



ΕΛΛΗΝΙΚΗ ΔΗΜΟΚΡΑΤΙΑ  
Εθνικόν και Καποδιστριακόν  
Πανεπιστήμιον Αθηνών

# Δίκτυα Επικοινωνιών I

Ενότητα 5: Επίπεδο Δικτύου  
-Επίπεδο ελέγχου

Διδάσκων: Νάσος Βάιος

Τμήμα Πληροφορικής και Τηλεπικοινωνιών  
Εθνικό & Καποδιστριακό Πανεπιστήμιο Αθηνών

# Δίκτυα Επικοινωνιών Ι

## Τμήμα Πληροφορικής και Τηλεπικοινωνιών



### Θεματικές Ενότητες (ΘΕ) μαθήματος:

**ΘΕ1: Εισαγωγή**

(Κεφ. 1 του βιβλίου)

**ΘΕ2: Επίπεδο Εφαρμογής**

(Κεφ. 2 του βιβλίου)

**ΘΕ3: Επίπεδο Μεταφοράς**

(Κεφ. 3 του βιβλίου)

**ΘΕ4: Επίπεδο Δικτύου**

**Επίπεδο Δεδομένων (Κεφ. 4 του βιβλίου)**

**Επίπεδο Ελέγχου (Κεφ. 5 του βιβλίου)**

Συνιστώμενο Βιβλίο:

**Computer Networking: A Top-Down Approach, by Kurose & Ross, Addison-Wesley, 8<sup>η</sup> Έκδοση**

Εκδόσεις : Μ. Γκιούρδας



Οι περισσότερες από τις διαφάνειες αυτής της ενότητας αποτελούν προσαρμογή και απόδοση στα ελληνικά των διαφανειών που συνοδεύουν το βιβλίο Computer Networking : A Top-Down Approach, J.F Kurose and K.W. Ross, 8/Ε, Addison-Wesley.

# Επίπεδο δικτύου (επίπεδο ελέγχου): Στόχοι Κεφαλαίου

- κατανόηση των βασικών αρχών πίσω από τις υπηρεσίες του επιπέδου δικτύου (επίπεδο ελέγχου):
  - παραδοσιακά πρωτόκολλα δρομολόγησης
  - SDN ελεγκτές
  - διαχείριση δικτύων
- πραγμάτωση, υλοποίηση στο Διαδίκτυο:
  - OSPF, BGP
  - OpenFlow
  - Internet Control Message Protocol: ICMP
  - SNMP

# Επίπεδο δικτύου: επίπεδο ελέγχου

- **εισαγωγή**
- αλγόριθμοι δρομολόγησης
  - κατάσταση ζεύξης
  - διάνυσμα απόστασης
- δρομολόγηση ενδο-αυτόνομου συστήματος (intra-ISP routing): OSPF
- δρομολόγηση ανάμεσα σε ISPs: BGP
- επίπεδο ελέγχου SDN
- Internet Control Message Protocol



- διαχείριση δικτύου
  - SNMP
  - NETCONF/YANG

# Λειτουργίες επιπέδου δικτύου

- **πρώθηση:** μεταφέρει πακέτα από την είσοδο του δρομολογητή σε μια κατάλληλη έξοδο

*Επίπεδο δεδομένων*

- **δρομολόγηση:** καθορίζει την διαδρομή που θα ακολουθήσουν τα πακέτα από την πηγή στον προορισμό

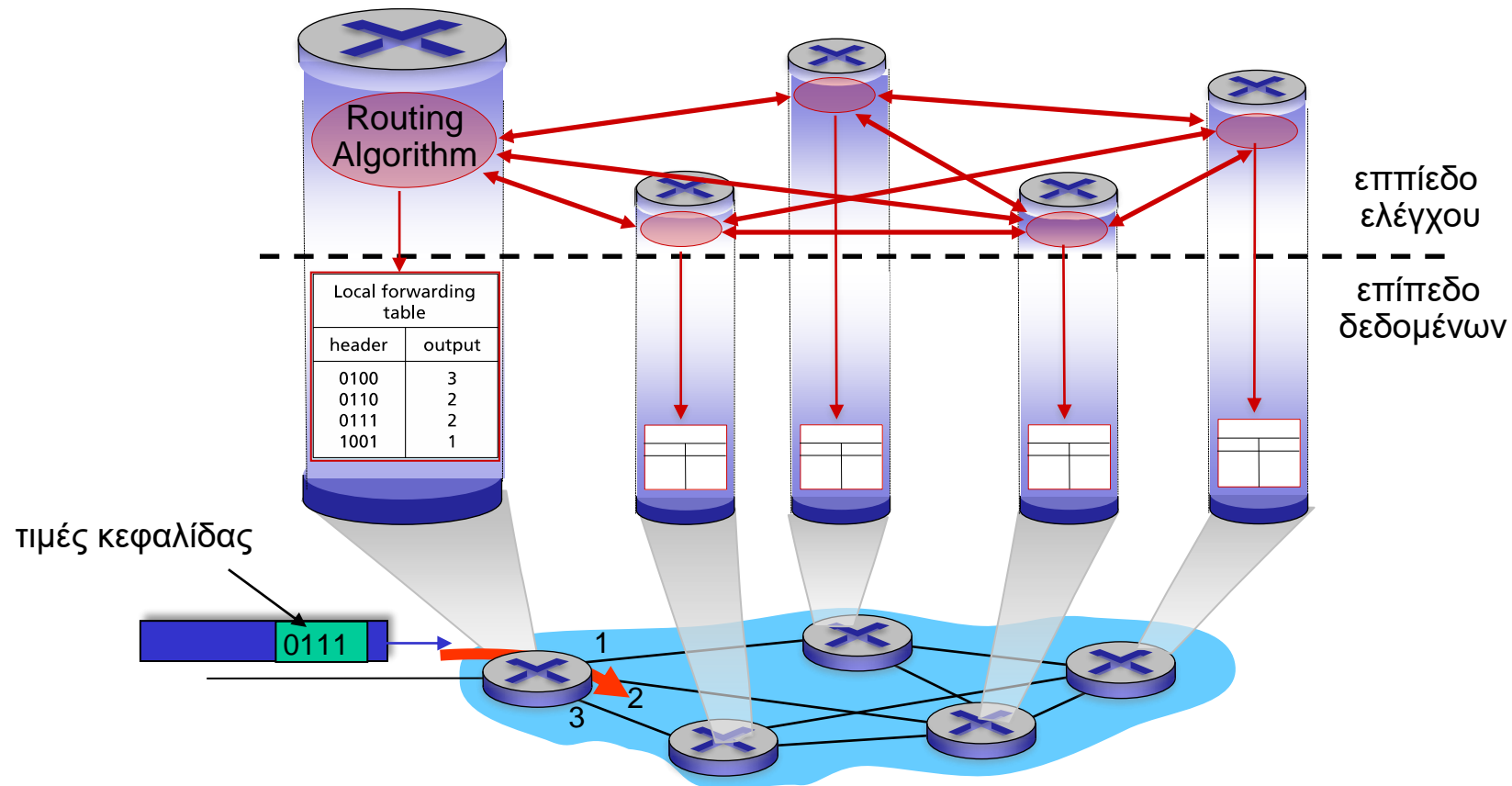
*Επίπεδο ελέγχου*

## **Δύο προσεγγίσεις:**

- έλεγχος ανά δρομολογητή (παραδοσιακά)
- λογικά κεντριοποιημένος έλεγχος (software defined networking)

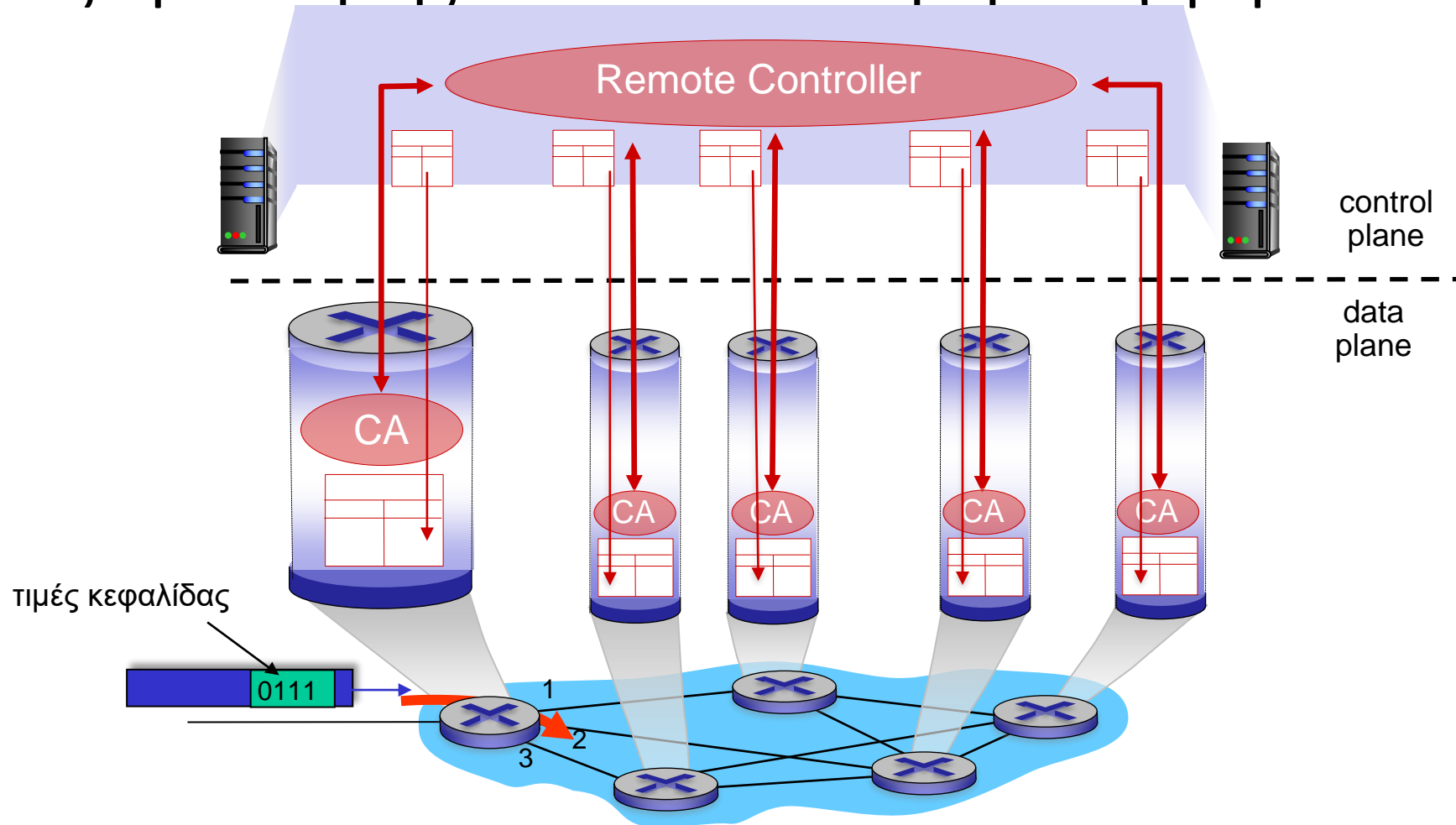
# Επίπεδο ελέγχου ανά δρομολογητή

Ξεχωριστά συστατικά στοιχεία αλγορίθμου δρομολόγησης *σε κάθε έναν δρομολογητή* αλληλεπιδρούν στο επίπεδο ελέγχου



# Λογικά κεντριοποιημένος έλεγχος (SDN)

Ένας λογικά κεντριοποιημένος ελεγκτής υπολογίζει και διανέμει τους πίνακες προώθησης σε κάθε έναν δρομολογητή



# Επίπεδο δικτύου: επίπεδο ελέγχου

- εισαγωγή
- αλγόριθμοι δρομολόγησης
  - κατάσταση ζεύξης
  - διάνυσμα απόστασης
- δρομολόγηση ενδο-αυτόνομου συστήματος (intra-ISP routing): OSPF
- δρομολόγηση ανάμεσα σε ISPs: BGP
- επίπεδο ελέγχου SDN
- Internet Control Message Protocol



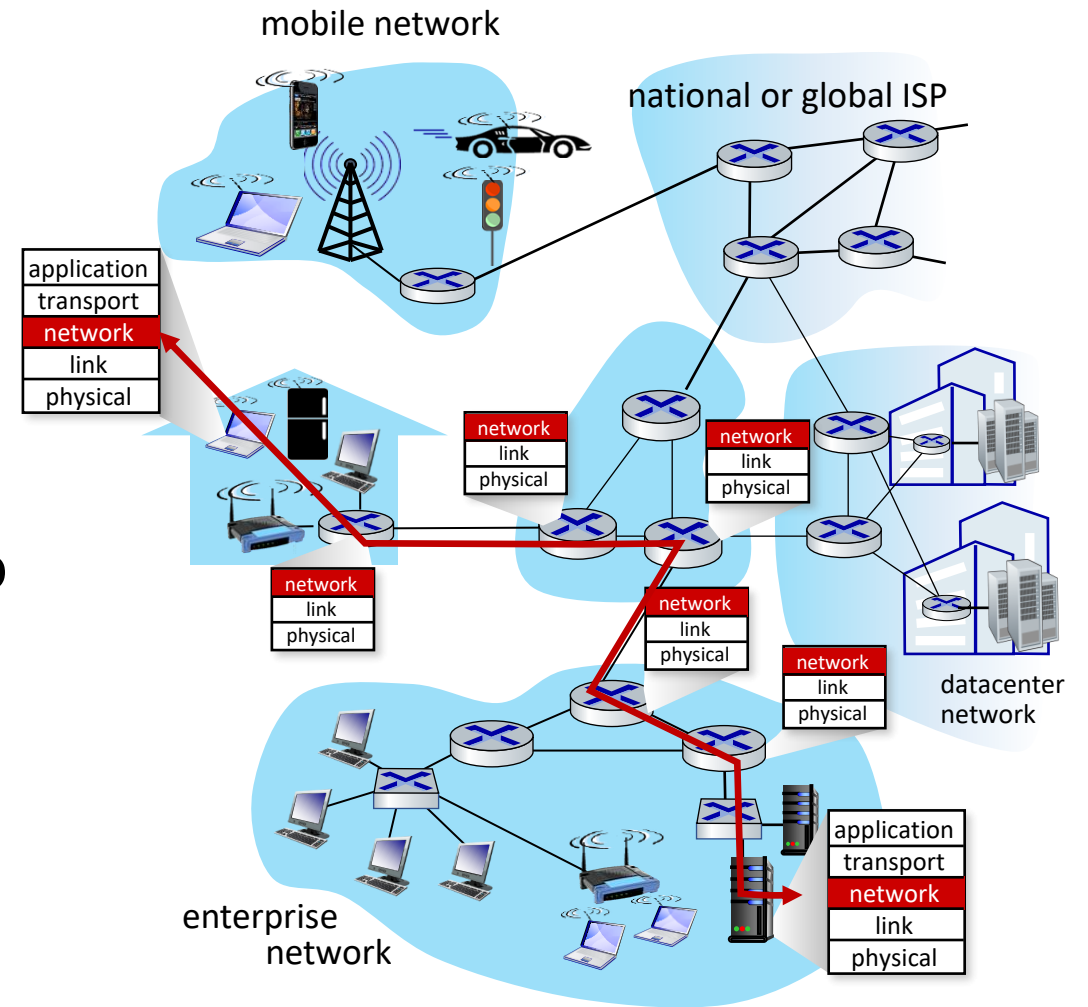
- διαχείριση δικτύου
  - SNMP
  - NETCONF/YANG



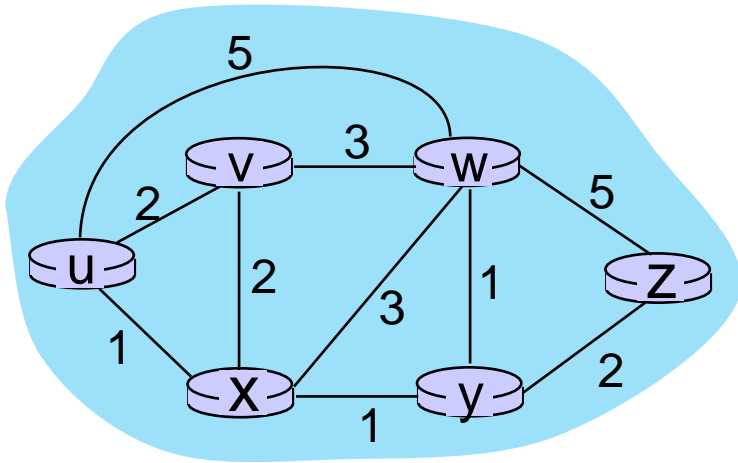
# Πρωτόκολλα δρομολόγησης

**Στόχος των πρωτοκόλλων δρομολόγησης:** καθορίζουν «καλά» μονοπάτια (διαδρομές) από πηγές σε προορισμούς

- **μονοπάτια:** ακολουθία των δρομολογητών από όπου περνάνε τα πακέτα για να οδηγηθούν από μία αρχική πηγή προς τον τελικό προορισμό
- **«καλά»:** ελαχίστου κόστους, γρηγορότερα, με τη μικρότερη συμφόρηση
- δρομολόγηση: ανάμεσα στις 10 πιο δημοφιλείς προκλήσεις στα δίκτυα!



# Αφηρημένο μοντέλο γράφων: κόστος ζεύξεων



$c_{a,b}$ : κόστος *απευθείας* ζεύξης που συνδέει τον  $a$  με τον  $b$

π.χ.,  $c_{w,z} = 5$ ,  $c_{u,z} = \infty$

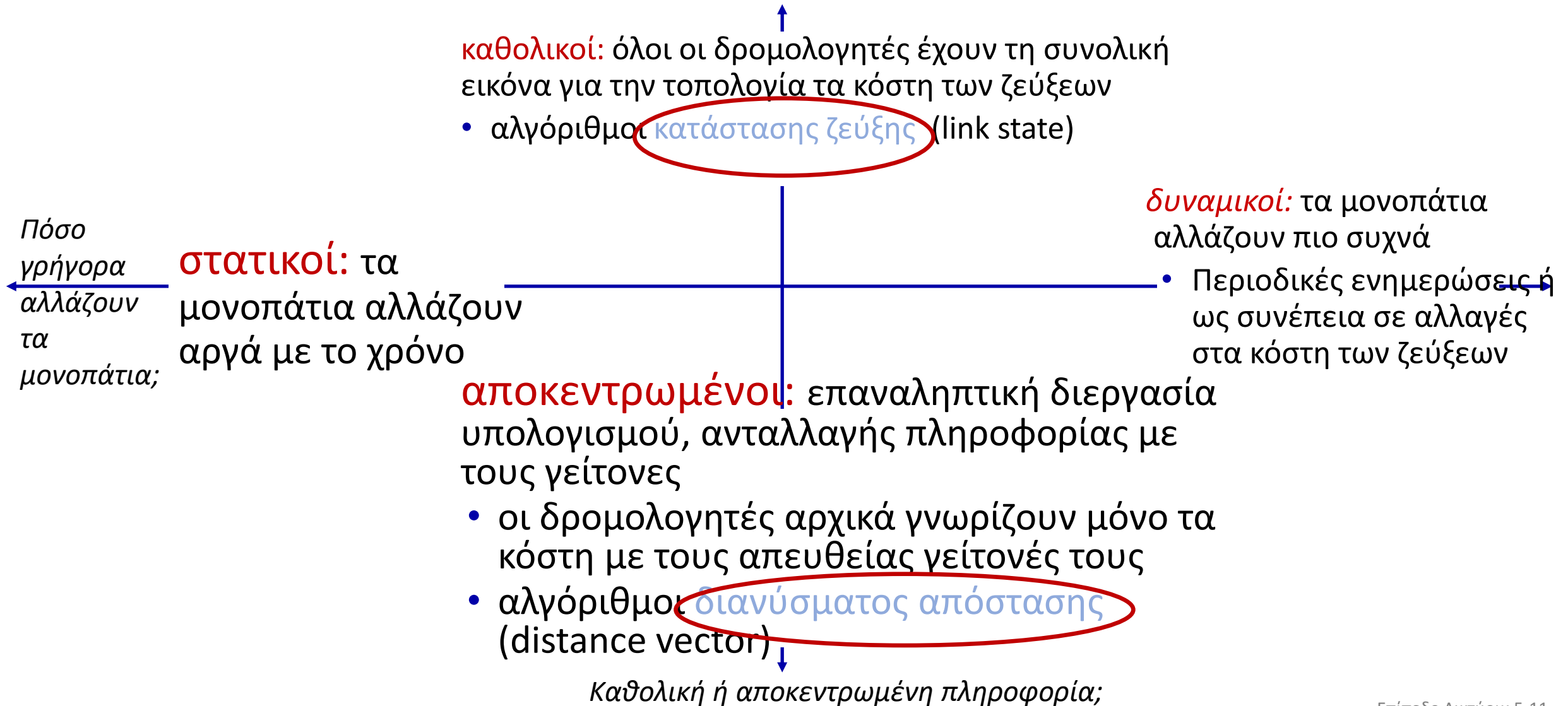
το κόστος καθορίζεται από την διαχειριστή του δικτύου: θα μπορούσε να είναι πάντα 1, ή αντίστροφα συσχετισμένο με το εύρος ζώνης ή τη συμφόρηση

γράφος:  $G = (N, E)$

$N$ : σύνολο δρομολογητών =  $\{ u, v, w, x, y, z \}$

$E$ : σύνολο ζεύξεων =  $\{ (u,v), (u,x), (v,x), (v,w), (x,w), (x,y), (w,y), (w,z), (y,z) \}$

# Κατάταξη αλγορίθμων δρομολόγησης



# Dijkstra: Αλγόριθμος κατάστασης ζεύξης

- **κεντριοποιημένος:** η δικτυακή τοπολογία, τα κόστη των ζεύξεων είναι γνωστά σε όλους τους κόμβους
  - μέσω ευρυεκπομπής (broadcast)
  - όλοι οι κόμβοι έχουν την ίδια πληροφορία
- υπολογίζει τις διαδρομές ελάχιστου κόστους από ένα κόμβο (προέλευση) προς όλους τους άλλους κόμβους
  - δίνει *πίνακα δρομολόγησης* για αυτό τον κόμβο
- **επαναληπτικός:** μετά από  $k$  επαναλήψεις, είναι γνωστές οι ελαχίστου κόστους διαδρομές προς  $k$  προορισμούς

## συμβολισμοί

- $c_{x,y}$ : απευθείας κόστος ζεύξης από τον κόμβο  $x$  στον  $y$ ;  $= \infty$  αν όχι άμεσοι γείτονες
- $D(v)$ : τρέχουσα εκτίμηση του κόστους της διαδρομής από την προέλευση στον προορισμό  $v$
- $p(v)$ : προηγούμενος από τον  $v$  κόμβο κατά μήκος της διαδρομής από την προέλευση στον  $v$
- $N'$ : σύνολο κόμβων για τους οποίους η ελαχίστου κόστους διαδρομή έχει σαφώς καθοριστεί

# Αλγόριθμος Dijkstra

1 *Initialization:*

2  $N' = \{u\}$  /\* compute least cost path from  $u$  to all other nodes \*/

3 for all nodes  $v$

4 if  $v$  adjacent to  $u$  /\*  $u$  initially knows direct-path-cost only to direct neighbors \*/

5 then  $D(v) = c_{u,v}$  /\* but may not be *minimum* cost! \*/

6 else  $D(v) = \infty$

7

8 *Loop*

9 find  $w$  not in  $N'$  such that  $D(w)$  is a minimum

10 add  $w$  to  $N'$

11 update  $D(v)$  for all  $v$  adjacent to  $w$  and not in  $N'$  :

12  $D(v) = \min ( D(v), D(w) + c_{w,v} )$

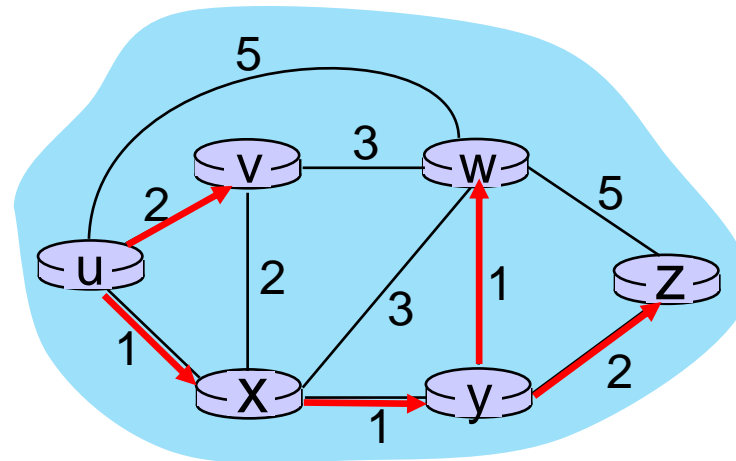
13 /\* new least-path-cost to  $v$  is either old least-cost-path to  $v$  or known

14 least-cost-path to  $w$  plus direct-cost from  $w$  to  $v$  \*/

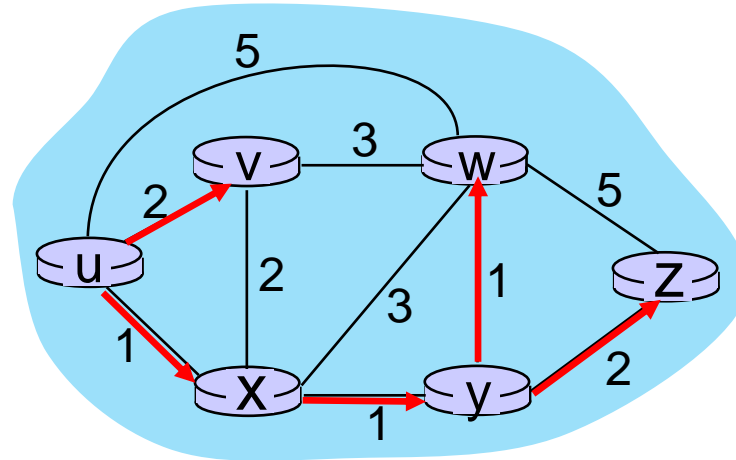
15 *until all nodes in  $N'$*

# Αλγόριθμος Dijkstra: ένα παράδειγμα

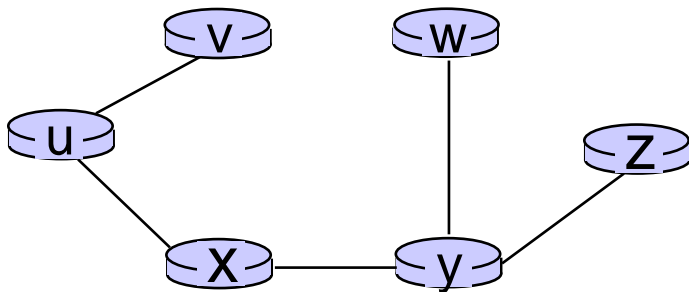
Step	N'	D(v),p(v)	D(w),p(w)	D(x),p(x)	D(y),p(y)	D(z),p(z)
0	u	2,u	5,u	1,u	$\infty$	$\infty$
1	ux	2,u	4,x	1,u	2,x	$\infty$
2	uxy	2,u	3,y	1,u	2,x	4,y
3	uxyv	2,u	3,y	1,u	2,x	4,y
4	uxyvw	2,u	3,y	1,u	2,x	4,y
5	uxyvwz	2,u	3,y	1,u	2,x	4,y



# Αλγόριθμος Dijkstra: ένα παράδειγμα



Δέντρο διαδρομών ελαχίστου κόστους από τον u



πίνακας προώθησης στον u:

destination	outgoing link
v	(u,v)
x	(u,x)
y	(u,x)
w	(u,x)
x	(u,x)

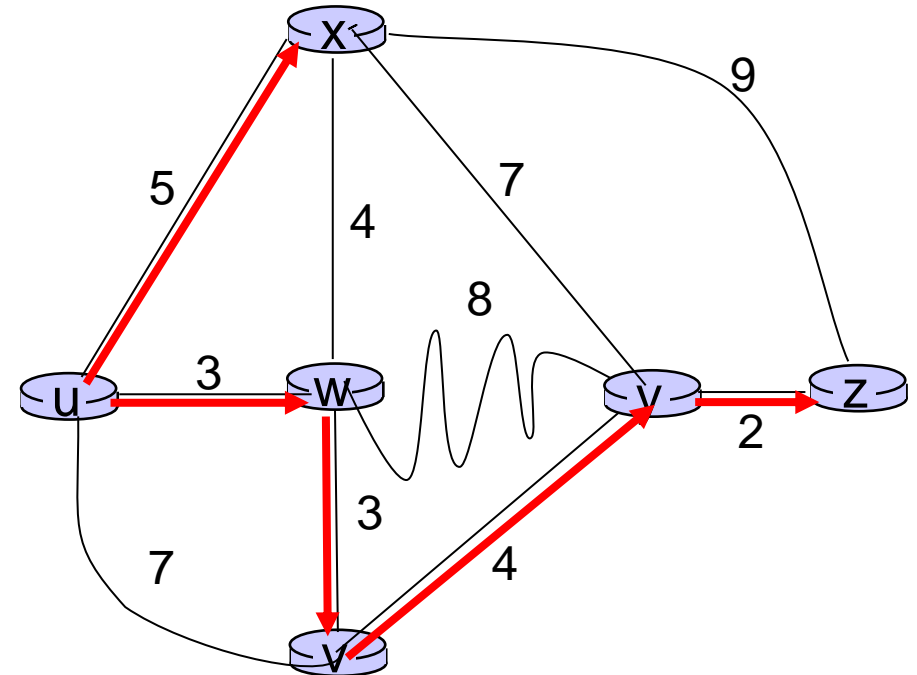
Απευθείας ζεύξη προς v

διαδρομή από τον u στους υπόλοιπους προορισμούς μέσω του x

ΤΟΥ Χ

# Αλγόριθμος Dijkstra: κι άλλο παράδειγμα

Step	$N'$	$D(v), p(v)$	$D(w), p(w)$	$D(x), p(x)$	$D(y), p(y)$	$D(z), p(z)$
0	u	7, u	3, u	5, u	$\infty$	$\infty$
1	uw	6, w	3, u	5, u	11, w	$\infty$
2	uwvx	6, w	3, u	5, u	11, w	14, x
3	uwxv	6, w	3, u	5, u	10, v	14, x
4	uwxvy	6, w	3, u	5, u	10, v	12, y
5	uwxvyz	6, w	3, u	5, u	10, v	12, y



