

# Introduction to Routing in Internet

Internet basics

IPv4 and ICMP

Internet Addressing

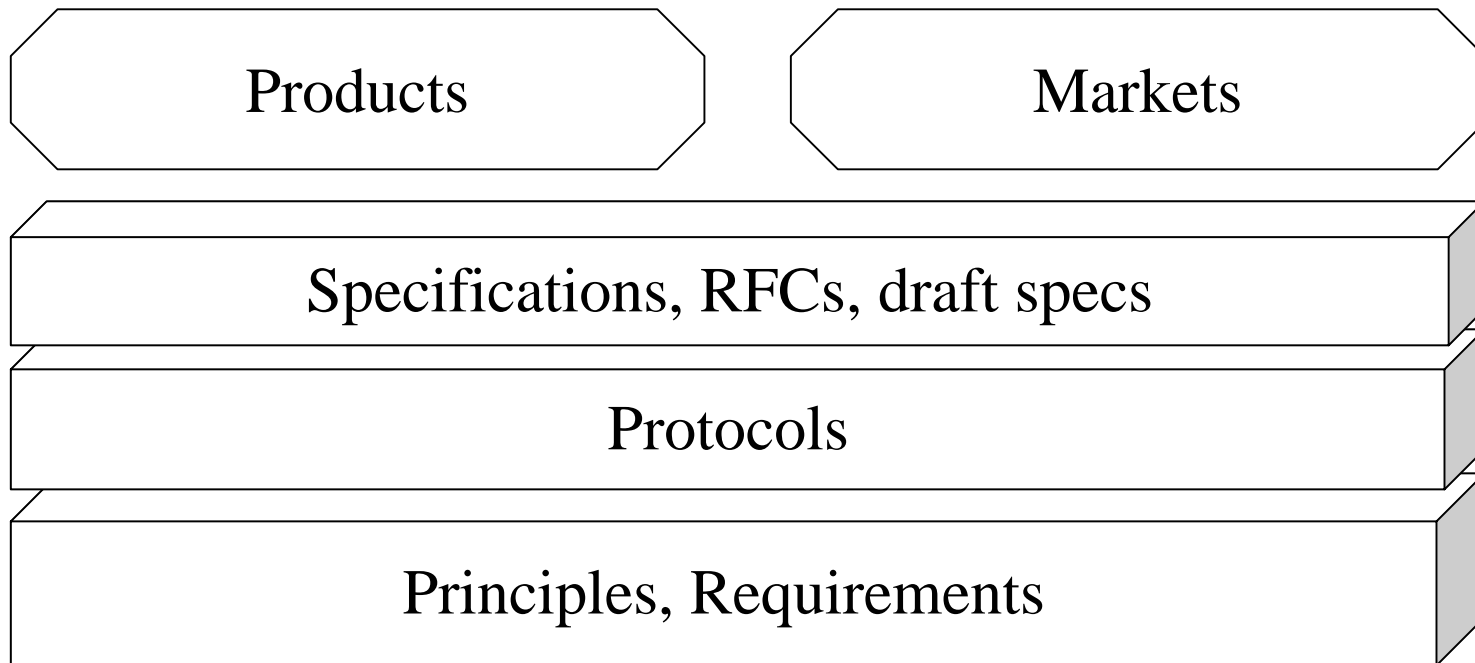
ARP - Address Resolution Protocol

Routing Information (Distance Vector ) Protocol Principles

# Internetin reititys perustuu reititysprotokolliin, joilla kerätään lähtötiedot

- Internetiin ei liity off-line reitityssuunnittelua, off-line tehdään ainoastaan mitoitus
- Itse reititys toimii kokonaan automaattisesti.
- Reititys jaetaan sisäiseen ja ulkoiseen.

# Levels of analysis - we deal with principles, protocols and specifications



# Internet Architecture Principles

- Does it exist?
- End-to-end principle *by Dave Clark*
  - all e.g. error and flow control in end stations
  - trusting networks is redundant
  - more reliable transport works for IP
  - no state information/connection in the network
  - same as distributed systems

