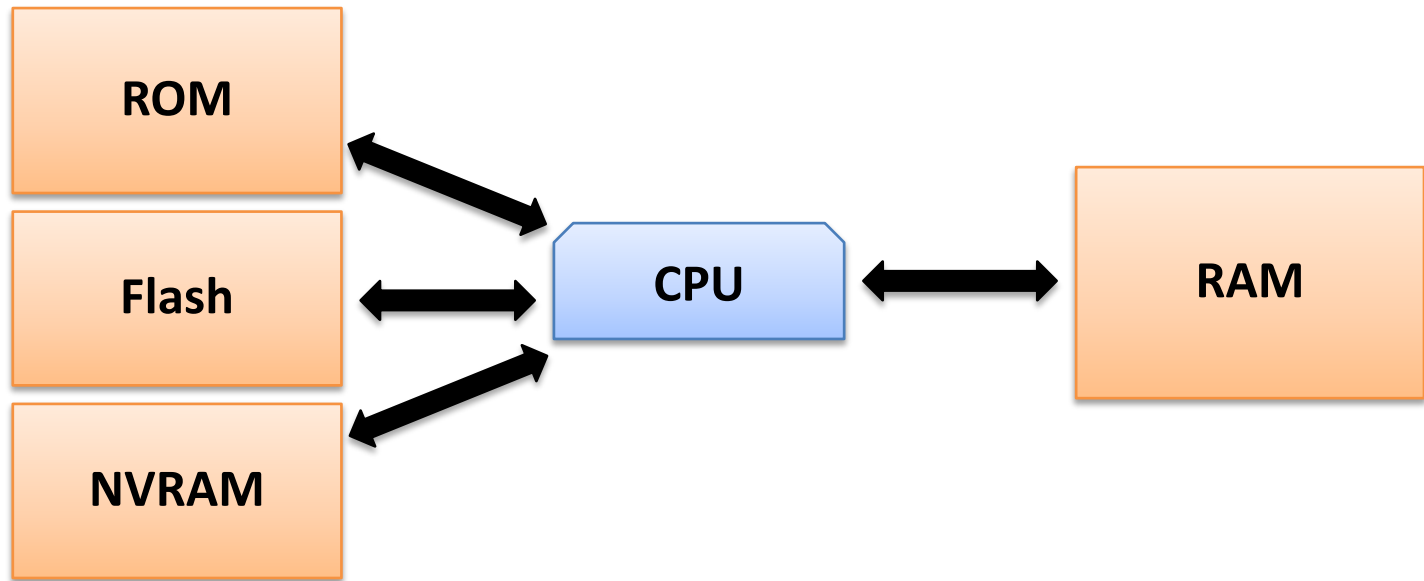
- **Ruterul** este un echipament intermediar ce are rolul de a ghida traficul pachetelor în Internet în mod cât mai eficient

Ce este un ruter

- Din punct de vedere arhitectural, ruterul este un calculator specializat; orice ruter este caracterizat prin:
 - Procesor
 - Memorie
 - Dispozitive de I/O (consolă, linii virtuale)
- Diferențe față de un calculator obișnuit sunt:
 - ASIC-uri pentru realizarea rapidă a procesului de rutare
 - Număr mare de interfețe de rețea și posibilitatea adăugării de noi module de interfețe
 - Sistem de operare optimizat pentru controlul procesului de rutare
 - Funcții specializate de monitorizare
- Un calculator obișnuit poate fi configurat să se comporte ca un ruter

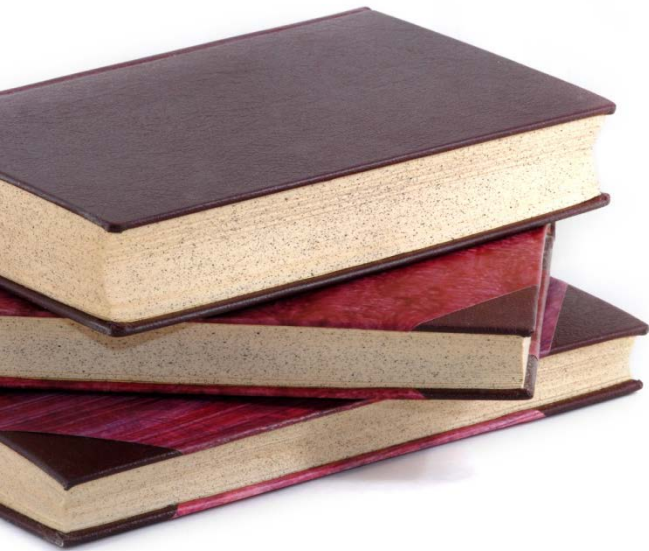


Schema bloc a unui ruter dedicat



Funcții ale ruterele moderne

- Pe lângă funcția de bază de a trimite pachete pe calea optimă, ruterele moderne mai pot îndeplini și o serie de alte funcții:
 - filtrarea traficului în funcție de anteturile de nivel 3 sau 4 (ACL-uri)
 - translatarea de adrese (NAT și PAT)
 - stabilirea de tuneluri
 - atribuire de adrese (server DHCP)
 - proxy ARP
- Presupunând că nu există rutere, ar fi o soluție organizarea Internetului ca o rețea imensă de switch-uri și host-uri aflate în același domeniu de broadcast?
 - R: **Nu**, un singur domeniu de broadcast nu ar face față traficului (~5.73 Tb/secundă în 2009)

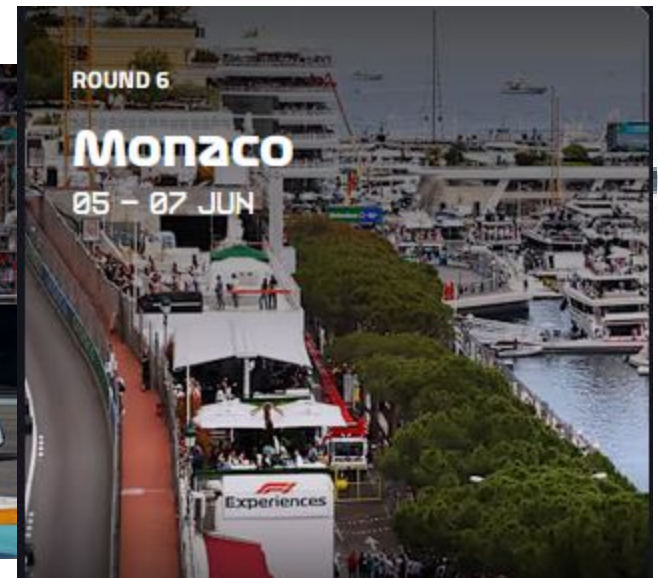


Procesul de rutare

- Definiții
- Tabela de rutare
- Surse de rute
- Procesul de rutare
- Classless vs classful routing
- Protocoale dinamice de rutare
- Exemplu

Definiție

- Procesul prin care un ruter alege calea optimă pentru trimiterea unui pachet poartă numele de **rutare**
- Setul (destinație, direcție, distanță) poartă numele de **rută**
 - Exemplu: (Monza, Spa, 666 km)
 - Setul sumarizează exprimarea “Pentru a ajunge din locația curentă la destinația Monza putem trece prin Monaco; distanța totală va fi de 120km”)
 - Un astfel de set ajută și în luarea unei decizii când există mai multe posibilități

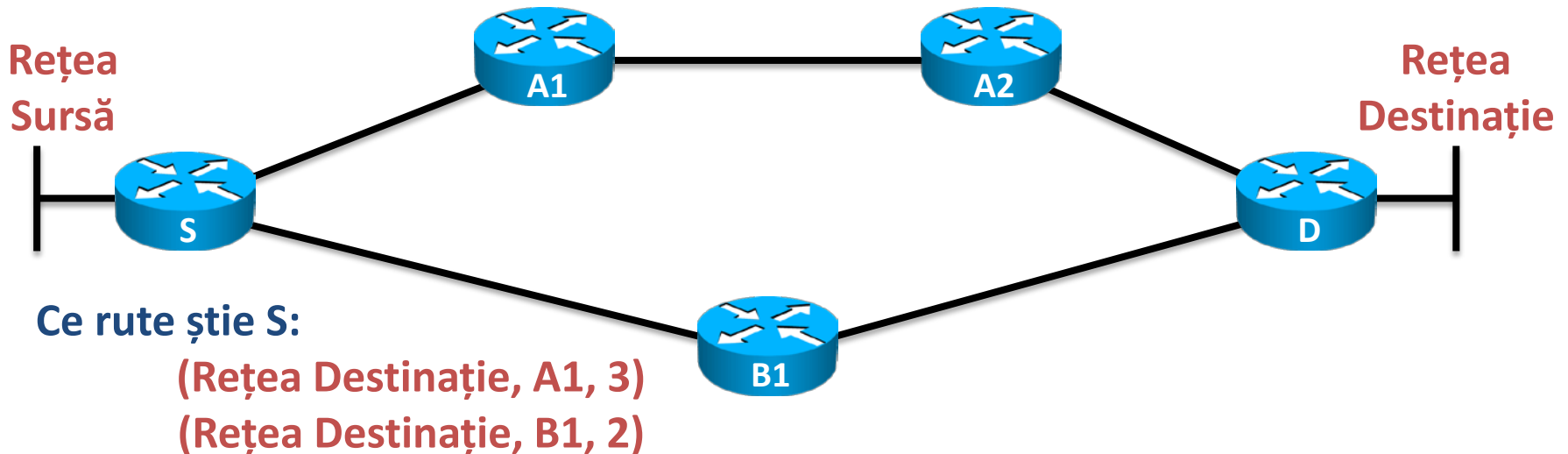


Metrică

- Deoarece distanța nu este o mărime foarte utilă în rețele, trebuie găsite alte mărimi ce descriu mai bine calitatea unei rute
- Mărimea asociată unei rute poartă numele de **metrică**
- Metrici utile sunt:
 - hop count (numărul de rutere până la destinație)
 - lățimea de bandă a legăturii
 - încărcarea unei legături
 - fiabilitatea (reliability)
 - costul
 - latența
- Metricile de bază pot fi compuse pentru a crea metrici noi

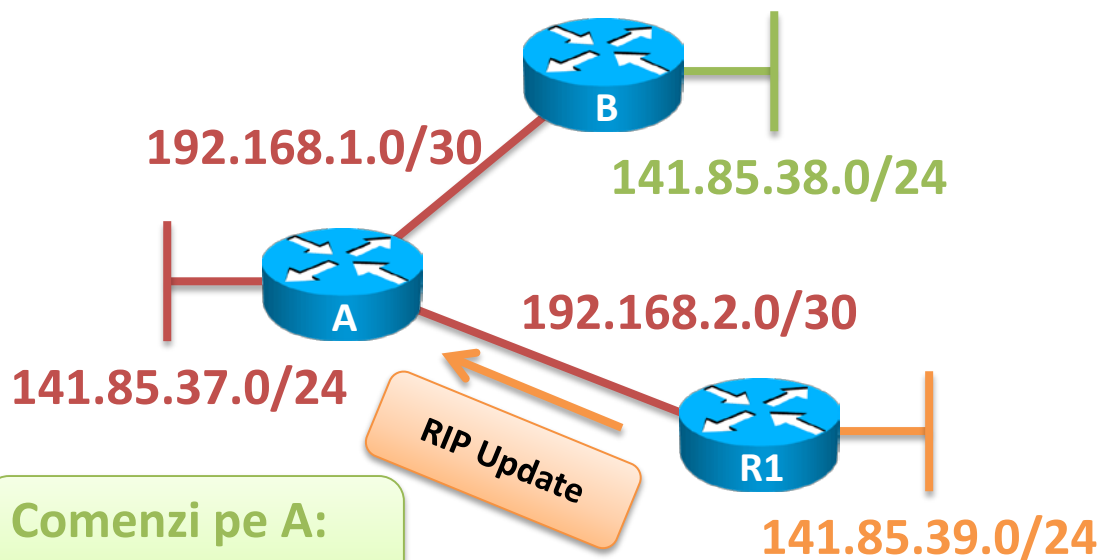
Metrica hop-count

- Metrica ajută un ruter în a lua o decizie când există mai multe căi către destinație
- În cazul acesta, calea prin **B1** este mai bună
- Direcția poate fi reprezentată printr-un **IP** sau printr-o **interfață**