## παρατηρήσεις:

- κατασκευή δέντρου συντομότερης διαδρομής εντοπίζοντας τους προηγούμενους κόμβους
- μπορεί να υπάρχουν «ισοπαλίες» (σπάνε αυθαίρετα)



# Αλγόριθμος Dijkstra: συζήτηση

**υπολογιστική πολυπλοκότητα:**  $n$  κόμβοι (χωρίς να μετράμε την προέλευση)

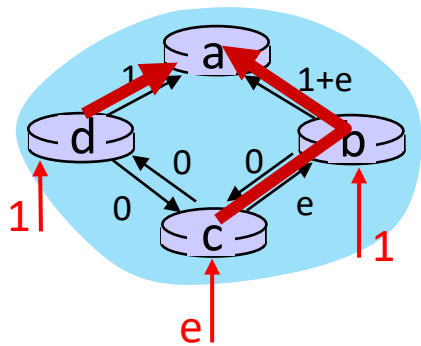
- κάθε επανάληψη: πρέπει να εξεταστούν όλοι οι κόμβοι,  $w$ , που δεν ανήκουν στο  $N$
- $n(n+1)/2$  συγκρίσεις: πολυπλοκότητα  $O(n^2)$
- είναι δυνατές πιο αποδοτικές υλοποιήσεις:  $O(n \log n)$

**πολυπλοκότητα μηνυμάτων:**

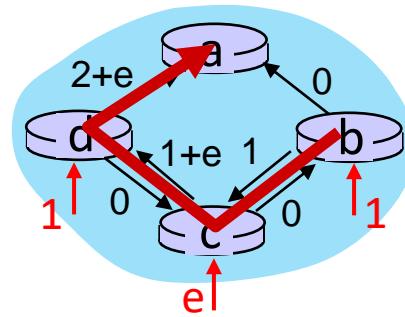
- κάθε δρομολογητής πρέπει να *εκπέμψει* την πληροφορία ζεύξης σε όλους τους υπόλοιπους δρομολογητές
- αποδοτικοί (και ενδιαφέροντες!) αλγόριθμοι ευρυεκπομπής: περνά από  $O(n)$  ζεύξεις για την διάδοση ενός τέτοιου μηνύματος από έναν δρομολογητή.
- κάθε μήνυμα δρομολογητή περνά από  $O(n)$  ζεύξεις: συνολική πολυπλοκότητα:  $O(n^2)$

# Αλγόριθμος Dijkstra: πιθανές ταλαντώσεις

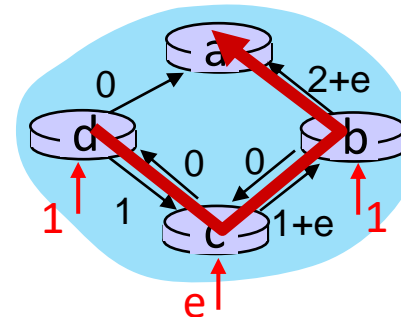
- όταν τα κόστη ζεύξεων εξαρτώνται από τον όγκο της κίνησης, πιθανές οι **ταλαντώσεις**
- για παράδειγμα:
  - δρομολόγηση προς τον προορισμό a, με κίνηση από b, c, d, με ρυθμό 1, e (<1), 1
  - κόστη ζεύξεων ξεχωριστά ανά κατεύθυνση



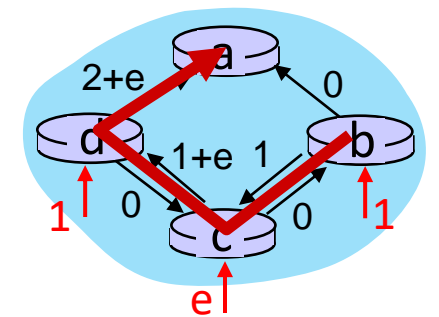
αρχικά



με δεδομένα αυτά τα κόστη,  
βρες νέα δρομολόγηση...  
προκύπτουν νέα κόστη



με δεδομένα αυτά τα κόστη,  
βρες νέα δρομολόγηση...  
προκύπτουν νέα κόστη



με δεδομένα αυτά τα κόστη,  
βρες νέα δρομολόγηση...  
προκύπτουν νέα κόστη

# Επίπεδο δικτύου: επίπεδο ελέγχου

- εισαγωγή
- αλγόριθμοι δρομολόγησης
  - κατάσταση ζεύξης
  - διάνυσμα απόστασης
- δρομολόγηση ενδο-αυτόνομου συστήματος (intra-ISP routing): OSPF
- δρομολόγηση ανάμεσα σε ISPs: BGP
- επίπεδο ελέγχου SDN
- Internet Control Message Protocol



- διαχείριση δικτύου
  - SNMP
  - NETCONF/YANG

# Αλγόριθμος Διανύσματος Απόστασης (Distance Vector)

Βασίζεται στην εξίσωση *Bellman-Ford* (δυναμικός προγραμματισμός):

Bellman-Ford εξίσωση

Έστω  $D_x(y)$ : κόστος της διαδρομής ελαχίστου κόστους από τον  $x$  στο  $y$ .

Τότε:

$$D_x(y) = \min_v \{ c_{x,v} + D_v(y) \}$$

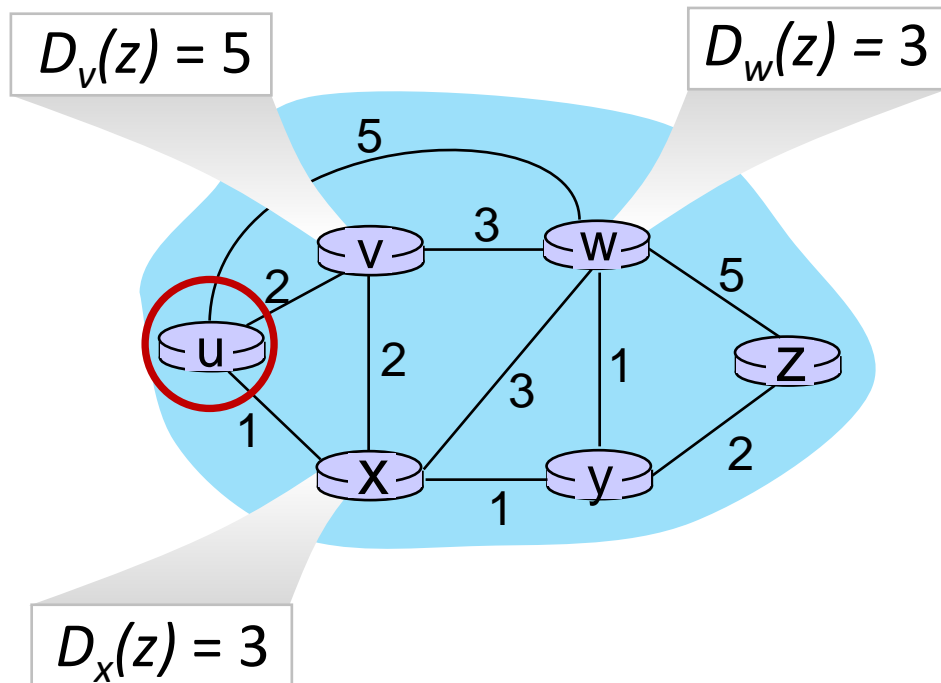
ελάχιστο κόστος από τον  $v$  στον  $y$

το ελάχιστο για όλους τους γείτονες  
 $v$  του  $x$

κόστος για το γείτονα  $v$

# Παράδειγμα Bellman-Ford

Υποθέτουμε ότι οι γείτονες του  $u$ , οι κόμβοι  $x, v, w$ , γνωρίζουν πως για τον προορισμό  $z$ :



Η εξίσωση Bellman-Ford δίνει:

$$\begin{aligned} D_u(z) &= \min \{ c_{u,v} + D_v(z), \\ &\quad c_{u,x} + D_x(z), \\ &\quad c_{u,w} + D_w(z) \} \\ &= \min \{ 2 + 5, \\ &\quad 1 + 3, \\ &\quad 5 + 3 \} = 4 \end{aligned}$$

ο κόμβος που επιτυγχάνει το ελάχιστο ( $x$ ) είναι το επόμενο άλμα στη βραχύτερη διαδρομή που υπολογίζεται για τον  $z$

# Αλγόριθμος Διανύσματος Απόστασης

## η ιδέα-κλειδί:

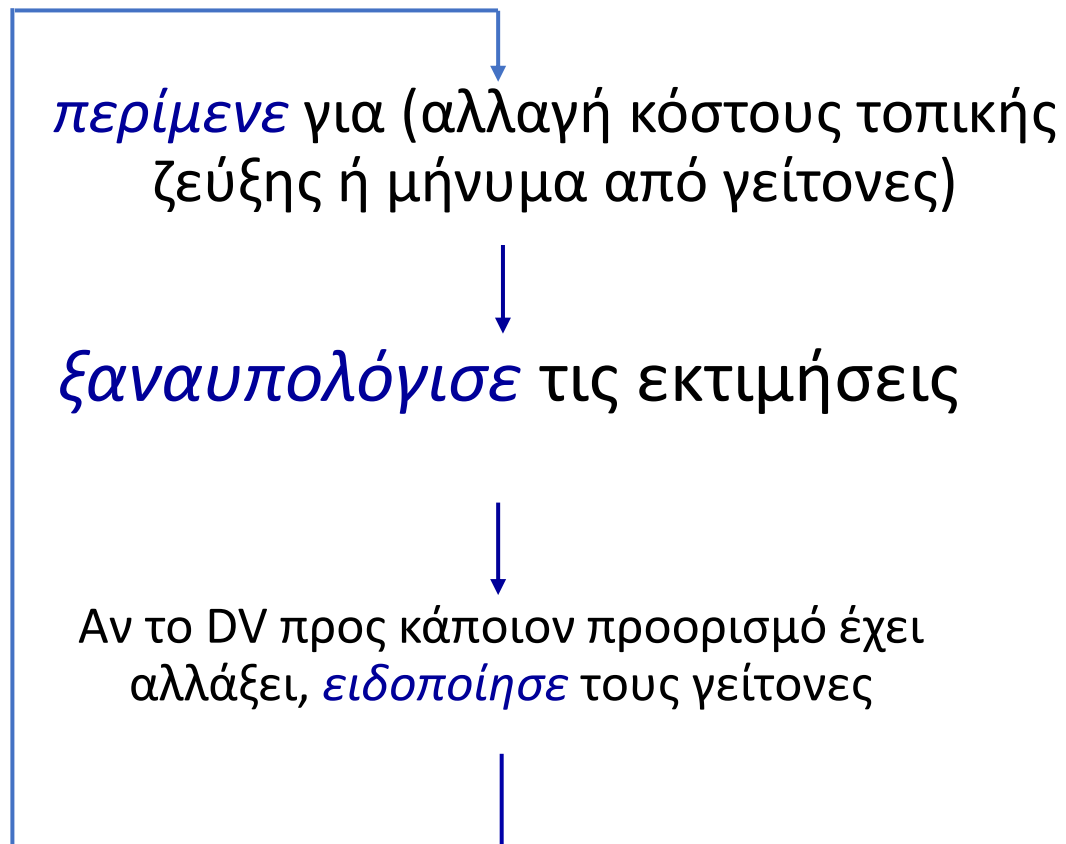
- από καιρό σε καιρό, κάθε κόμβος στέλνει τις δικές του εκτιμήσεις (διάνυσμα απόστασης) στους γείτονές του
- όταν ο  $x$  λαμβάνει νέα εκτίμηση (DV) από οποιοδήποτε γείτονα, ενημερώνει το δικό του διάνυσμα απόστασης σύμφωνα με την εξίσωση Bellman-Ford:

$$D_x(y) \leftarrow \min_v \{c_{x,v} + D_v(y)\} \text{ για κάθε κόμβο } y \in N$$

- υπό φυσιολογικές συνθήκες αναμένουμε την εκτίμηση  $D_x(y)$  να συγκλίνει στο πραγματικό ελάχιστο κόστος  $d_x(y)$

# Αλγόριθμος Διανύσματος Απόστασης

κάθε κόμβος:



επαναληπτικός, ασύγχρονος:

κάθε τοπική επανάληψη προκαλείται από:

- αλλαγή κόστους τοπικής ζεύξης
- μήνυμα ενημέρωσης DV από γείτονα

κατανεμημένος, αυτό-τερματιζόμενος:

κάθε κόμβος ειδοποιεί τους γείτονες μόνο όταν το DV του αλλάζει

- οι γείτονες τότε ειδοποιούν τους γείτονές τους – μόνο αν απαιτείται
- δεν λαμβάνονται ειδοποιήσεις, δεν γίνονται ενέργειες!

# Διάνυσμα Απόστασης: παράδειγμα

$$D_x()$$

	cost to		
	x	y	z
from x	0	2	7
from y	$\infty$	$\infty$	$\infty$
from z	$\infty$	$\infty$	$\infty$

	cost to		
	x	y	z
from x	0	2	3
from y	2	0	1
from z	7	1	0

$$D_x(z) = \min\{c_{x,y} + D_y(z), c_{x,z} + D_z(z)\}$$

$$= \min\{2+1, 7+0\} = 3$$

$$D_y()$$

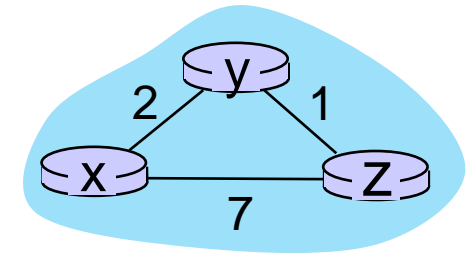
	cost to		
	x	y	z
from x	$\infty$	$\infty$	$\infty$
from y	2	0	1
from z	$\infty$	$\infty$	$\infty$

$$D_x(y) = \min\{c_{x,y} + D_y(y), c_{x,z} + D_z(y)\}$$

$$= \min\{2+0, 7+1\} = 2$$

$$D_z()$$

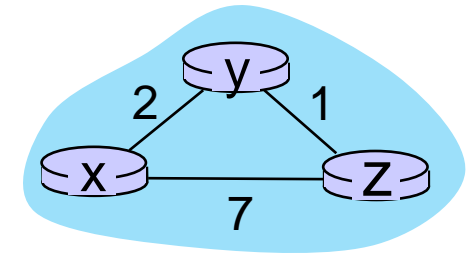
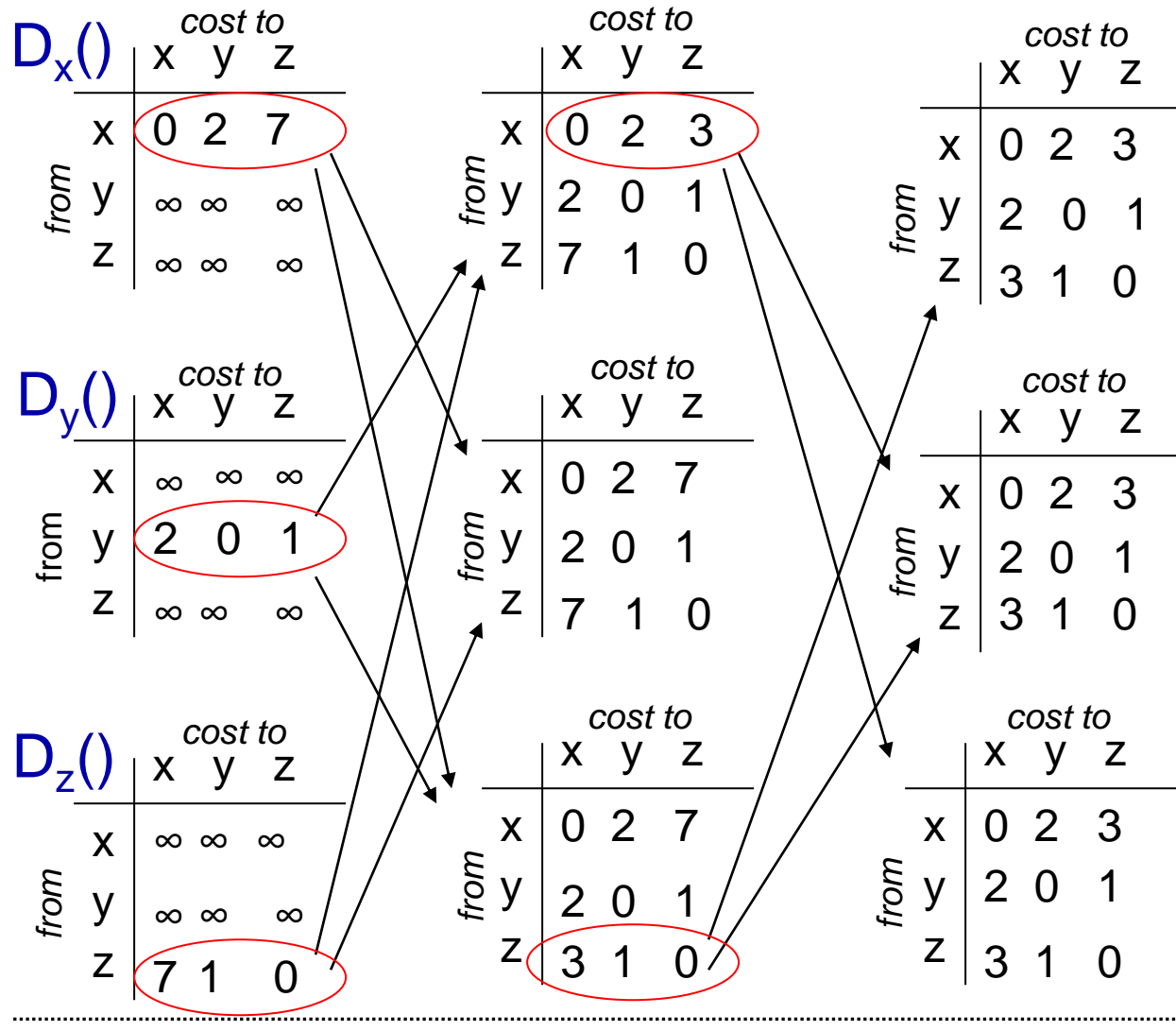
	cost to		
	x	y	z
from x	$\infty$	$\infty$	$\infty$
from y	$\infty$	$\infty$	$\infty$
from z	7	1	0



.....→ time



# Διάνυσμα Απόστασης: παράδειγμα



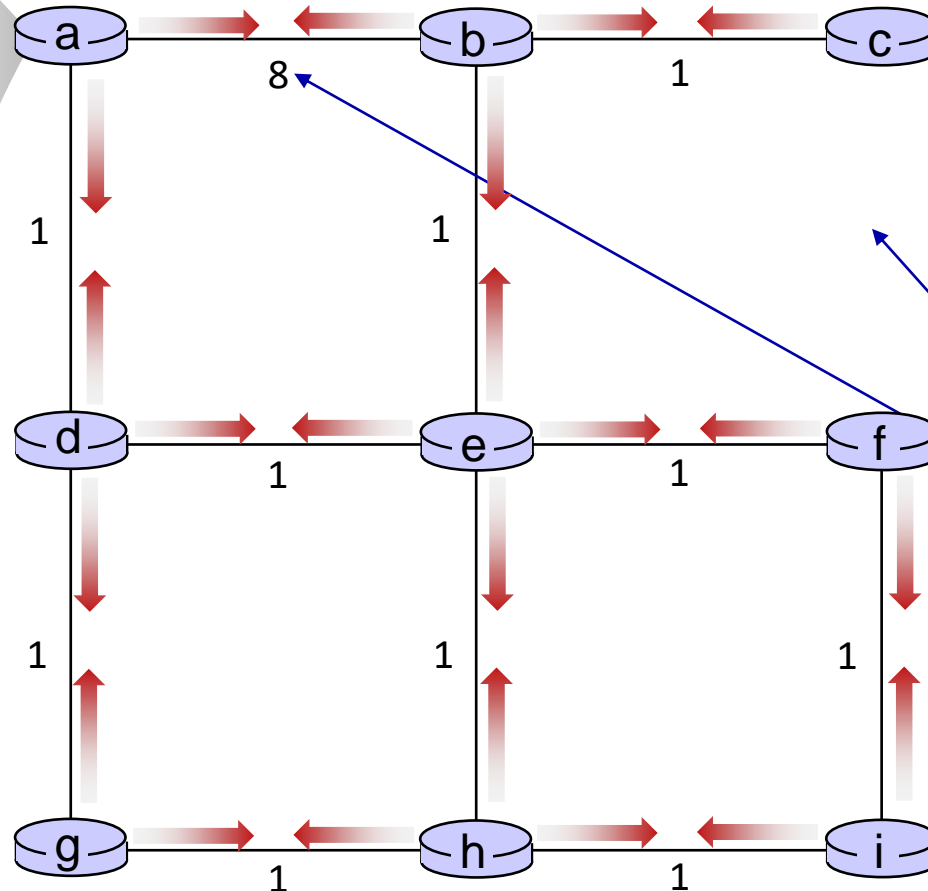
# Διάνυσμα Απόστασης: παράδειγμα



t=0

- Όλοι οι κόμβοι έχουν τις εκτιμήσεις για τις αποστάσεις με τους κοντινούς τους γείτονες (μόνο)
- Όλοι οι κόμβοι στέλνουν τα διανύσματά τους στους γείτονές τους

DV του a:
$D_a(a)=0$
$D_a(b)=8$
$D_a(c)=\infty$
$D_a(d)=1$
$D_a(e)=\infty$
$D_a(f)=\infty$
$D_a(g)=\infty$
$D_a(h)=\infty$
$D_a(i)=\infty$



Asymmetries:  
■ lacks pair  
■ larger cost

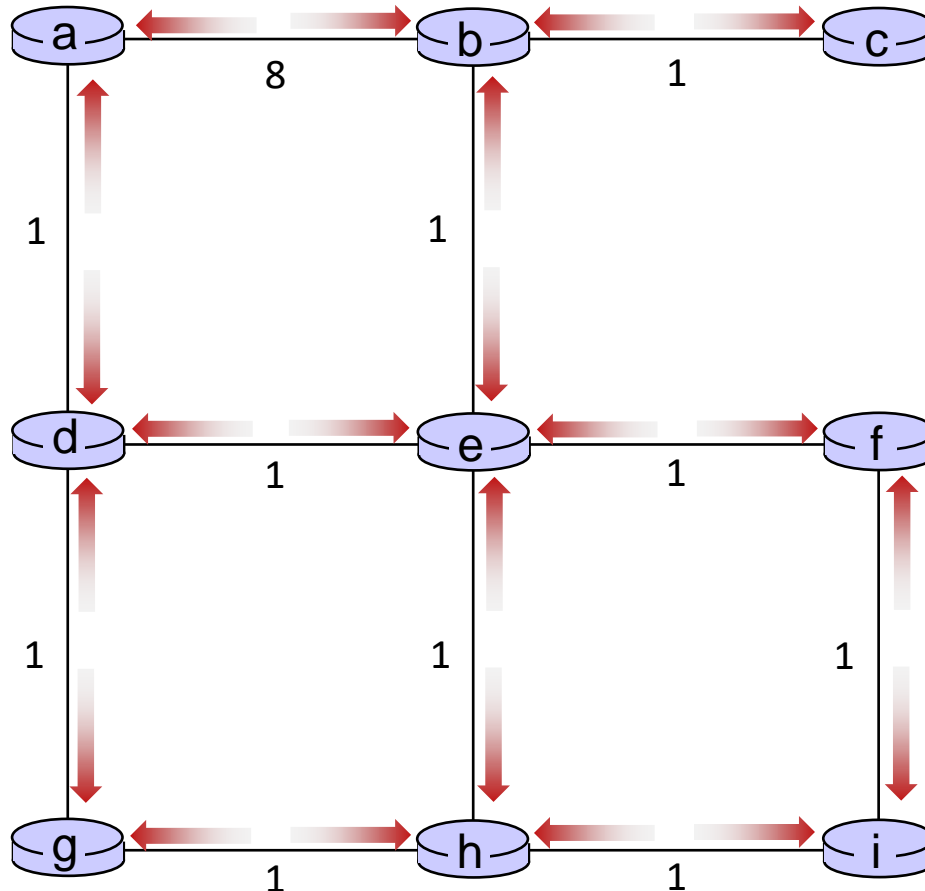
# Παράδειγμα Διανύσματος Απόστασης: επανάληψη



$t=1$

Όλοι οι κόμβοι:

- λαμβάνουν διανύσματα απόστασης από γείτονες
- υπολογίζουν το νέο τους διάνυσμα απόστασης
- στέλνουν το νέο τους διάνυσμα απόστασης στους γείτονές τους



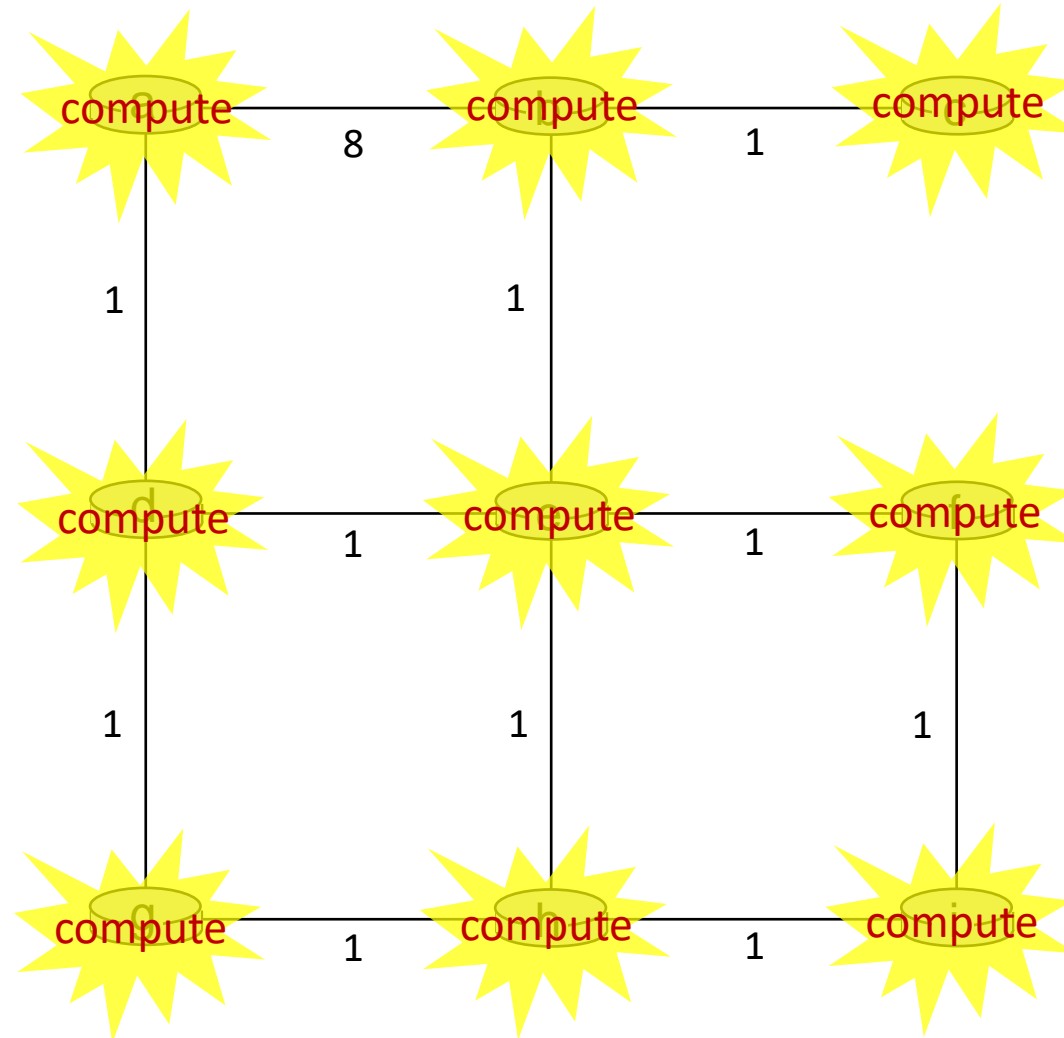
# Παράδειγμα Διανύσματος Απόστασης: επανάληψη



$t=1$

Όλοι οι κόμβοι:

- λαμβάνουν διανύσματα απόστασης από γείτονες
- υπολογίζουν το νέο τους διάνυσμα απόστασης
- στέλνουν το νέο τους διάνυσμα απόστασης στους γείτονές τους



