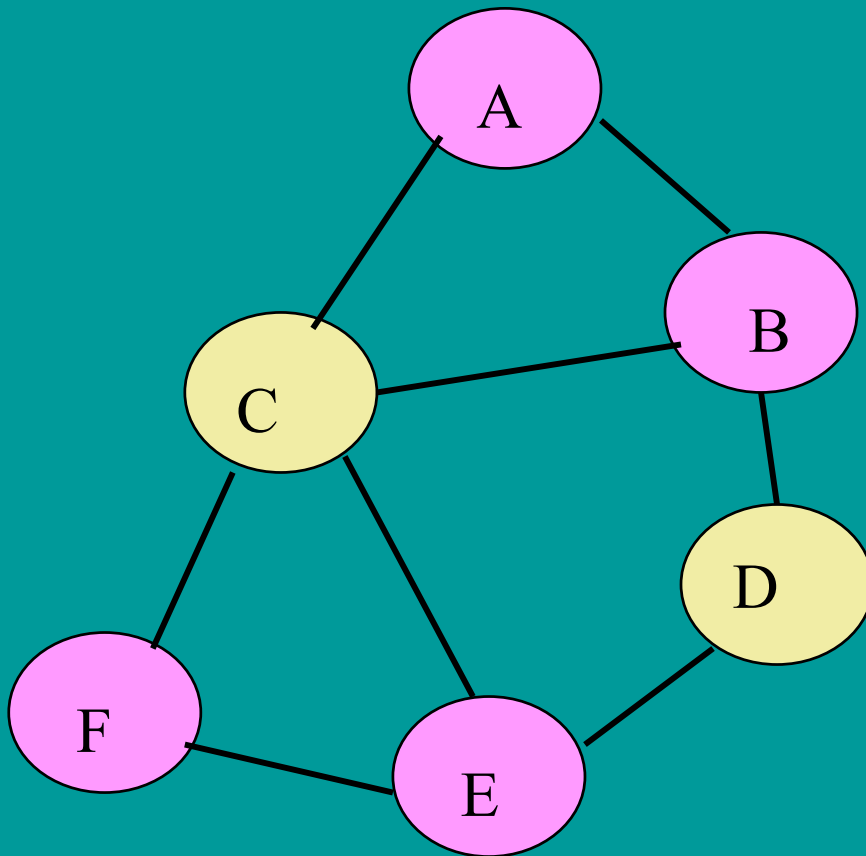


Monilähetysreititys (multicast routing)

- **Ongelma:**

- Reitittimien on kyettävä rakentamaan ‘optimaaliset’ reitit ryhmän kaikille vastaanottajille
 - kun mikä tahansa kone voi toimia lähettäjänä
 - ryhmään voi kuulua eri määrä vastaanottajia
 - lähes kaikki isäntäkoneet
 - vain muutama isäntäkone
 - ryhmän jäsenyys voi olla hyvin dynaamista
- Tavoitteena on löytää mahdollisimman optimaalinen linkkipuu, joka yhdistää kaikki ryhmän jäsenet
 - sanomien reititys puun linkkejä pitkin