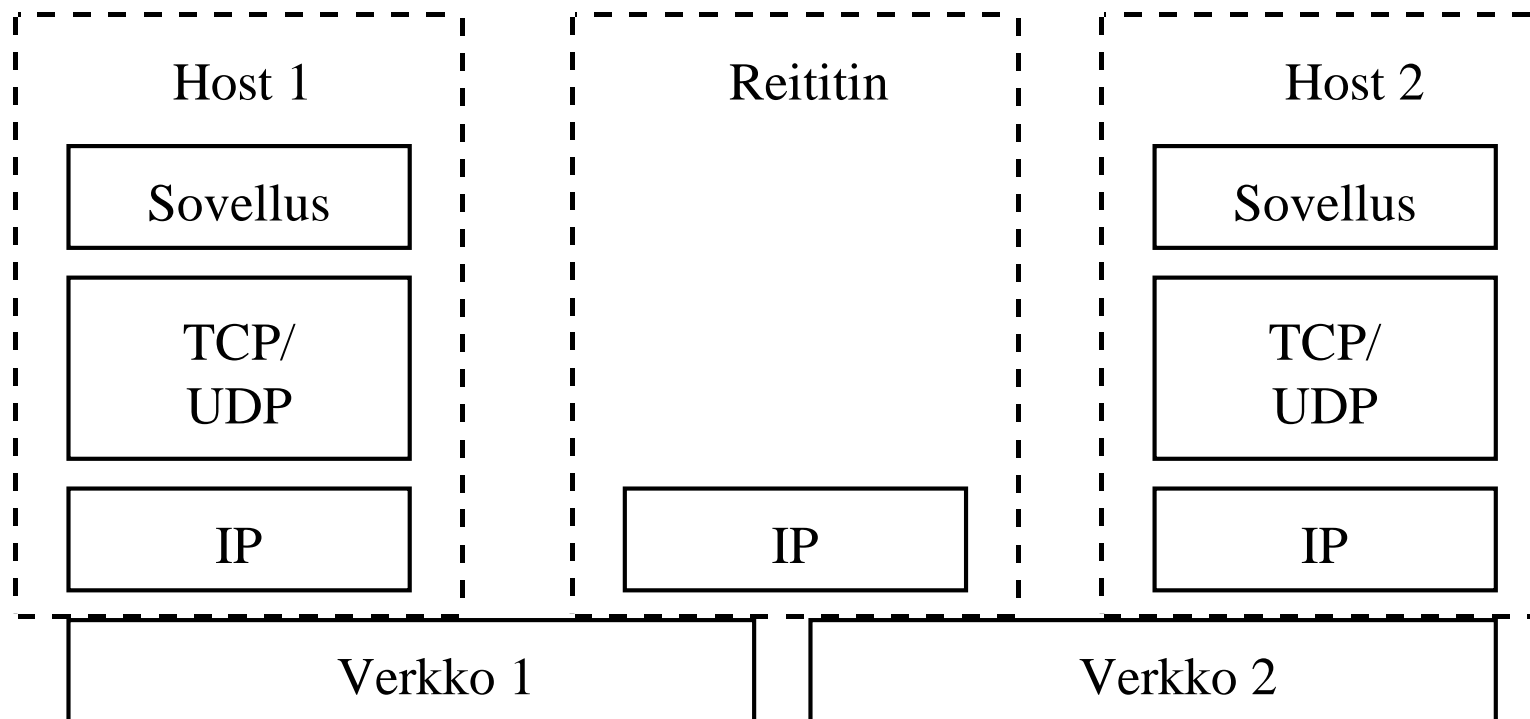
# Principles - IP over everything

- Interconnection based on IP overlay over all kinds of networks *by Winston Cerf*
  - framing or encapsulation
  - address mapping  
for each transport technology
- interconnection based on translation:
  - e.g. signalling interworking - imperfect mapping

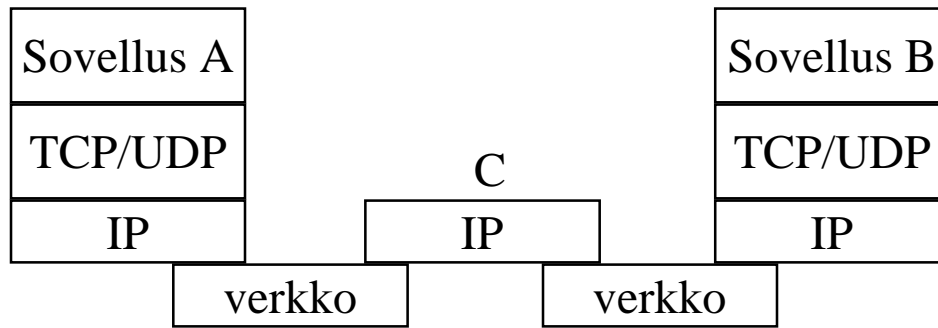
# Principles - Connectivity is its own reward

- Be liberal with what you receive, conservative with what you send
  - try to make your best to understand what you receive
  - maximum adherence to standard when sending
- snowballing effect keeps all interested in connectivity thus keeps adhering to standards

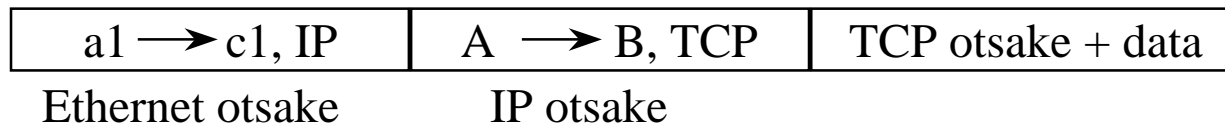
# Internet kerrosmalli - isäntäkoneet ja reitittimet



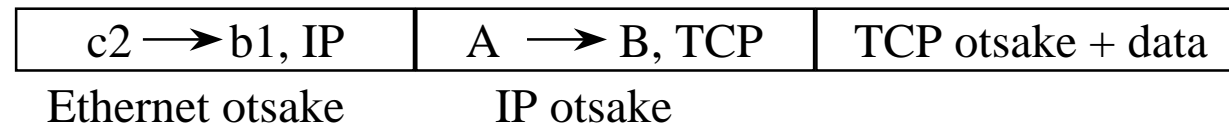
# Internet kerrosmalli - sanomien välitys



Paketointi:

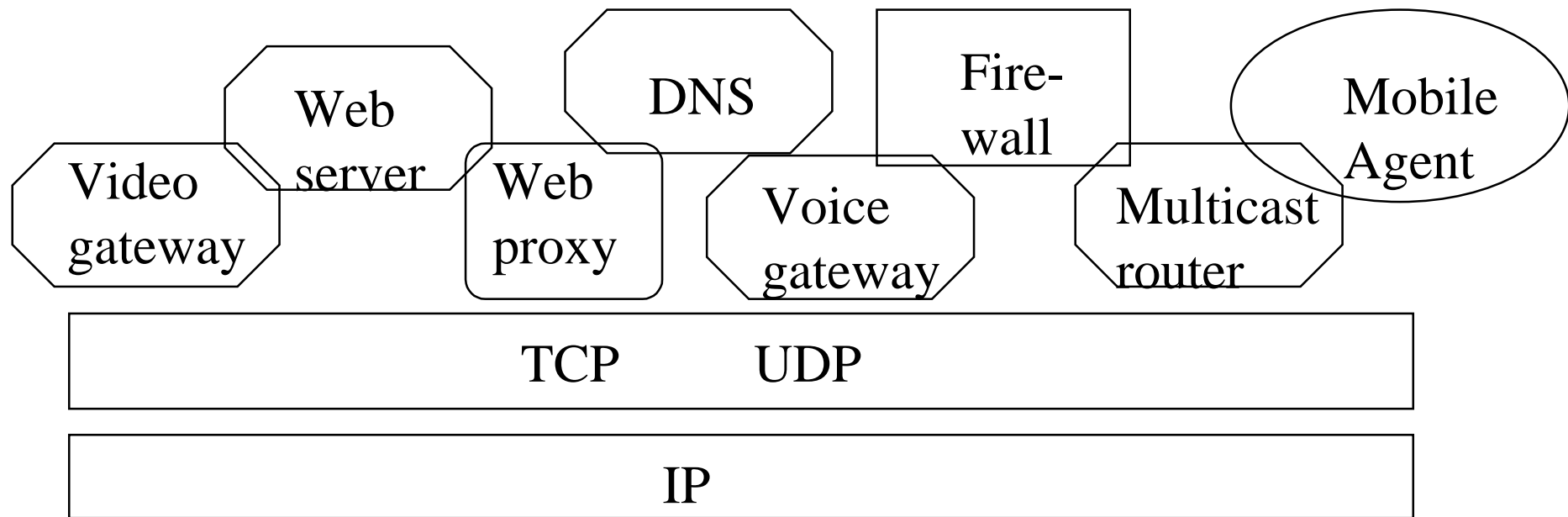


Paketointi:



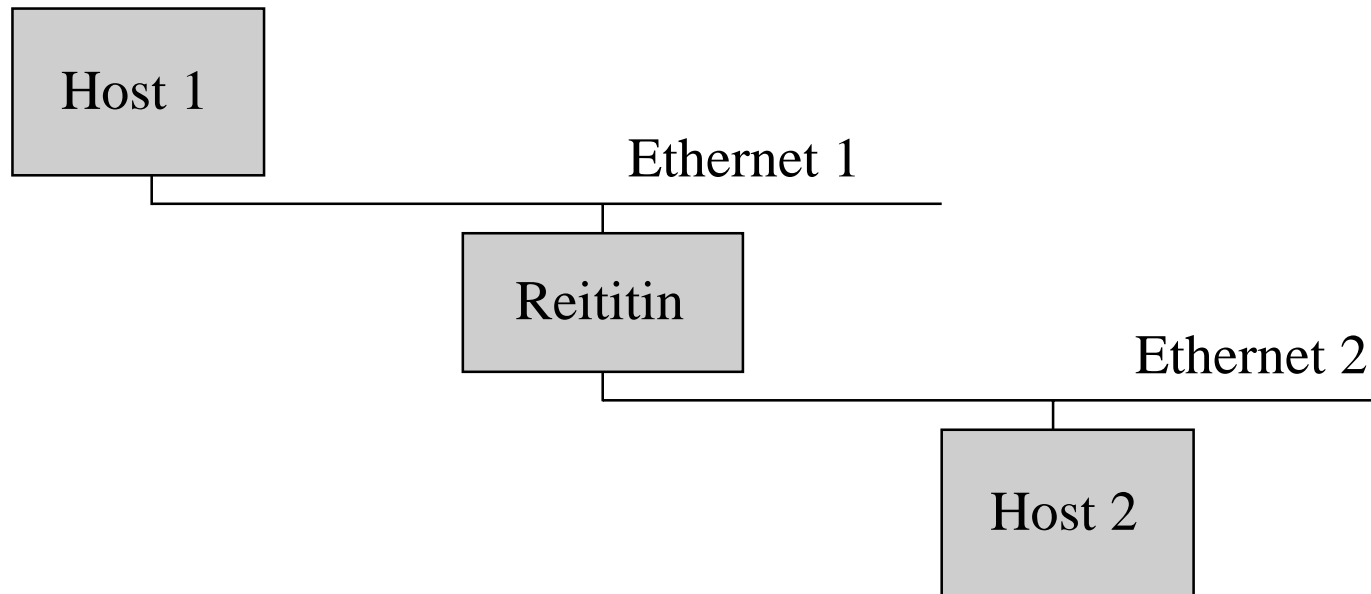


# Internet arkkitehtuuri sisältää nykyään joukon TCP/IP:n päällä olevia palvelutason komponentteja

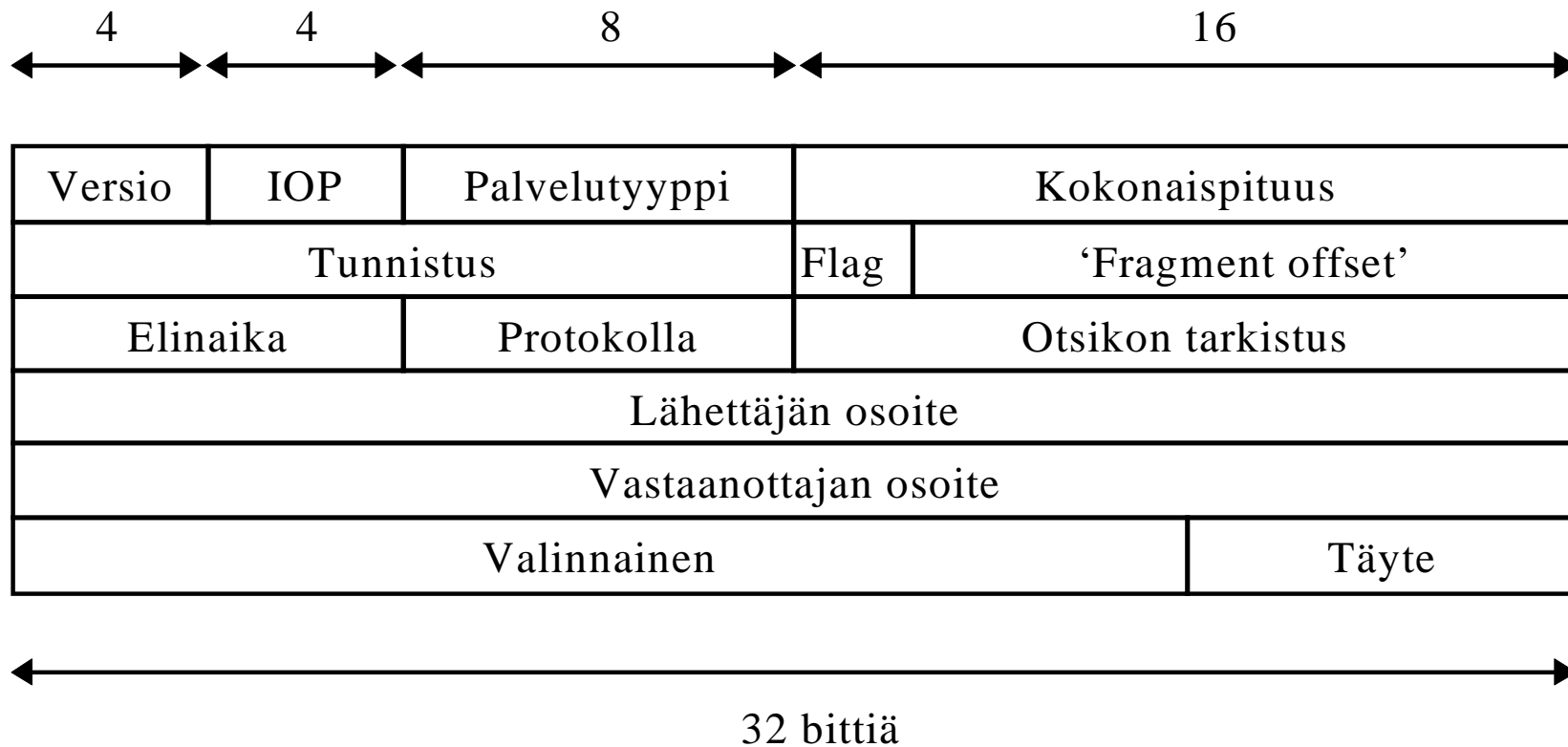


*Tässä kurssissa puututaan yo komponentteihin vain sikäli kuin ne liittyvät reititykseen.*

# Yhdistämällä Ethernet segmentit reitittimellä segmenttien liikenteet pystytään erottamaan



# IP paketin otsikko



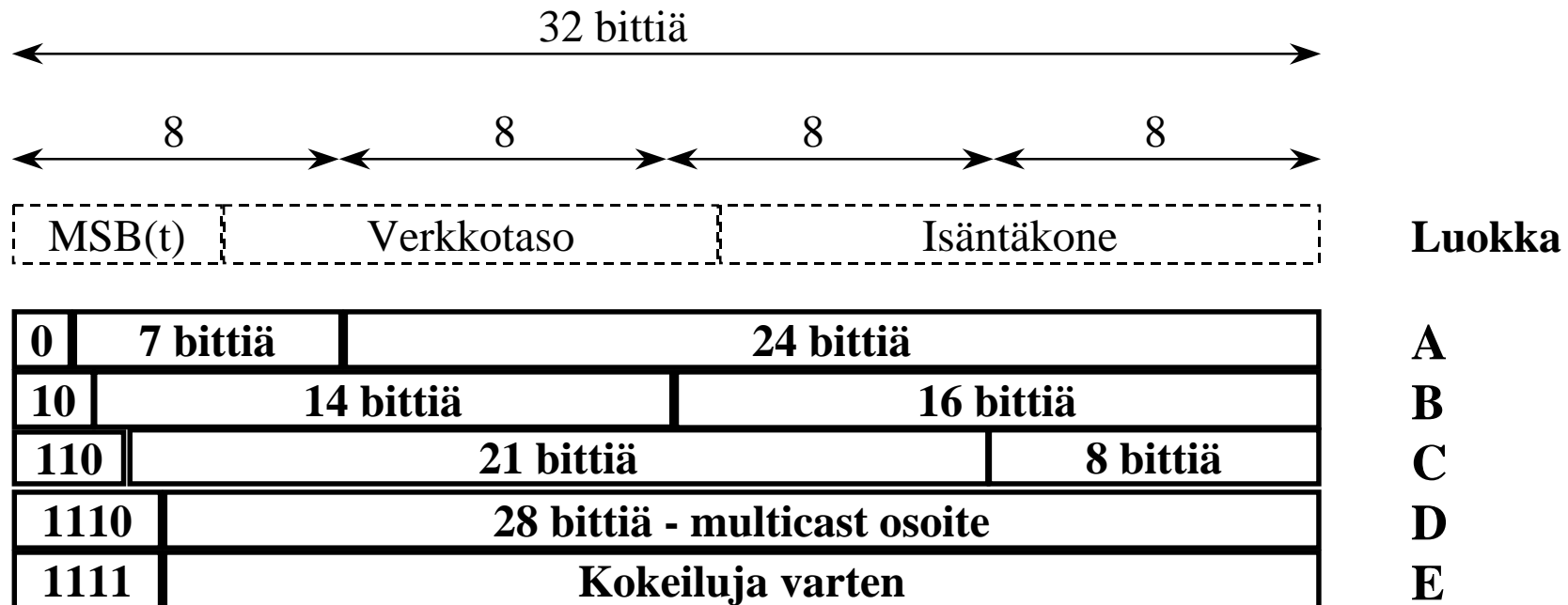
*Oletus: Lähettäjä tietää oman osoitteensa*

*jos ei --> itsekonfigurointi (RARP, BOOTP, DHCP - dynamic host conf. protocol)*

IP versio	IP version numero. Nykyinen versionumero on 4
IHL	Internet header's length. IP -otsikon pituus ilmaistuna 32 -bittisten sanojen määränä
Palvelutyyppe	Sisältää kaksi alikenttää, 3MSB:paketin prioriteettia + palvelutyyppe.
Paketin kokonaispituus	Koko datapaketin, IP -otsikko mukaanlukien kokonaispituus ilmaistuna oktettien määränä
Identifikaatio, liput ja Offset	Käytetään jaettaessa isoja datapaketteja pienemmiksi osiksi tiettyjä siirtotien osia varten.
TTL	Datapaketin elinikä. Luku, josta vähennetään verkon laatua kuvaava kokonaisluku kaikissa kohdatuissa reitittimissä. Paketin reitittäminen päättyy, kun TTL saa arvon 0.