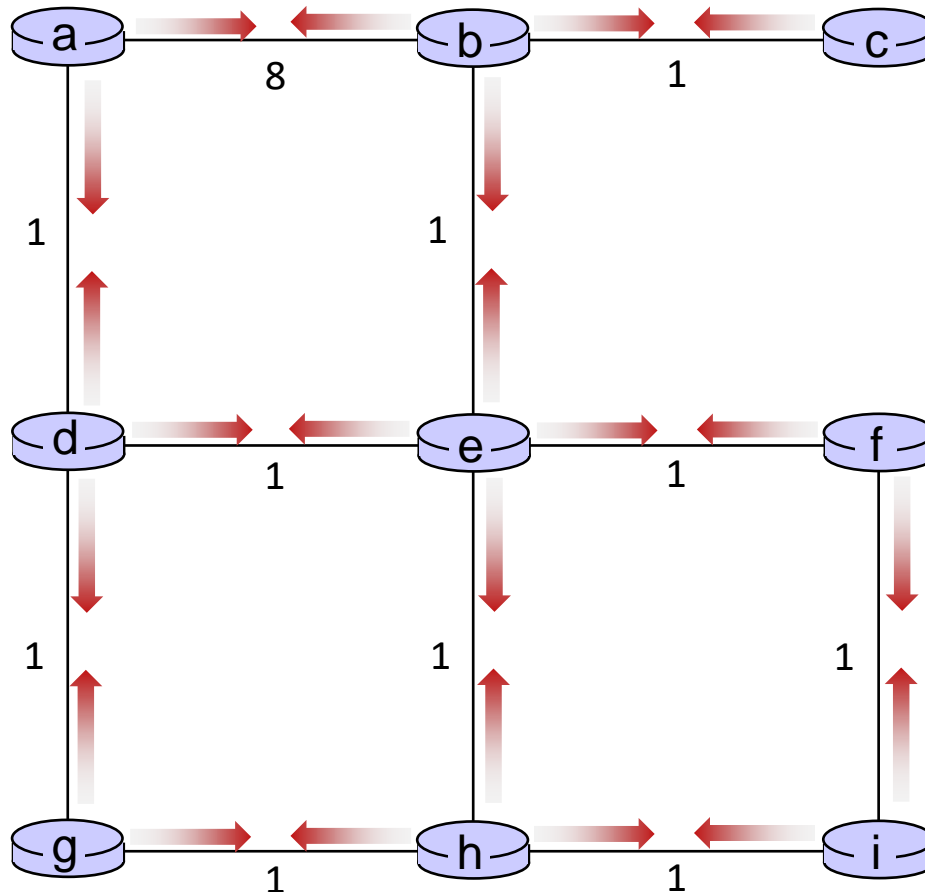
# Παράδειγμα Διανύσματος Απόστασης: επανάληψη



$t=1$

Όλοι οι κόμβοι:

- λαμβάνουν διανύσματα απόστασης από γείτονες
- υπολογίζουν το νέο τους διάνυσμα απόστασης
- στέλνουν το νέο τους διάνυσμα απόστασης στους γείτονές τους



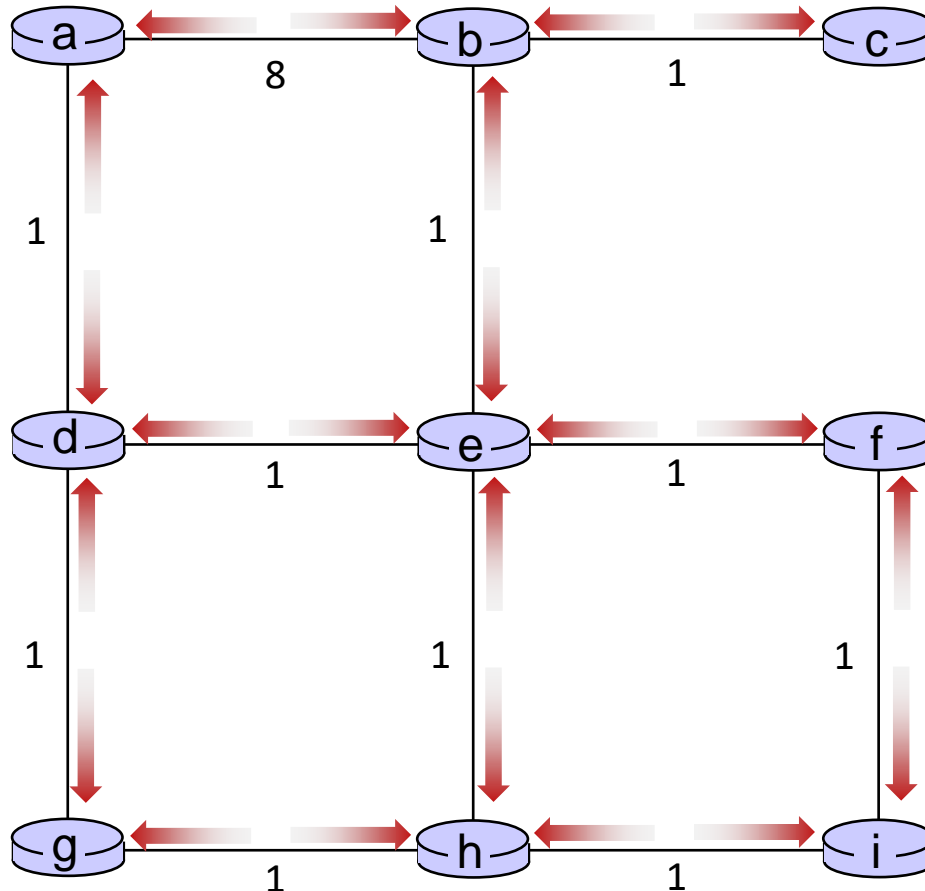
# Παράδειγμα Διανύσματος Απόστασης: επανάληψη



$t=2$

Όλοι οι κόμβοι:

- λαμβάνουν διανύσματα απόστασης από γείτονες
- υπολογίζουν το νέο τους διάνυσμα απόστασης
- στέλνουν το νέο τους διάνυσμα απόστασης στους γείτονές τους



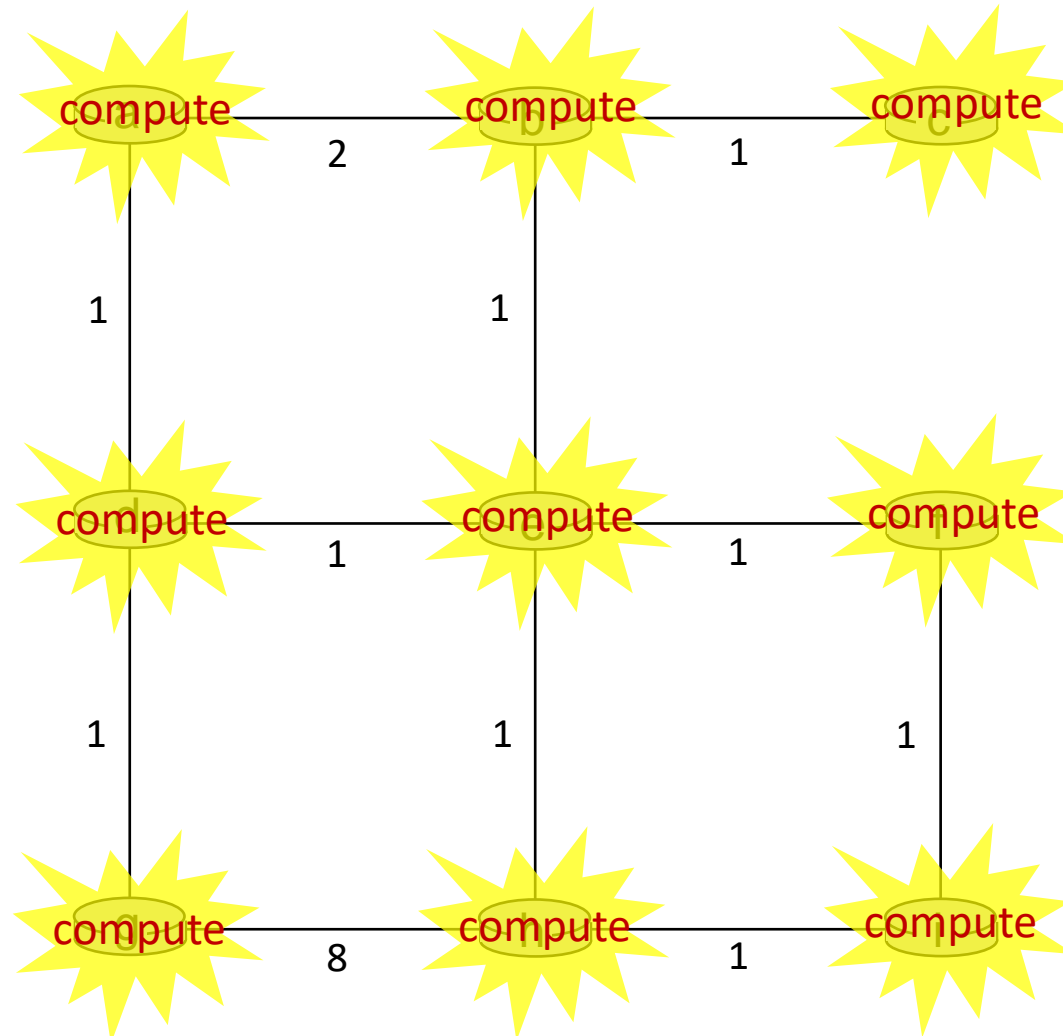
# Παράδειγμα Διανύσματος Απόστασης: επανάληψη



$t=2$

Όλοι οι κόμβοι:

- λαμβάνουν διανύσματα απόστασης από γείτονες
- υπολογίζουν το νέο τους διάνυσμα απόστασης
- στέλνουν το νέο τους διάνυσμα απόστασης στους γείτονές τους



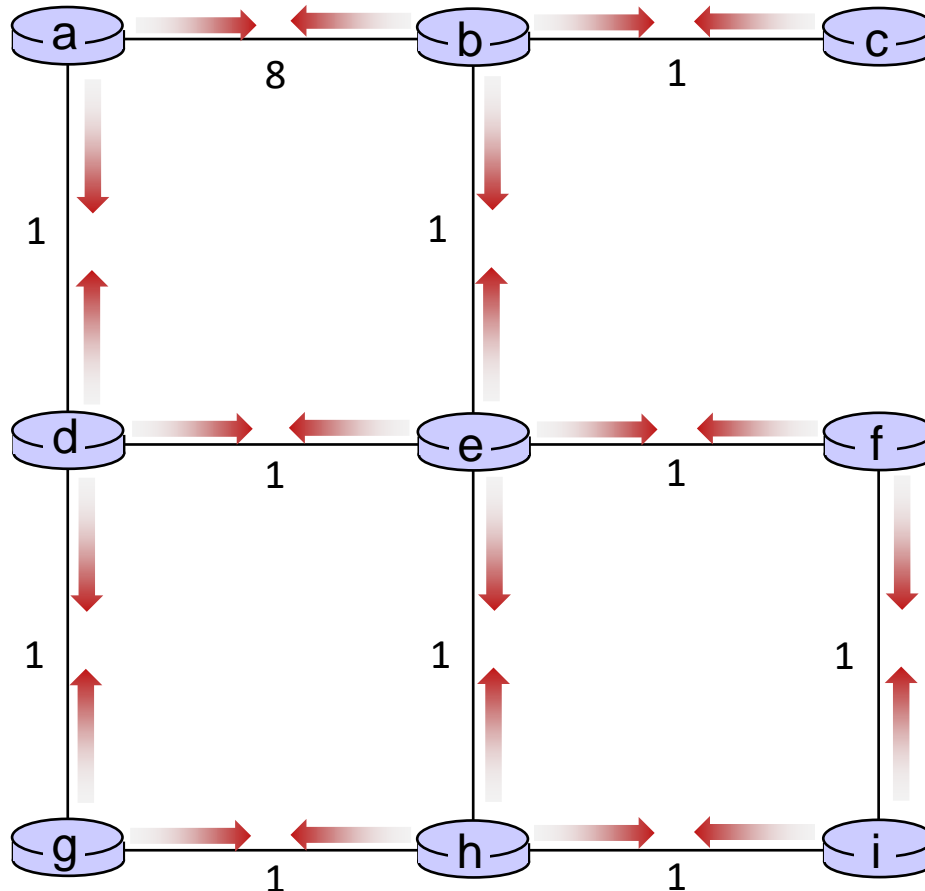
# Παράδειγμα Διανύσματος Απόστασης: επανάληψη



t=2

Όλοι οι κόμβοι:

- λαμβάνουν διανύσματα απόστασης από γείτονες
- υπολογίζουν το νέο τους διάνυσμα απόστασης
- στέλνουν το νέο τους διάνυσμα απόστασης στους γείτονές τους





# Παράδειγμα Διανύσματος Απόστασης: επανάληψη

.... και συνεχίζει έτσι

Ας δούμε τους υπολογισμούς που λαμβάνουν χώρα επαναληπτικά στους κόμβους

# Παράδειγμα Διάνυσημα Απόστα



**t=1**

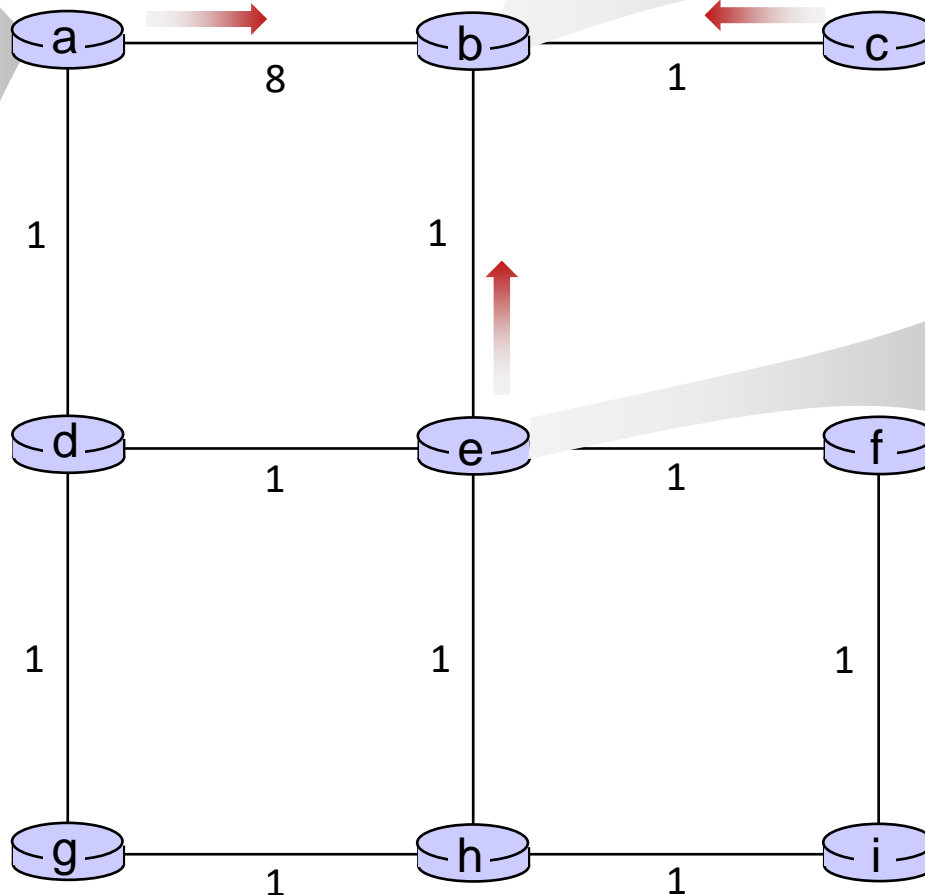
- Ο κόμβος b λαμβάνει τα DVs από a, c, e

DV a:
$D_a(a)=0$
$D_a(b)=8$
$D_a(c)=\infty$
$D_a(d)=1$
$D_a(e)=\infty$
$D_a(f)=\infty$
$D_a(g)=\infty$
$D_a(h)=\infty$
$D_a(i)=\infty$

DV b:	
$D_b(a)=8$	$D_b(f)=\infty$
$D_b(c)=1$	$D_b(g)=\infty$
$D_b(d)=\infty$	$D_b(h)=\infty$
$D_b(e)=1$	$D_b(i)=\infty$

DV c:
$D_c(a)=\infty$
$D_c(b)=1$
$D_c(c)=0$
$D_c(d)=\infty$
$D_c(e)=\infty$
$D_c(f)=\infty$
$D_c(g)=\infty$
$D_c(h)=\infty$
$D_c(i)=\infty$

DV e:
$D_e(a)=\infty$
$D_e(b)=1$
$D_e(c)=\infty$
$D_e(d)=1$
$D_e(e)=0$
$D_e(f)=1$
$D_e(g)=\infty$
$D_e(h)=1$
$D_e(i)=\infty$



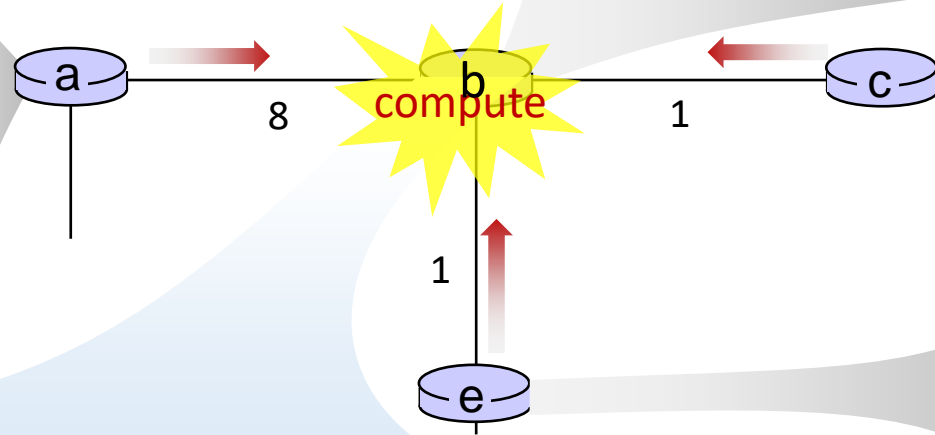
# Παράδειγμα Διάνυση Απόστα



**t=1**

- Ο b λαμβάνει από τους a, c, e, και υπολογίζει:

DV a:
$D_a(a)=0$
$D_a(b)=8$
$D_a(c)=\infty$
$D_a(d)=1$
$D_a(e)=\infty$
$D_a(f)=\infty$
$D_a(g)=\infty$
$D_a(h)=\infty$
$D_a(i)=\infty$



DV b:	
$D_b(a)=8$	$D_b(f)=\infty$
$D_b(c)=1$	$D_b(g)=\infty$
$D_b(d)=\infty$	$D_b(h)=\infty$
$D_b(e)=1$	$D_b(i)=\infty$

DV c:
$D_c(a)=\infty$
$D_c(b)=1$
$D_c(c)=0$
$D_c(d)=\infty$
$D_c(e)=\infty$
$D_c(f)=\infty$
$D_c(g)=\infty$
$D_c(h)=\infty$
$D_c(i)=\infty$

DV e:
$D_e(a)=\infty$
$D_e(b)=1$
$D_e(c)=\infty$
$D_e(d)=1$
$D_e(e)=0$
$D_e(f)=1$
$D_e(g)=\infty$
$D_e(h)=1$
$D_e(i)=\infty$

$$D_b(a) = \min\{c_{b,a}+D_a(a), c_{b,c}+D_c(a), c_{b,e}+D_e(a)\} = \min\{8, \infty, \infty\} = 8$$

$$D_b(c) = \min\{c_{b,a}+D_a(c), c_{b,c}+D_c(c), c_{b,e}+D_e(c)\} = \min\{\infty, 1, \infty\} = 1$$

$$D_b(d) = \min\{c_{b,a}+D_a(d), c_{b,c}+D_c(d), c_{b,e}+D_e(d)\} = \min\{9, 2, \infty\} = 2$$

$$D_b(e) = \min\{c_{b,a}+D_a(e), c_{b,c}+D_c(e), c_{b,e}+D_e(e)\} = \min\{\infty, \infty, 1\} = 1$$

$$D_b(f) = \min\{c_{b,a}+D_a(f), c_{b,c}+D_c(f), c_{b,e}+D_e(f)\} = \min\{\infty, \infty, 2\} = 2$$

$$D_b(g) = \min\{c_{b,a}+D_a(g), c_{b,c}+D_c(g), c_{b,e}+D_e(g)\} = \min\{\infty, \infty, \infty\} = \infty$$

$$D_b(h) = \min\{c_{b,a}+D_a(h), c_{b,c}+D_c(h), c_{b,e}+D_e(h)\} = \min\{\infty, \infty, 2\} = 2$$

$$D_b(i) = \min\{c_{b,a}+D_a(i), c_{b,c}+D_c(i), c_{b,e}+D_e(i)\} = \min\{\infty, \infty, \infty\} = \infty$$

DV b:	
$D_b(a)=8$	$D_b(f)=2$
$D_b(c)=1$	$D_b(g)=\infty$
$D_b(d)=2$	$D_b(h)=2$
$D_b(e)=1$	$D_b(i)=\infty$

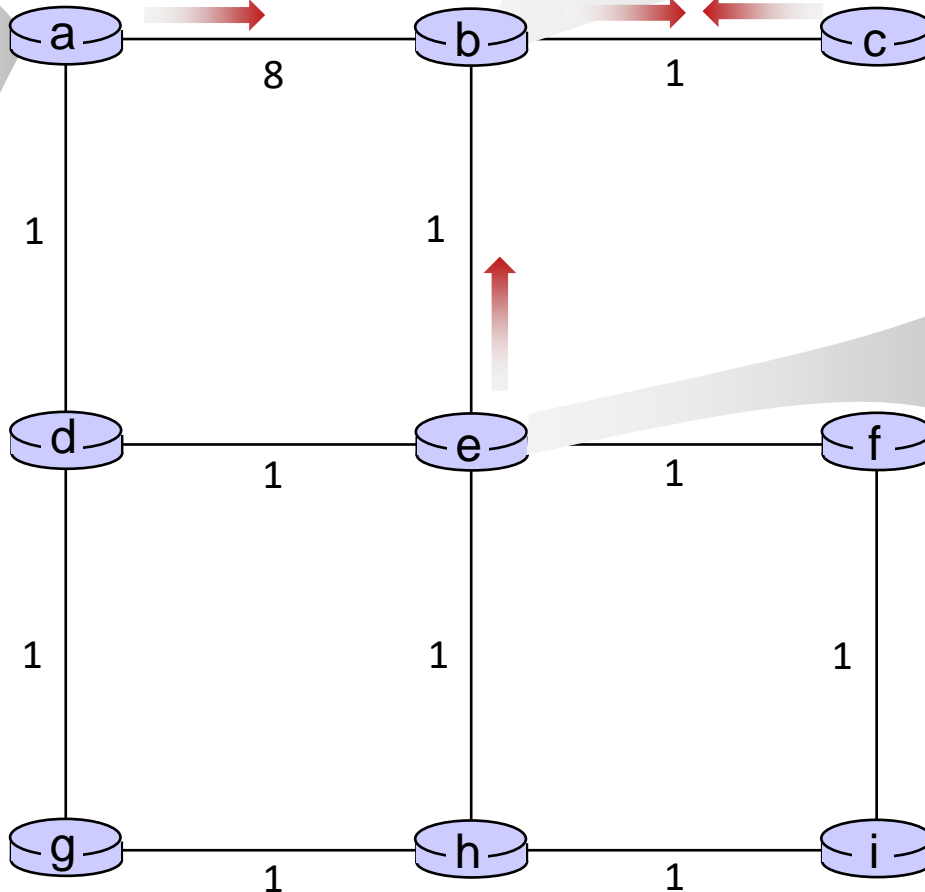
# Παράδειγμα Διάνυση Απόστα



**t=1**

- Ο c λαμβάνει DVs από τον b

DV a:
$D_a(a)=0$
$D_a(b)=8$
$D_a(c)=\infty$
$D_a(d)=1$
$D_a(e)=\infty$
$D_a(f)=\infty$
$D_a(g)=\infty$
$D_a(h)=\infty$
$D_a(i)=\infty$



DV b:	
$D_b(a)=8$	$D_b(f)=\infty$
$D_b(c)=1$	$D_b(g)=\infty$
$D_b(d)=\infty$	$D_b(h)=\infty$
$D_b(e)=1$	$D_b(i)=\infty$

DV c:
$D_c(a)=\infty$
$D_c(b)=1$
$D_c(c)=0$
$D_c(d)=\infty$
$D_c(e)=\infty$
$D_c(f)=\infty$
$D_c(g)=\infty$
$D_c(h)=\infty$
$D_c(i)=\infty$

DV e:
$D_e(a)=\infty$
$D_e(b)=1$
$D_e(c)=\infty$
$D_e(d)=1$
$D_e(e)=0$
$D_e(f)=1$
$D_e(g)=\infty$
$D_e(h)=1$
$D_e(i)=\infty$

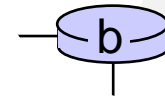
# Παράδειγμα Διάνυσμα Απόστα



t=1

- Ο c λαμβάνει DVs από τον b και υπολογίζει:

$$\begin{aligned} D_c(a) &= \min\{c_{c,b} + D_b(a)\} = 1 + 8 = 9 \\ D_c(b) &= \min\{c_{c,b} + D_b(b)\} = 1 + 0 = 1 \\ D_c(d) &= \min\{c_{c,b} + D_b(d)\} = 1 + \infty = \infty \\ D_c(e) &= \min\{c_{c,b} + D_b(e)\} = 1 + 1 = 2 \\ D_c(f) &= \min\{c_{c,b} + D_b(f)\} = 1 + \infty = \infty \\ D_c(g) &= \min\{c_{c,b} + D_b(g)\} = 1 + \infty = \infty \\ D_c(h) &= \min\{c_{c,b} + D_b(h)\} = 1 + \infty = \infty \\ D_c(i) &= \min\{c_{c,b} + D_b(i)\} = 1 + \infty = \infty \end{aligned}$$



1

compute

DV b:	
$D_b(a) = 8$	$D_b(f) = \infty$
$D_b(c) = 1$	$D_b(g) = \infty$
$D_b(d) = \infty$	$D_b(h) = \infty$
$D_b(e) = 1$	$D_b(i) = \infty$

DV c:
$D_c(a) = \infty$
$D_c(b) = 1$
$D_c(c) = 0$
$D_c(d) = \infty$
$D_c(e) = \infty$
$D_c(f) = \infty$
$D_c(g) = \infty$
$D_c(h) = \infty$
$D_c(i) = \infty$

DV c:
$D_c(a) = 9$
$D_c(b) = 1$
$D_c(c) = 0$
$D_c(d) = 2$
$D_c(e) = \infty$
$D_c(f) = \infty$
$D_c(g) = \infty$
$D_c(h) = \infty$
$D_c(i) = \infty$

# Παράδειγμα Διάνυση Απόστα



$t=1$

- Ο e λαμβάνει τα DVs από τους b, d, f, h

DV d:
$D_c(a) = 1$
$D_c(b) = \infty$
$D_c(c) = \infty$
$D_c(d) = 0$
$D_c(e) = 1$
$D_c(f) = \infty$
$D_c(g) = 1$
$D_c(h) = \infty$
$D_c(i) = \infty$

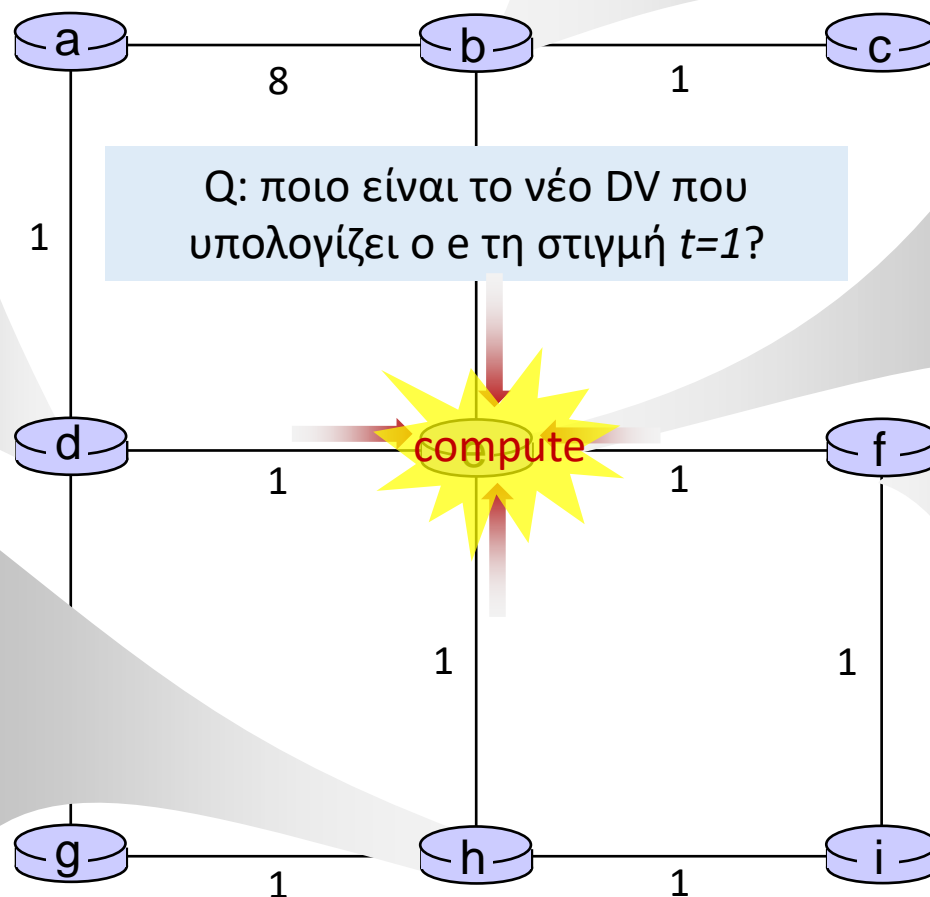
DV h:
$D_c(a) = \infty$
$D_c(b) = \infty$
$D_c(c) = \infty$
$D_c(d) = \infty$
$D_c(e) = 1$
$D_c(f) = \infty$
$D_c(g) = 1$
$D_c(h) = 0$
$D_c(i) = 1$