A, B, E ja F:
reitittimillä
ryhmän jäseniä

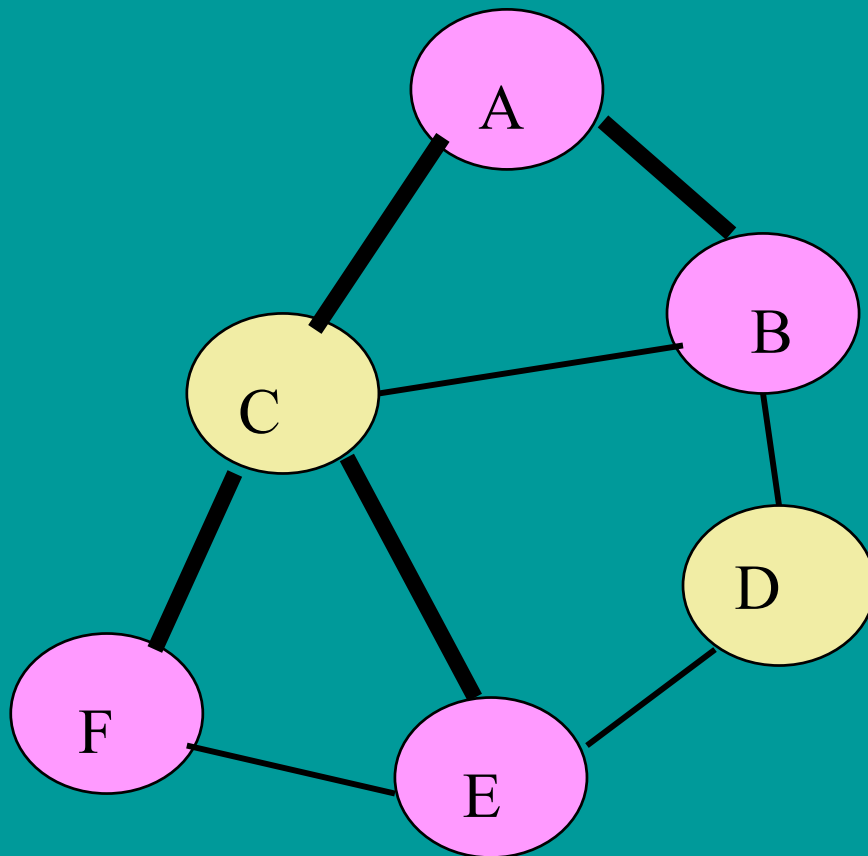


C ja D:
reitittimillä ei
ole ryhmän
jäseniä

Monireitityspuun rakentaminen

- **Kaksi erilaista lähestymistapaa**
 - yksi puu koko ryhmälle (group shared tree)
 - kuka tahansa toimii lähettäjänä, niin reitityksessä käytetään samaa puuta
 - jokaiselle lähettäjälle oma puu (source-based tree)
 - jos ryhmässä on n jäsentä, niin muodostetaan n eri puuta
 - jokaisen lähettäjän sanomat reititetään sen oman linkkipuun avulla