Protokolla	Protokolla, jolla vastaanottavan Hostin tulee käsitellä datapaketti. Esim. TCP
Tarkistus-summa	Otsikon tarkistussumma, lasketaan 16 bittisenä yhden komplementtina.
Lähdeosoite	Paketin lähettäneen Hostin osoite
Kohdeosoite	Sen Hostin osoite, jolle paketti on lähetetty
Optiot	Käytetään erityisinformaation lähettämiseen. Yksittäise paketit voivat sisältää useita optiokenttiä

# IPv4 osoiteformaatit



*IP -osoite määrittelee rajapinnan (interface).*

*Aliverkkomaskilla erotetaan verkko-osa ja aliverkko+isäntäkoneosa.*

# IP paketin otsikon reitityksen kannalta tärkeät tiedot ovat kohdeosoite ja TTL

Versio	IHL	<i>Palvelun tyyppi</i>	Kokonaispituus	
Tunnistus			Liput	Viipaleen siirtymä
<i>TTL - elinaika</i>	Protokolla		Otsakkeen tarkistussumma	
<i>Lähdeosoite</i>				
<i>Kohdeosoite</i>				
Optiot				Täytebitit

*palvelun tyyppi = reitin valintakriteeri: D - viiveen minimointi tai*  
*T - siirtokapasiteetin maksimointi tai*  
*R - luotettavuuden maksimointi tai*  
*C - kustannusten minimointi*

*prioriteetti - suurin arvo --> otetaan jonosta ensin reititettäväksi*

*Optiot: mm. source routing - käytetään harvoin.*

# Internet address structure

network number	subnet	host
----------------	--------	------

Mask	Address	Net	Subnet	Host
0xFFFF0000	10.27.32.100	A: 10	27	32.100
0xFFFFFE00	136.27.33.100	B: 136.27	16 (32)	1.100
	136.27.34.141	136.27	17(34)	0.141
0xFFFFFC0	193.27.32.197	C: 193.27.32	3(192)	5

High order bits:

- 0 ..... 0 - 127. --> A-class
- 10.... 128. - 191. --> B-class
- 110...192. - 223. --> C-class

without right zeroes (and with right zeroes)



# ICMP - Internet Control Message Protocol

antaa lähettäjälle palautetta verkon toiminnasta

- Kaikkien hostien ja reitittimien täytyy tukea ICMP:tä.
- ICMP paketti lähetetään takaisinpäin, jos esim.  
vastaanottajaa ei tavoiteta  
reititin tuhoaa paketin  
TTL = 0
- Jos ICMP viesti tuhotaan, ei generoida uutta ICMP-viestiä,  
jottei tule “lumivyöryä“

# ICMP viestejä

- Tyyppi = 0 - Kaikuvastaus  
**3 - Kohde saavuttamaton**  
 4 - (Hiljennä tahtia)  
**5 - Uudelleenohjaus**  
 8 - Kaiku  
**9 - Reititin mainos**  
**10 - Reititin mainoksen pyyntö**  
**11 - Elin aika loppui**  
 12 - Parametrionglema  
 13 - Aikaleima  
 14 - Aikaleimavastaus  
 15 - Informaatiopyyntö  
 16 - Informaatiovastaus



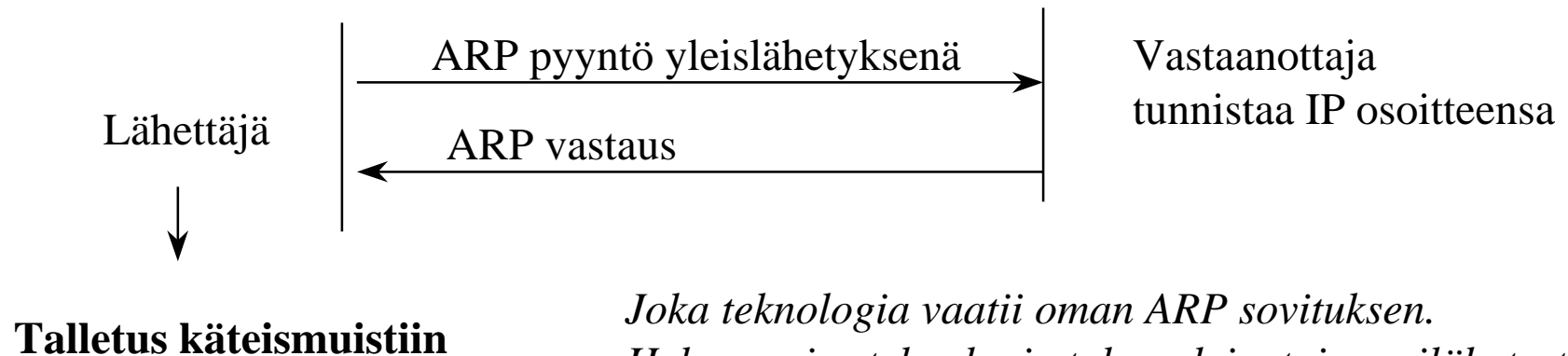
- Koodi =
- 0 - verkko saavuttamaton
  - 1 - isäntäkone saavuttamaton
  - 2 - protokolla saavuttamaton
  - 3 - portti saavuttamaton
  - 4 - sanoma paloitetava
  - 5 - lähdereitti viallinen

(4 - source quench has been dropped from recommendations)