Surse de rute

- Când un ruter neconfigurat este pornit, acesta nu cunoaște nicio rută
- Rutele trebuie învățate din diferite surse; acestea sunt:
 - Rețelele direct conectate (marcate prin simbolul **C** – **connected**)
 - Rute statice configurate de administrator (marcate prin simbolul **S** – **static**)
 - Rute învățate de la alte rutere prin protocoale dinamice de rutare (**R**, **D**, **O**)



Comenzi pe A:
ip route

Ce rute știe A:

C (141.85.37.0/24, Fa0/0)

C (192.168.1.0/30, Fa0/1)

S (141.85.38.0/24, B)

C (192.168.2.0/30, Fa0/2)

R (141.85.39.0/24, R1, 1)

Rute statice

- Rutele statice sunt rute configurate manual de administrator
- O rută statică poate folosi ca direcție:
 - **O interfață** – dacă interfața nu aparține unui mediu multi-acces

Exemplu - legătură serială:

```
A# ip route 192.168.10.0/24 Se0/0
```



- **IP Next hop** – poate fi folosită în orice situație, dar este mai lentă

Exemplu – legătură Ethernet:

```
A# ip route 192.168.10.0/24 10.0.0.3
```

Problemă: De ce nu merge soluția

```
A# ip route 192.168.10.0/24 Fa0/0 ?
```

R: Nu știm ce MAC destinație să punem în cadru.

Poate funcționa doar cu ajutorul Proxy ARP

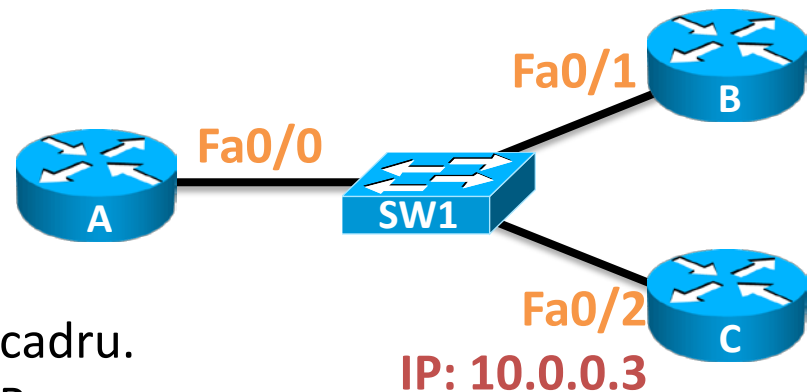


Tabela de rutare

- Pe măsură ce ruterul învață rute își alcătuiește pe baza acestora **tabela de rutare**
- Tabela de rutare este harta ruterului către rețeaua din jur; toate deciziile de dirijare vor fi luate pe baza acestei tabele
- Când există mai multe posibilități de a ajunge într-o rețea destinație, doar ruta optimă va ajunge în tabela de rutare
- Tabela de rutare este o versiunea eficientă a tuturor rutelor pe care un ruter le cunoaște

Tabela de rutare: Exemplu

- Se consideră un ruter cu următoarele rute cunoscute:

1. C (10.0.0.8/30, Fa0/0)

2. R (141.85.37.0/24, <IP>, 3)

3. C (10.0.0.4/30, Fa0/1)

4. R (141.85.37.0/24, <IP>, 2)

5. D (200.0.0.0/16, <IP>, 31452)

6. R (200.0.0.0/16, <IP>, 3)



1. C (10.0.0.8/30, Fa0/0)

3. C (10.0.0.4/30, Fa0/1)

4. R (141.85.37.0/24, <IP>, 2)

5. D (200.0.0.0/16, <IP>, 31452)

- Care din aceste rute ar trebui să ajungă în tabela de rutare?

- R: **1.** și **3.** vor ajunge pentru că sunt singurele rute către destinațiile 10.0.0.8/30 și 10.0.0.4/30
- R: **2.** și **4.** duc spre aceeași destinație, însă **4.** are o metrică mai bună
- R: **5.** și **6.** duc spre aceeași destinație și **6.** pare să aibă o metrică mai bună, însă sursele sunt diferite; este metrica un criteriu valid pentru clasificare în acest caz?

Distanță administrativă

- Atunci când există mai multe protocoale ce oferă căi către aceeași destinație trebuie să existe o metodă de a le putea clasifica
- Mărimea folosită în acest caz este **distanța administrativă**
- Distanță administrativă este specifică sursei rutei:

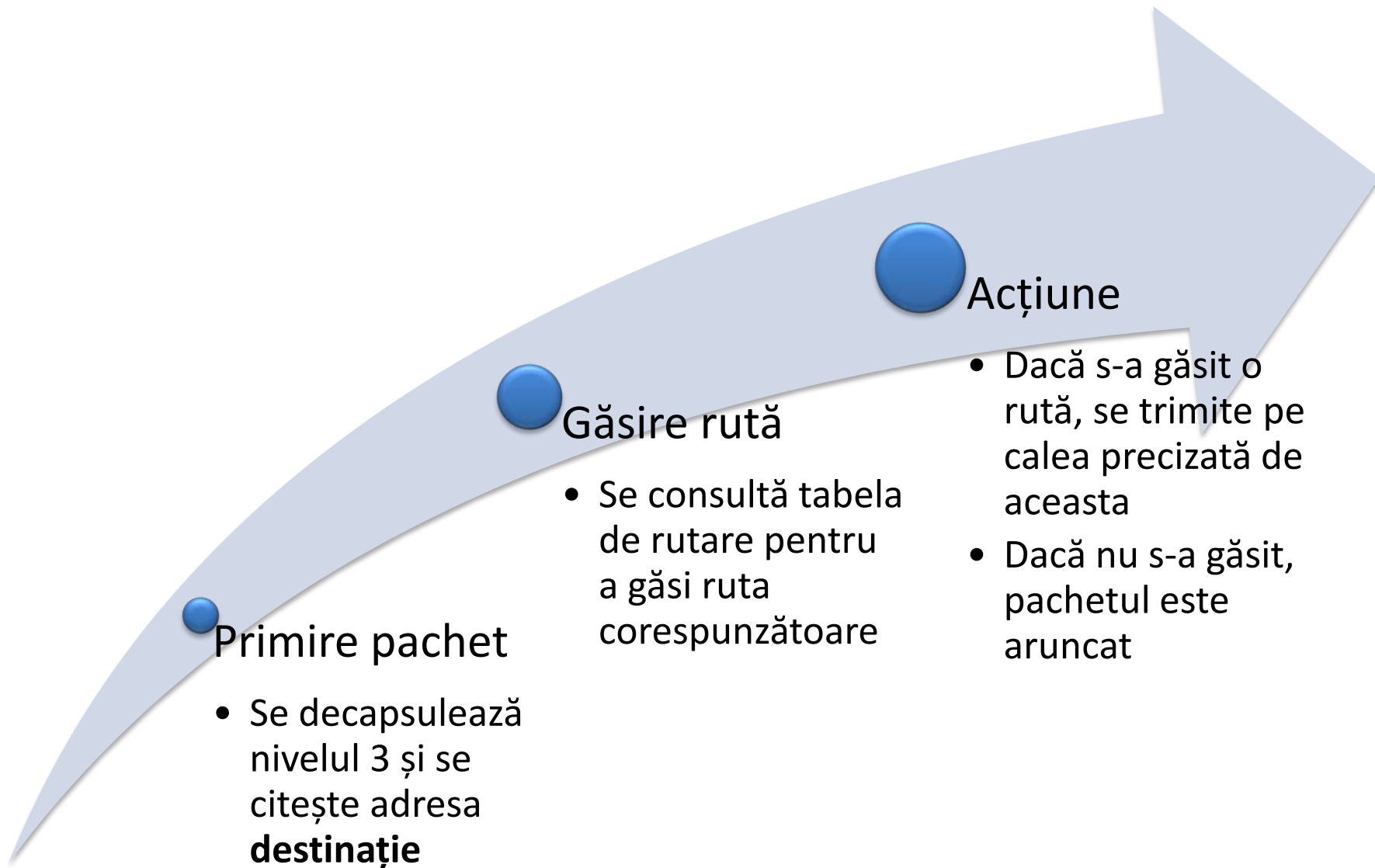
Simbol	Nume	AD
C	Connected	0
S	Static route	1
D	EIGRP	90
O	OSPF	110
i	IS-IS	115
R	RIP	120

- Rutele adăugate în tabela de rutare sunt cele cu un AD cât mai **mic**

Ordonarea tabelii de rutare

- Tabela de rutare este organizată de la rutele cele mai specifice (cu mască mare) către cele mai generale:
 1. R (10.0.0.8/30, <IP>, 3)
 2. R (11.0.0.0/26, <IP>, 2)
 3. R (12.0.0.0/20, <IP>, 2)
 4. R (13.0.0.0/14, <IP>, 4)
 5. R (13.0.0.0/8, <IP>, 3)
- Această organizare ajută în eficientizarea procesului de rutare— se va încerca trimiterea pachetelor pe cea mai specifică rută spre destinație

Procesul de rutare



Consultarea tabelii de rutare

- Căutarea rutei se face secvențial, pe baza adresei **IP destinație**
- Pentru fiecare rută din tabelă se face **AND** între mască și adresa IP destinație a pachetului
- Dacă rezultatul corespunde cu rețeaua din rută pachetul este trimis pe calea respectivă

IP Sursă: 192.168.10.1; IP Dest: 12.0.15.23

12.0.15.23 & 255.255.255.252 = 12.0.15.20

12.0.15.23 & 255.255.255.192 = 12.0.15.0

12.0.15.23 & 255.255.240.0 = 12.0.0.0

~~1. R (10.0.0.8/30, <IP>, 3)~~

~~2. R (11.0.0.0/26, <IP>, 2)~~

3. R (12.0.0.0/20, <IP>, 2)

4. R (13.0.0.0/14, <IP>, 4)

5. R (13.0.0.0/8, <IP>, 3)

Ruta default

- Ruta default este o rută specială care face match pe orice destinație
- Mai este denumită și ruta quad-zero datorită formatului:

S (0.0.0.0/0, Se0/0)

- Unde ar fi plasată această rută într-o tabelă de rutare?
 - R: pe ultima poziție deoarece are cea mai generală mască
- De ce face match pe orice destinație?