DV b:	
$D_b(a) = 8$	$D_b(f) = \infty$
$D_b(c) = 1$	$D_b(g) = \infty$
$D_b(d) = \infty$	$D_b(h) = \infty$
$D_b(e) = 1$	$D_b(i) = \infty$

ιοί






DV e:
$D_e(a) = \infty$
$D_e(b) = 1$
$D_e(c) = \infty$
$D_e(d) = 1$
$D_e(e) = 0$
$D_e(f) = 1$
$D_e(g) = \infty$
$D_e(h) = 1$
$D_e(i) = \infty$

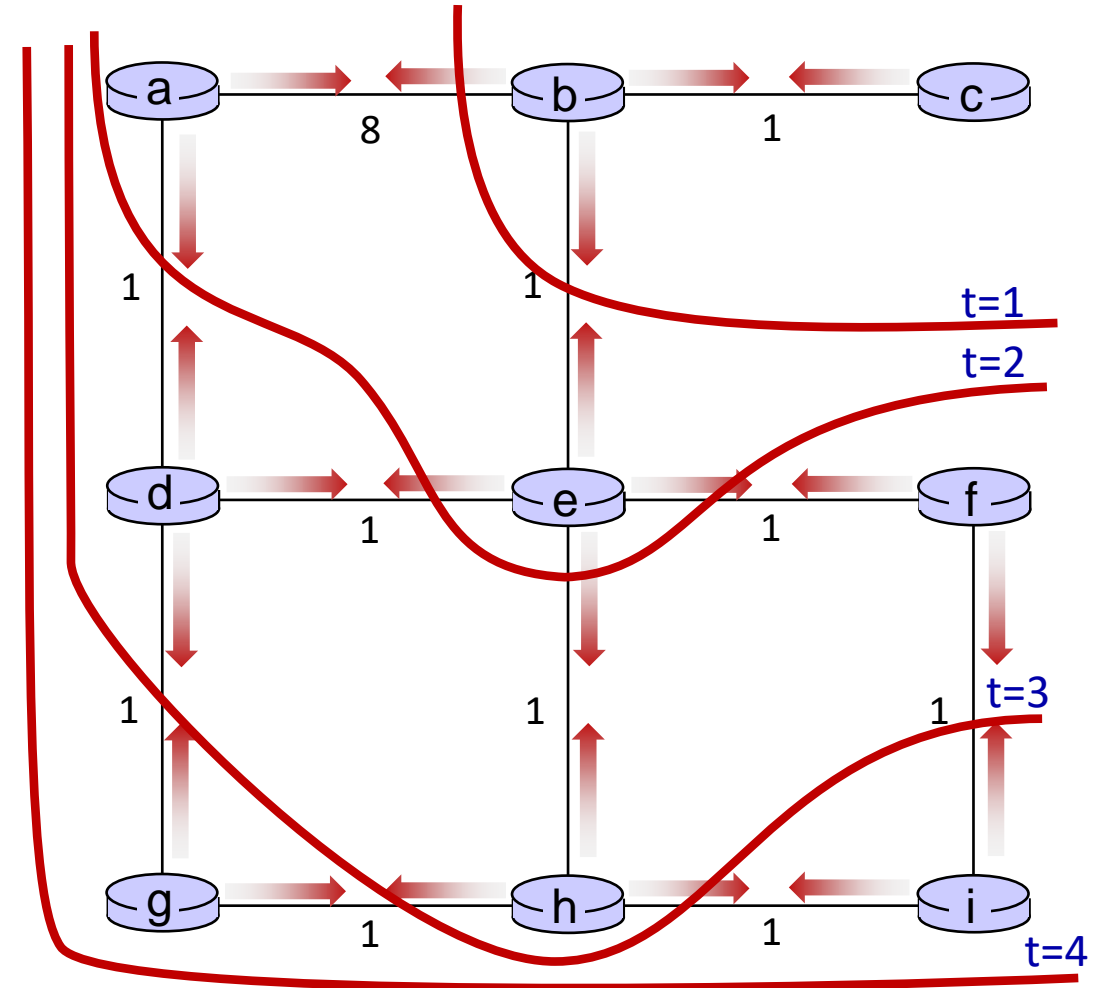
DV f:
$D_c(a) = \infty$
$D_c(b) = \infty$
$D_c(c) = \infty$
$D_c(d) = \infty$
$D_c(e) = 1$
$D_c(f) = 0$
$D_c(g) = \infty$
$D_c(h) = \infty$
$D_c(i) = 1$



# Διάνυσμα Απόστασης: διάδοση της πληροφορίας κατάστασης

Τα βήματα υπολογισμού διαδίδουν την πληροφορία στο δίκτυο:

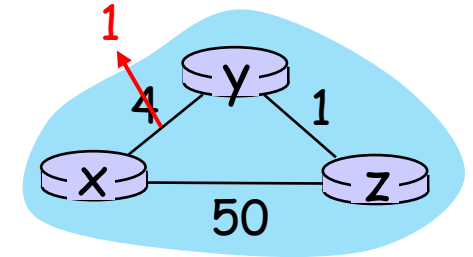
-   $t=0$  η πληροφορία του  $c$  σε χρόνο  $t=0$  βρίσκεται μόνο στον  $c$
-   $t=1$  η κατάσταση του  $c$  σε χρόνο  $t=0$  έχει διαδοθεί στον  $b$ , και μπορεί να επηρεάσει τους υπολογισμούς μέχρι **1** άλμα μακριά π.χ., στον  $b$
-   $t=2$  η κατάσταση του  $c$  σε χρόνο  $t=0$  μπορεί να επηρεάσει τους υπολογισμούς μέχρι **2** άλματα μακριά, π.χ., στον  $b$  και πλέον  $a, e$
-   $t=3$  η κατάσταση του  $c$  σε χρόνο  $t=0$  μπορεί να επηρεάσει τους υπολογισμούς μέχρι **3** άλματα μακριά, π.χ., στους  $b, a, e$  και  $c, f, h$
-   $t=4$  η κατάσταση του  $c$  σε χρόνο  $t=0$  μπορεί να επηρεάσει τους υπολογισμούς μέχρι **4** άλματα μακριά, π.χ., σε  $b, a, e, c, f, h$  και πλέον  $g, i$



# Διάνυσμα απόστασης: Αλλαγές κόστους ζεύξης

## Αλλαγές κόστους ζεύξης:

- Ο κόμβος ανιχνεύει αλλαγή κόστους στην τοπική ζεύξη
- Ενημερώνει την πληροφορία δρομολόγησης, επαναυπολογίζει το DV
- Αν αλλάξει το DV, ενημερώνει τους γείτονες



“τα καλά νέα ταξιδεύουν γρήγορα”

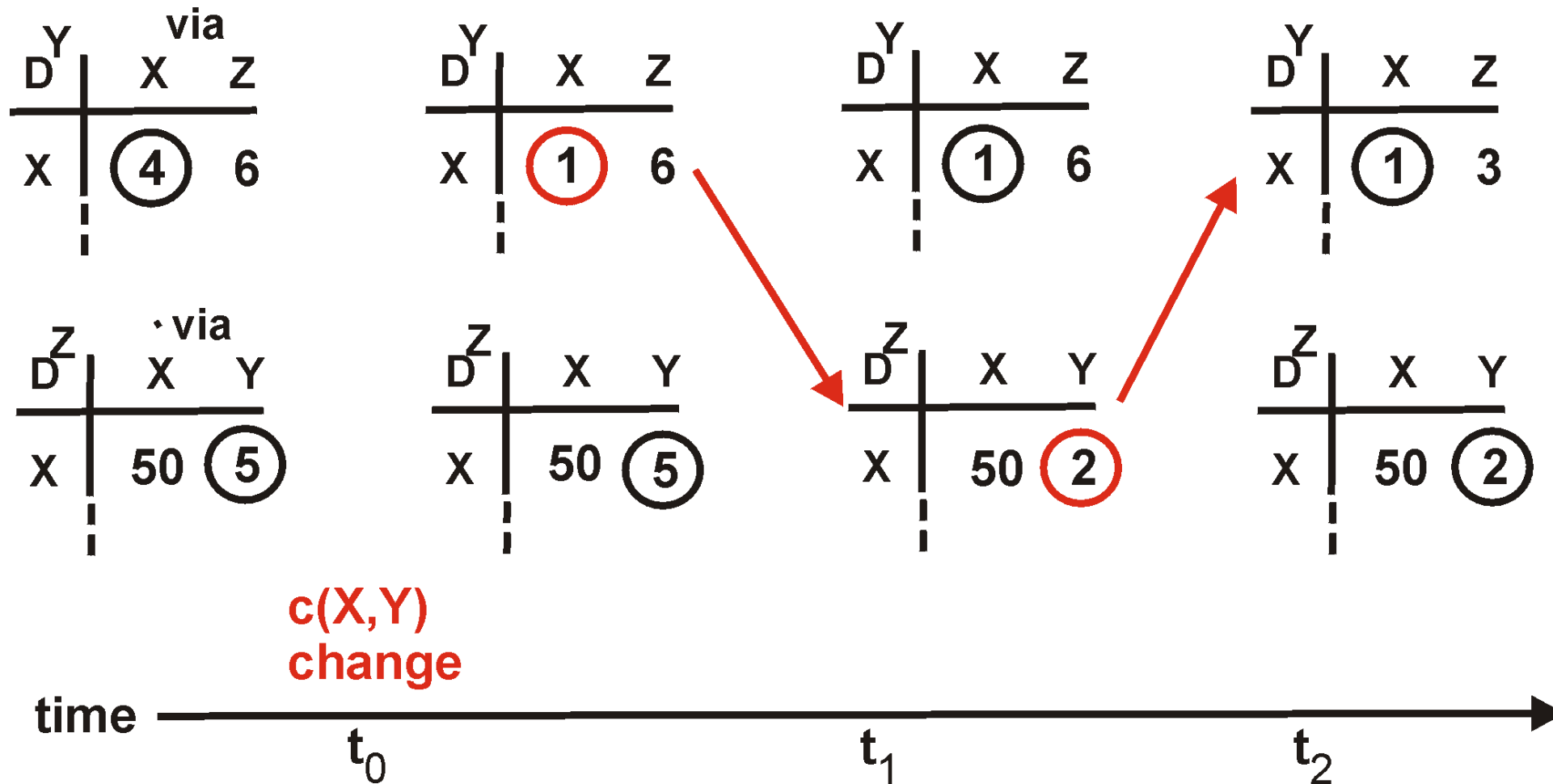
$t_0$ : ο  $y$  ανιχνεύει την αλλαγή στο κόστος ζεύξης, ενημερώνει το DV του, και ενημερώνει τους γείτονές του

$t_1$ : ο  $z$  δέχεται την ενημέρωση από τον  $y$ , ενημερώνει τον πίνακά του, υπολογίζει νέο ελάχιστο κόστος προς τον  $x$ , και στέλνει στους γείτονές του το DV του.

$t_2$ : ο  $y$  δέχεται την ενημέρωση του  $z$  κι ενημερώνει τον πίνακα αποστάσεων. Τα ελάχιστα κόστη του  $y$  δεν αλλάζουν και έτσι ο  $y$  δεν στέλνει κανένα μήνυμα στον  $z$ .



Για δρομολόγηση προς τον x (γραμμή στον πίνακα) αφορά πίνακες των κόμβων y και z) :



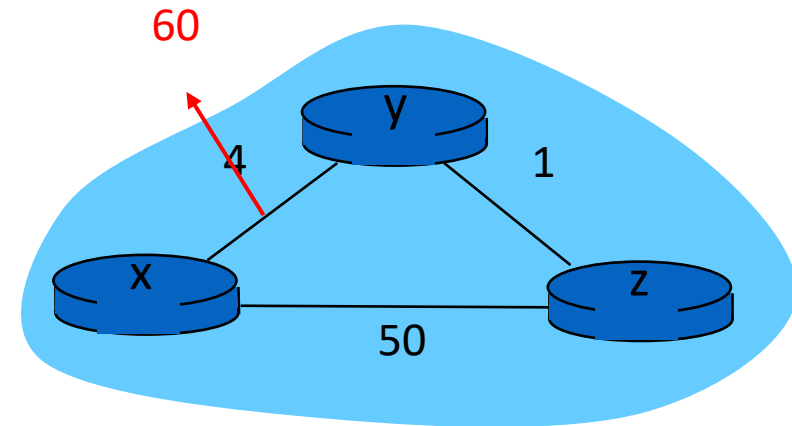
# Διάνυσμα απόστασης: αλλαγές κόστους ζεύξης

## Αλλαγές κόστους ζεύξης:

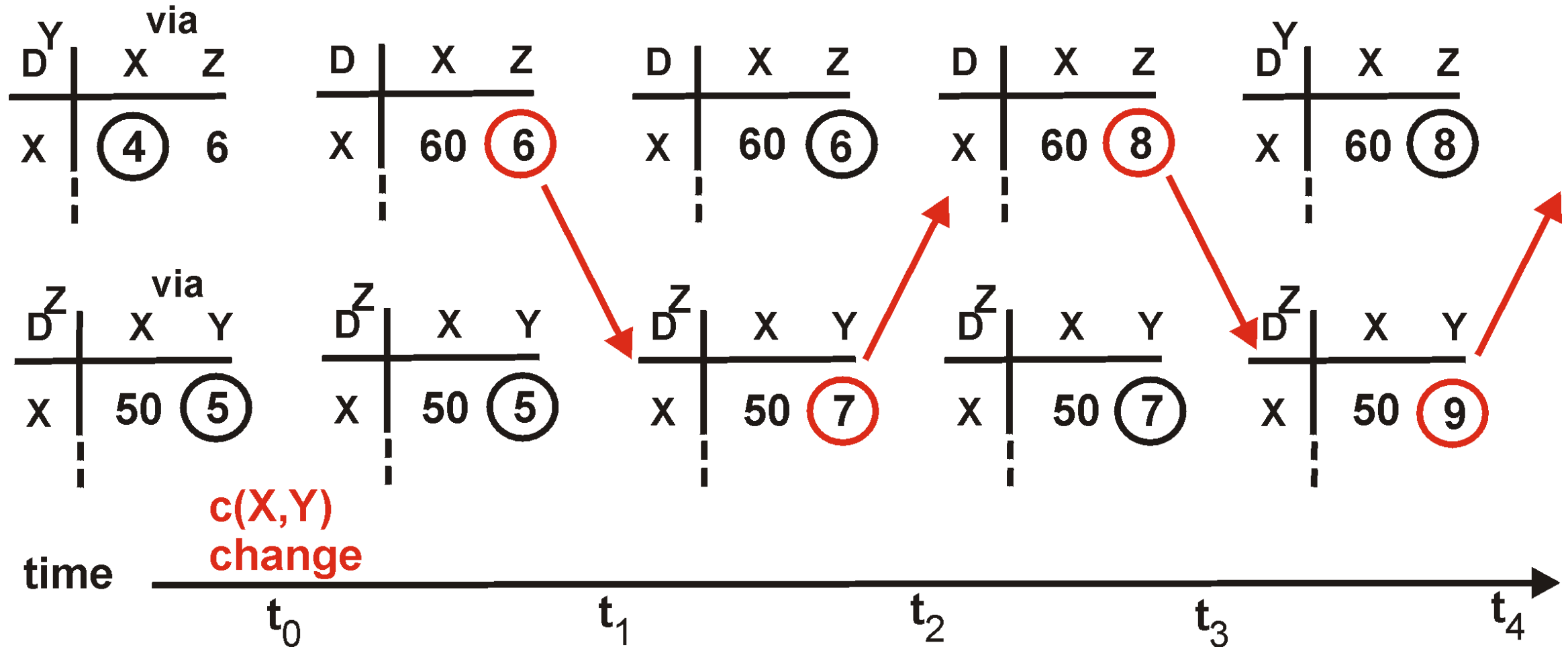
- ❑ Ο κόμβος ανιχνεύει αλλαγή κόστους τοπικής ζεύξης
- ❑ Τα κακά νέα ταξιδεύουν αργά – πρόβλημα “μέτρησης μέχρι το άπειρο”!
- ❑ 44 επαναλήψεις πριν σταθεροποιηθεί ο αλγόριθμος (γιατί;)

## “poisoned reverse”:

- ❑ Αν ο Z δρομολογεί μέσω του Y για να φτάσει στον X
  - Ο Z λέει στον Y ότι η απόστασή του από τον X είναι άπειρη (ώστε ο Y να μη δρομολογεί στον X μέσω του Z)



Για δρομολόγηση προς τον χ (γραμμή στον πίνακα) αφορά πίνακες των κόμβων γ και z :



# Σύγκριση των LS και DV αλγόριθμων

## Πολυπλοκότητα μηνύματος

LS: με  $n$  κόμβους,  $E$  ζεύξεις, στέλνονται  $O(nE)$  μηνύματα

DV: ανταλλαγή μόνο μεταξύ γειτόνων

- ο χρόνος σύγκλισης ποικίλει

## Ταχύτητα σύγκλισης

LS:  $O(n^2)$  ο αλγόριθμος απαιτεί  $O(nE)$  μηνύματα

- μπορεί να έχει ταλαντώσεις

DV: ο χρόνος σύγκλισης ποικίλει

- μπορεί να υπάρχουν βρόχοι δρομολόγησης
- πρόβλημα μέτρησης μέχρι το άπειρο

Ευρωστία: τι συμβαίνει αν ένας δρομολογητής δυσλειτουργεί;

LS:

- ο κόμβος μπορεί να εκπέμψει λάθος κόστος **ζεύξης**
- κάθε κόμβος υπολογίζει μόνο το δικό του πίνακα

DV:

- ο DV κόμβος μπορεί να εκπέμψει εσφαλμένο κόστος διαδρομής
- ο πίνακας κάθε κόμβου χρησιμοποιείται από άλλους
- τα λάθη διαδίδονται μέσω του δικτύου

# Επίπεδο δικτύου: επίπεδο ελέγχου

- εισαγωγή
- αλγόριθμοι δρομολόγησης
  - κατάσταση ζεύξης
  - διάνυσμα απόστασης
- δρομολόγηση ενδο-αυτόνομου συστήματος (intra-ISP routing): OSPF
- δρομολόγηση ανάμεσα σε ISPs: BGP
- επίπεδο ελέγχου SDN
- Internet Control Message Protocol



- διαχείριση δικτύου
  - SNMP
  - NETCONF/YANG

# Ιεραρχική Δρομολόγηση

Η μελέτη της δρομολόγησης ως τώρα εξιδανικευμένη:

- όλοι οι δρομολογητές πανομοιότυποι
  - «επίπεδο» (flat) δίκτυο
- ... δεν ισχύει στην πράξη

**Κλίμακα:** με 600 εκατομμύρια προορισμούς

- ❑ δεν μπορεί να αποθηκευτούν όλοι οι προορισμοί στους πίνακες δρομολόγησης!
- ❑ η ανταλλαγή των πινάκων δρομολόγησης θα κατάκλυζε τις ζεύξεις

**Διαχειριστική αυτονομία**

- ❑ Διαδίκτυο = δίκτυο δικτύων
- ❑ κάθε διαχειριστής δικτύου ενδέχεται να θέλει να ελέγχει τη δρομολόγηση στο δικό του δίκτυο

# Ιεραρχική Δρομολόγηση

- ομαδοποίηση δρομολογητών σε περιοχές, «αυτόνομα συστήματα» (“autonomous systems” (AS))
- δρομολογητές του ίδιου AS τρέχουν το ίδιο πρωτόκολλο (εσωτερικής) δρομολόγησης
  - πρωτόκολλο δρομολόγησης “intra-AS” [πρωτόκολλο δρομολόγησης ενδοαυτόνομου συστήματος]
  - δρομολογητές σε διαφορετικά AS μπορούν να τρέχουν διαφορετικά intra-AS πρωτόκολλα δρομολόγησης

## Δρομολογητής πύλης (Gateway router)

- Στην “άκρη” του δικού του AS
- Έχει ζεύξη με δρομολογητή σε άλλο AS

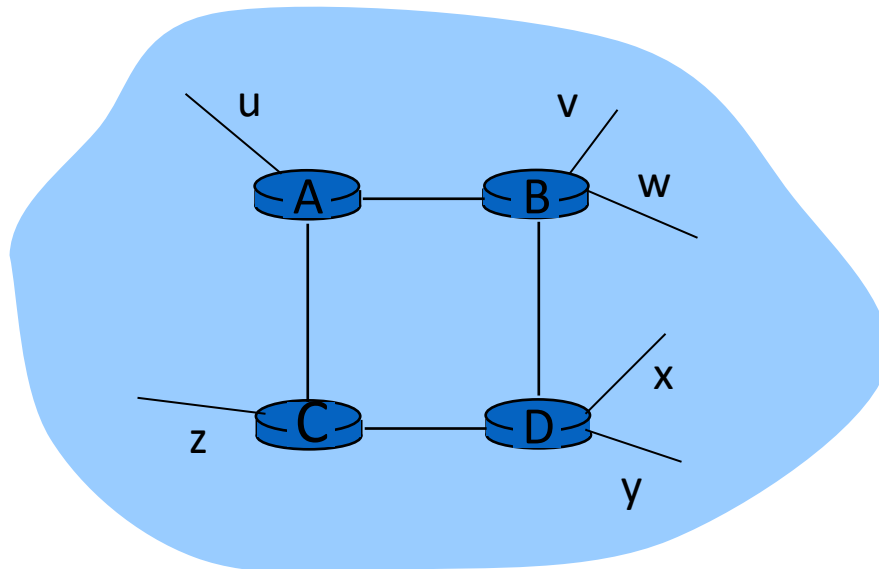
# Δρομολόγηση intra-AS

- Γνωστά και ως **Interior Gateway Protocols (IGP)** (πρωτόκολλα εσωτερικής πύλης/δρομολόγησης)
  
- Τα πιο κοινά Intra-AS πρωτόκολλα δρομολόγησης :
  - RIP: Routing Information Protocol
  - OSPF: Open Shortest Path First
  - IGRP: Interior Gateway Routing Protocol (ιδιοταγές της Cisco)



# RIP ( Routing Information Protocol)

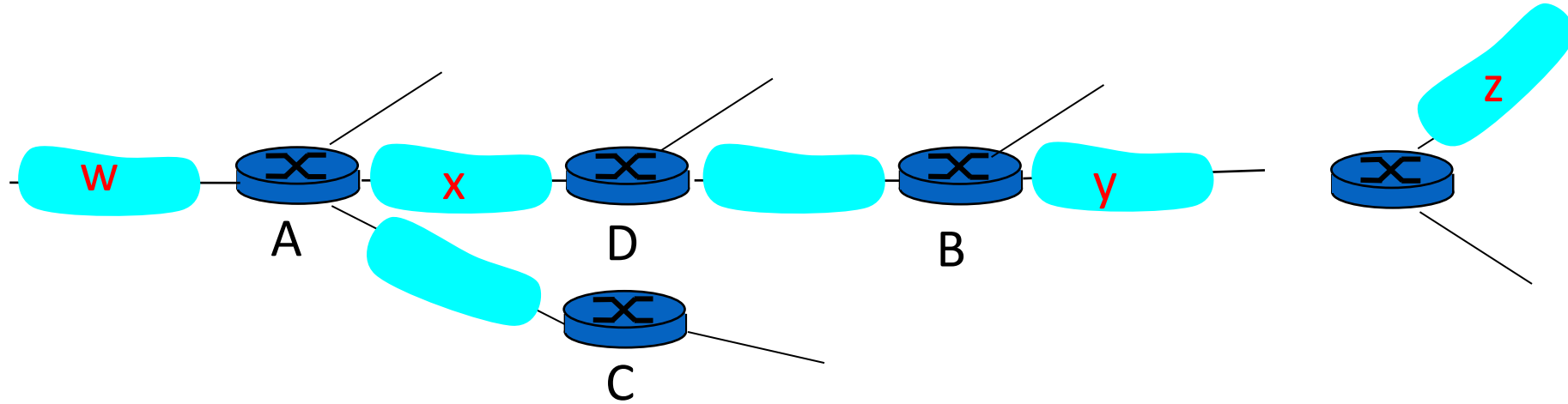
- ❑ Περιελήφθη στη διανομή BSD-UNIX το 1982
- ❑ Αλγόριθμος διανύσματος απόστασης
  - Μετρική απόστασης: # αλμάτων (max = 15 hops), κάθε ζεύξη έχει κόστος 1
  - Ανταλλάσσει DVs με τους γείτονες κάθε 30 sec σε απαντητικό μήνυμα (δημοσιοποίηση)
  - Κάθε δημοσιοποίηση: λίστα μέχρι και 25 υποδικτύα προορισμοί (υπό την έννοια της IP διευθυνσιοδότησης)



Από τον δρομολογητή A στα υποδίκτυα προορισμού:

<u>Προορισμός</u>	<u>Άλματα</u>
u	1
v	2
w	2
x	3
y	3
z	2

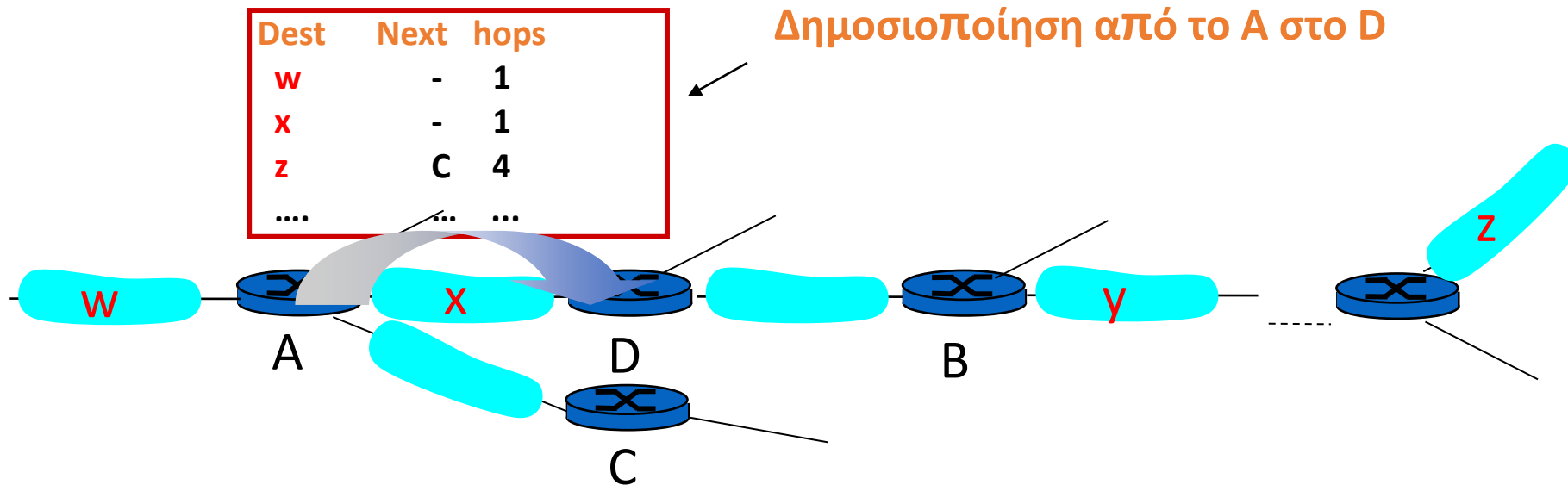
# RIP: Παράδειγμα



Υποδίκτυο προορισμού	Επόμενος δρομολογητής	Πλήθος αλμάτων για προορισμό
<b>W</b>	<b>A</b>	<b>2</b>
<b>Y</b>	<b>B</b>	<b>2</b>
<b>Z</b>	<b>B</b>	<b>7</b>
<b>X</b>	--	<b>1</b>
....	....	....

Πίνακας δρομολόγησης στο δρομολογητή D

# RIP: Παράδειγμα



Υποδίκτυο προορισμού	Επόμενος δρομολογητής	Πλήθος αλμάτων στον προορισμό
w	A	2
y	B	2
z	<del>B</del> A	<del>7</del> 5
x	--	1
....	....	....