# ARP - Address resolution protocol (RFC-826) sovittaa IP:n allaolevaan verkkoon.

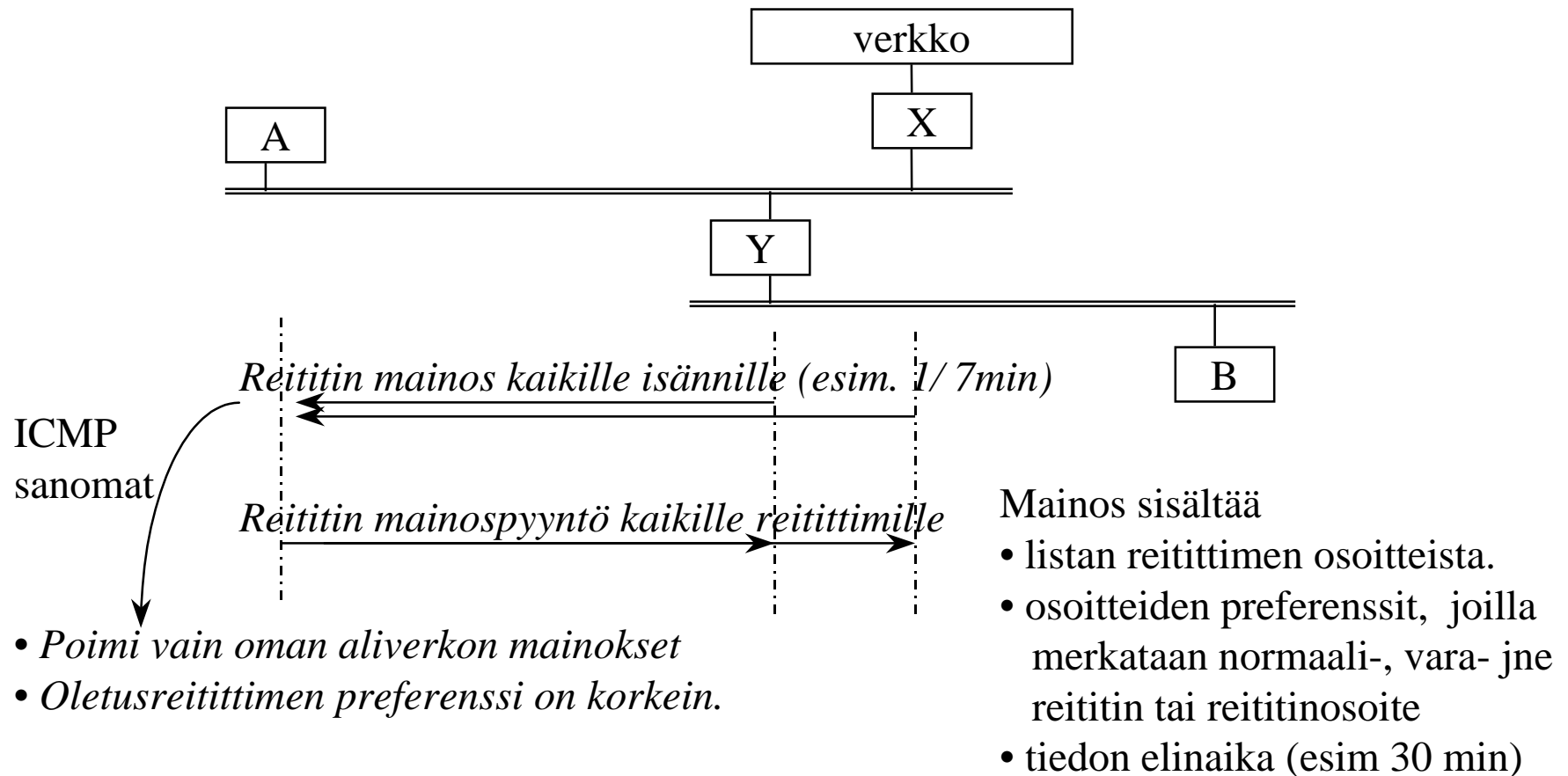
Lähettäjän toiminta:

1. Tutkitaan onko kohdeosoite omassa aliverkossa vertaamalla oman ja kohdeosoitteen maskattuja arvoja. Jos =, kohde on samassa aliverkossa, jos ei viesti pitää lähettää reitittimelle.
2. Etsitään kohteen mediaosoite

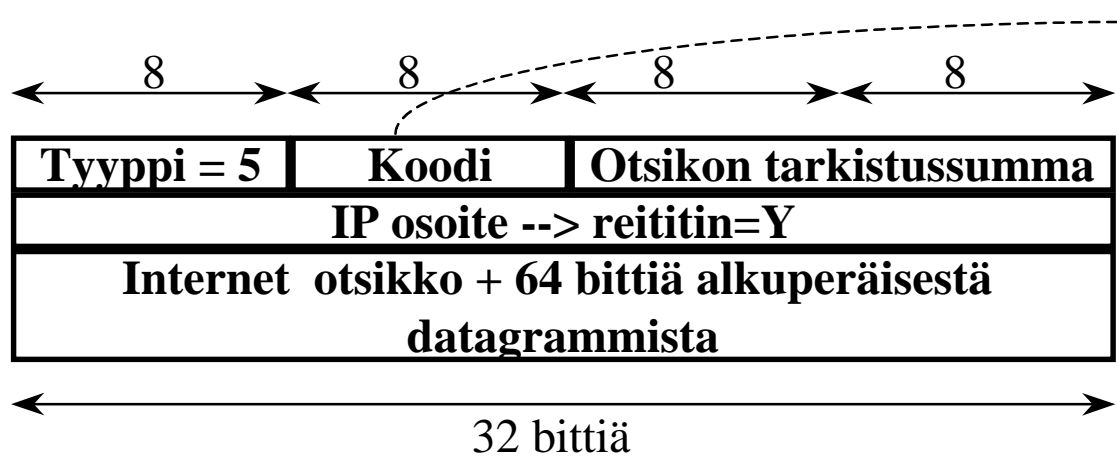
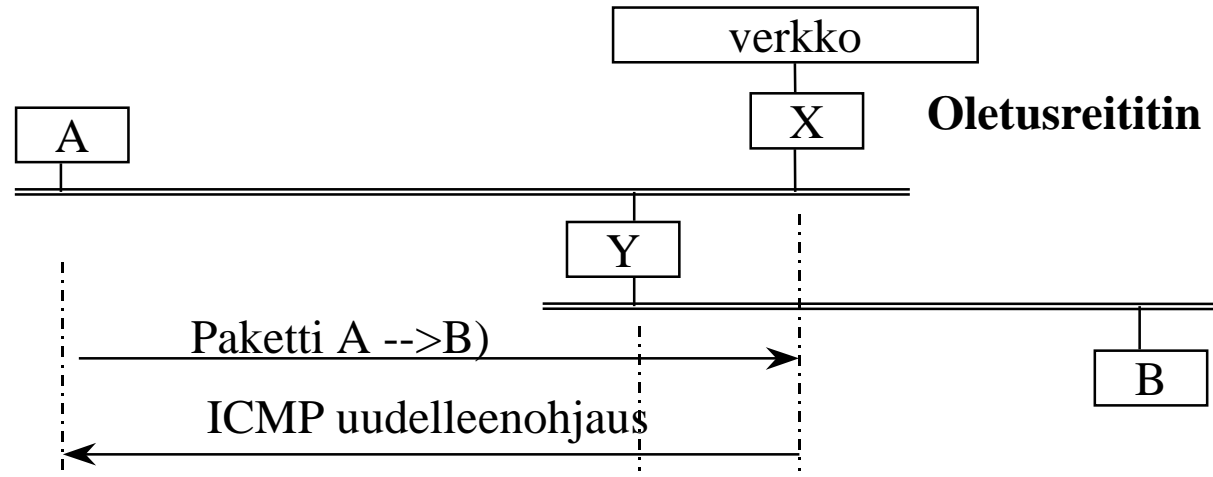