12.0.15.23 & 0.0.0.0 = 0.0.0.0

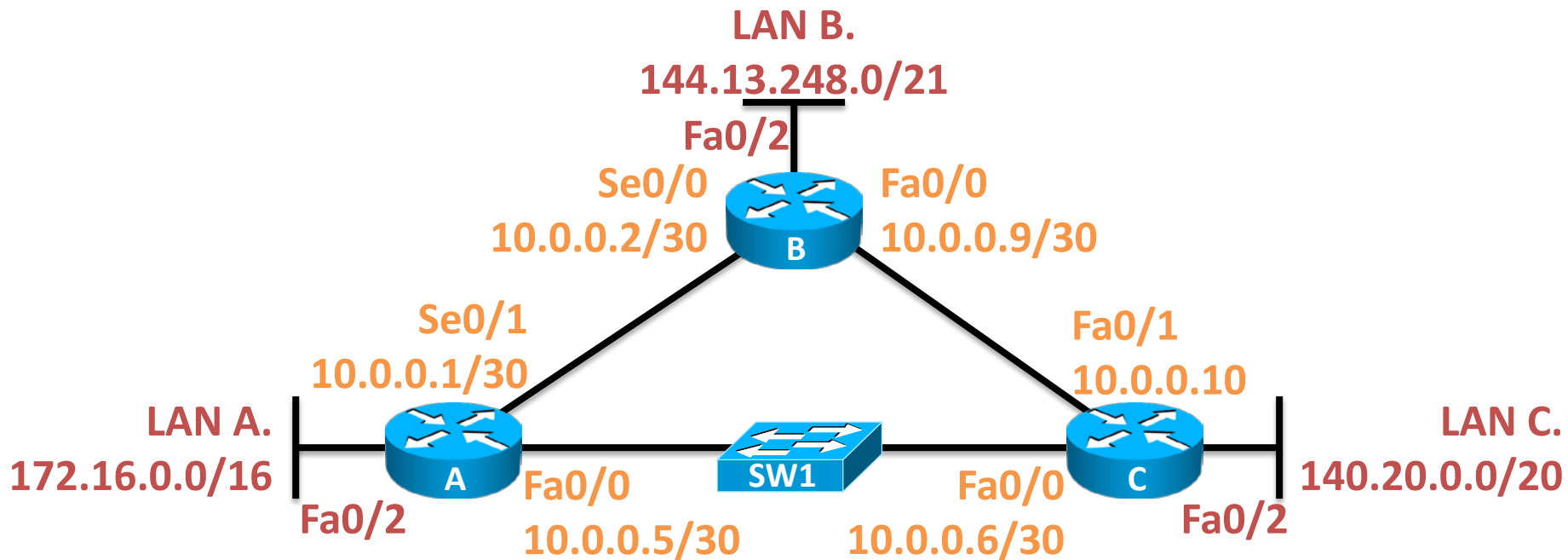
S (0.0.0.0/0, Se0/0)

Legile rutării

Rutarea se face individual, pentru fiecare pachet în parte

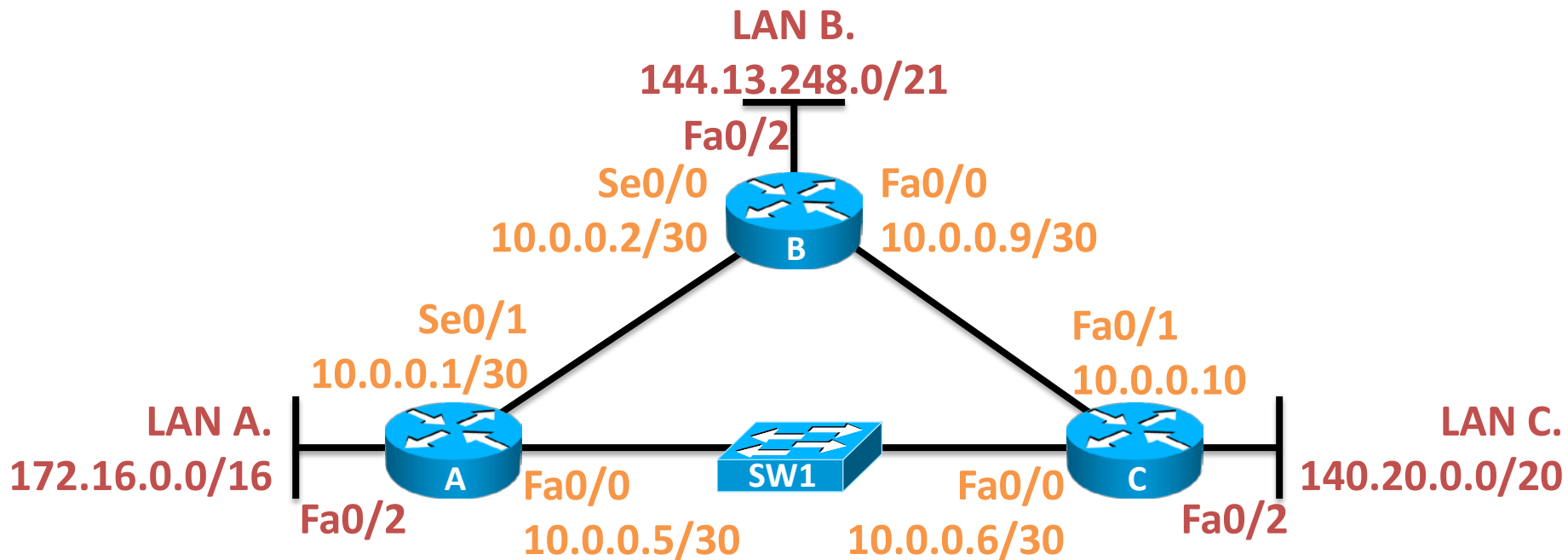
Fiecare ruter ia decizia **doar** pe baza propriei sale tabele de rutare

Exercițiu



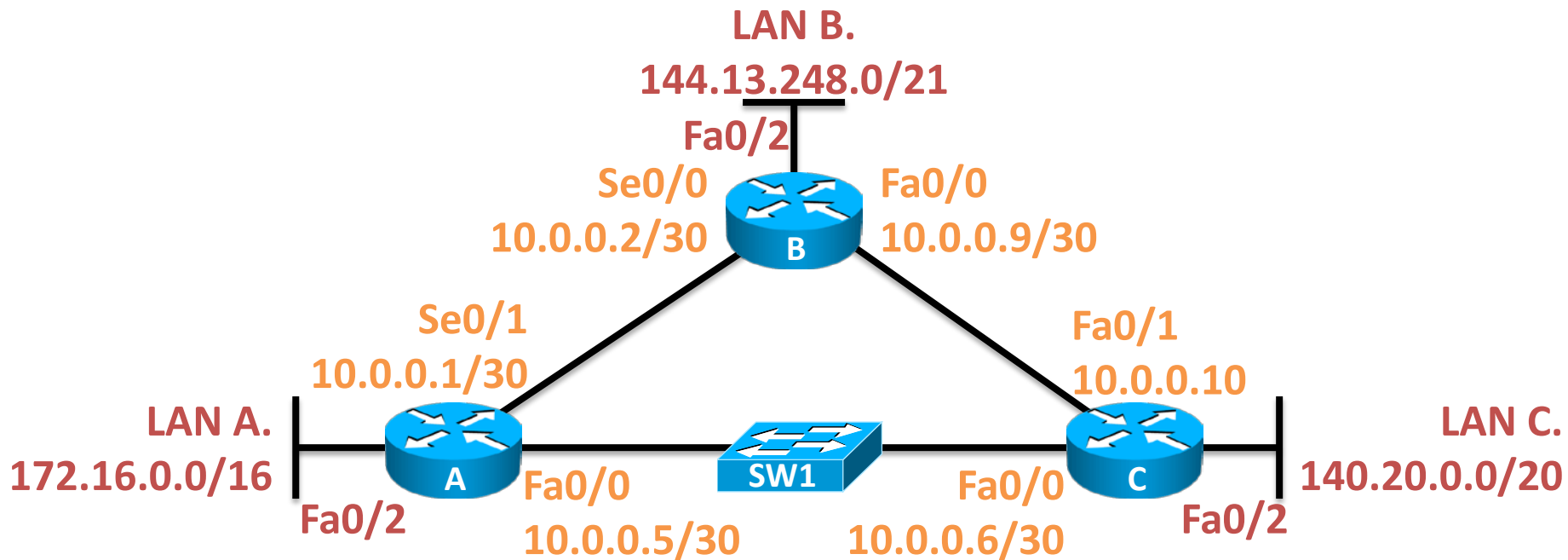
- Routerul **A** abia a fost pornit cu o configurație vidă. Ce va conține tabela sa de rutare după ce interfețele sunt pornite?
 - R: **C (172.16.0.0/16, Fa0/2)**
C (10.0.0.1, Se0/1)
C (10.0.0.5, Fa0/0)

Exercițiu



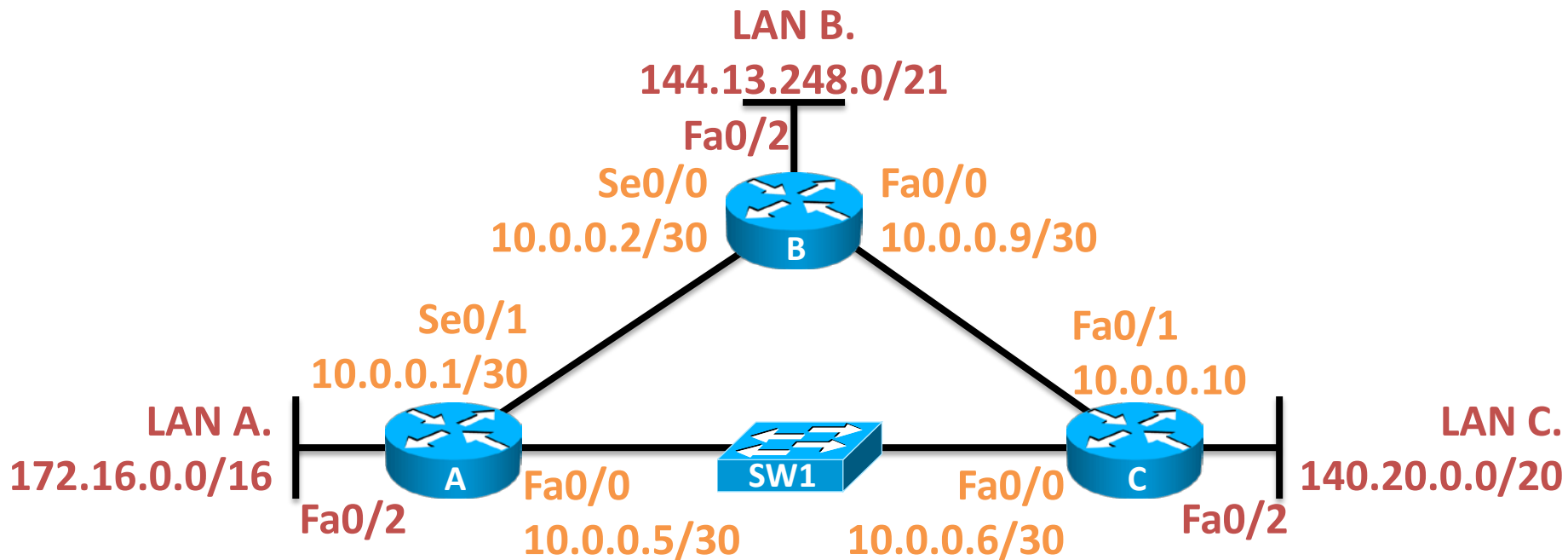
- Configurați rute statice a.î. **LAN A** să aibă conectivitate cu **LAN C**. Folosiți calea optimă.
 - R: **A# ip route 140.20.0.0/20 10.0.0.6**
C# ip route 172.16.0.0/16 10.0.0.5
- De ce nu funcționează varianta cu interfață de ieșire?