Yksi puu koko ryhmälle



A, B, E ja F:
reitittimillä
ryhmän jäseniä

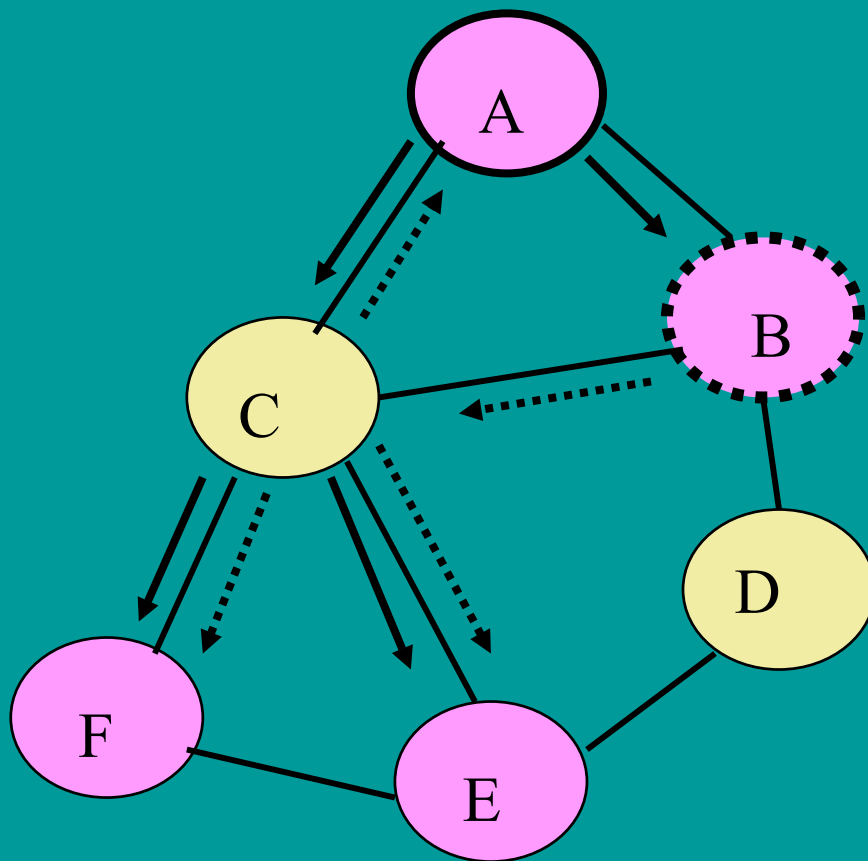


C ja D:
reitittimillä ei
ole ryhmän
jäseniä



reitityslinkki

Eri lähettäjäille omat puut



A, B, E ja F
:reitittimillä
ryhmän jäseniä



C ja D:
reitittimillä ei
ole ryhmän
jäseniä



A:n lähettäessä

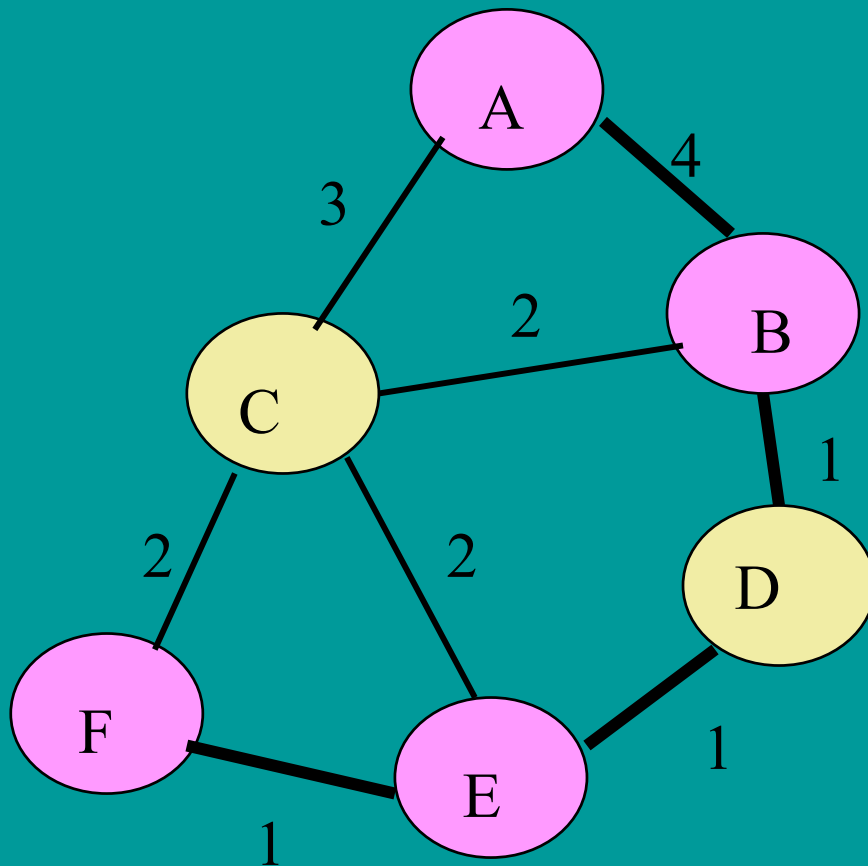


B:n lähettäessä

Reititys käyttäen yhtä puuta koko ryhmälle

- **Löydettävä puu, joka yhdistää kaikki ryhmän reitittimet**
 - mukana myös muita reitittämiä
 - puun kustannus on sen linkkien kustannusten summa
- **pienimmän kustannuksen puu**
 - NP-täydellinen ongelma (Steiner tree problem)
 - suht.koht. hyviä heuristisia ratkaisuja on
 - ei ole käytössä Internetissä
 - tiedettävä kaikki kaikki linkkikustannukset
 - kustannusten muuttuessa laskettava uudelleen
 - jo muutenkin laskettujen kustannusten hyödyntäminen