*Joka teknologia vaatii oman ARP sovituksen.  
Helppoa, jos teknologia tukee yleis- tai monilähetystä.  
Esim ATM:ssä tarvitaan ARP - palvelin.*

# Verkossa voi olla useita reitittimiä, joista pitää löytää se, joka on lähinnä kohdetta



# Reititin voi lähettää uudelleenohjauksen osoittaakseen lyhyemmän reitin kohteeseen



- 0 - uudelleenohjaus verkolle
- 1 - uudelleenohjaus kohteelle
- 2 - uo palvelutyyppille ja verkolle
- 4 - uo palvelutyyppille ja isäntäkoneelle

# Redirect is a slow mechanism. Hot-standby addressing is an improvement

- Virtual router redundancy protocol (RFC 2338 - 4/98)
  - a router may have a virtual IP address
  - a router can take the IP and MAC addresses of a failed router (in the same segment)
  - After recovery routers negotiate about address assignments
  - Clients are configured with a static (virtual) router address
  - Cisco and DEC have equivalent proprietary protocols
- Host can listen to RIP or OSPF
  - not recommended but used sometimes anyway

Isäntäkoneen täytyy saada palautetta ensimmäiseltä reitittimeltä, jotta se ei lähettäisi “mustaan aukkoon”

Palautteeksi kelpaa

- TCP tason kuittaukset
- Reititinmainokset
- ARP-vastaukset
- ICMP kaikuvasaus

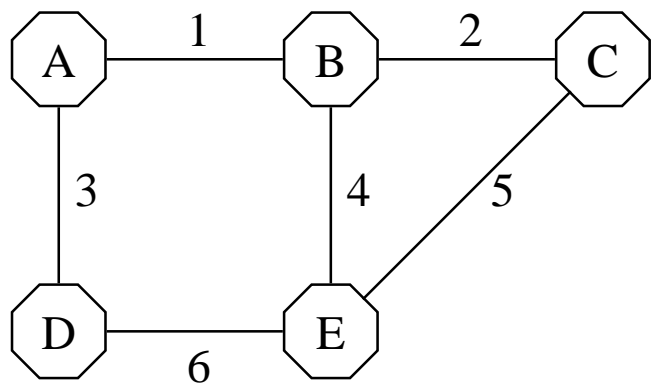
*Reitittimien välillä reititysprotokollat huolehtivat viallisten reitittimien paljastamisesta*

# Routing can be static or dynamic

- Static routing is based on manually configured routing tables.
  - Static routing is used when e.g. two peer providers do not trust each other
  - Static routing is difficult to maintain
- Dynamic routing is based on routing protocols which create and maintain the routing tables automatically
  - examples of routing protocols are RIP, OSPF, BGP...



# RIP - Routing Information Protocol on sisäisen reitityksen perusprotokolla



Esimerkkiverkko, jossa solmut A, B, C ...

RIP on etäisyysvektoriprotokolla.

*Tarkastellaan EV-protokollien toimintaperiaatetta*

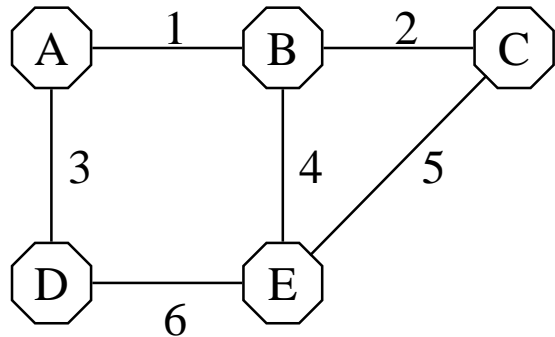
Alkutila: Noodit tuntevat omat osoitteensa ja liitännänsä, mutta ei muuta

Solmu A muodostaa reititystaulun:

A:sta solmuun	Linkki	Kustannus
A	paikallinen	0

Taulua vastaa etäisyysvektori  $A=0$ .

# Reititystaulujen muodostus käynnistyy, kun kaikki solmut lähettävät toisilleen omat etäisyysvektorinsa kaikista liitännöistä