Exercițiu



- Configurați rute statice a.î. LAN B să aibă conectivitate cu LAN C. Folosiți calea optimă.
 - R: **B# ip route 140.20.0.0/20 10.0.0.10**
C# ip route 144.13.248.0/21 10.0.0.9
- Ar funcționa varianta cu interfață de ieșire în acest caz?

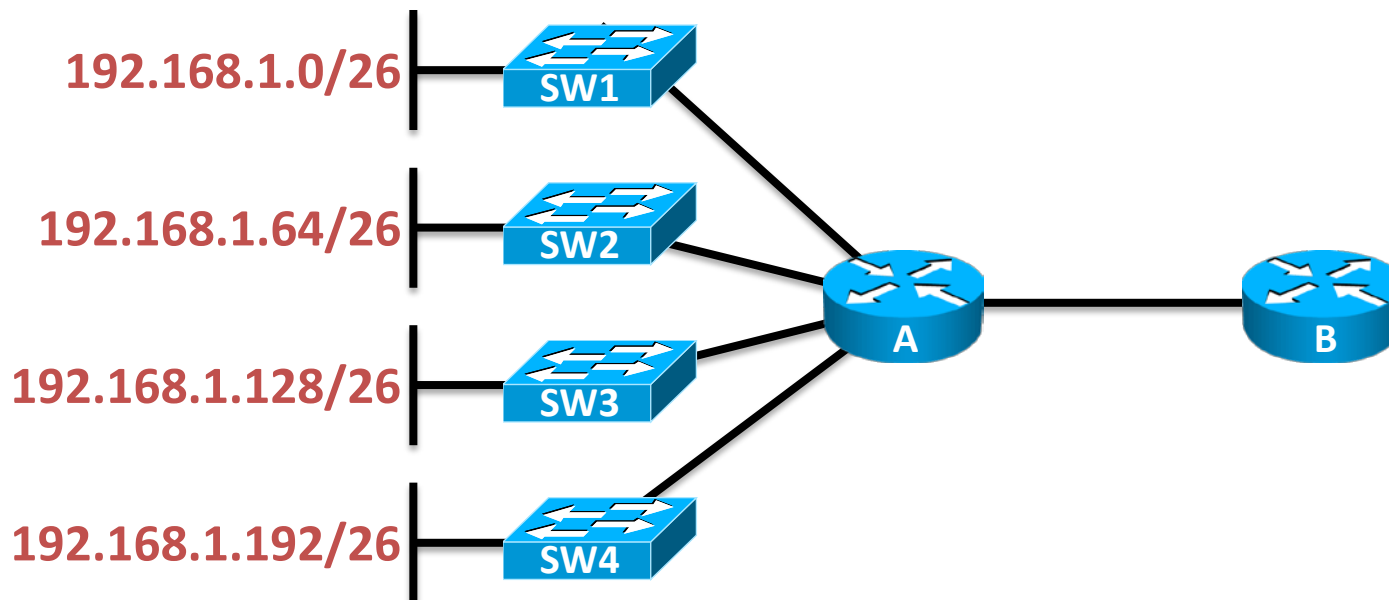
Exercițiu



- Creați o buclă de rutare a.î. pachetele din LAN A să nu ajungă niciodată în LAN B.
 - R: **A# ip route 144.13.248.0/21 10.0.0.6**
C# ip route 144.13.248.0/21 10.0.0.5
- Vor circula la infinit pachetele acestea?

Sumarizarea rutelor

- Se consideră următoarea topologie:



- În loc de 4 rute statice același efect poate fi obținut cu o singură rută:

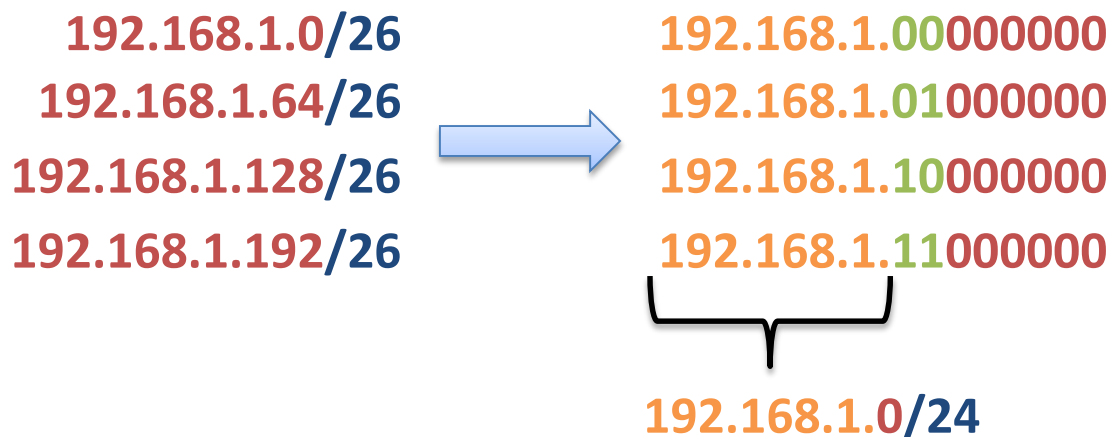
192.168.1.0/26
192.168.1.64/26
192.168.1.128/26
192.168.1.192/26

↔

192.168.1.0/24

Sumarizarea rutelor

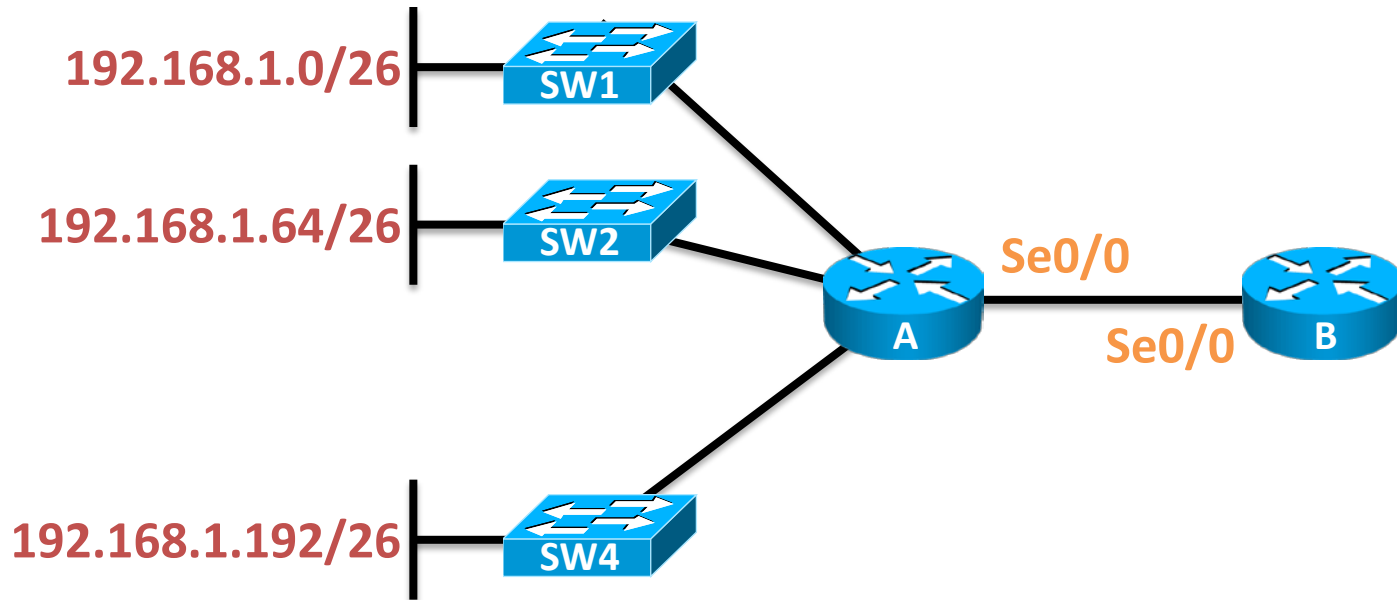
- Procesul poartă numele de **sumarizarea rutelor**
- Rutele sumarizate se calculează prin transformarea în baza 2 și observarea segmentului comun între adresele de rețea:



- Avantajul este micșorarea tabelii de rutare care duce la căutări mult mai rapide
- Există protocoale de rutare care pot sumariza automat

Interfețe nule

- Uneori este necesară forțarea aruncării unui pachet

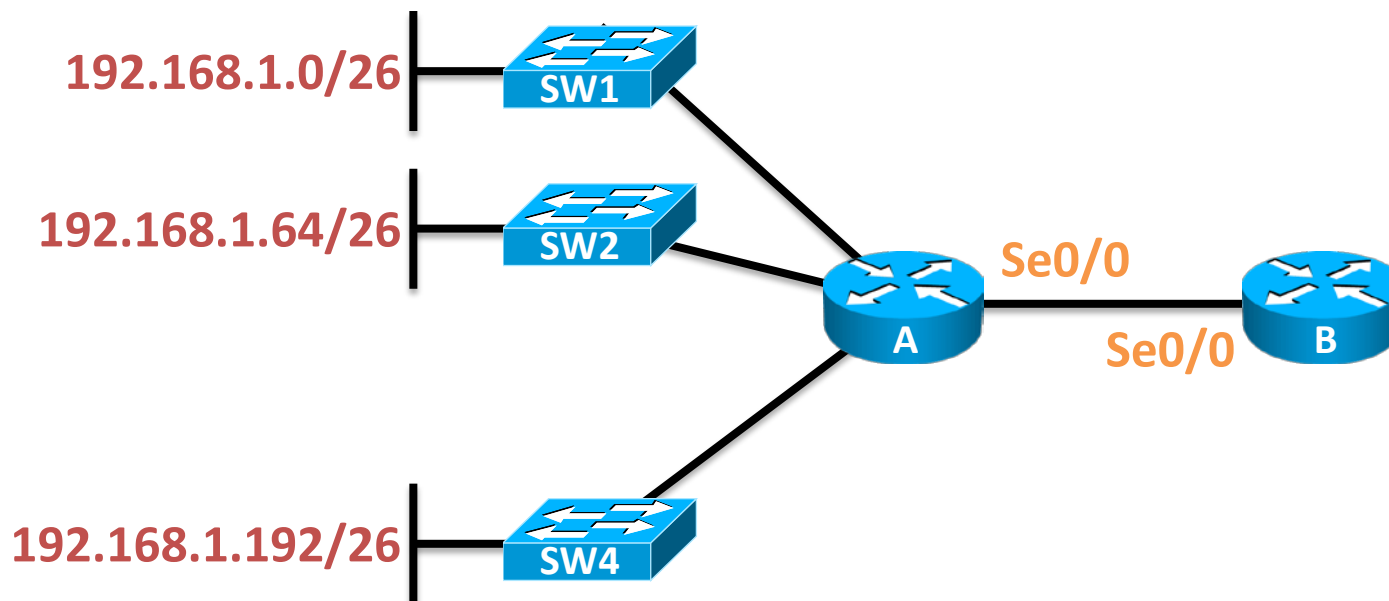


- Considerăm că pe ruterul B a fost configurată ruta statică:

S (192.168.1.0/24, Se0/0)

- Această rută va trimite lui A pachetele destinate rețelei **192.168.1.128/26**, chiar dacă aceasta nu mai există

Interfețe nule



- În această situație putem adăuga pe **B** următoarea rută statică:
S (192.168.1.128/26, Null0)
- Pachetele ce vor face match pe această rută vor fi aruncate direct de către **B** (nu vor mai ajunge la **A**)
- În Linux interfața logică nulă este **/dev/null**

Exercițiu

- Se dă următoarea tabelă de rutare:
 1. **C** (172.30.14.0/30, Fa0/0)
 2. **C** (172.30.14.4/30, Fa0/1)
 3. **S** (192.168.3.0/24, Null0)
 4. **S** (192.168.5.0/24, Null0)
 5. **S** (192.168.0.0/20, 172.30.14.2)
 6. **S** (0.0.0.0/0, 172.30.14.6)
- Pe ce regulă vor face match următoarele destinații și ce se va întâmpla cu fiecare pachet?