Pienimmän kustannuksen monilähetyspuu



A, B, E ja F:
reitittimillä
ryhmän jäseniä

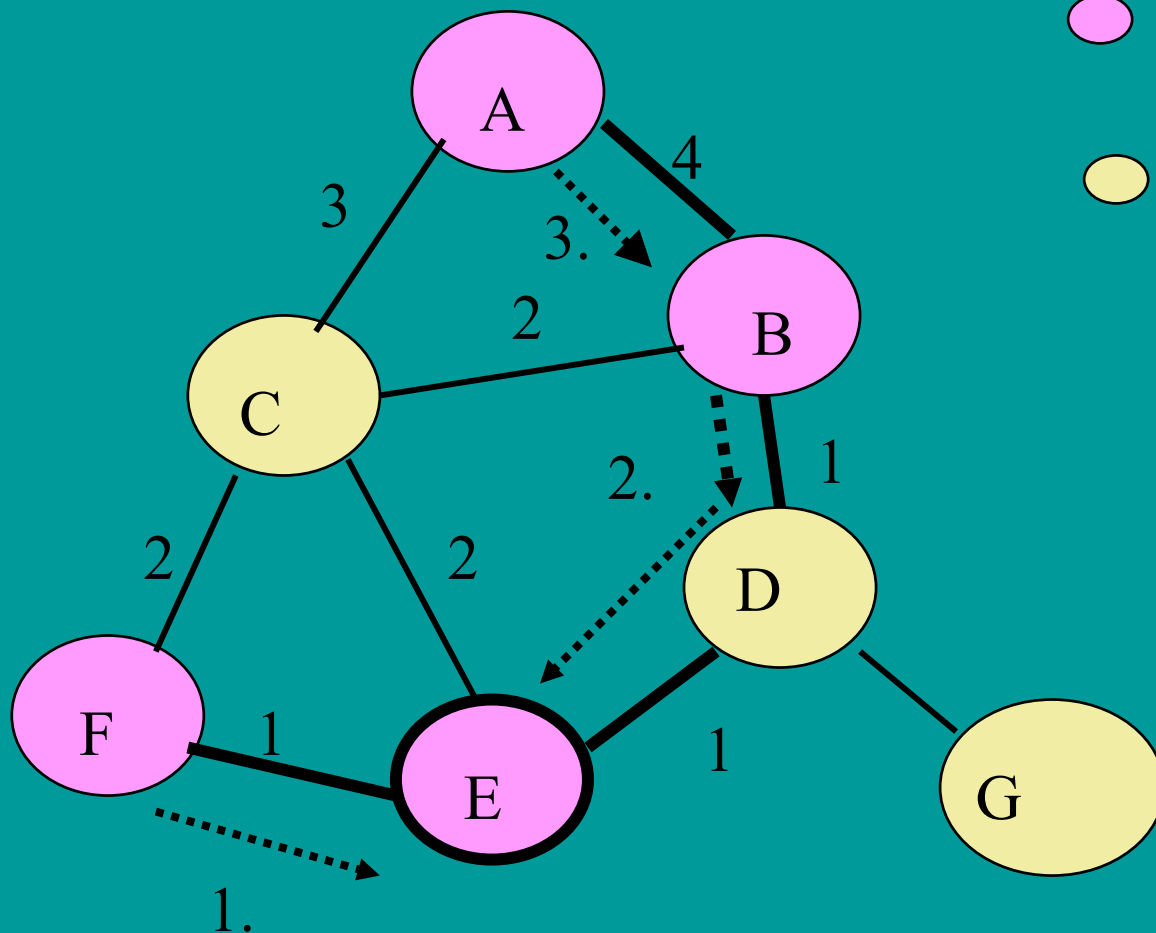


C ja D:
reitittimillä ei
ole ryhmän
jäseniä

Keskuspohjainen reititys (Center-based routing)

- **Ryhmän puun keskuksena on jokin solmu, johon muut myöhemmin liittyvät**
 - ensin saadaan selville keskussolmu
 - muut liittyvät siihen JOIN-sanomilla
 - yksilähetyksiä keskussolmulle
 - Miten keskussolmu valitaan?
 - Valitaan siten, että puu on melko lähellä optimia

Keskuspohjainen monilähetyspuu



- A, B, E ja F: reitittimillä ryhmän jäseniä
- C ja D: reitittimillä ei ole ryhmän jäseniä

Ratkaisevaa on keskussolmun järkevä valinta

Jokaiselle lähettäjälle oma puu

- **Tavallisessa reitityksessä jo yleensä lasketaan pienimmän kustannuksen puu lähettäjältä muihin solmuihin**
 - Dijkstra => reititystaulu
 - least unicast-cost path tree = näiden polkujen yhdistelmä
- **Reverse path forwarding**
 - “Älä turhaan lähetä tänne” (pruning)
- **paljon puita**
 - N lähettäjä => N puuta
 - reitityksessä käytetty puu valitaan lähettäjän mukaan₉