$A=0$  →

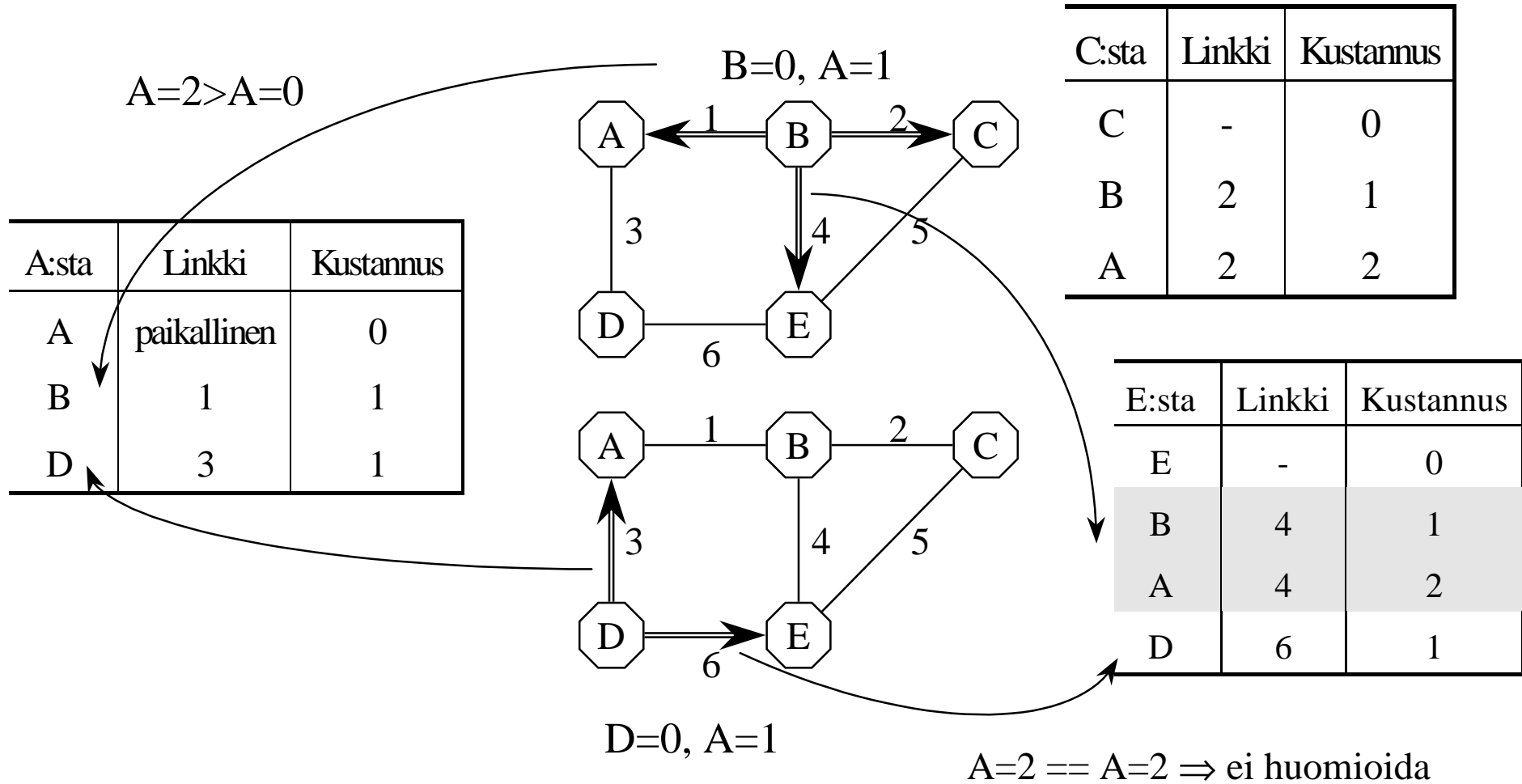
Tarkastellaan vastaottoa solmussa B

B:sta solmuun	Linkki	Kustannus
B	paikallinen	0

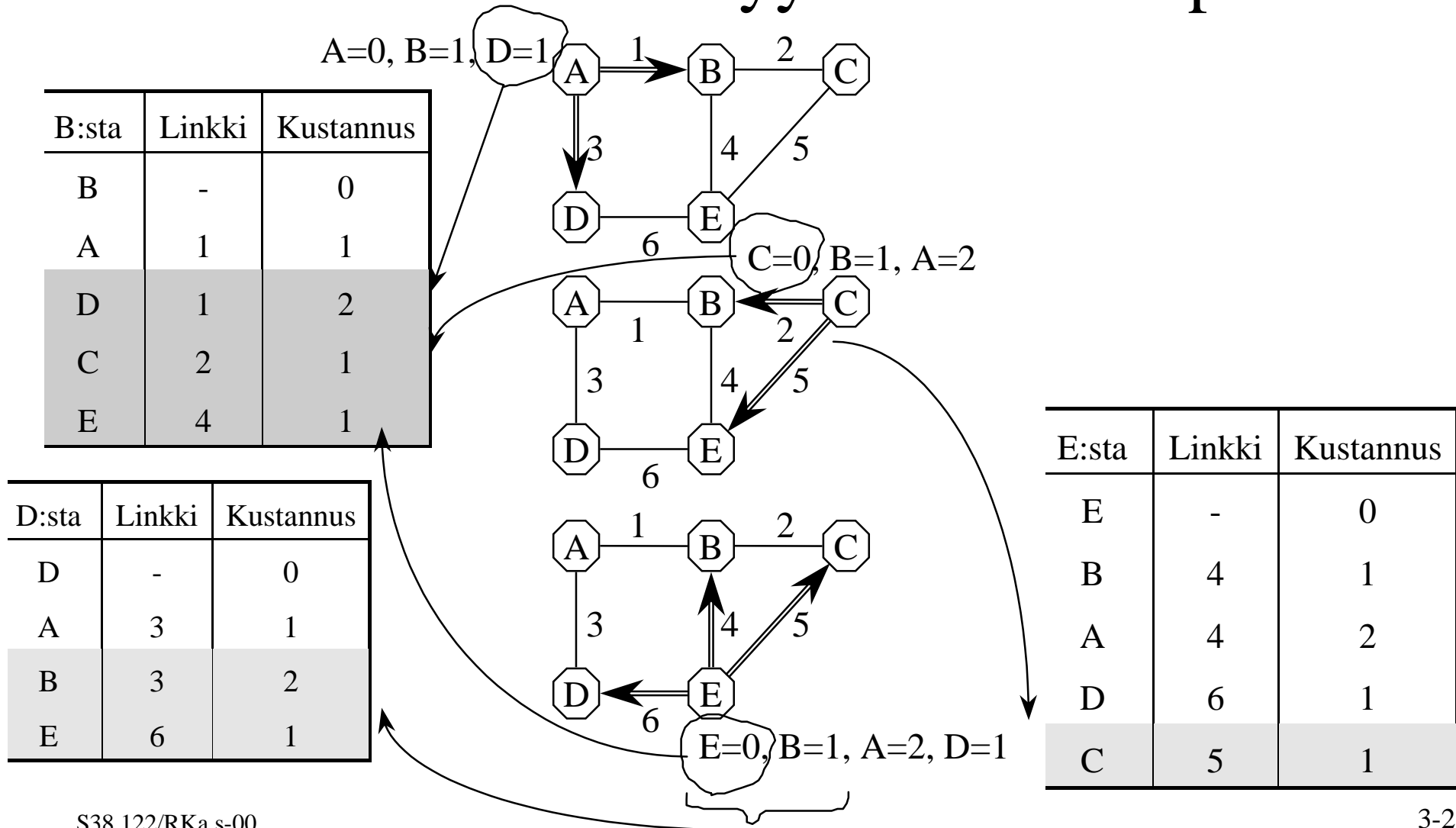
1. B lisää heti etäisyysvektoriin + 1 --> A=1 ja
2. B etsii tulosta omasta taulusta, ei löydy
3. B lisää saamansa tiedon reititystauluunsa, tulos on

B:sta solmuun	Linkki	Kustannus
B	paikallinen	0
A	1	1

# B muodostaa oman vektorinsa ja lähettää sen kaikille naapureille



# Solmut, joiden reititystaulut muuttuivat lähettävät uudet etäisyysvektorit naapureille



# Muutokset lähetetään taas ...