MAC - Dest	MAC - Sursă	IP - Sursă	IP - Dest
00:02:16:87:16:01	00:02:17:6D:B9:96	172.30.14.2	192.168.32.6

R: 7; forward

00:02:16:87:16:01	00:02:17:2F:F1:04	172.30.14.4	192.168.3.6
-------------------	-------------------	-------------	-------------

R: 4; drop

Exercițiu

- Se dă următoarea tabelă de rutare:
 1. **C** (172.30.14.0/30, Fa0/0)
 2. **C** (172.30.14.4/30, Fa0/1)
 3. **S** (192.168.9.0/24, Null0)
 4. **S** (192.168.3.0/24, Null0)
 5. **S** (192.168.5.0/24, Null0)
 6. **S** (192.168.0.0/20, 172.30.14.2)
 7. **S** (0.0.0.0/0, 172.30.14.6)
- Pe ce regulă vor face match următoarele destinații și ce se va întâmpla cu fiecare pachet?

MAC - Dest	MAC - Sursă	IP - Sursă	IP - Dest
00:02:16:87:16:01	00:02:17:04:8A:16	200.0.1.255	172.30.14.9

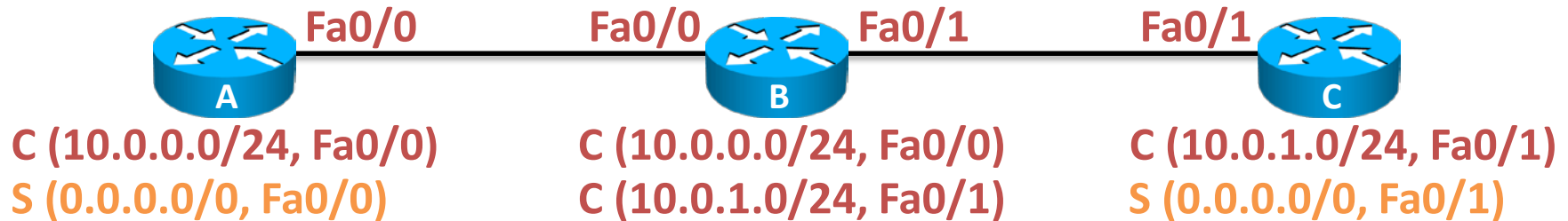
R: 7; forward

00:02:16:87:16:01	00:02:17:F0:B4:64	172.30.14.2	192.168.9.14
-------------------	-------------------	-------------	--------------

R: 6; forward

Proxy ARP

- Se dă următoarea rețea, cu următoarele rute în tabelă:



- Se consideră că interfețele au următoarele adrese configurate:

Fa0/0: 10.0.0.2/24

Fa0/0: 10.0.0.1/24

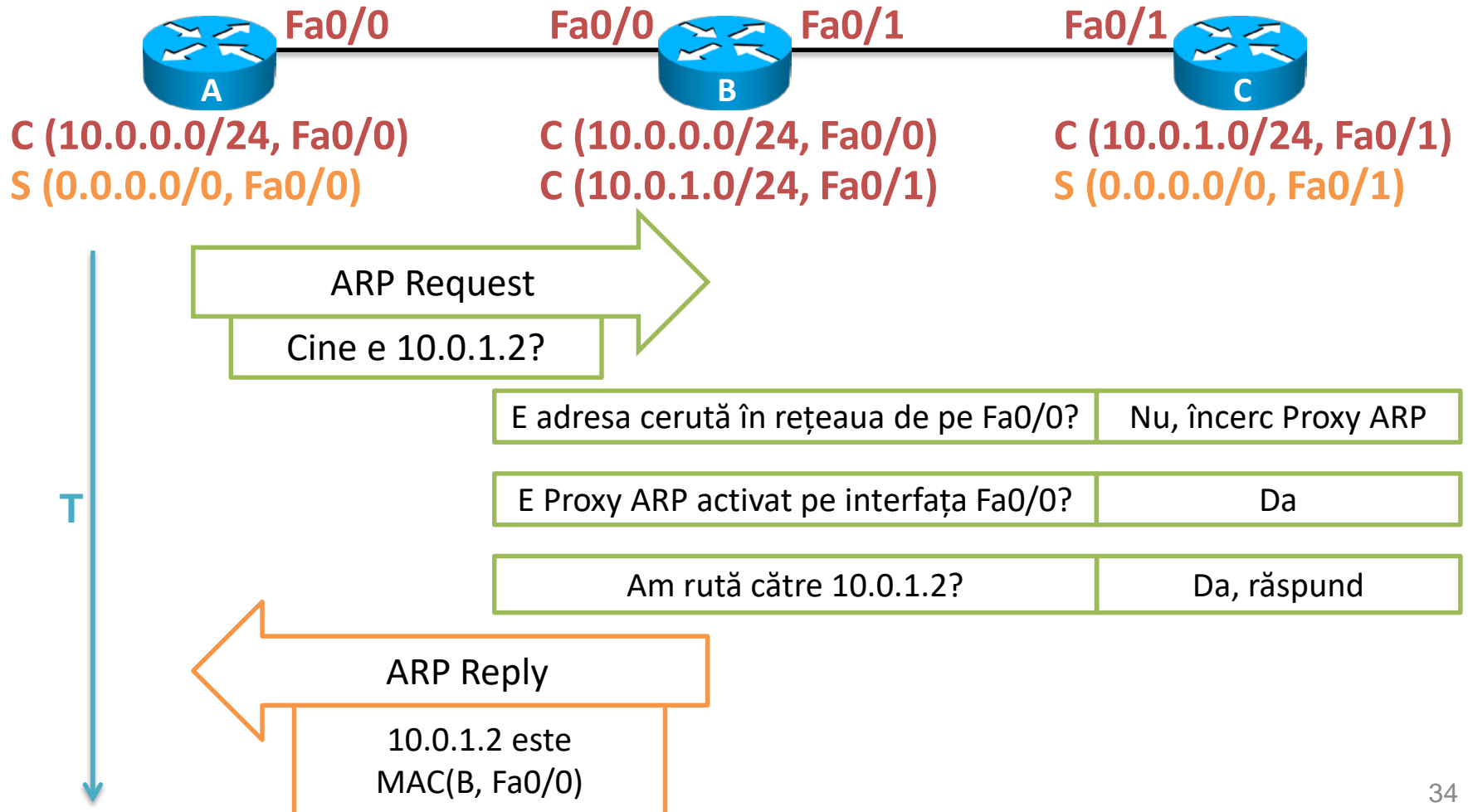
Fa0/0: 10.0.1.2/24

Fa0/1: 10.0.1.1/24

- O problemă în această rețea este reprezentată de cele două rute statice ce au ca interfață de ieșire un mediu multi-acces
- În absența Proxy ARP pe **B** nu ar putea da ping **A** în **C**

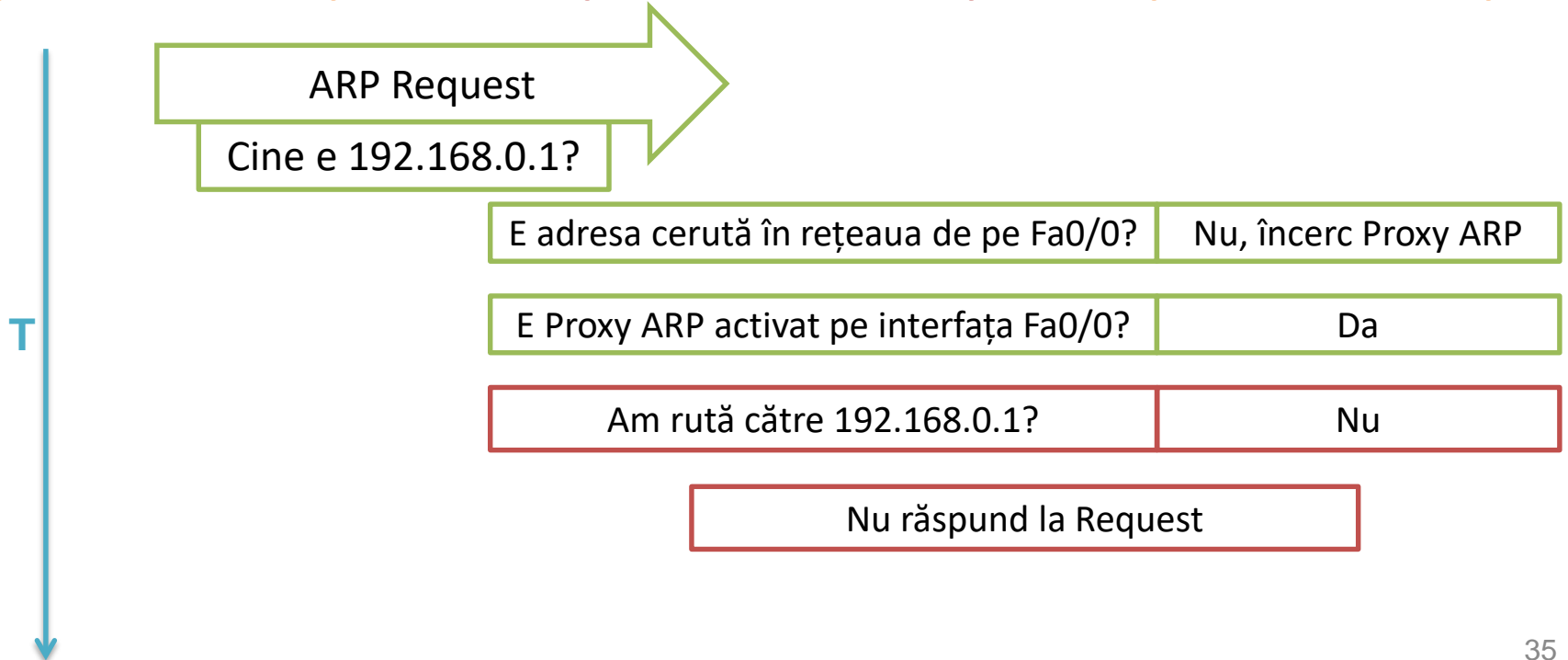
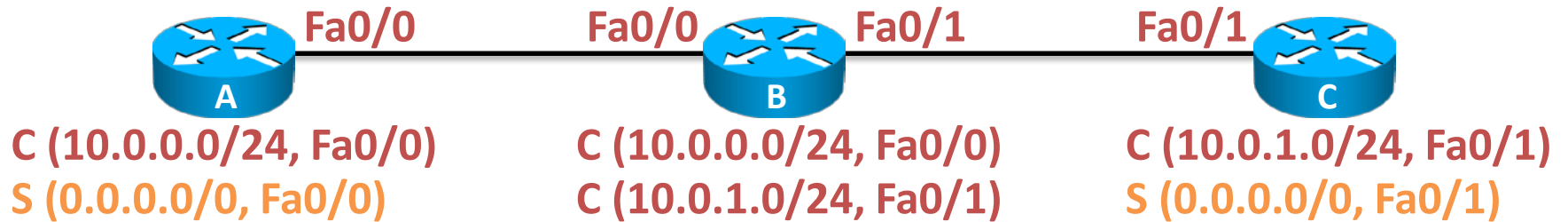
Proxy ARP

- Ce se întâmplă dacă **A** dă ping în **C**?