Πίνακας δρομολόγησης στο δρομολογητή D

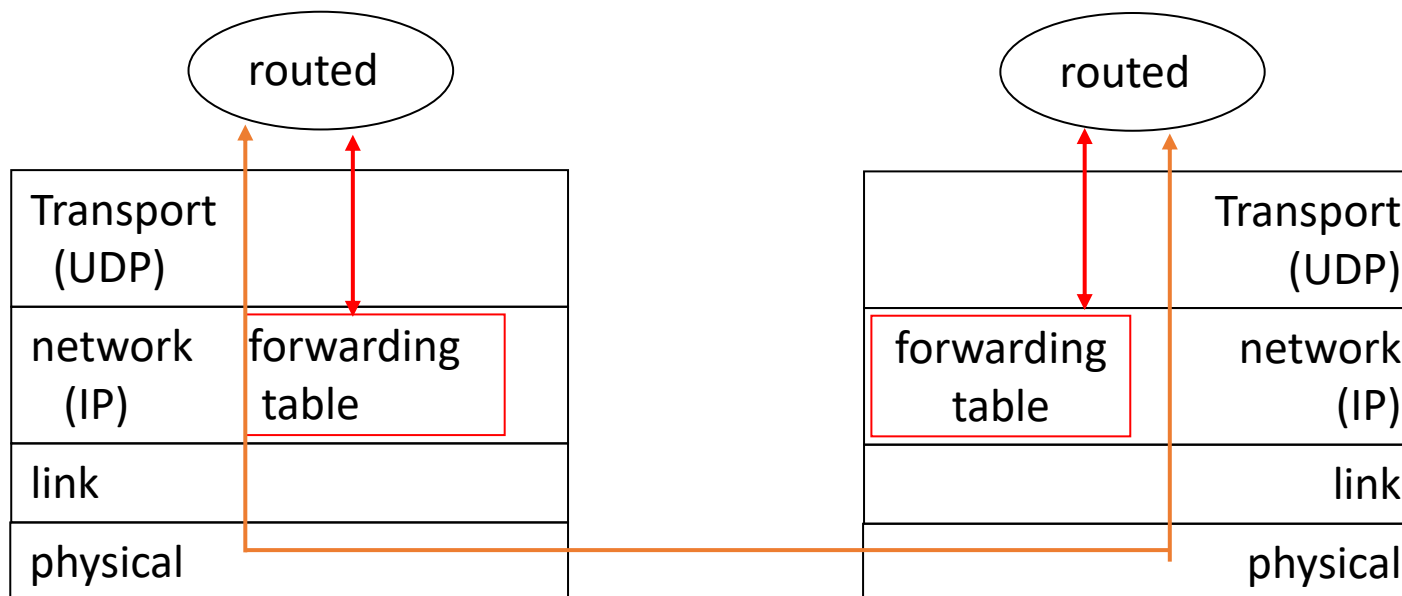
# RIP: Αποτυχία ζεύξης και ανάνηψη

Αν δεν ακουστεί διαφήμιση μετά από 180 sec --> ο γείτονας/η ζεύξη θεωρείται νεκρός

- Οι διαδρομές μέσω του γείτονα παύουν να είναι έγκυρες
- Νέες διαφημίσεις στέλνονται στους γείτονες
- Οι γείτονες με τη σειρά τους στέλνουν νέες διαφημίσεις (αν άλλαξαν οι πίνακες)
- Η πληροφορία αποτυχίας της ζεύξης διαδίδεται γρήγορα (;) σε ολόκληρο το δίκτυο
- *Χρησιμοποιείται poison reverse* για την αποφυγή βρόχων ring-pong (άπειρη απόσταση = 16 άλματα)

# Επεξεργασία πίνακα RIP

- ❑ Η διαχείριση των πινάκων δρομολόγησης του RIP γίνεται από διεργασία **επιπέδου εφαρμογής** που ονομάζεται route-d (daemon)
- ❑ Οι διαφημίσεις στέλνονται σε πακέτα UDP, επαναλαμβάνονται περιοδικά και χρησιμοποιείται ο αριθμός θύρας 520



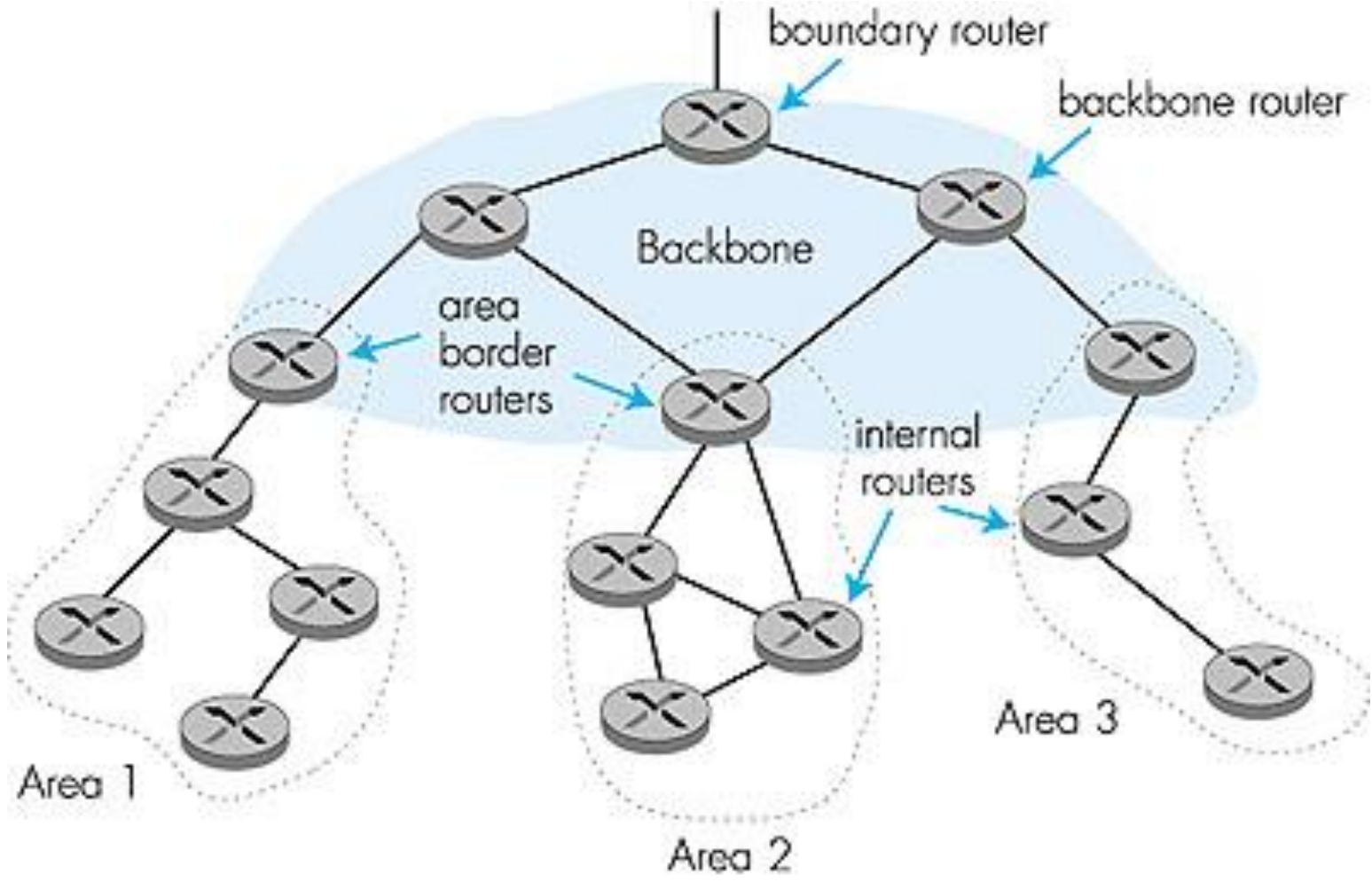
# OSPF (Open Shortest Path First)

- ❑ “Ανοικτό”: δημόσια διαθέσιμη προδιαγραφή
- ❑ Χρησιμοποιεί αλγόριθμο κατάστασης ζεύξης (link state)
  - Διάδοση πακέτου LS
  - Χάρτης τοπολογίας σε κάθε κόμβο
  - Υπολογισμός διαδρομής με χρήση του αλγορίθμου του Dijkstra
- ❑ Η δημοσιοποίηση του OSPF μεταφέρει μόνο μία καταχώριση για κάθε γειτονικό δρομολογητή
- ❑ Οι δημοσιοποιήσεις διαδίδονται σε **ολόκληρο** το AS (μέσω πλημμύρας (flooding))
  - Μεταφέρονται σε μηνύματα του OSPF απευθείας επάνω στο IP (αντί για TCP ή UDP), αριθμός πρωτοκόλλου: 89
- ❑ **Πρωτόκολλο δρομολόγησης IS-IS**: σχεδόν πανομοιότυπο με το OSPF

# «Προηγμένα» χαρακτηριστικά του OSPF (όχι στο RIP)

- ❑ **Ασφάλεια:** όλα τα μηνύματα του OSPF αυθεντικοποιούνται (για αποφυγή κακόβουλων εισβολών)
- ❑ Επιτρέπει **πολλαπλές διαδρομές** ίδιου κόστους (μόνο μία διαδρομή στο RIP)
- ❑ Για κάθε ζεύξη, πολλαπλές μετρικές κόστους για διαφορετικό **TOS** (π.χ., κόστος δορυφορικής ζεύξης θεωρείται «χαμηλό» για TOS βέλτιστης προσπάθειας, υψηλό για TOS πραγματικού χρόνου)
- ❑ Ενσωματωμένη υποστήριξη μονο- και πολυ-εκπομπής (unicast/multicast):
  - Το Multicast OSPF (MOSPF) (OSPF πολυ-εκπομπής) χρησιμοποιεί την ίδια βάση δεδομένων τοπολογίας με το OSPF
- ❑ **Ιεραρχικό** OSPF σε μεγάλους τομείς (domains)

# Ιεραρχικό OSPF





# Ιεραρχικό OSPF

- **Δύο επίπεδα ιεραρχίας:** τοπική περιοχή, δικτυακός κορμός.
  - Δημοσιοποιήσεις κατάστασης ζεύξης μόνο στην περιοχή
  - Κάθε κόμβος έχει τη λεπτομερή τοπολογία; Γνωρίζει μόνο την κατεύθυνση (βραχύτερη διαδρομή) προς δίκτυα σε άλλες περιοχές
- **Παραμεθόριοι συνοριακοί δρομολογητές (area border routers):** “συνοψίζουν” τις αποστάσεις προς δίκτυα στη δική τους περιοχή, τις δημοσιοποιούν σε άλλους παραμεθόριους συνοριακούς δρομολογητές
- **Δρομολογητές δικτυακού κορμού (backbone routers):** τρέχουν δρομολόγηση OSPF περιορισμένη στον δικτυακό κορμό.
- **Συνοριακοί δρομολογητές (boundary routers):** συνδέουν με άλλα AS.

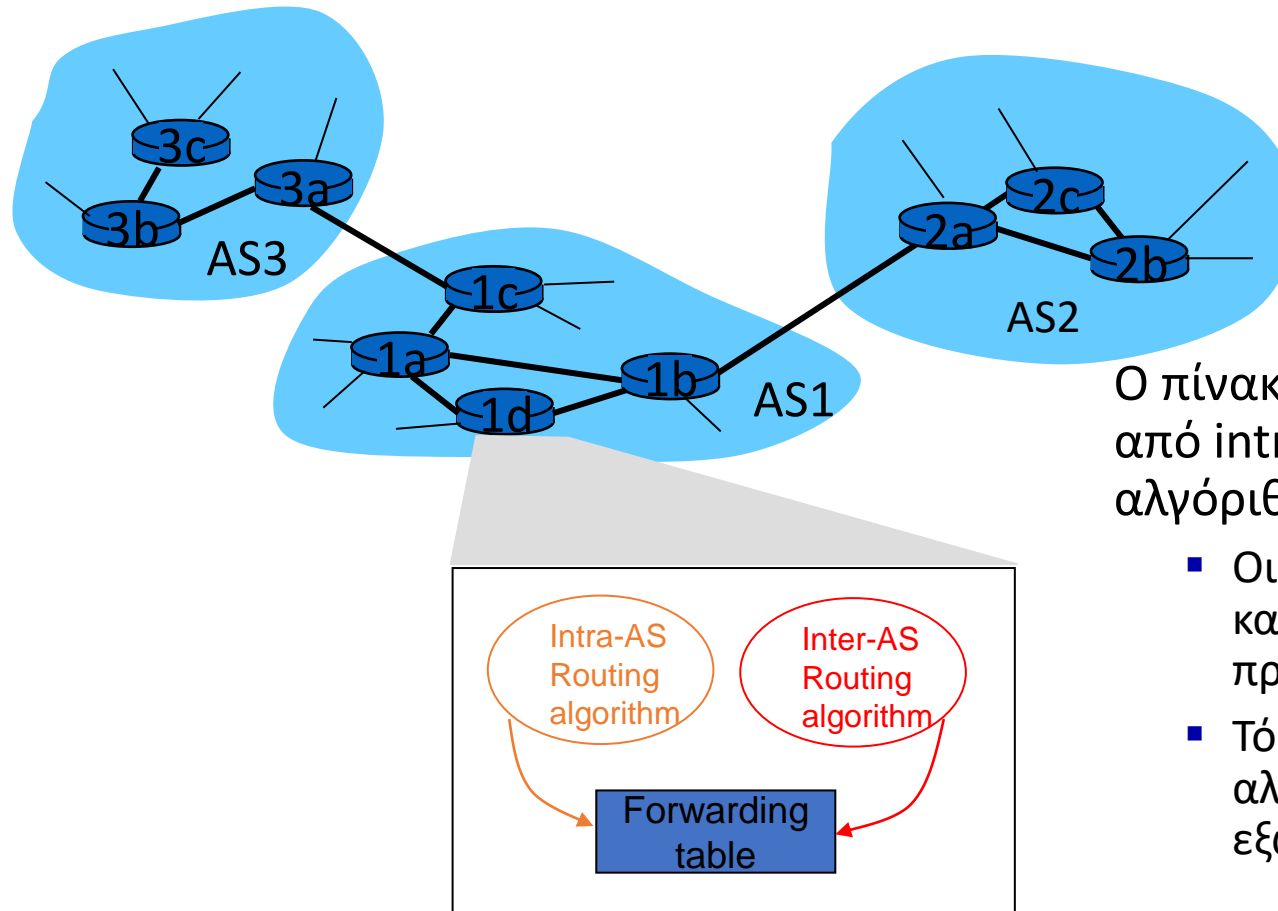
# Επίπεδο δικτύου: επίπεδο ελέγχου

- εισαγωγή
- αλγόριθμοι δρομολόγησης
  - κατάσταση ζεύξης
  - διάνυσμα απόστασης
- δρομολόγηση ενδο-αυτόνομου συστήματος (intra-ISP routing): OSPF
- δρομολόγηση ανάμεσα σε ISPs: BGP
- επίπεδο ελέγχου SDN
- Internet Control Message Protocol



- διαχείριση δικτύου
  - SNMP
  - NETCONF/YANG

# Διασυνδεδεμένα AS



Ο πίνακας προώθησης διαμορφώνεται από intra- (ενδο-) και inter- (δια-) AS αλγόριθμους δρομολόγησης

- Οι intra-AS αλγόριθμοι ορίζουν καταχωρίσεις για εσωτερικούς προορισμούς
- Τόσο οι inter-AS, όσο και οι intra-AS αλγόριθμοι ορίζουν καταχωρίσεις για εξωτερικούς προορισμούς

# Εργασίες Inter-AS

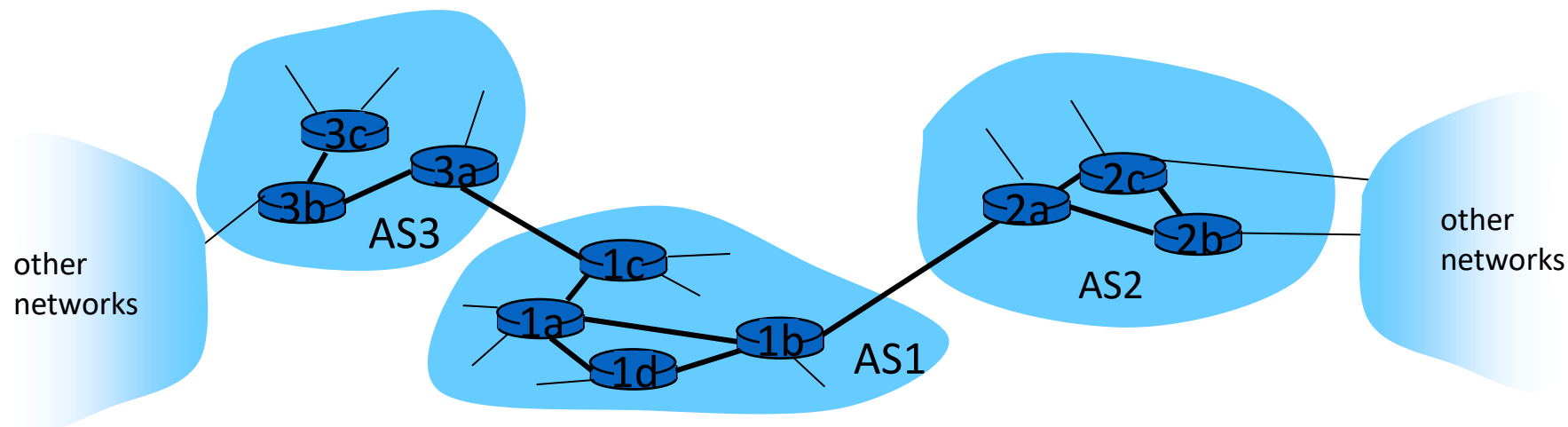
Έστω ότι δρομολογητής στο AS1 λαμβάνει datagram με προορισμό εκτός του AS1:

- Ο δρομολογητής θα έπρεπε να προωθήσει το πακέτο σε δρομολογητή πύλης (gateway router), αλλά σε ποιόν;

Το AS1 πρέπει:

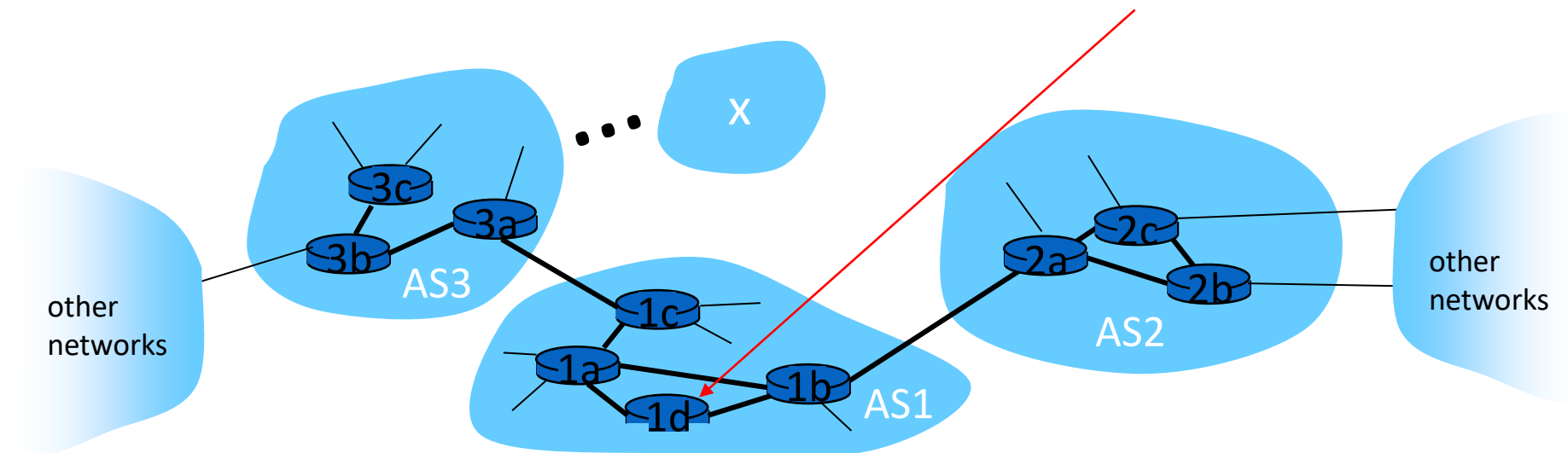
1. Να μάθει ποιοί προορισμοί είναι προσεγγίσιμοι μέσω του AS2 και ποιοί μέσω του AS3
2. Να διαδώσει την πληροφορία προσέγγισης σε όλους τους δρομολογητές στο AS1

Δουλειά της inter-AS δρομολόγησης!



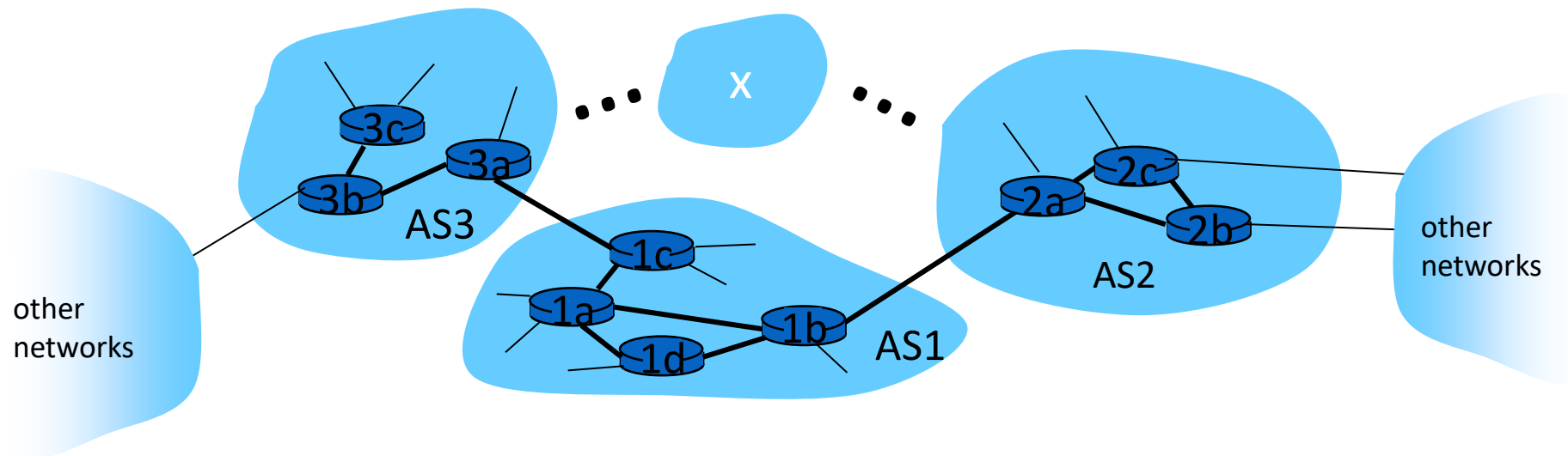
# Παράδειγμα: Καθορισμός του πίνακα προώθησης στο δρομολογητή 1d

- Έστω ότι το AS1 μαθαίνει (μέσω πρωτοκόλλου inter-AS) ότι το υποδίκτυο  $x$  είναι προσεγγίσιμο μέσω του AS3 (δρομολογητής πύλης 1c) αλλά όχι μέσω του AS2.
  - Το πρωτόκολλο inter-AS διαδίδει την πληροφορία προσέγγισης σε όλους τους εσωτερικούς δρομολογητές
- Ο δρομολογητής 1d καθορίζει μέσω της intra-AS πληροφορίας δρομολόγησης ότι η διεπαφή του  $l$  είναι στη διαδρομή ελάχιστου κόστους προς το 1c.
  - Εισάγει καταχώριση στον πίνακα προώθησης  $(x, l)$



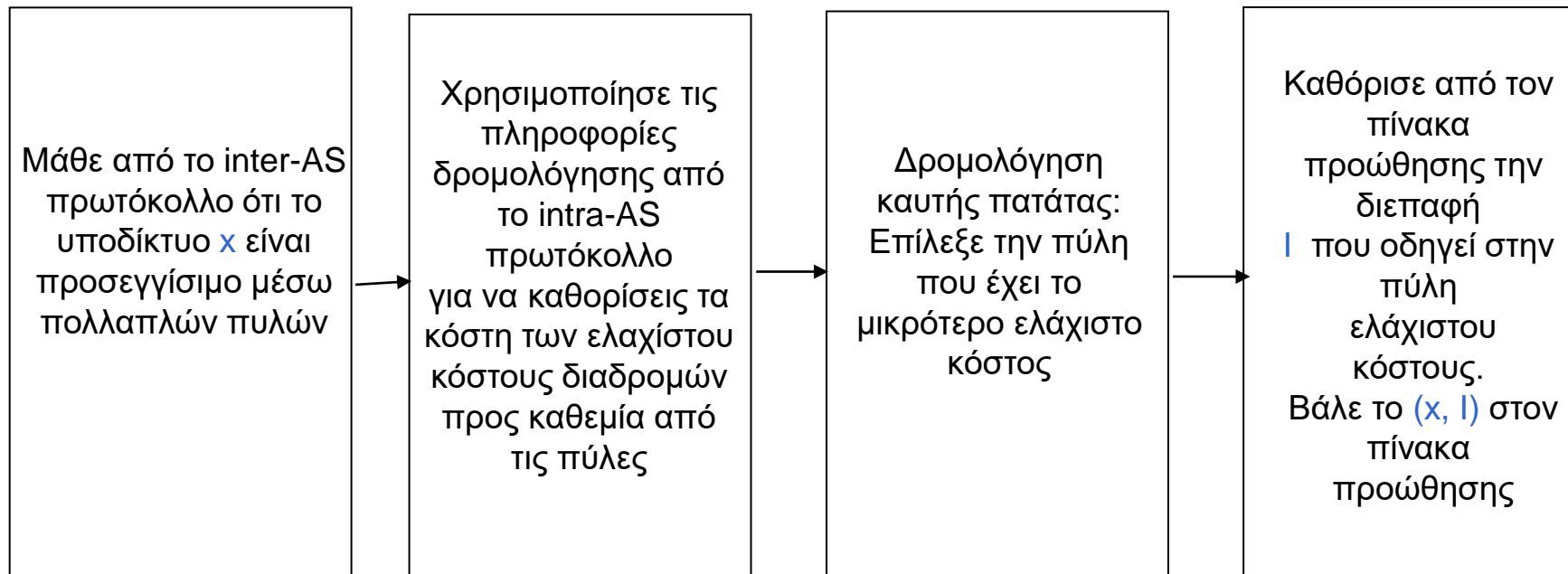
# Παράδειγμα: Διαλέγοντας μεταξύ πολλαπλών AS

- Έστω τώρα ότι το AS1 μαθαίνει από το inter-AS πρωτόκολλο ότι το υποδίκτυο  $x$  είναι προσεγγίσιμο από το AS3 *και* από το AS2
- Για τη διαμόρφωση του πίνακα προώθησης, ο δρομολογητής 1d πρέπει να καθορίσει προς ποιόν δρομολογητή πύλης θα πρέπει να προωθεί τα πακέτα για τον προορισμό  $x$ 
  - Αυτό είναι επίσης δουλειά του inter-AS πρωτοκόλλου δρομολόγησης!