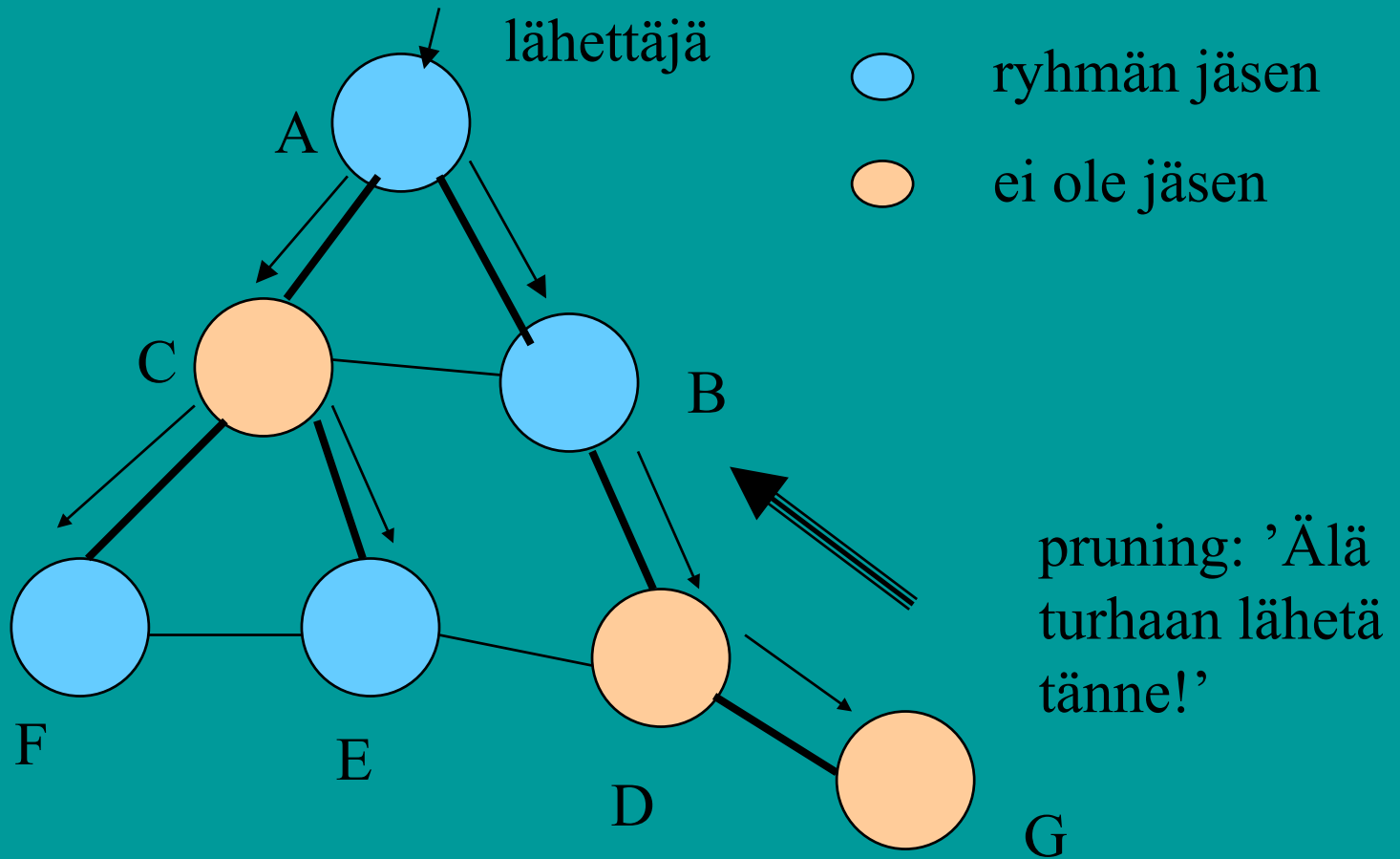
Reverse path forwarding -algoritmi

- **idea**

- tuliko paketti portista, josta normaalisti lähetetään paketin aloittaneelle solmulle?
 - jos tuli, paketti kopioidaan kaikkiin muihin portteihin
 - jos ei tullut paketti tuhoetaan kaksoiskappaleena

- **edut**

- tehokas ja helppo toteuttaa
- ei tarvitse tuntea virittävää puuta
- ei ylim. yleisrasitetta (kohdelista, lisäbittejä)
- tulvitus päättyy itsestään