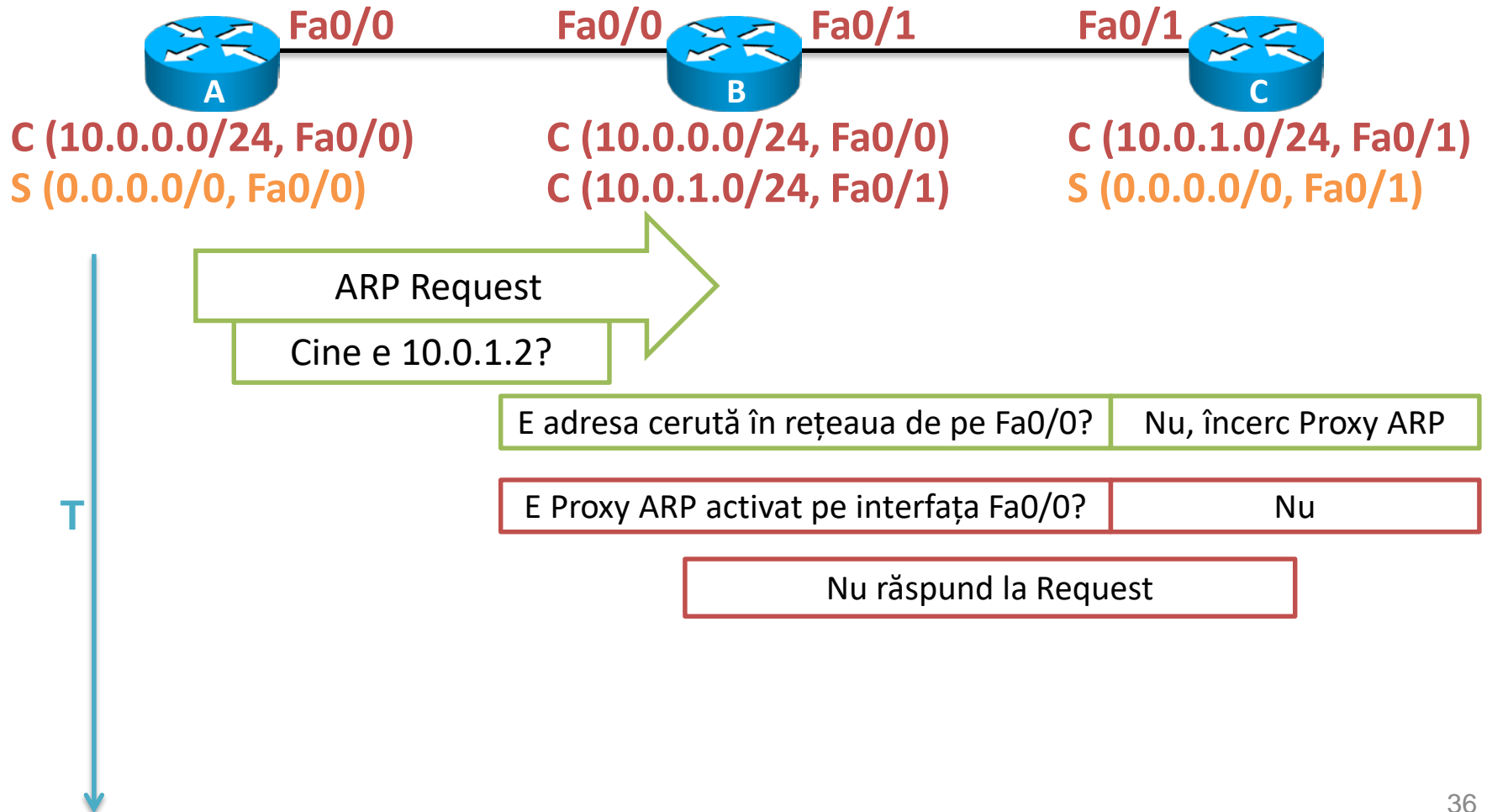
Proxy ARP

- Când nu răspunde **B** la cererea ARP?

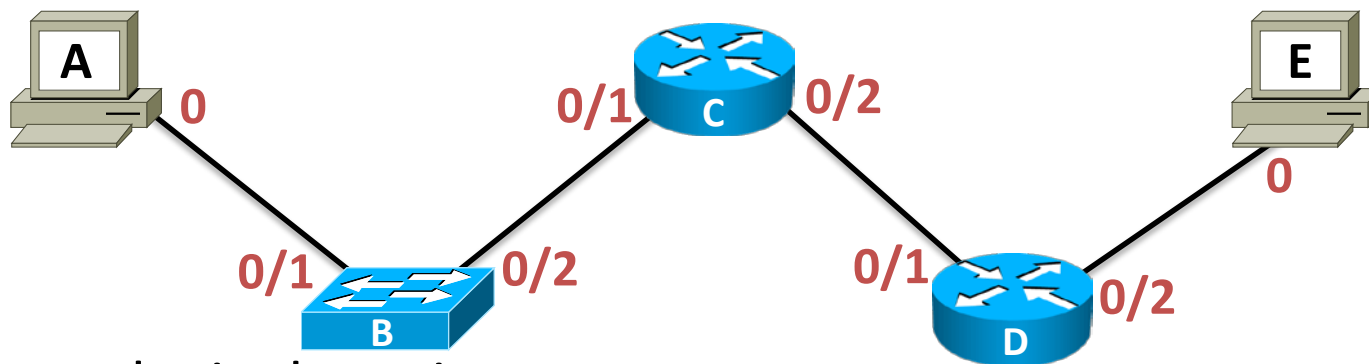


Proxy ARP

- Când nu răspunde **B** la cererea ARP?



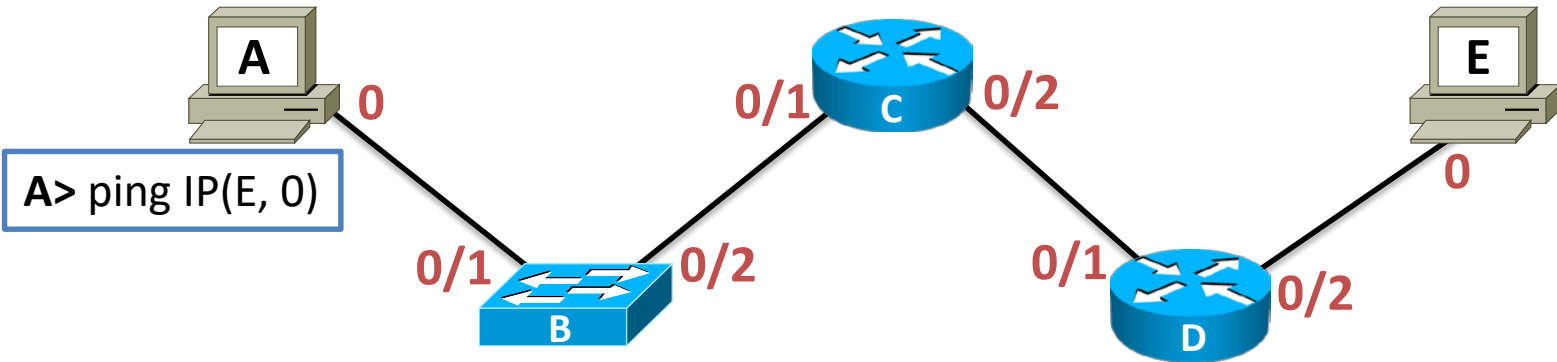
Adrese sursă și destinație

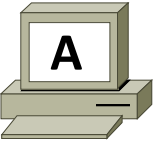







- În topologia de mai sus:
 - adresele de nivel 2 sunt de forma MAC(C, 0/1)
 - adresele de nivel 3 sunt de forma IP(C, 0/1)
 - toate dispozitivele abia au fost inițializate și switch-ul nu rulează STP
 - host-urile au setate default gateway-uri corecte
 - ruterele cunosc toate rețelele prin rute statice cu next-hop
- Ce adrese MAC și IP sursă și destinație vor avea pachetele din rețea la rularea comenzii?