# Παράδειγμα: Διαλέγοντας μεταξύ πολλαπλών AS

- Δρομολόγηση καυτής πατάτας (hot potato routing): στείλε το πακέτο στον πιο κοντινό από τους δύο δρομολογητές



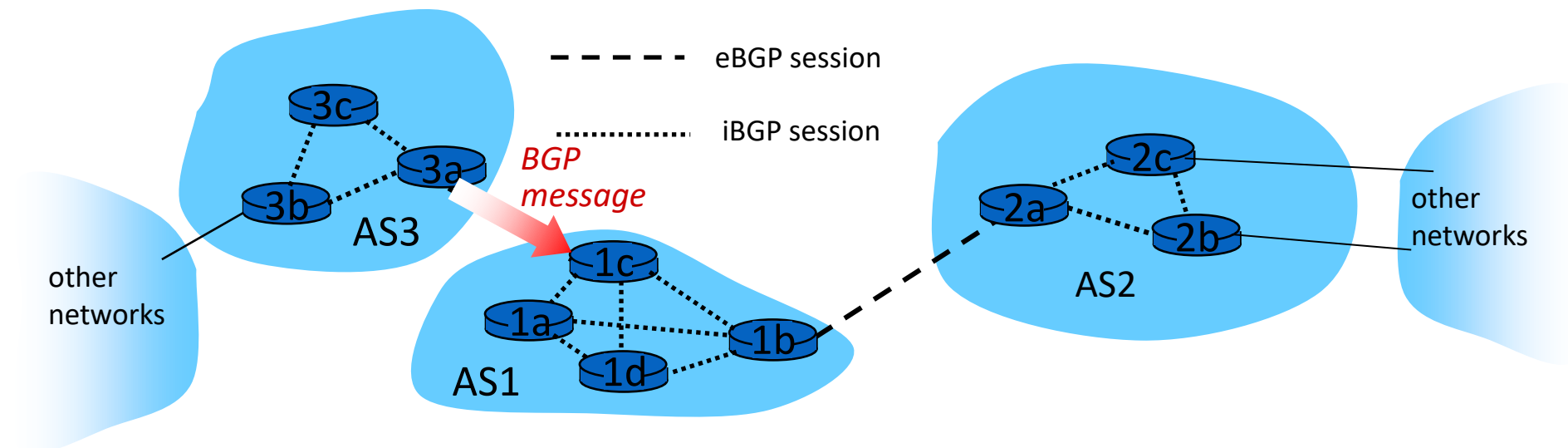
# Δρομολόγηση inter-AS στο Διαδίκτυο: BGP

- ❑ **BGP (Border Gateway Protocol):** το de facto πρωτόκολλο δρομολόγησης μεταξύ τομέων (domains)
- ❑ Το BGP παρέχει σε κάθε AS ένα τρόπο για:
  - **eBGP:** να λαμβάνει πληροφορίες προσέγγισης υποδικτύου από γειτονικά AS.
  - **iBGP:** να διαδίδει τις πληροφορίες προσέγγισης σε όλους τους δρομολογητές που είναι εσωτερικοί στο AS
  - Να καθορίζει τις “καλές” διαδρομές προς άλλα δίκτυα με βάση τις πληροφορίες προσέγγισης και μια πολιτική
- ❑ Επιτρέπει σε κάθε υποδίκτυο να διαφημίζει την ύπαρξή του στο υπόλοιπο Διαδίκτυο: *“Είμαι εδώ”*



# Τα βασικά του BGP

- **Σύνοδοι BGP (BGP sessions):** 2 BGP δρομολογητές (ομότιμοι) ανταλλάσσουν BGP μηνύματα:
  - Δημοσιοποιούν **διαδρομές** προς διαφορετικού προορισμού δικτυακά προθέματα (πρωτόκολλο “διανύσματος διαδρομής”)
  - Ανταλλάσσονται πάνω από ημι-μόνιμες TCP συνδέσεις (θύρα 179)
- Όταν το AS3 διαφημίζει ένα πρόθεμα στον AS1:
  - Το AS3 **υπόσχεται** ότι θα προωθήσει datagrams προς αυτό το πρόθεμα.
  - Το AS3 μπορεί να ομαδοποιήσει τα προθέματα στη δημοσιοποίησή του

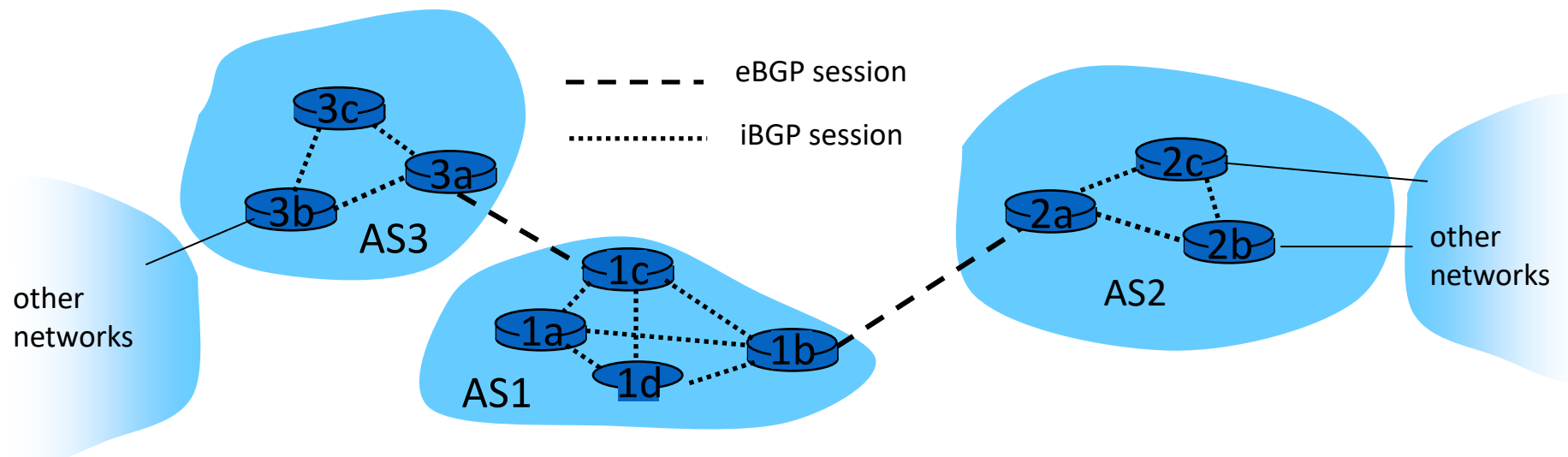


# Μηνύματα BGP

- ❑ Τα μηνύματα BGP ανταλλάσσονται μεταξύ ομότιμων πάνω από TCP σύνδεση.
- ❑ Μηνύματα BGP:
  - **OPEN**: ανοίγει σύνδεση TCP προς τον ομότιμο και ταυτοποιεί τον αποστολέα
  - **UPDATE**: δημοσιοποιεί νέα διαδρομή (ή αποσύρει παλιά)
  - **KEEPALIVE**: διατηρεί τη σύνδεση ζωντανή ελλείψει UPDATES. Επίσης επιβεβαιώνει αίτηση OPEN
  - **NOTIFICATION**: αναφέρει σφάλματα σε προηγούμενο μήνυμα και χρησιμοποιείται για το κλείσιμο σύνδεσης

# Διανομή πληροφορίας προσέγγισης

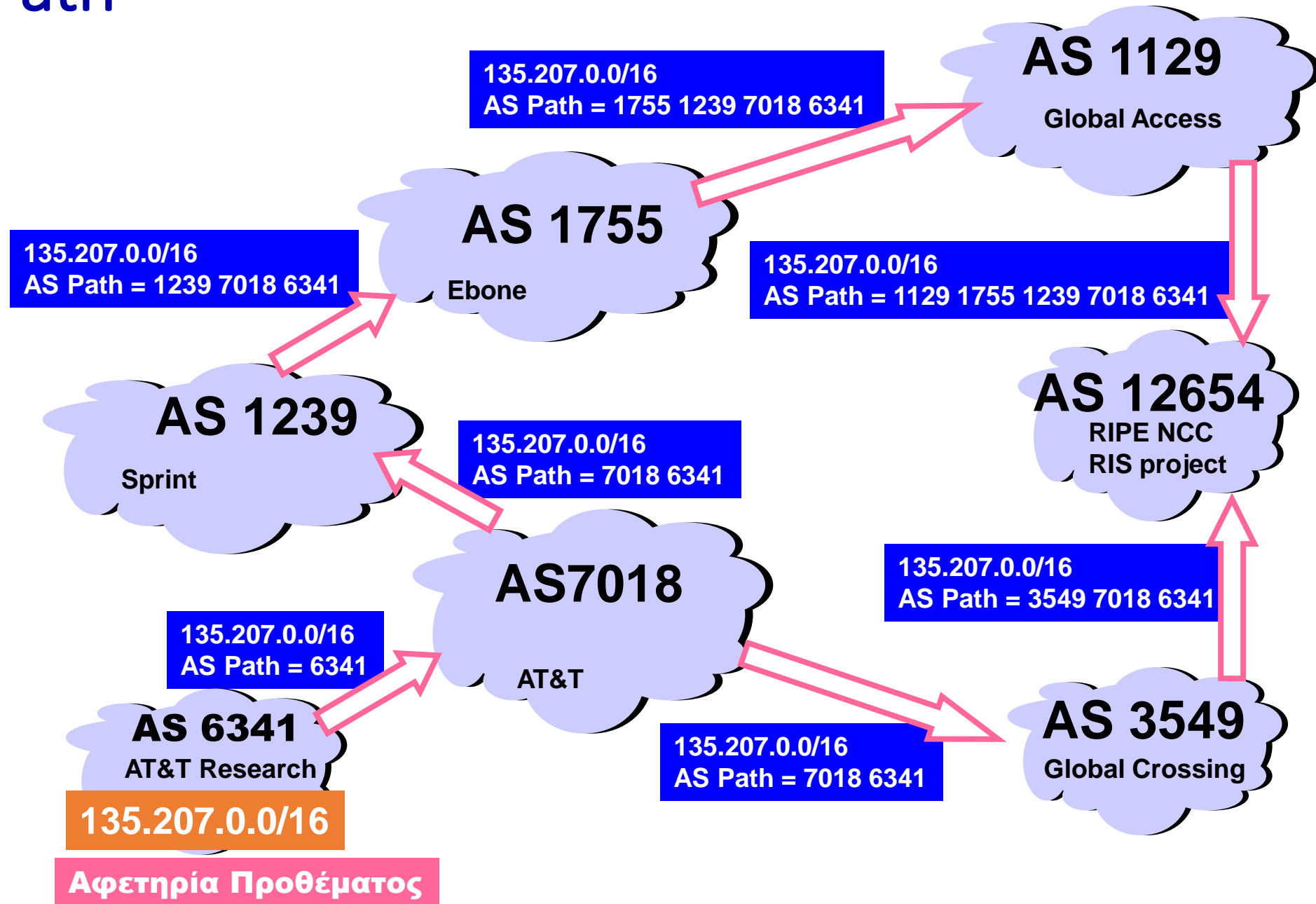
- Χρησιμοποιώντας σύνοδο eBGP μεταξύ των 3a και 1c, το AS3 στέλνει πληροφορίες δυνατότητας προσέγγισης προθέματος στο AS1.
  - Το 1c μπορεί τότε να χρησιμοποιήσει iBGP για να διανείμει νέες πληροφορίες προθέματος σε όλους τους δρομολογητές του AS1
  - Το 1b μπορεί τότε να διαφημίσει εκ νέου τις νέες πληροφορίες προσέγγισης στο AS2 μέσω της συνεδρίας eBGP από το 1b-στο-2a
- Όταν κάποιος δρομολογητής μαθαίνει για νέο πρόθεμα δημιουργεί καταχώριση για το πρόθεμα στον πίνακα προώθησής του



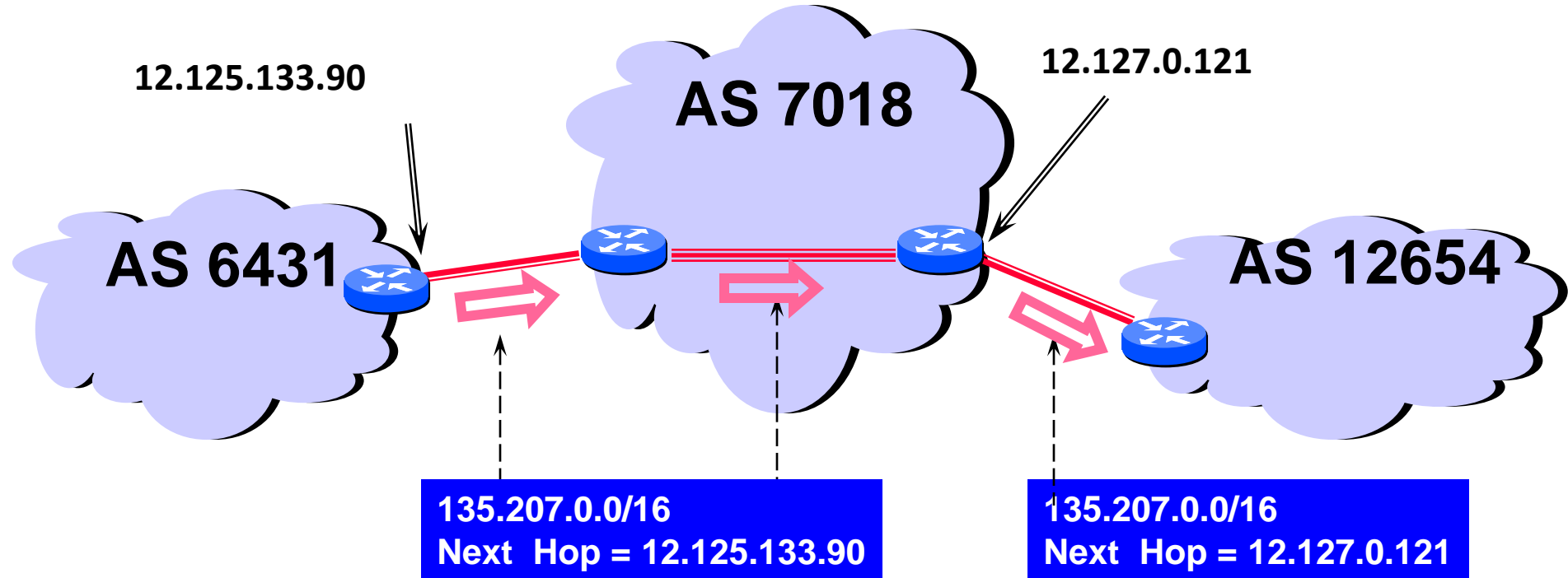
# Ιδιοχαρακτηριστικά διαδρομής & διαδρομές BGP

- Ένα δημοσιοποιούμενο πρόθεμα περιλαμβάνει χαρακτηριστικά του BGP
  - Πρόθεμα + ιδιοχαρακτηριστικά = “διαδρομή” (route)
- Δύο σημαντικά ιδιοχαρακτηριστικά:
  - **AS Path**: περιλαμβάνει τα AS μέσω των οποίων έχει περάσει η διαφήμιση για το πρόθεμα: π.χ., AS 67, AS 17
  - **Next Hop**: υποδεικνύει συγκεκριμένο εσωτερικό στο AS δρομολογητή προς το AS επόμενου άλματος (ενδέχεται να υπάρχουν πολλαπλές ζεύξεις από το τρέχον AS στο AS επόμενου άλματος)
- Όταν ένας δρομολογητής πύλης λαμβάνει τη διαφήμιση διαδρομής, χρησιμοποιεί την **πολιτική εισαγωγής (import policy)** για να δεχτεί/απορρίψει
  - π.χ., ποτέ μη δρομολογείς μέσω του AS x
  - δρομολόγηση βασισμένη σε κάποια πολιτική

# AS Path

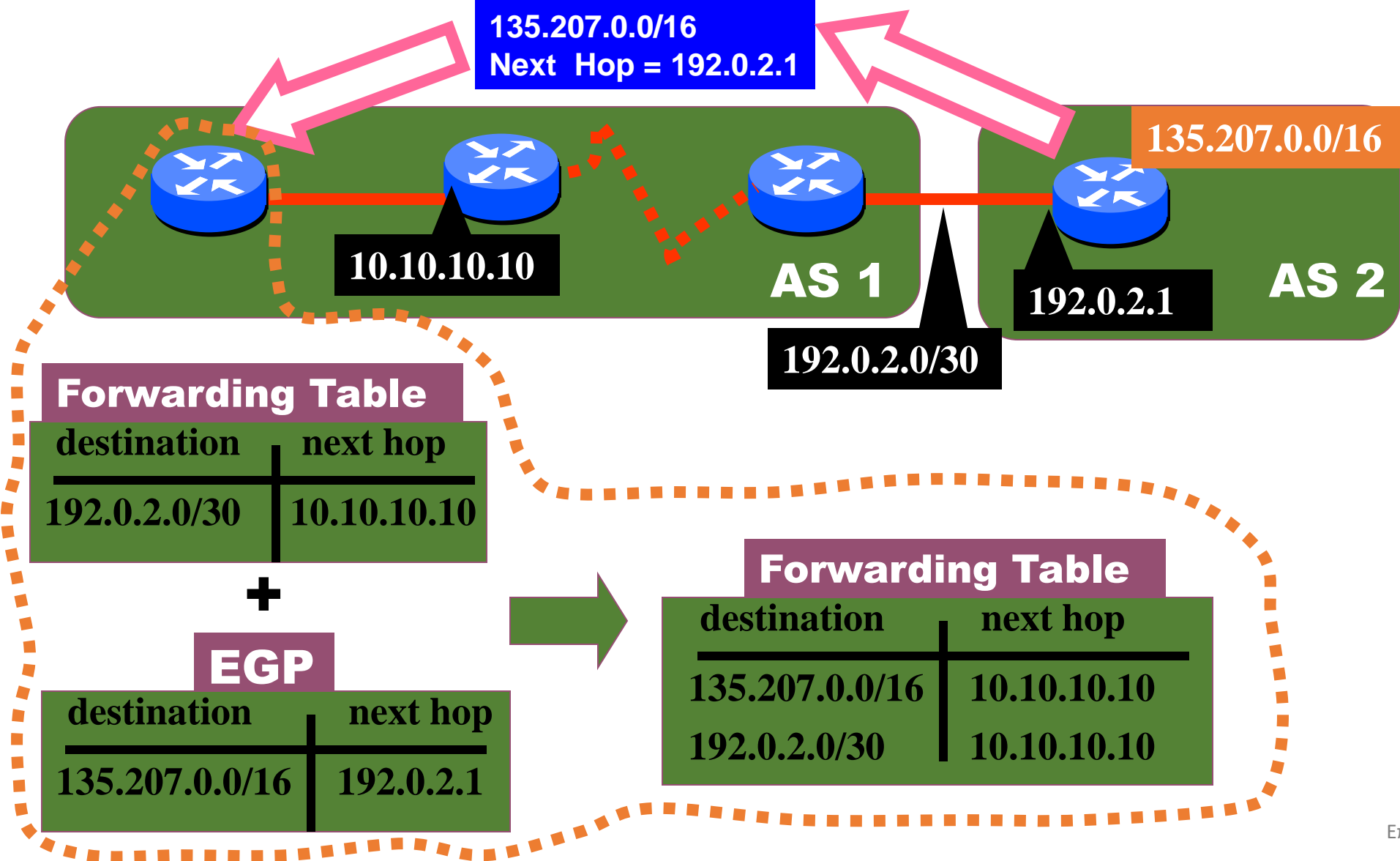


# Next Hop



Κάθε φορά που μια ανακοίνωση προθέματος διασχίζει τα σύνορα ενός ΑΣ, το Next Hop attribute αλλάζει στην IP address του συνοριακού δρομολογητή που ανακοίνωσε την διαδρομή.

# Πίνακας Προώθησης: Συνδυασμός intra-AS και inter-AS δρομολόγησης για προώθηση



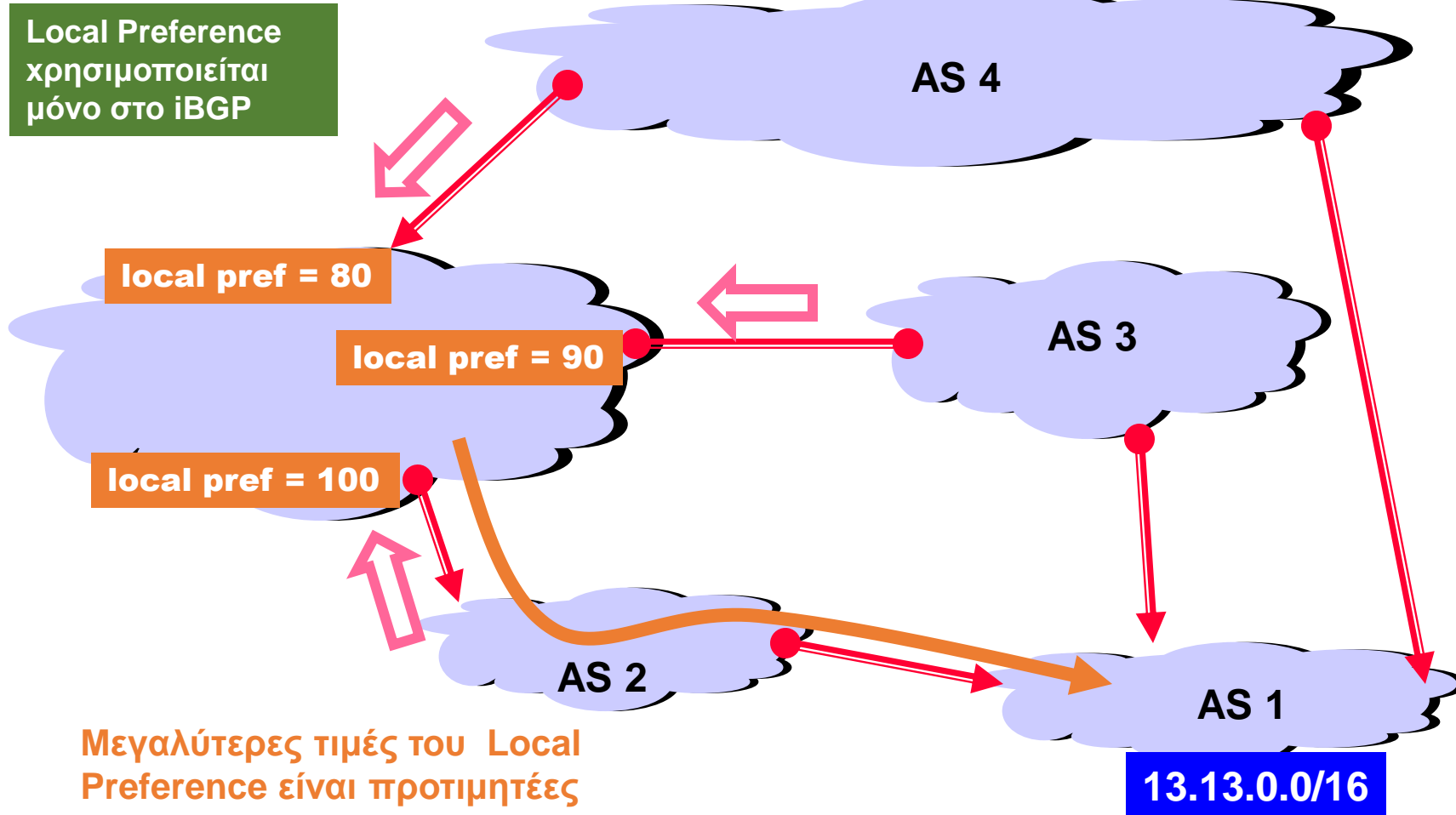
# Επιλογή διαδρομής BGP

Ο δρομολογητής ενδέχεται να μάθει για περισσότερες από μία διαδρομές για το AS προορισμού. Επιλέγει διαδρομή βασισμένος σε:

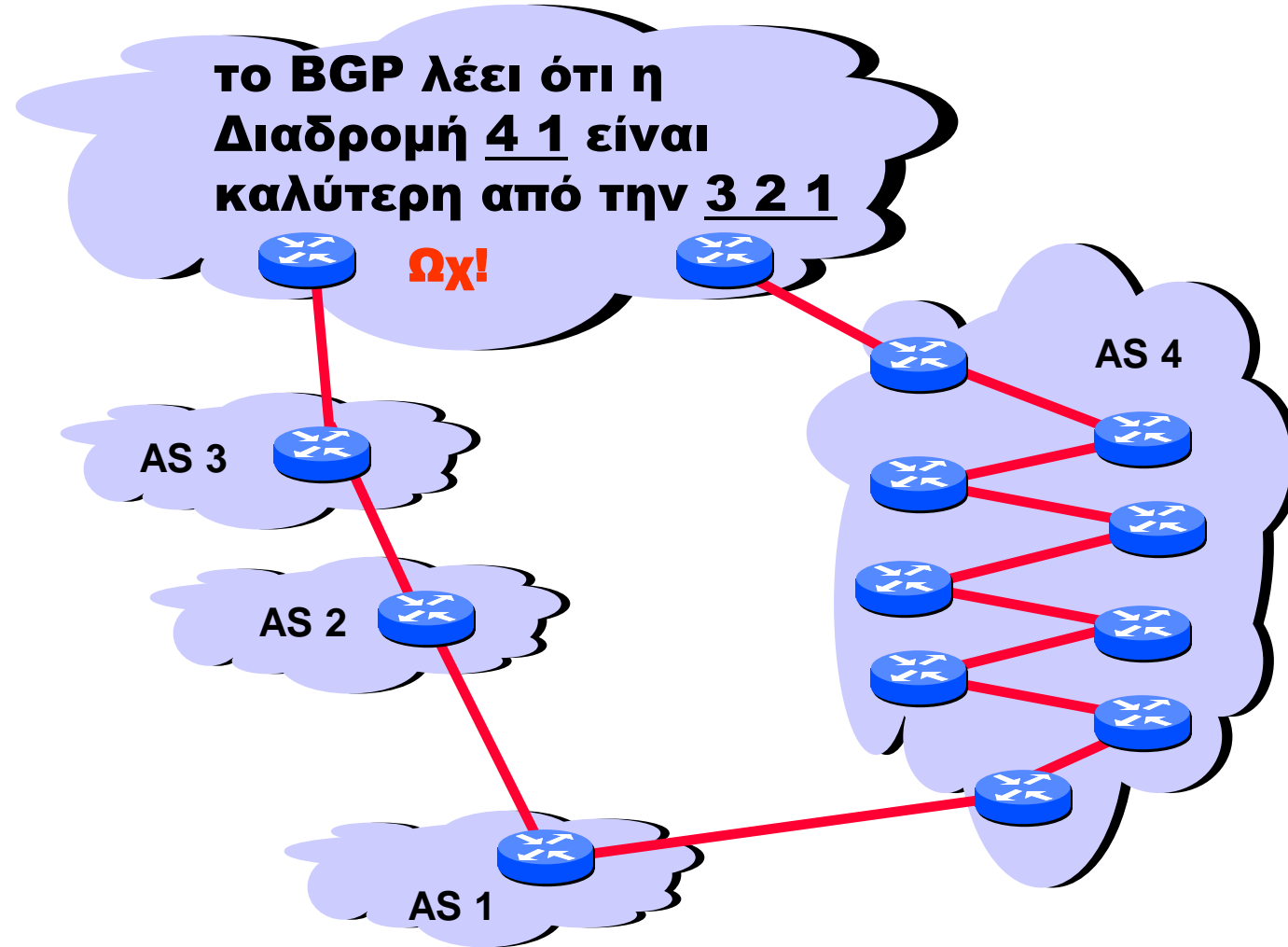
1. Ιδιοχαρακτηριστικό τιμής τοπικής προτίμησης: απόφαση πολιτικής
2. Βραχύτερο AS-PATH
3. Πλησιέστερος δρομολογητής NEXT-HOP: δρομολόγηση καυτής πατάτας
4. Πρόσθετα κριτήρια



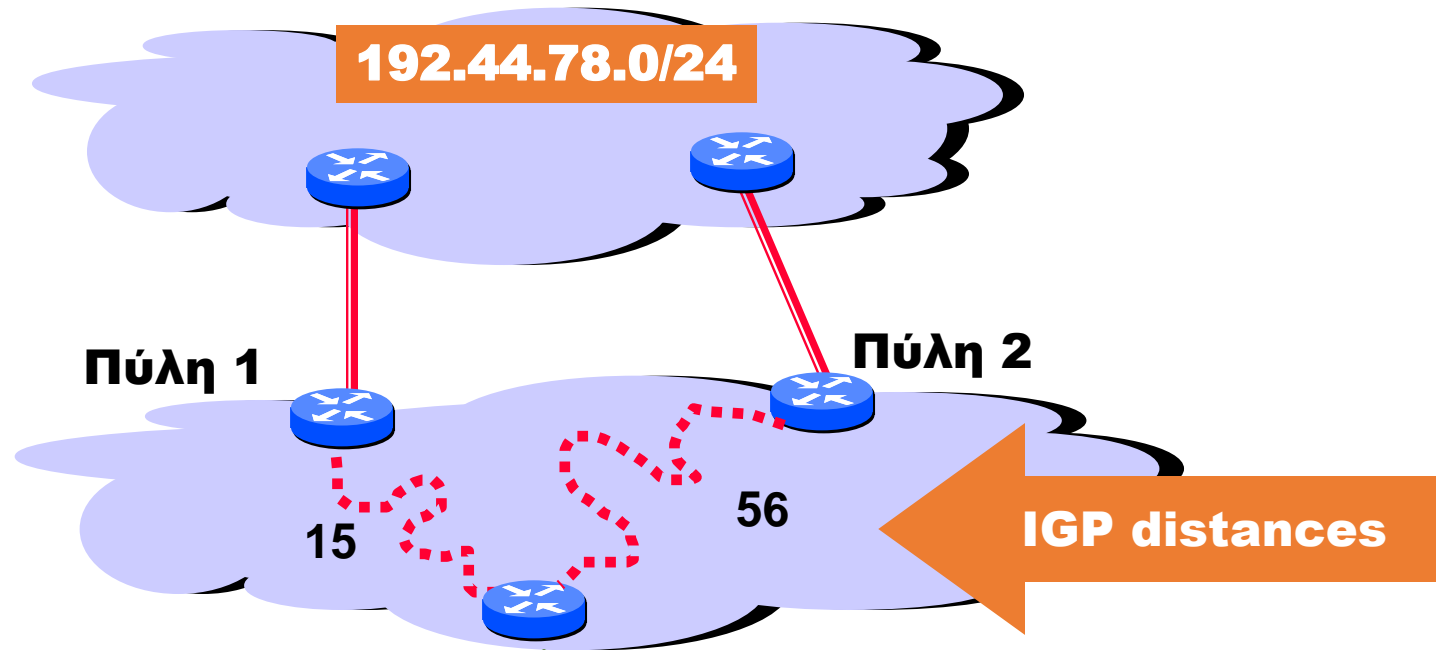
# Τοπική Προτίμηση (Local Preference)



# Συντομότερο δεν σημαίνει πάντα ...συντομότερο!



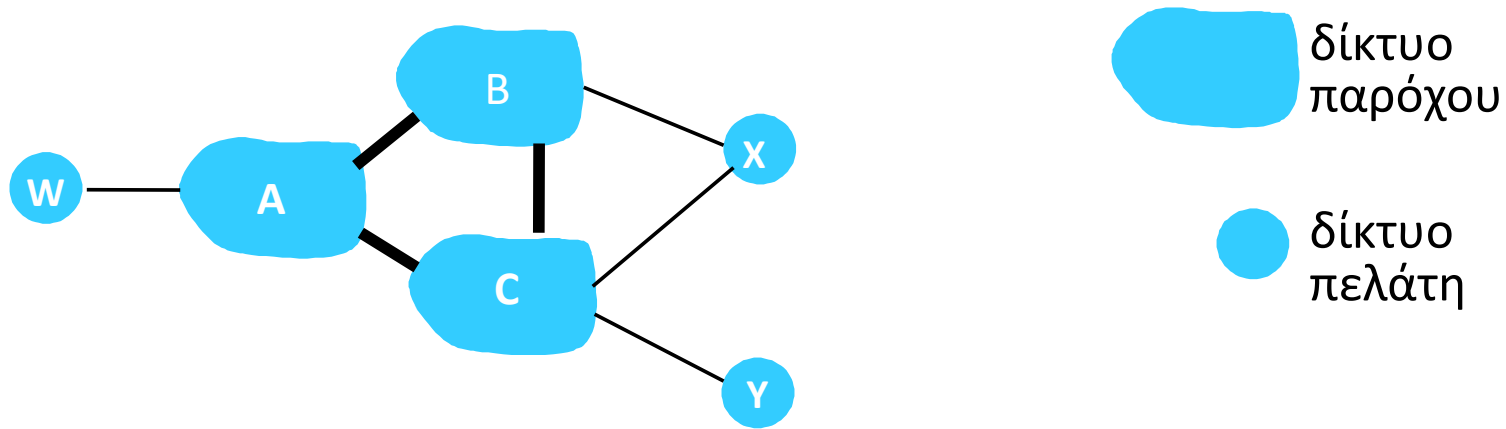
# Hot Potato Routing: Επίλεξε το πλησιέστερο σημείο εξόδου



**Αυτός ο δρομολογητής έχει 2 BGP διαδρομές προς 192.44.78.0/24**

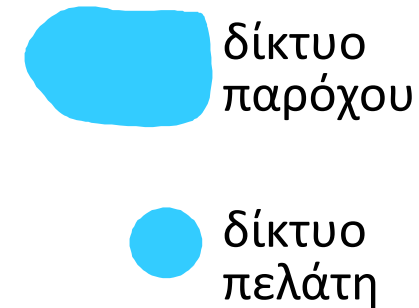
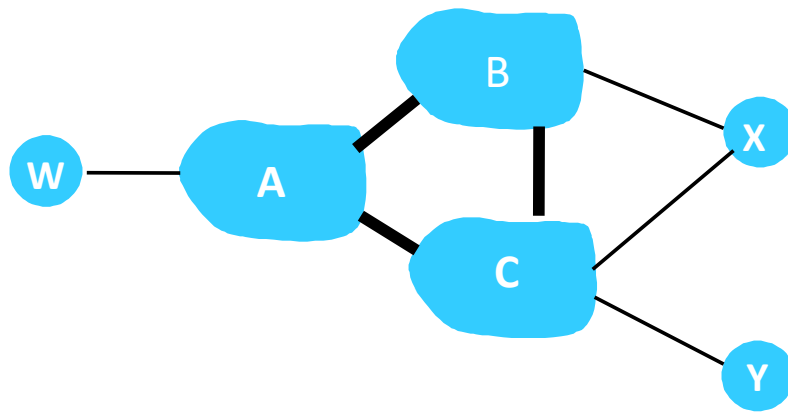
**Hot potato: ξεφορτώσου την κίνηση από το δίκτυό σου το συντομότερο δυνατό. Επίλεξε Πύλη 1!**

# Πολιτική δρομολόγησης BGP



- ❑ Τα A,B,C είναι δίκτυα παρόχων (provider networks)
- ❑ Τα X,W,Y είναι πελάτες (των δικτύων παρόχων)
- ❑ Το X είναι διεστιακό (dual-homed): συνδέεται σε δύο δίκτυα
  - Το X δεν θέλει να δρομολογεί από το B μέσω του X προς το C
  - .. έτσι το X δεν θα δημοσιοποιήσει στο B μια διαδρομή προς το C

# Πολιτική δρομολόγησης BGP (2)



- ❑ Το A διαφημίζει τη διαδρομή AW στο B
- ❑ Το B διαφημίζει τη διαδρομή BAW στο X
- ❑ Θα πρέπει το B να διαφημίσει τη διαδρομή BAW στο C;
  - Σε καμία περίπτωση! Το B δεν έχει όφελος από τη δρομολόγηση CBAW, καθώς ούτε το W ούτε το C είναι πελάτες του B
  - Το B θέλει να εξαναγκάσει το C να δρομολογεί προς το W μέσω του A
  - Το B θέλει να δρομολογεί *μόνο* προς/από τους πελάτες του!