Monilähetysreititys Internetissä

- **DVMRP (Distance Vector Multicast Routing Protocol) (RFC 1075)**
 - kullekin lähteelle oma puu
 - käyttäen ‘reverse path forwarding’-menetelmää ja karsimista (pruning) ja lisäämistä (graft)
 - etäisyysvetorialgoritmin avulla kukin reititin laskee lyhyimmän polun jokaiseen mahdolliseen lähteeseen ja tallettaa linkin (next hop)
 - tieto puussa ‘alavirtaan’ sijaitsevista reitittimistä, jotta tiedetään, milloin haara voidaan karsia
 - kun kaikki reitittimet ilmoittavat, etteivät enää ole kiinnostuneita

Muita

- **MOSPF** (Multicast Open Shortest Path First) (RFC 1584)
 - OSPF:ää käyttävissä AS:issä
 - linkkitilailmoituksissa myös tieto monilähetysryhmien jäsenyydestä
 - kaikki reitittimet tietävät, mihin monilähetysryhmiin muiden reittimien isäntäkoneet kuuluvat
 - voivat laskea kullekin lähteelle oman ennaltakarsitun lyhyimmän polun puun kullekin monilähetysryhmälle

Muita monilähetysprotokollia: CBT

- **CBT (Core-based Trees) (RFC 2201, RFC 2189)**
 - kaksisuuntainen yhteiskäyttöinen puu, jossa yksi keskus
 - sanomia
 - JOIN_REQUEST keskussolmulle, kun haluaa liittyä ryhmään
 - JOIN_ACK keskussolmu tai lähin jo ryhmässä oleva reititin
 - ECHO_REQUEST vieläkö mukana ryhmässä
 - ECHO_REPLY vielä mukana
 - FLUSH_TREE poistetaan ryhmästä

Muita: PIM

- **PIM** (Protocol Independent Multicast) (RFC 2362)
 - dense mode ~ DVMRP
 - tulvita ja karsi sopii hyvin, jos vastaanottajia on ‘tiheään’
 - sparse mode ~ CBT
 - JOIN-sanomia, jotka ohjataan yksilähetyksenä keskussolmuun
 - polulla olevat reitittimet monilähetysmoodiin
 - keskussolmu lähettää monilähetyksenä muille
 - yksi puu \Leftrightarrow lähettäjälle oma puu

Mobile IP

- **IP-reititys IP-osoitteen perusteella**
 - koneen osoite riippuu verkosta, jossa kone sijaitsee
 - kun kone siirtyy toiseen verkkoon tilapäisesti, osoite ei ole enää voimassa
 - koneelle uusi osoite tässä verkossa?
 - Tieto uudesta osoitteesta muille?
 - TCP-yhteys katkeaa
 - saumaton siirtyminen tuntumattomasti ei ole mahdollinen
 - kaikille koneille verkosta riippumaton osoite?