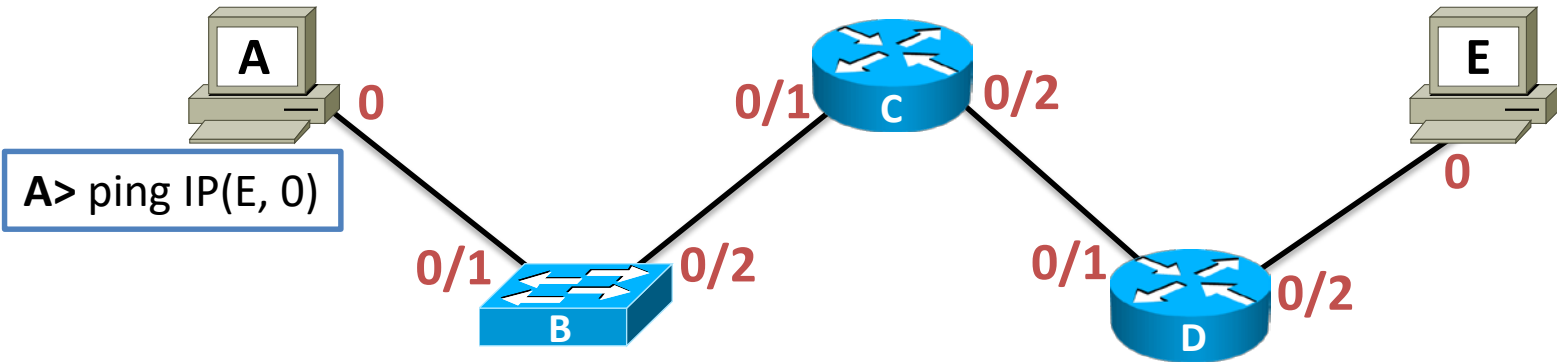
```
A> ping IP(E, 0)
```


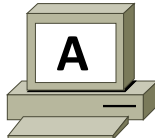




Adrese sursă și destinație



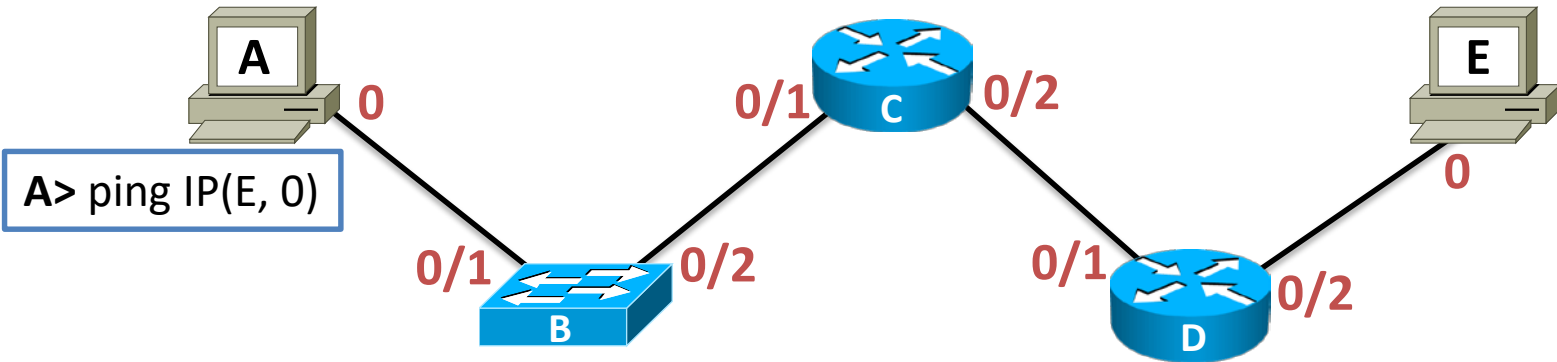
Sursă	Destinație	Protocol superior	Adrese sursă (IP și Ethernet)		Adrese destinație (IP și Ethernet)	
		ARP	MAC(A, 0)		FF.FF.FF.FF.FF.FF	
		ARP	MAC(A, 0)		FF.FF.FF.FF.FF.FF	
		ARP	MAC(C, 0/1)		MAC(A, 0)	







Adrese sursă și destinație



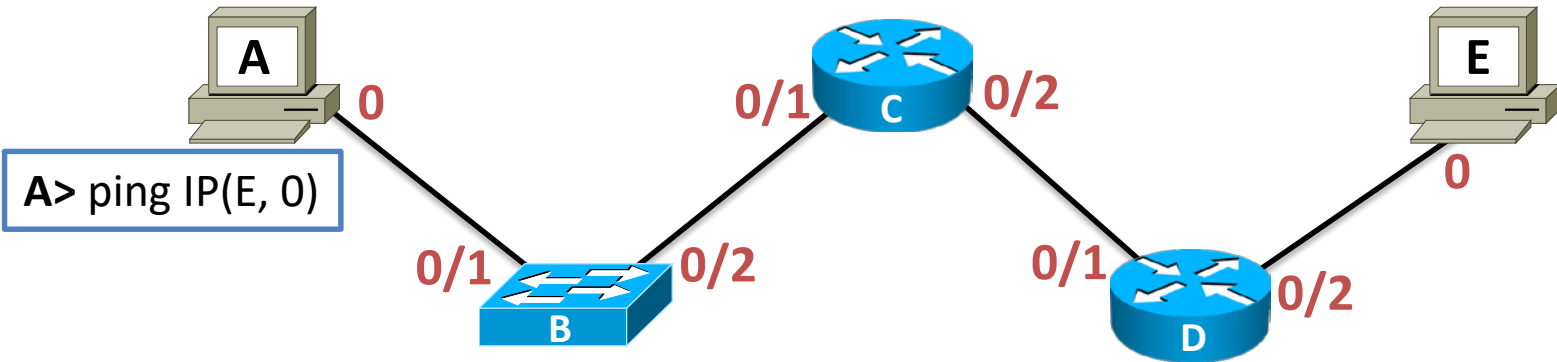
Sursă	Destinație	Protocol superior	Adrese sursă (IP și Ethernet)		Adrese destinație (IP și Ethernet)	
		ARP	[Redacted]		MAC(A, 0)	
		ICMP	IP(A, 0)		IP(E, 0)	
			MAC(A, 0)		MAC(C, 0/1)	
		ICMP	IP(A, 0)		IP(E, 0)	
			MAC(A, 0)		MAC(C, 0/1)	


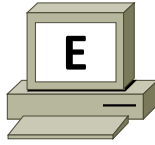
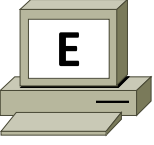


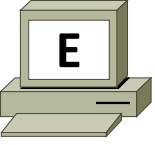
Adrese sursă și destinație



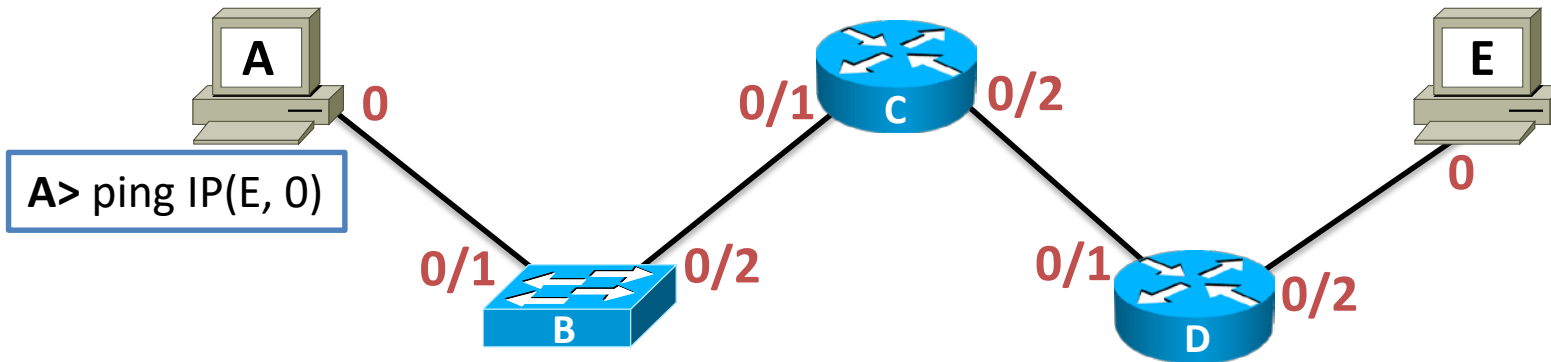
Sursă	Destinație	Protocol superior	Adrese sursă (IP și Ethernet)		Adrese destinație (IP și Ethernet)	
		ARP	MAC(C, 0/2)		FF.FF.FF.FF.FF.FF	
		ARP	MAC(D, 0/1)		MAC(C, 0/2)	
		ICMP	IP(A, 0)	IP(E, 0)	MAC(C, 0/2)	MAC(D, 0/1)

Adrese sursă și destinație



Sursă	Destinație	Protocol superior	Adrese sursă (IP și Ethernet)		Adrese destinație (IP și Ethernet)	
		ARP	MAC(D, 0/2)		FF.FF.FF.FF.FF.FF	
		ARP	MAC(E, 0)		MAC(D, 0/2)	
		ICMP	IP(A, 0)	IP(E, 0)	MAC(D, 0/2)	MAC(E, 0)

Adrese sursă și destinație



- Similar arată și traficul de întoarcere ICMP Echo-Reply
- Concluzii:
 - Adresele IP sursă și destinație rămân constante
 - Adresele MAC sursă și destinație variază pe fiecare segment Ethernet
 - Pot fi necesare multiple interogări ARP pentru ca pachetul să străbată toată calea

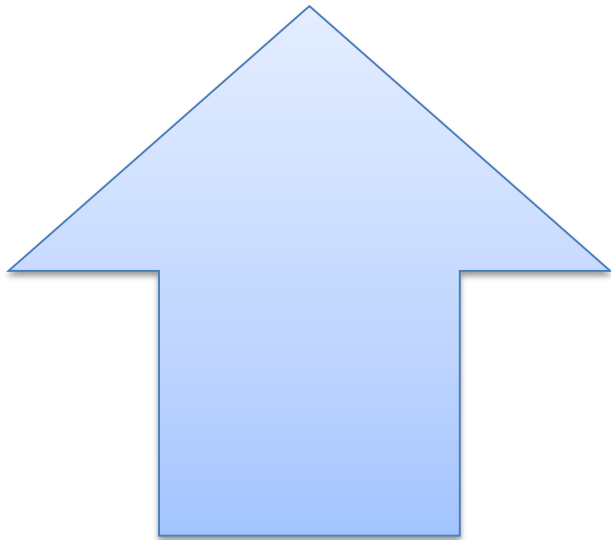
Procesoale dinamice de rutare

- Definiție
- Avantaje și dezavantaje
- Exemple

Protocoale dinamice de rutare

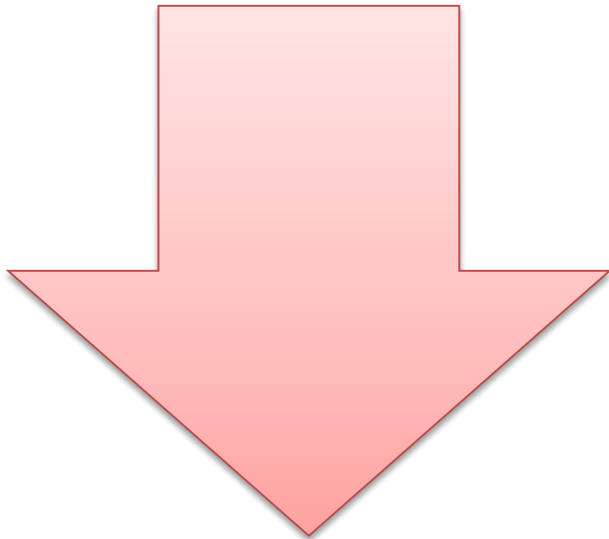
- Infrastructura Internetului este formată din mii de rutere și milioane de rețele
- Asigurarea conectivității între toate aceste rețele numai cu rute statice ar fi un pic cam complicat
- Protocoalele dinamice de rutare sunt folosite de rutere pentru a comunica automat între ele informații:
 - Despre rețelele cunoscute
 - Despre schimbările de topologie (de exemplu dacă o legătură pică)

Protocoale dinamice vs rute statice



Avantaje:

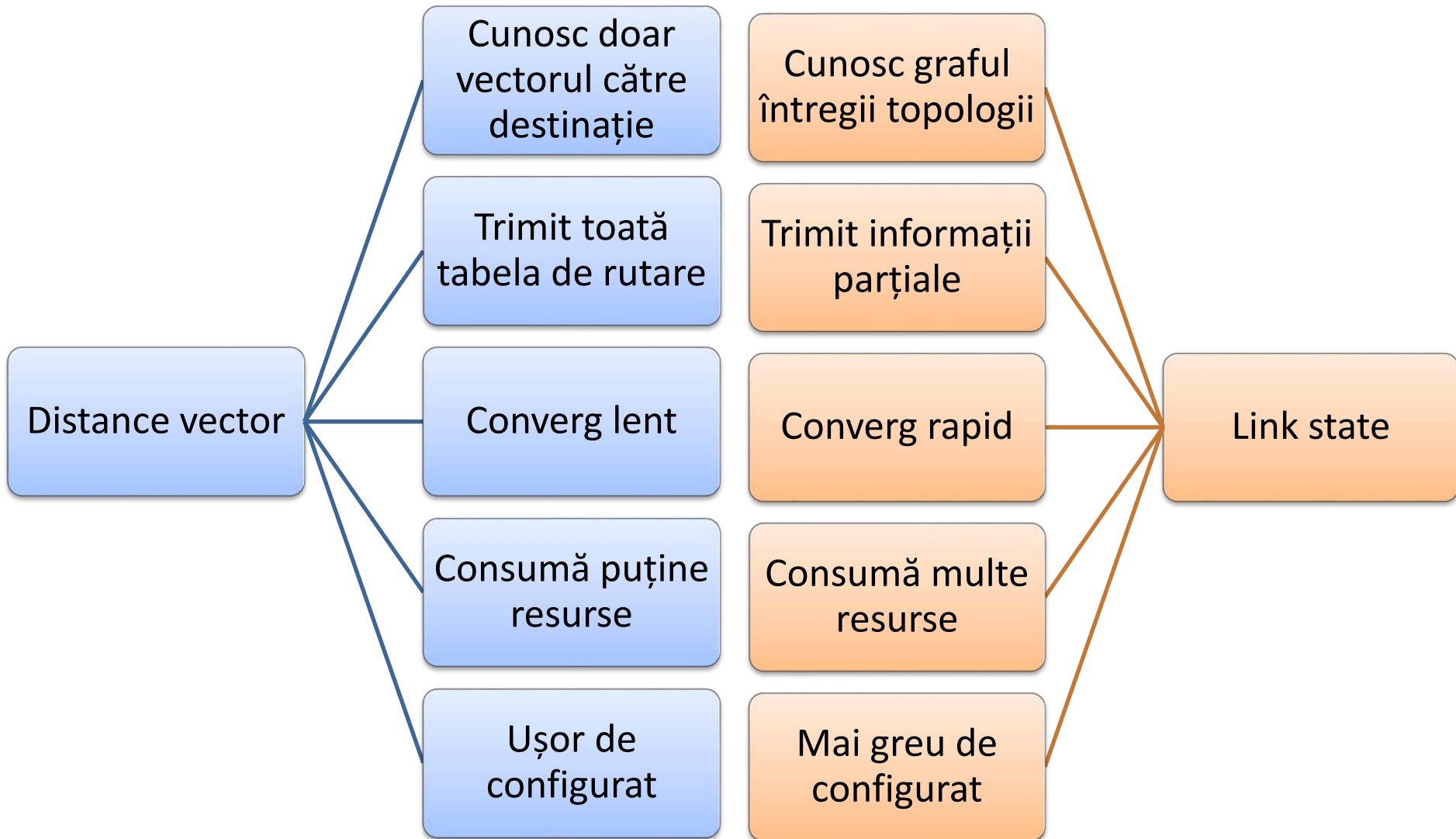
- Mai ușor de configurat pe rețele mari
- Scalabile
- Răspund automat la modificările de topologie
- Permit implementarea unor politici de rutare complexe



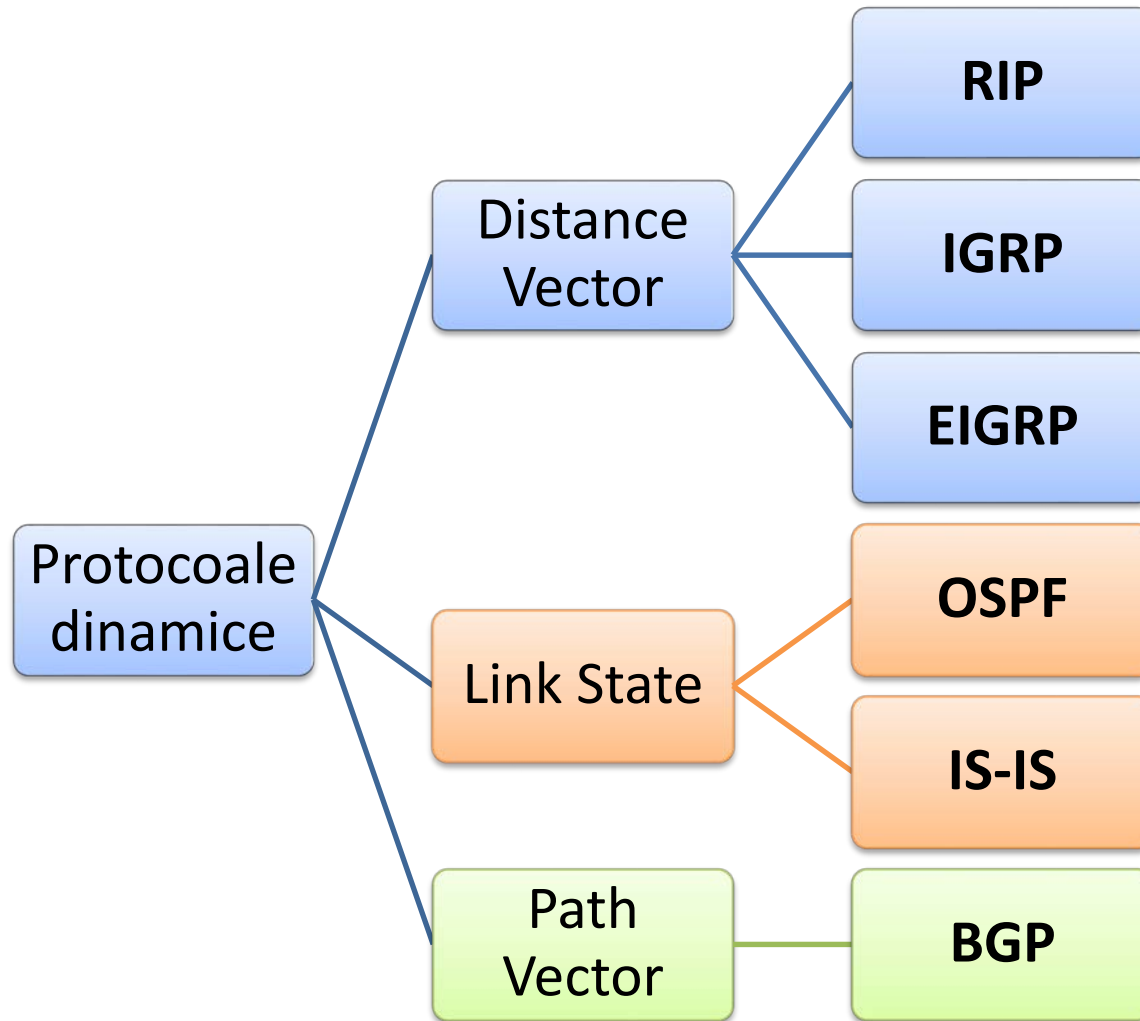
Dezavantaje:

- Consumă mai multe resurse pe rutere (memorie și procesare)
- Ruterele trebuie să fie capabile să ruleze respectivele protocoale
- Administratorul trebuie să fie familiarizat cu funcționarea protocoalelor

Protocele dinamice de rutare

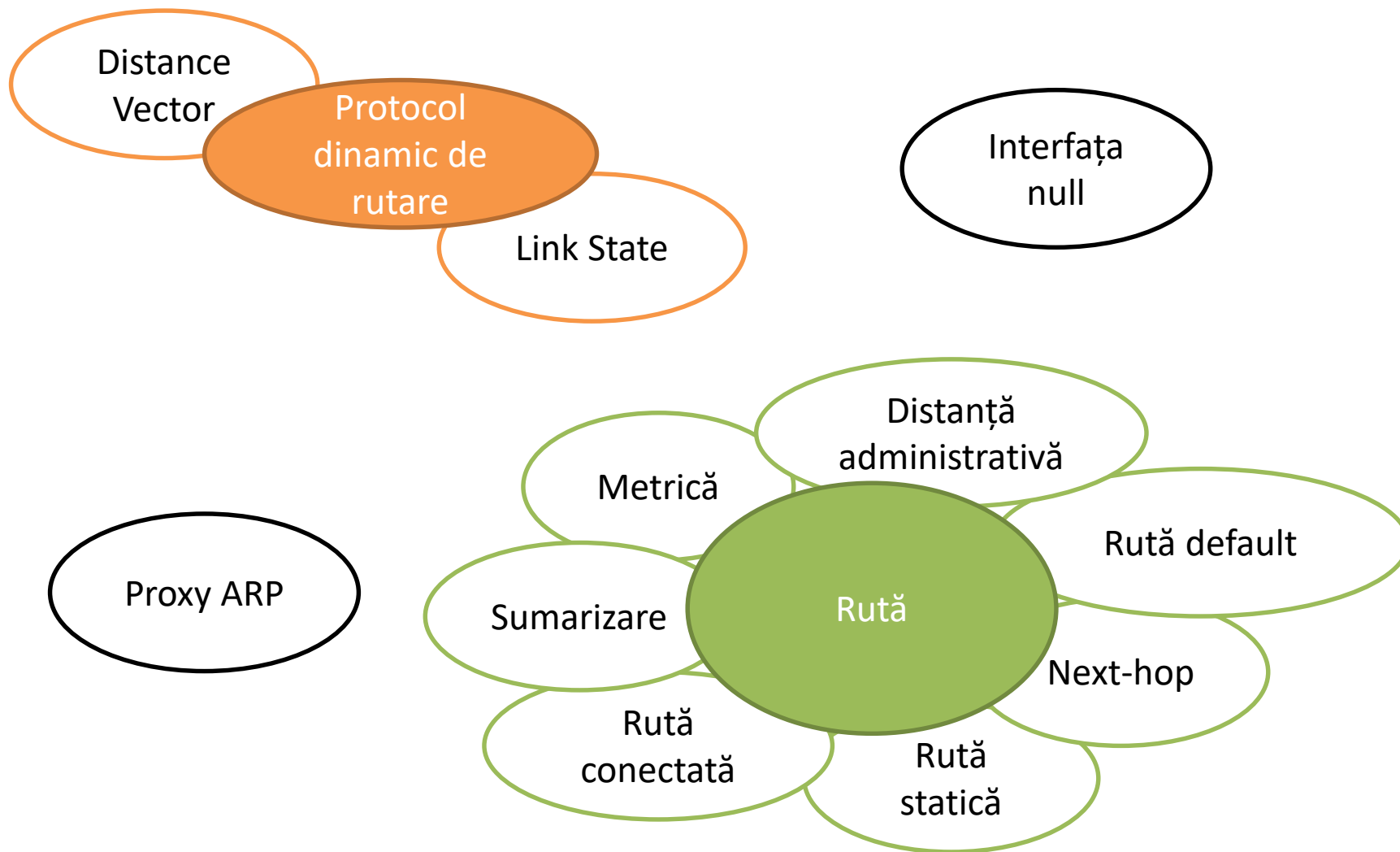


Exemple de protocoale dinamice de rutare



Protocoalele dinamice sunt studiate în detaliu la cursul de Proiectarea Rețelelor din anul IV.

Cuvinte cheie



The End

?