# Γιατί διαφορετική δρομολόγηση intra- και inter-AS (εσωτερική και εξωτερική);

## Πολιτική:

- inter-AS (εξωτερική): ο διαχειριστής θέλει έλεγχο στο πώς δρομολογείται η κίνησή του, ποιός δρομολογεί μέσω του δικτύου του
- intra-AS (εσωτερική): μοναδικός διαχειριστής, οπότε δεν χρειάζονται αποφάσεις πολιτικής

## Κλίμακα:

- Η ιεραρχική δρομολόγηση εξοικονομεί μέγεθος πίνακα, μειωμένη κίνηση ενημερώσεων

## Απόδοση:

- intra-AS: μπορεί να επικεντρώσει στην απόδοση
- inter-AS: η πολιτική μπορεί να κυριαρχήσει της απόδοσης

# Επίπεδο δικτύου: επίπεδο ελέγχου

- εισαγωγή
- αλγόριθμοι δρομολόγησης
  - κατάσταση ζεύξης
  - διάνυσμα απόστασης
- δρομολόγηση ενδο-αυτόνομου συστήματος (intra-ISP routing): OSPF
- δρομολόγηση ανάμεσα σε ISPs: BGP
- **επίπεδο ελέγχου SDN**
- Internet Control Message Protocol



- διαχείριση δικτύου
  - SNMP
  - NETCONF/YANG

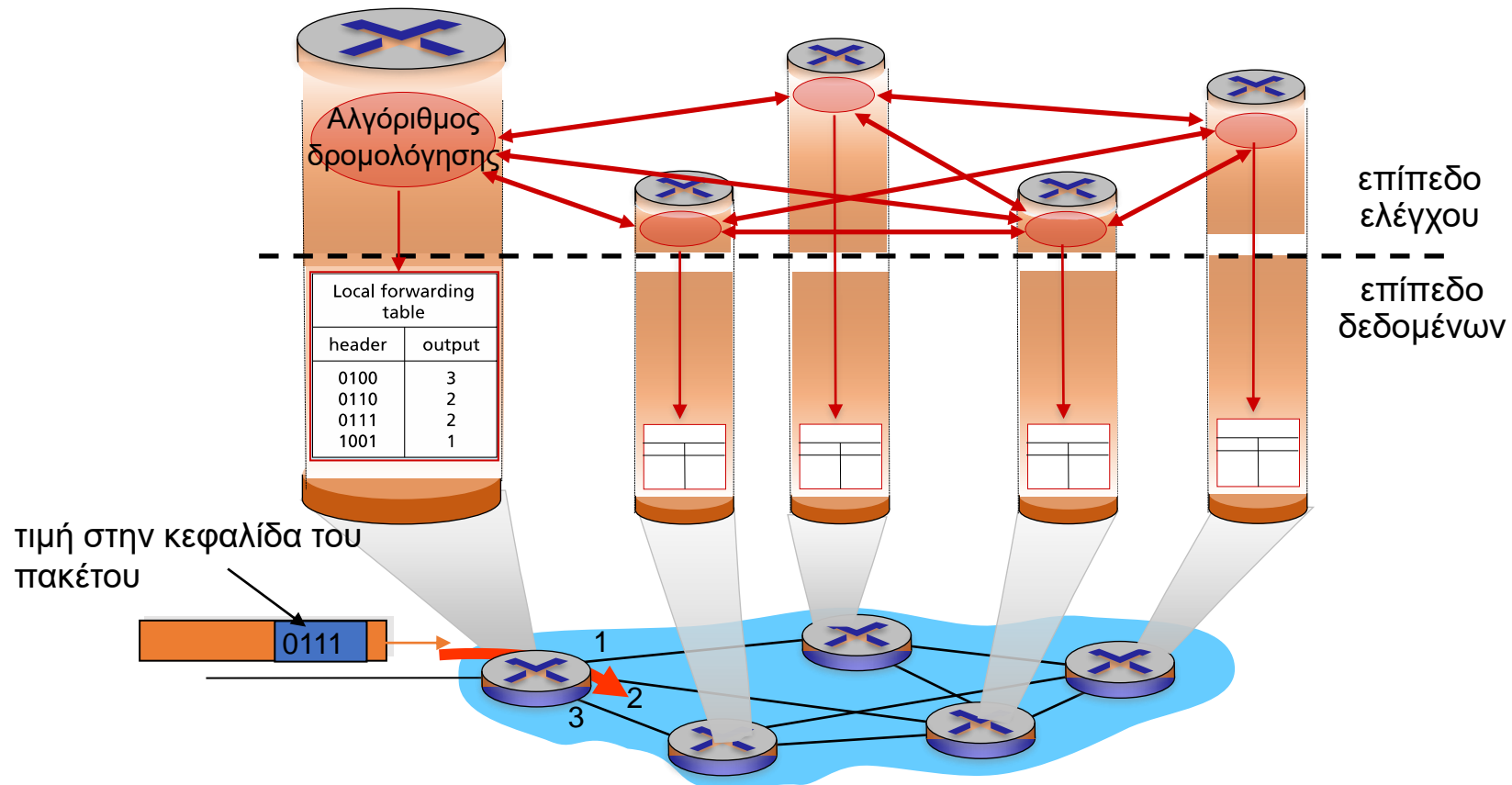
# Software defined networking (SDN)

- Το επίπεδο Δικτύου στο Διαδίκτυο: ιστορικά, αναπτύχθηκε μέσω κατανεμημένης προσέγγισης (ανά δρομολογητή)
  - ο παραδοσιακός δρομολογητής περιλαμβάνει υλικό μεταγωγής, «τρέχει» πρωτόκολλα δρομολόγησης κι έχει το δικό του λειτουργικό σύστημα
  - διαφορετικά ενδιάμεσα κουτιά – “middleboxes” – για διαφορετικές δικτυακές λειτουργίες: firewalls, load balancers, NAT boxes, ..
- ~2005: αναθερμάνθηκε το ενδιαφέρον για μια άλλη λογική στο επίπεδο ελέγχου



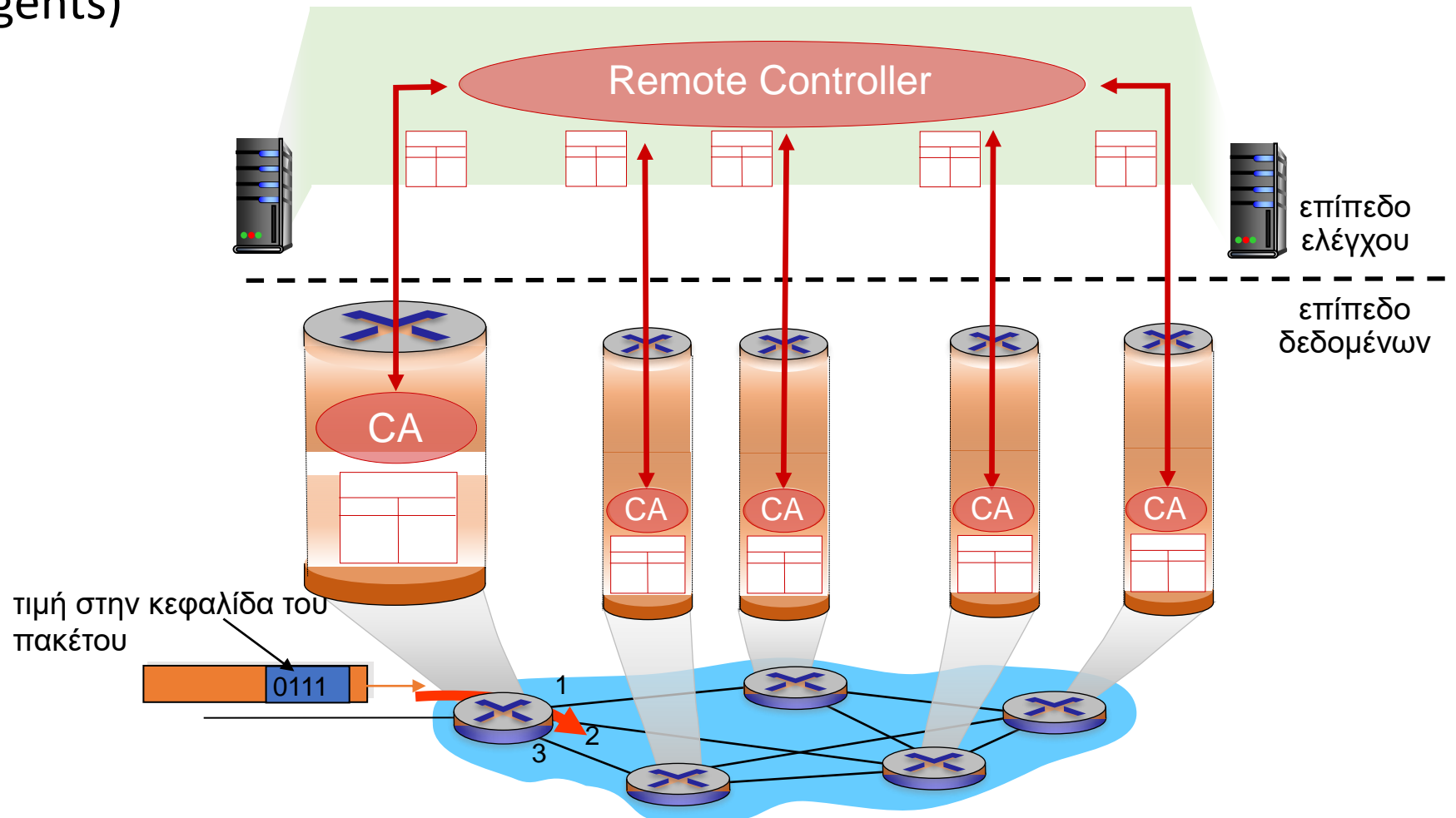
# Επίπεδο ελέγχου ανά δρομολογητή

Κάθε δρομολογητής τρέχει έναν αλγόριθμο δρομολόγησης



# Κεντριοποιημένο επίπεδο ελέγχου

Ξεχωριστός (τυπικά απομακρυσμένος) ελεγκτής αλληλοεπιδρά με κάθε δρομολογητή (control agents)



# Software defined networking (SDN)

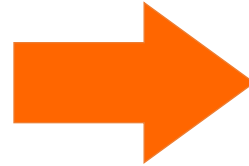
*Γιατί* να έχουμε ένα *λογικά κεντριοποιημένο* επίπεδο ελέγχου?

- ευκολότερη διαχείριση: αποφυγή κακών/εσφαλμένων ρυθμίσεων, μεγαλύτερη ευελιξία στις ροές της κίνησης
- γενικευμένη προώθηση με βάση πίνακες (OpenFlow API) που επιτρέπει τον «προγραμματισμό» των δρομολογητών
  - ευκολότερος ο κεντριοποιημένος «προγραμματισμός» δικτύου: κεντρικός υπολογισμός πινάκων και κατανομή
  - δυσκολότερος ο κατανεμημένος «προγραμματισμός» δικτύου: υπολογισμός πινάκων ως αποτέλεσμα ενός κατανεμημένου αλγορίθμου που εφαρμόζεται σε καθένα δρομολογητή
- ανοικτή εφαρμογή (όχι ιδιοταγής)

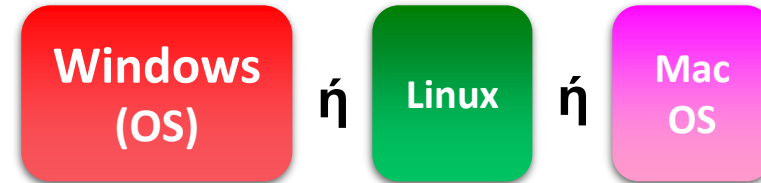
# Αναλογία με τις εφαρμογές στους Η/Υ



Κάθετη υλοποίηση  
Κλειστό, ιδιοταγές  
Αργή εξέλιξη και ανάπτυξη  
Μικρός ανταγωνισμός



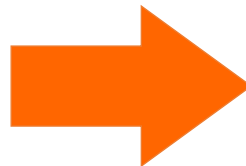
— Ανοικτή Διεπαφή —



— Ανοικτή Διεπαφή —

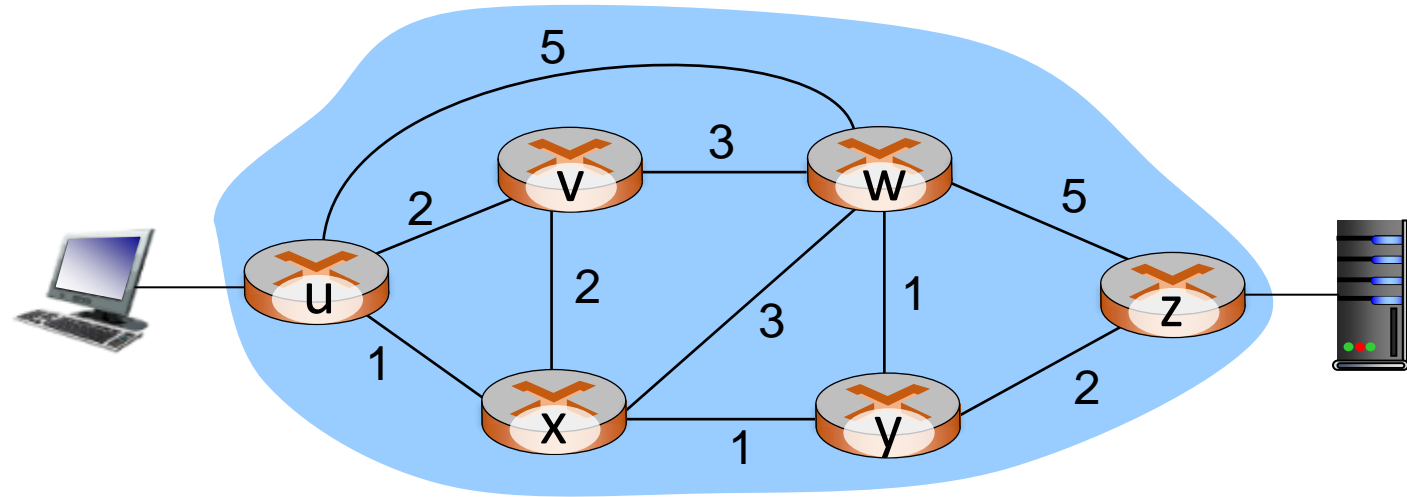


Μικροεπεξεργαστής



Οριζόντια υλοποίηση  
Ανοικτές διεπαφές  
Γρήγορη ανάπτυξη  
Τεράστια βιομηχανία

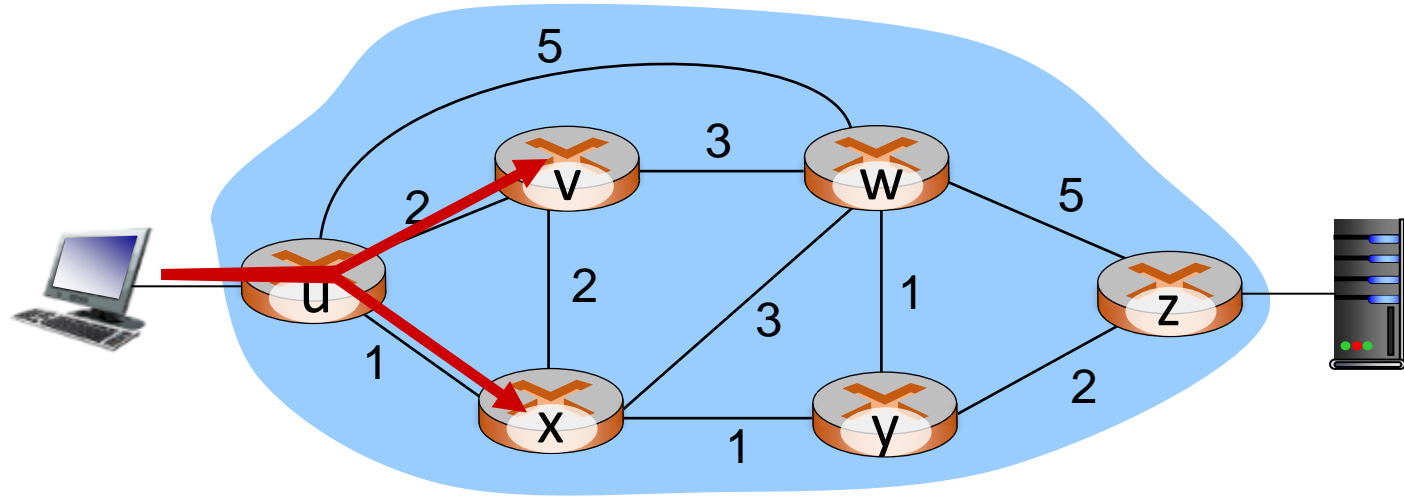
# Έλεγχος κίνησης δεδομένων: δύσκολη η παραδοσιακή δρομολόγηση



E: αν ο διαχειριστής ήθελε η κίνηση ανάμεσα στο u και z να ρέει πάνω στο μονοπάτι  $u \rightarrow v \rightarrow w \rightarrow z$ , και x με z να ρέει στο μονοπάτι  $x \rightarrow w \rightarrow y \rightarrow z$ ?

A: θα απαιτούνταν ο ορισμός βαρών στις ζεύξεις ώστε να τις συμπεριλάβει ένας αλγόριθμος δρομολόγησης υπόψη του!

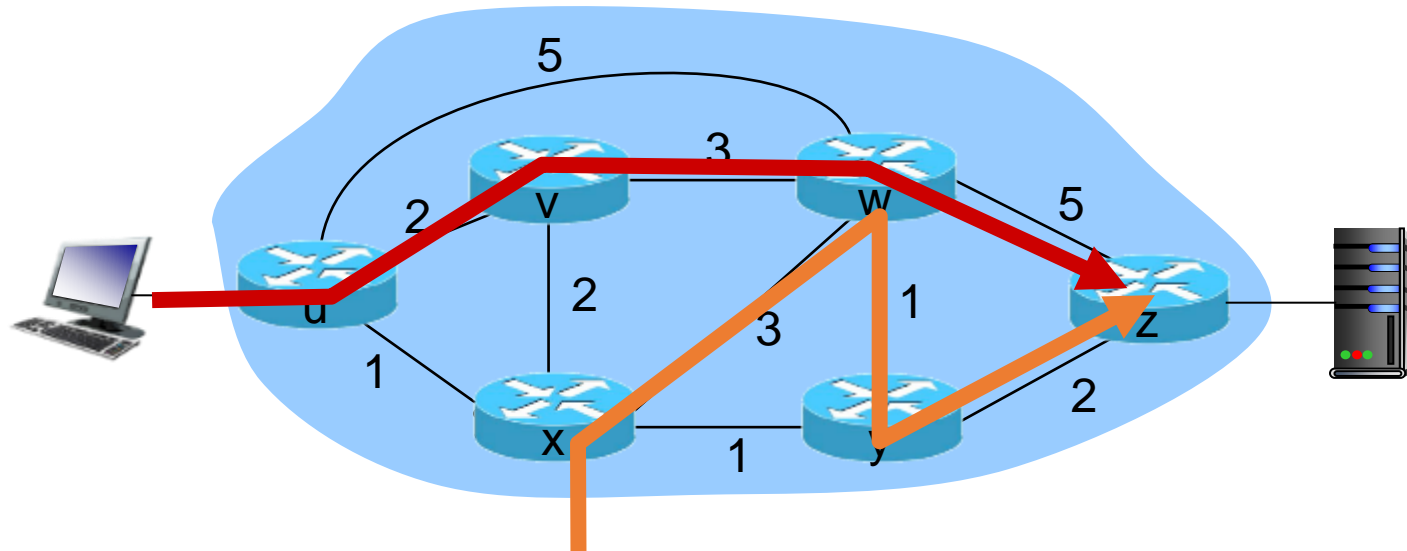
# Έλεγχος κίνησης δεδομένων: δύσκολη η παραδοσιακή δρομολόγηση



E: κι αν ο διαχειριστής ήθελε να χωρίσει για λόγους διαμοιρασμού του φορτίου την κίνηση ανάμεσα στον u και z μεταξύ των μονοπατιών  $u-v-w-z$  **και**  $u-x-z$ ;

A: δεν μπορεί να το κάνει (ή θα χρειαζόταν έναν νέο αλγόριθμο δρομολόγησης)

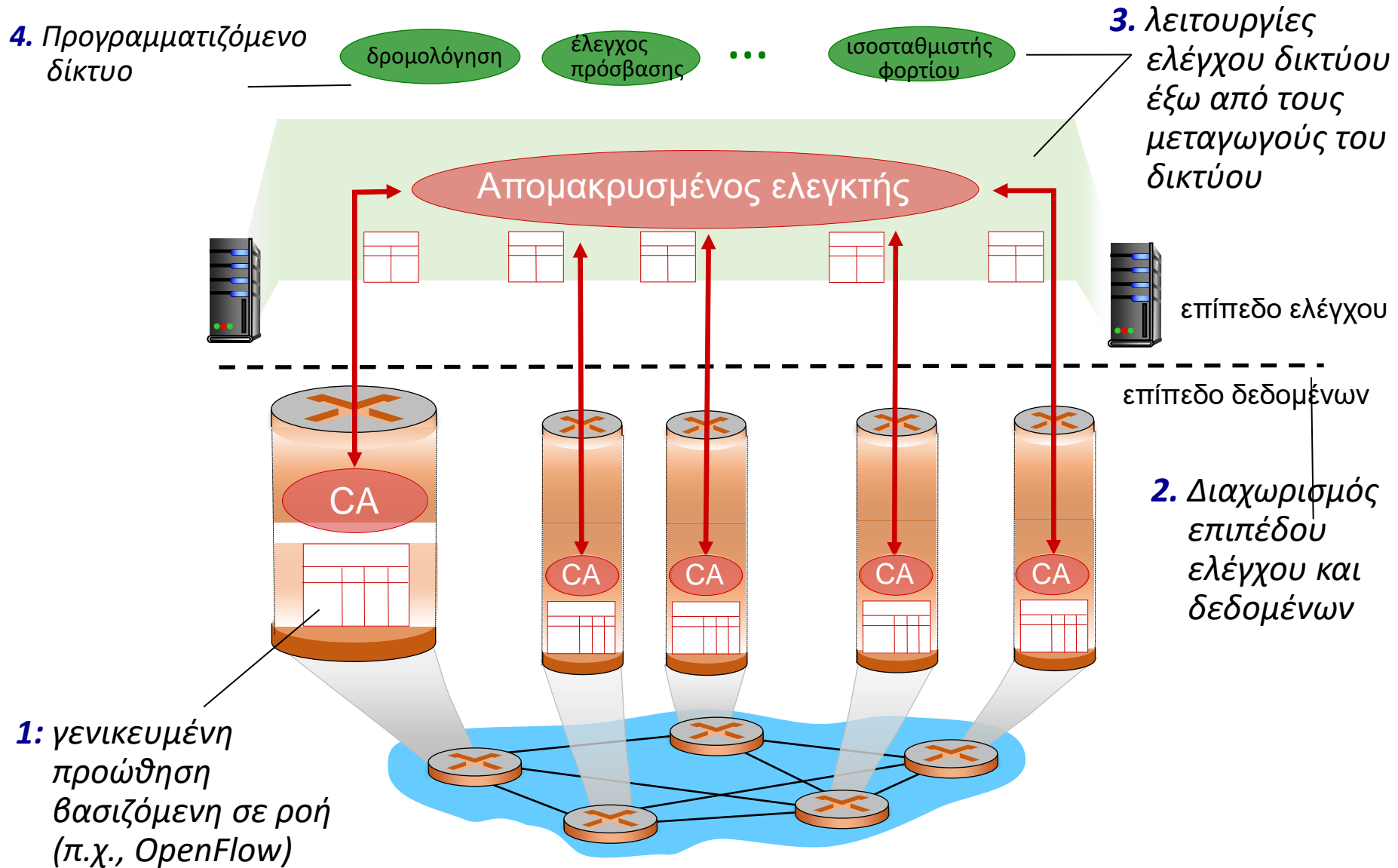
# Έλεγχος κίνησης δεδομένων: δύσκολη η παραδοσιακή δρομολόγηση



E: κι αν ήθελε να δρομολογήσει διαφορετικά την κόκκινη και πορτοκαλί ροή?

A: δεν θα μπορούσε (με την προώθηση με βάση τον προορισμό ή με τη λογική των αλγορίθμων κατάστασης ζεύξης ή διανύσματος απόστασης)

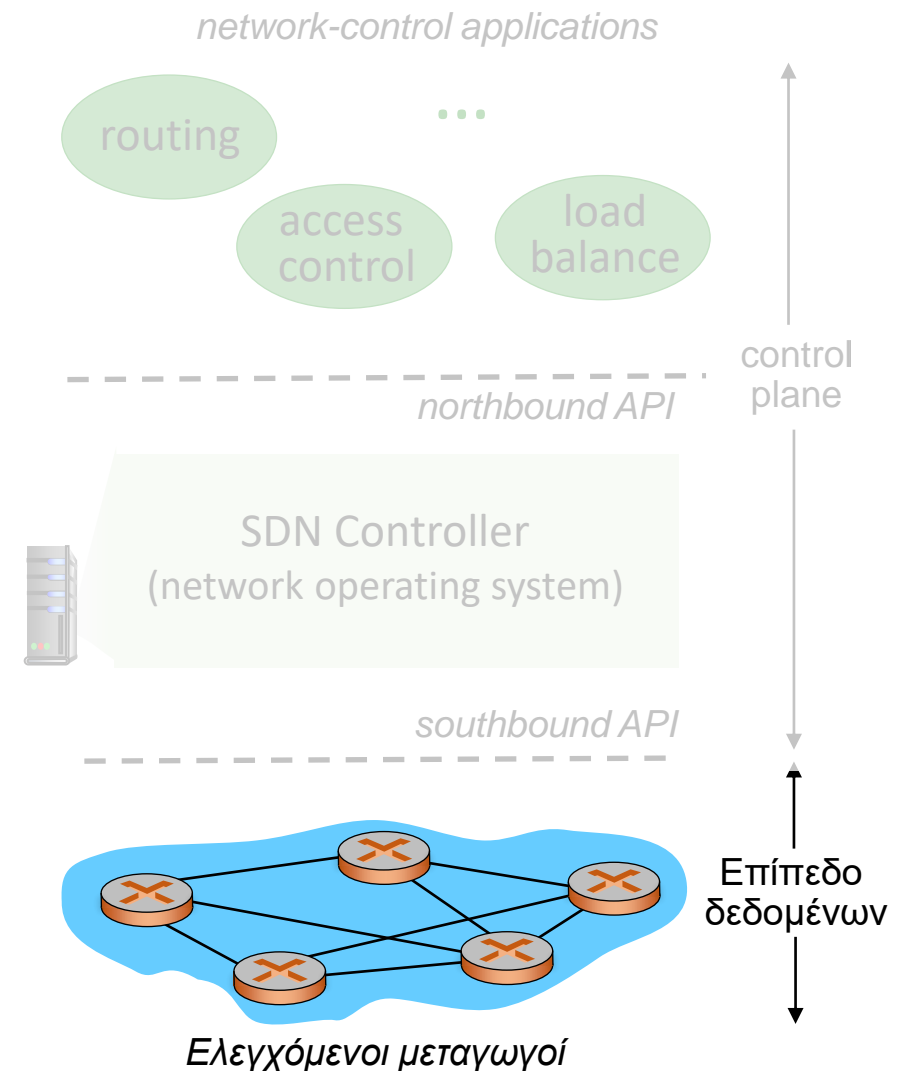
# Software defined networking (SDN)





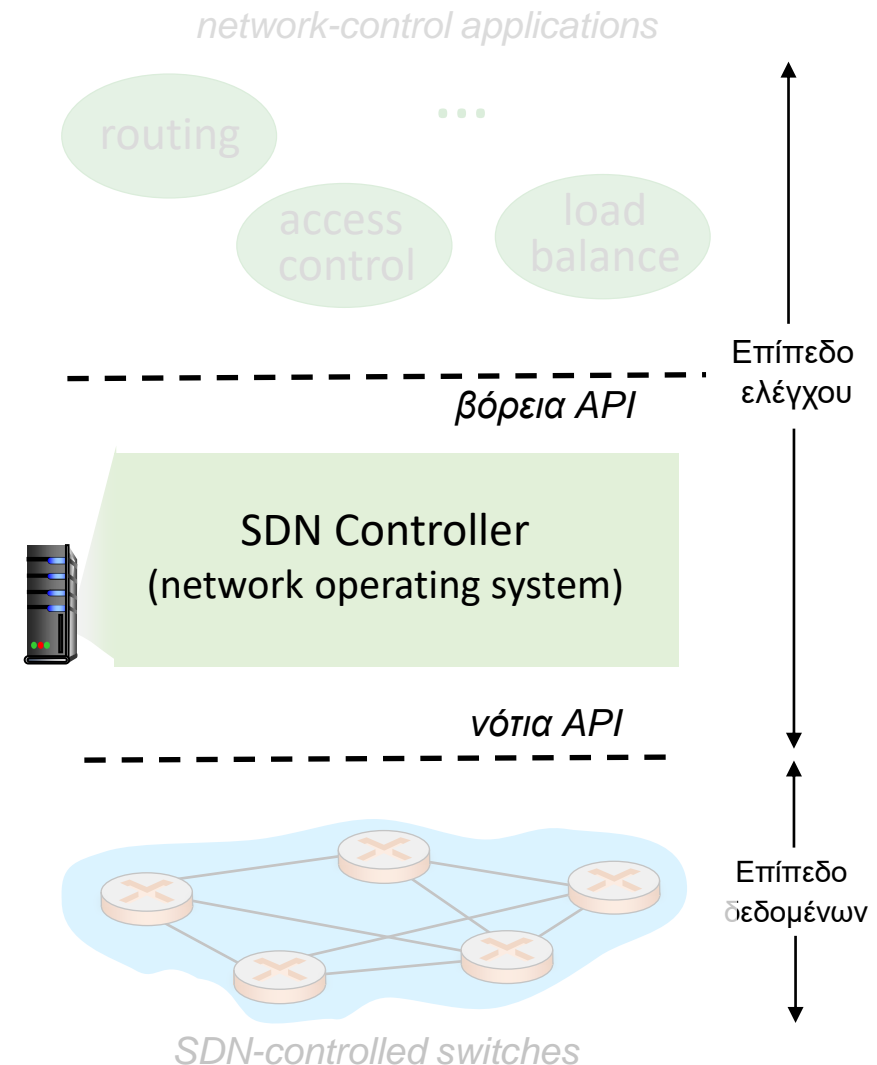
# SDN προσέγγιση: μεταγωγί επιπέδου δεδομένων

- γρήγορα, απλά κουτιά υλοποιώντας τη μεταγωγή με τη βοήθεια κυκλωμάτων (hardware)
- ο πίνακας ροής υπολογίζεται και εγκαθίσταται από τον ελεγκτή
- ένα API για τον έλεγχο των μεταγωγών (π.χ., OpenFlow)
  - ορίζει τί είναι ελέγξιμο και τί όχι
- ένα πρωτόκολλο για την επικοινωνία με τον ελεγκτή (π.χ., OpenFlow)



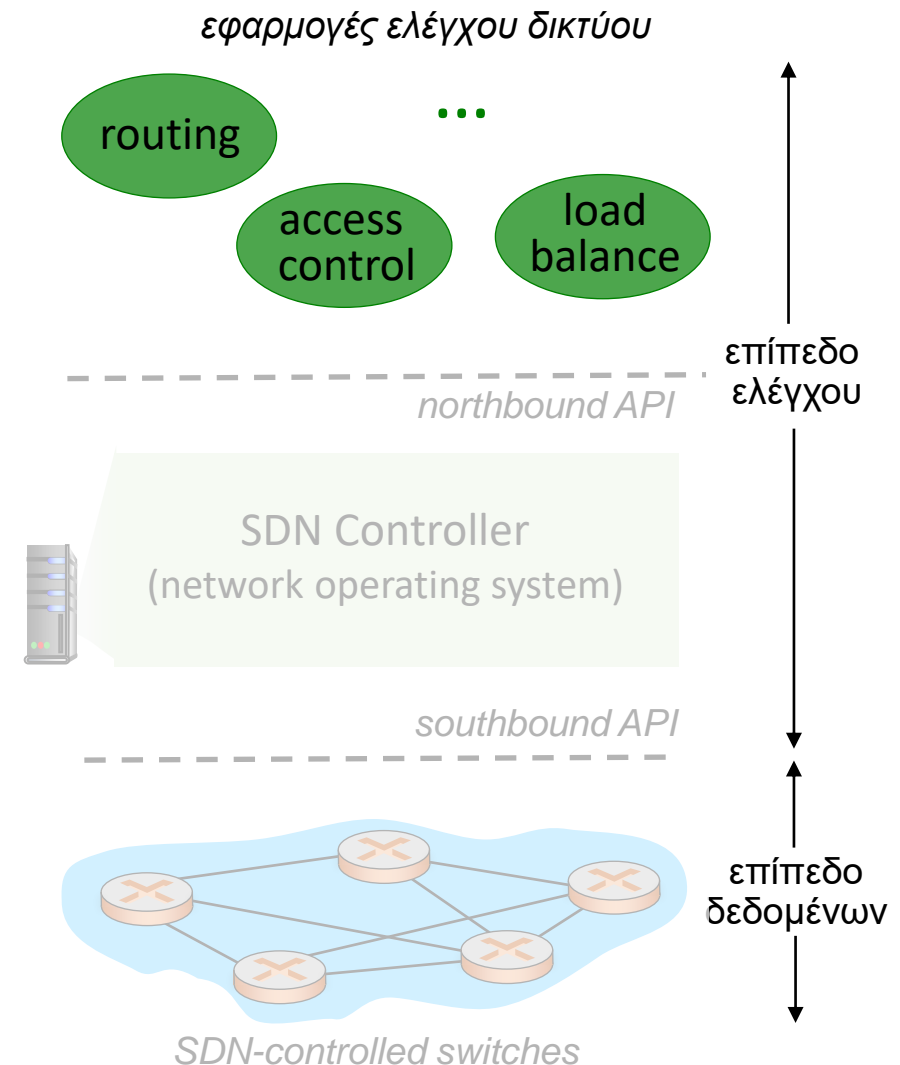
# SDN προσέγγιση: ελεγκτής

- διατηρεί πληροφορία κατάστασης για το δίκτυο
- αλληλεπιδρά με τις εφαρμογές ελέγχου από «πάνω» του (μέσω της βόρειας API)
- αλληλεπιδρά με τους μεταγωγούς δικτύου από «κάτω» του (μέσω της νότιας API)
- εφαρμογή ως κατανεμημένο σύστημα για απόδοση, κλιμάκωση, ευρωστία και ανθεκτικότητα σε σφάλματα

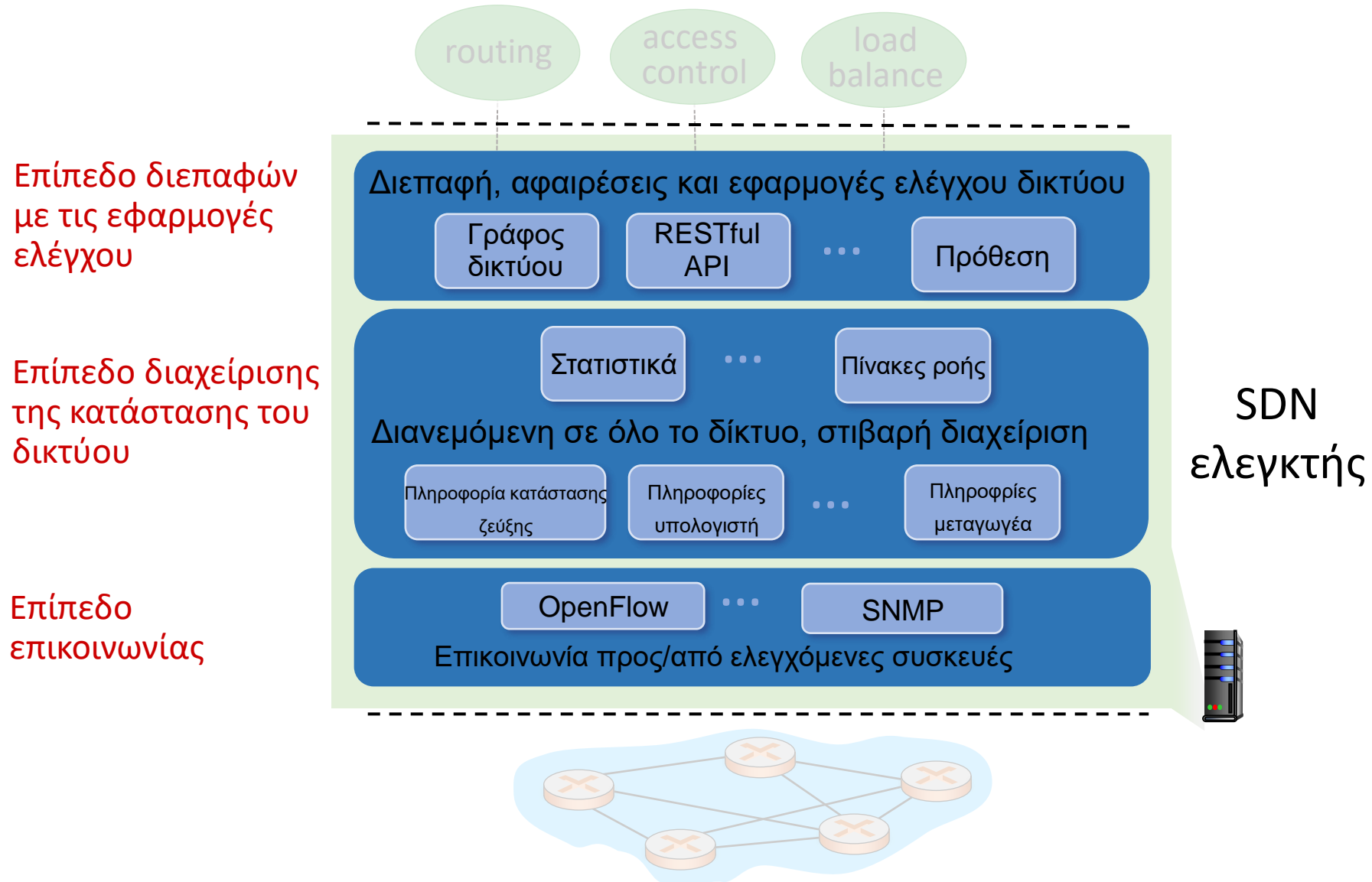


# SDN προσέγγιση: εφαρμογές ελέγχου

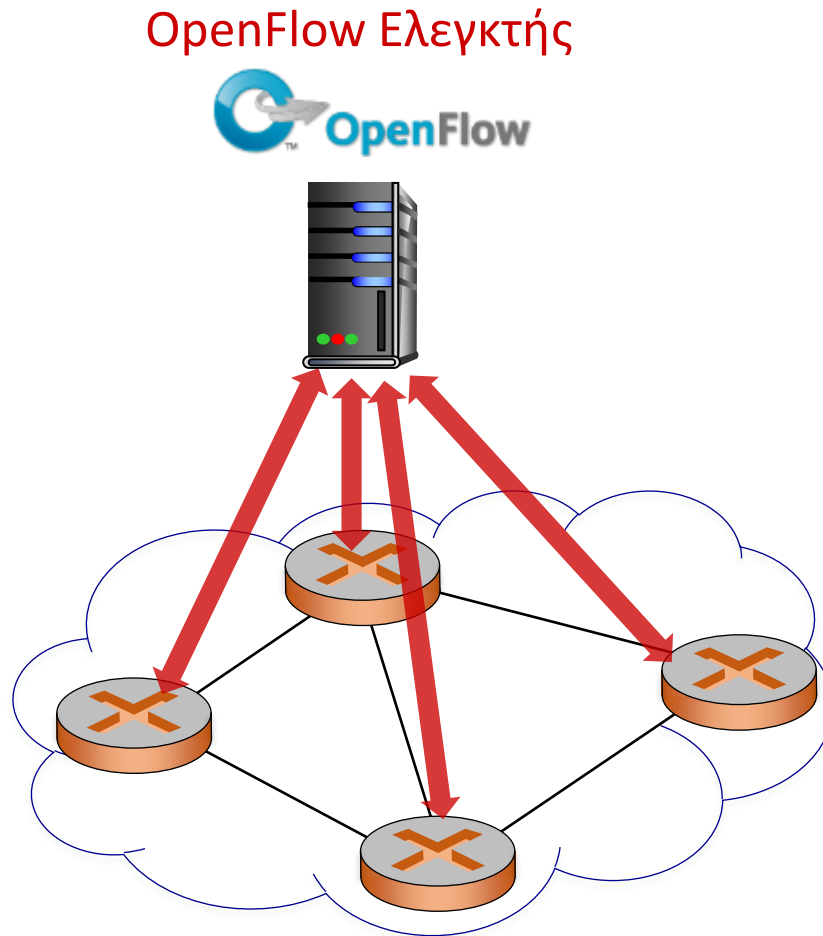
- το «μυαλό» του ελέγχου: εφαρμόζει τις λειτουργίες ελέγχου χρησιμοποιώντας υπηρεσίες χαμηλότερου επιπέδου, το API παρέχεται από τον ελεγκτή
- μπορεί να παρασχεθεί από τρίτους, διαφορετικούς από τους προμηθευτές του εξοπλισμού μεταγωγής ή του ελεγκτή



# Συστατικά στοιχεία του SDN ελεγκτή



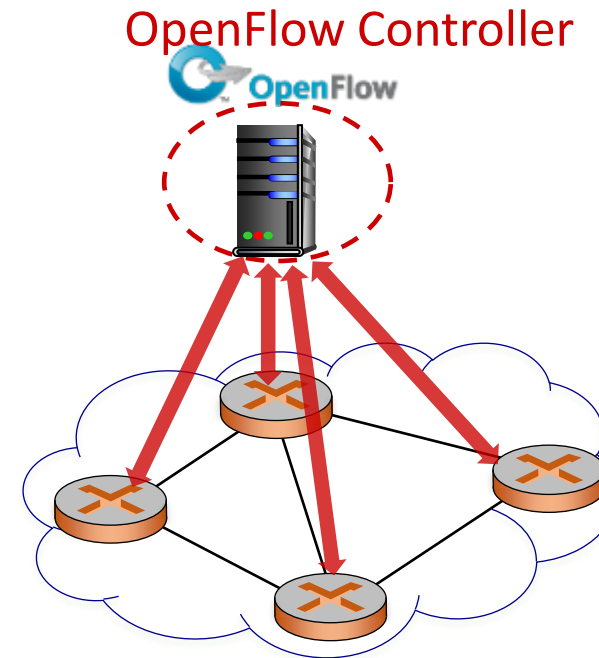
# OpenFlow πρωτόκολλο



- λειτουργεί ανάμεσα σε ελεγκτή, μεταγωγό
- το TCP χρησιμοποιείται για την ανταλλαγή μηνυμάτων (port:6653)
  - προαιρετική κρυπτογράφηση
- τρεις κλάσεις μηνυμάτων OpenFlow :
  - ελεγκτής σε μεταγωγό
  - ασύγχρονα (μεταγωγός σε ελεγκτή)
  - συμμετρικά (διάφορα)

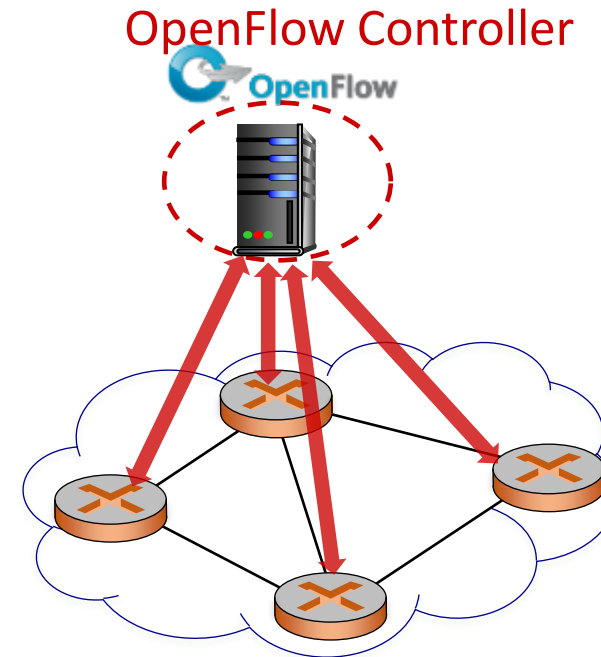
# OpenFlow: μηνύματα ελεγκτή σε μεταγωγό

- **κατάσταση ανάγνωσης:** συλλέγει στατιστικά στοιχεία και τιμές μετρητών από τον πίνακα ροής και τις θύρες του μεταγωγού
- **παραμετροποίηση:** υποβάλλει ερωτήματα και θέτει παραμέτρους
- **τροποποίηση κατάστασης:** προσθέτει, διαγράφει, τροποποιεί καταχωρίσεις στον πίνακα ροής
- **αποστολή πακέτου:** στέλνει ένα πακέτο σε μια καθορισμένη θύρα στον μεταγωγό

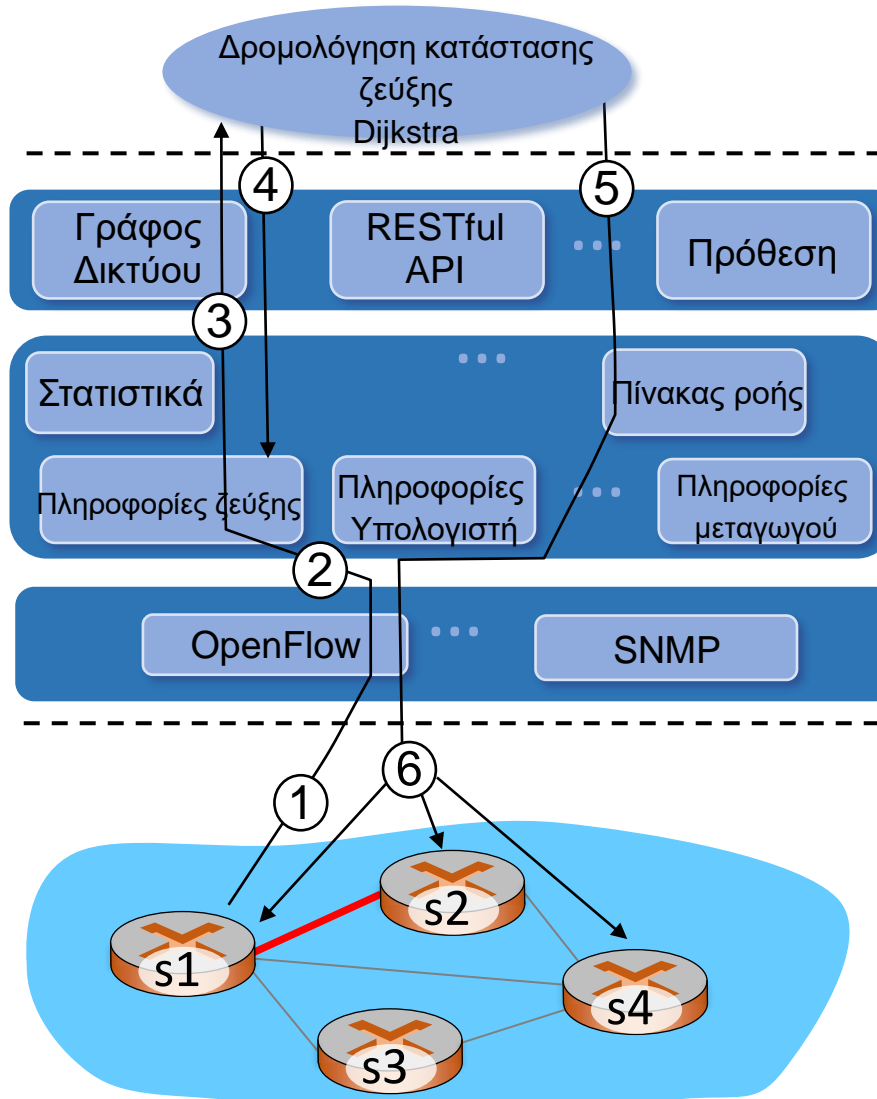


# OpenFlow: μηνύματα μεταγωγού προς ελεγκτή

- **εισερχόμενο πακέτο:** μεταφορά πακέτου που δεν ταιριάζει σε καταχώριση του πίνακα ροής
- **αφαίρεση ροής:** διαγραφή ροής πίνακα στο μεταγωγό
- **κατάσταση θύρας:** πληροφόρηση λόγω αλλαγής στην κατάσταση θύρας.



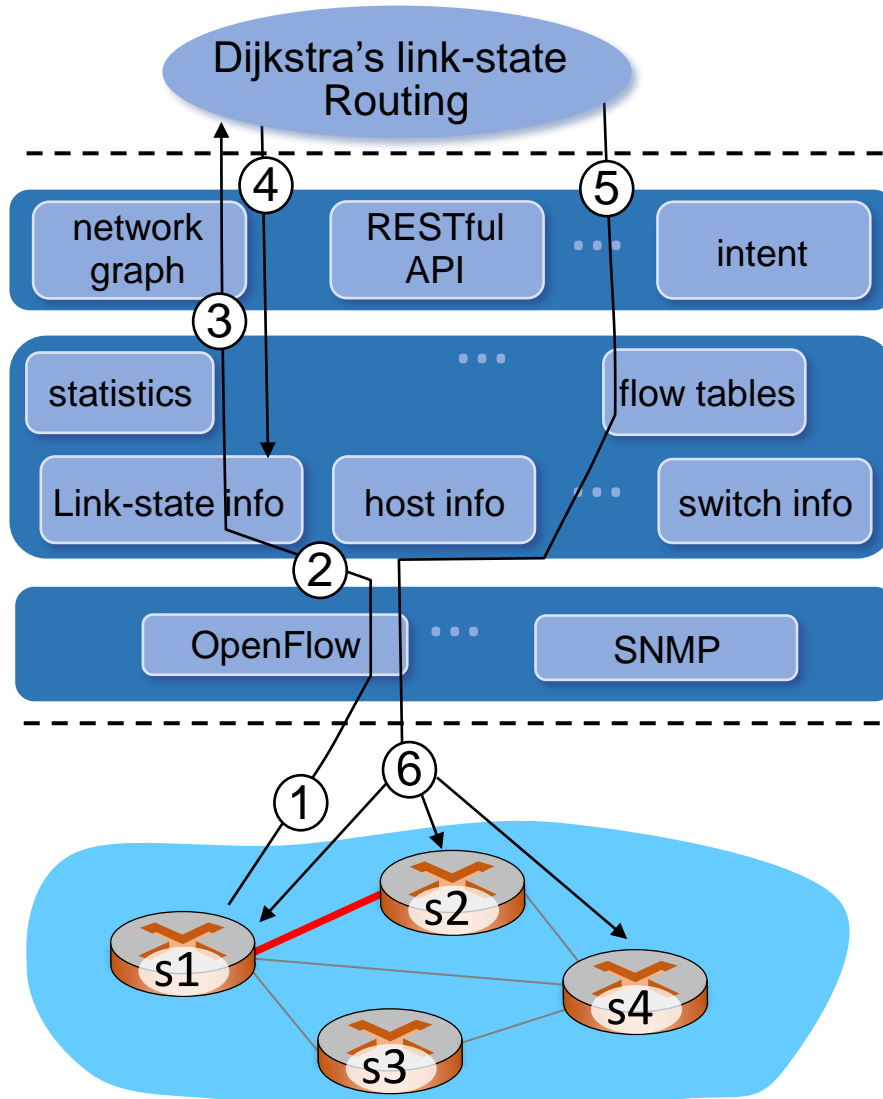
# SDN: αλληλεπίδραση επιπέδων ελέγχου και δεδομένων



- ① Ο S1 ενημερώνει τον ελεγκτή επειδή διαπίστωσε αστοχία στη ζεύξη του με S2
- ② Ο ελεγκτής ενημερώνει τον διαχειριστή κατάστασης ζεύξης
- ③ Η εφαρμογή ελέγχου δικτύου που υλοποιεί την δρομολόγηση κατάστασης ζεύξης ειδοποιείται για την αλλαγή.
- ④ Η εφαρμογή δρομολόγησης κατάστασης ζεύξης ελέγχει τις πληροφορίες και υπολογίζει νέες διαδρομές.



# SDN: αλληλεπίδραση επιπέδων ελέγχου και δεδομένων



- ⑤ Η εφαρμογή δρομολόγησης αλληλεπιδρά με τον διαχειριστή πίνακα ροής που καθορίζει τους πίνακες ροής που θα ενημερωθούν
- ⑥ Ο διαχειριστής πίνακα ροής χρησιμοποιεί το OpenFlow για να εγκαταστήσει νέους πίνακες στους μεταγωγούς που χρειάζονται ενημέρωση

# Επίπεδο δικτύου: επίπεδο ελέγχου

- εισαγωγή
- αλγόριθμοι δρομολόγησης
  - κατάσταση ζεύξης
  - διάνυσμα απόστασης
- δρομολόγηση ενδο-αυτόνομου συστήματος (intra-ISP routing): OSPF
- δρομολόγηση ανάμεσα σε ISPs: BGP
- επίπεδο ελέγχου SDN
- **Internet Control Message Protocol**



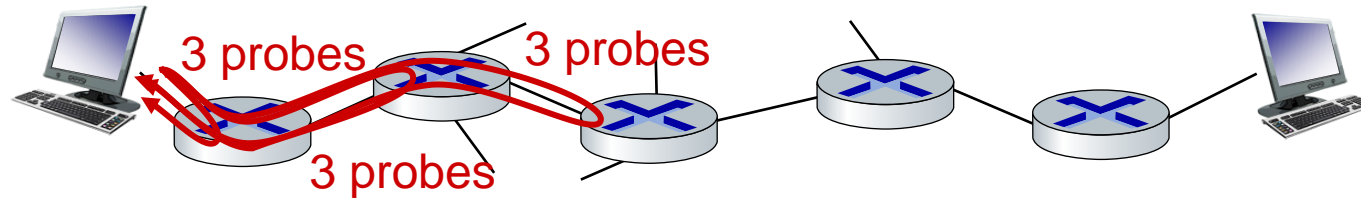
- διαχείριση δικτύου
  - SNMP
  - NETCONF/YANG

# ICMP: Internet Control Message Protocol

- Χρησιμοποιείται από τερματικά και δρομολογητές να επικοινωνήσουν πληροφορία επιπέδου δικτύου
  - αναφορά σφαλμάτων: μη προσβάσιμο τερματικό, δίκτυο, θύρα, πρωτόκολλο
  - αίτηση/απάντηση σε μηνύματα ηχούς (ping)
- Επίπεδο δικτύου «πάνω» από το IP:
  - τα μηνύματα ICMP μεταφέρονται σε IP δεδομενογράμματα
- *ICMP μήνυμα*: τύπος, κωδικός αριθμός μαζί με τα 8 πρώτα bytes του δεδομενογράμματος που παράγει το σφάλμα

<u>Type</u>	<u>Code</u>	<u>description</u>
0	0	echo reply (ping)
3	0	dest. network unreachable
3	1	dest host unreachable
3	2	dest protocol unreachable
3	3	dest port unreachable
3	6	dest network unknown
3	7	dest host unknown
4	0	source quench (congestion control - not used)
8	0	echo request (ping)
9	0	route advertisement
10	0	router discovery
11	0	TTL expired
12	0	bad IP header

# Traceroute



- η πηγή στέλνει ομάδες των 3 τμημάτων UDP στον προορισμό
    - η 1<sup>η</sup> τριάδα έχει TTL =1, η 2<sup>η</sup> έχει TTL=2, κλπ.
  - όταν ένα δεδομένογραμμα της N-στής ομάδας φθάνει στον N-στό δρομολογητή:
    - ο δρομολογητής απορρίπτει το δεδομένογραμμα και στέλνει μήνυμα ICMP (type 11, code 0)
    - το μήνυμα ICMP πιθανά περιλαμβάνει το όνομα και την IP διεύθυνση του δρομολογητή
  - όταν φθάνει το ICMP μήνυμα στην πηγή: καταγράφονται τα RTTs
- κριτήρια τερματισμού:**
- το UDP τμήμα τελικά φθάνει στον προορισμό
  - ο προορισμός επιστρέφει μήνυμα ICMP “port unreachable” (type 3, code 3)
  - η πηγή σταματάει

# Επίπεδο δικτύου: επίπεδο ελέγχου - Τέλος

- εισαγωγή
- αλγόριθμοι δρομολόγησης
  - κατάσταση ζεύξης
  - διάνυσμα απόστασης
- δρομολόγηση ενδο-αυτόνομου συστήματος (intra-ISP routing): OSPF
- δρομολόγηση ανάμεσα σε ISPs: BGP
- επίπεδο ελέγχου SDN
- Internet Control Message Protocol



- διαχείριση δικτύου
  - SNMP
  - NETCONF/YANG