Liikkuvien isäntäkoneiden reititys

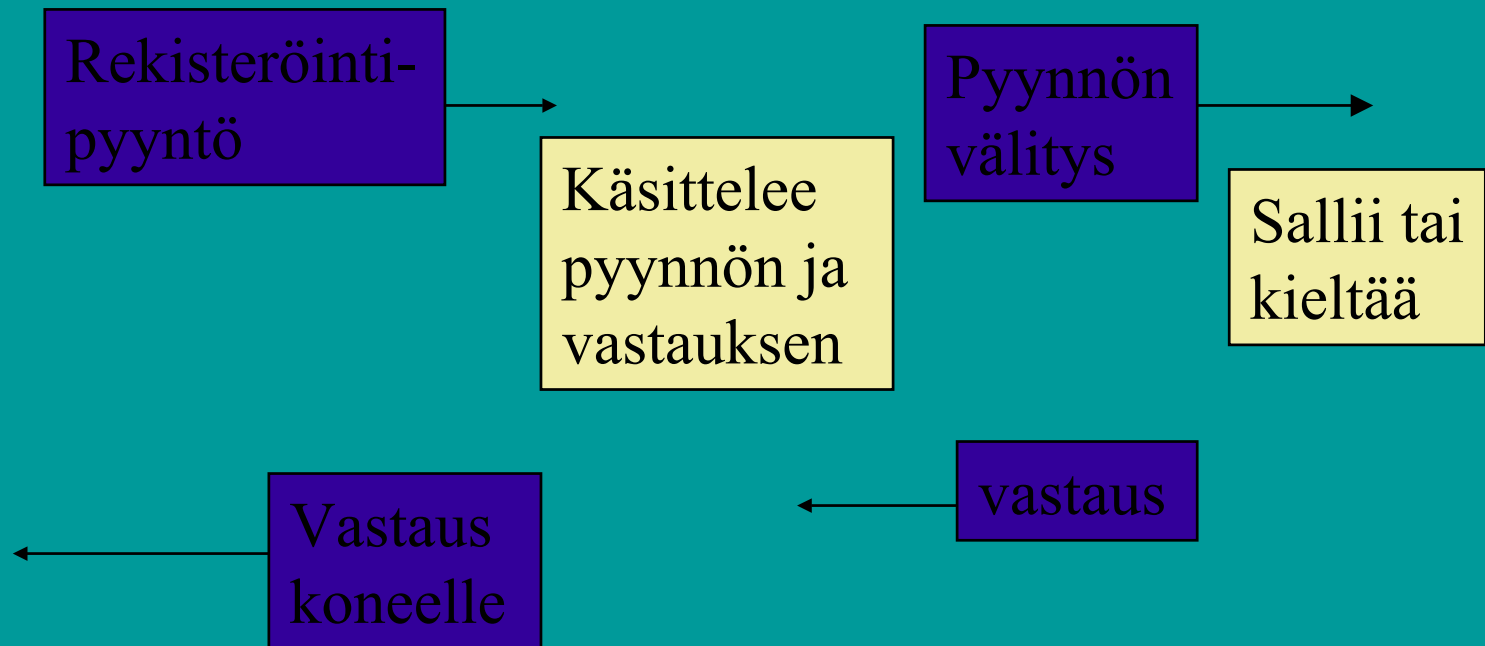
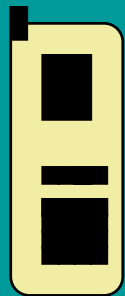
- **liikkuva kone (mobile host)**
 - kotiosoite (home address, home location)
 - kotiagentti (home agent)
 - tietää, missä omat liikkuvat ovat
- **kun ilmaantuu vieraille alueelle, se rekisteröityy**
 - vierasagentti (foreign agent)
 - hoitaa alueelle tulleet vieraat liikkuvat

Liikkuvien isäntäkoneiden reititys

- **liikkuva kone (mobile host)**
 - kotiosoite (home address, home location)
 - pysyvä osoite omassa verkossa,
 - aina tavoitettavissa tällä osoitteella
 - kotiagentti (home agent)
 - tietää, missä omat liikkuvat ovat
 - jos ei liikkuva kone ei ole kotiverkossa, kotiagentti osaa ohjata sille tulevat sanomat liikkuvan uuteen osoitteeseen

- **kun liikkuva kone ilmaantuu vieraalle alueelle, se rekisteröityy alueen**
 - vierasagentille (foreign agent)
 - joka hoitaa alueelle tulleet vieraat liikkuvat
 - antaa niille osoitteen (care of address)
 - tämän verkon osoite
 - tarkistaa vieraan tiedot sen kotiagentilta
 - ilmoittaa kotiagentille koneen uuden osoitteen
 - näin kotiagentti tietää uuden sijainnin

Uudelle alueelle rekisteröinti



Rekisteröintipyyntö sisältää:

- sanoman tyyppin (1)
- lippuja, mm. haluttu tunnelointitapa
- rekisteröinnin keston
- koneen kotiosoitteen, kotiagentin osoitteen ja koneen vierasverkon osoitteen
- rekisteröintipyynnön tunnisteiden
- laajennuksia, mm. autentikointilaajennus

Rekisteröintivastauksessa:

- **Sanoman tyyppi (3)**
- **hyväksyttiin vai hylättiin rekisteröintipyyntö, kuka hylkäsi kotiagentti vai vierasagentti**
- **hyväksytty rekisteröinnin kesto**
- **pyynnön tunniste**
 - liittyy vastauksen pyyntöön
- **laajennusosia mm. autentikointi**

Rekisteröinnissä ongelma on turvallisuus

- **Tekeytyminen vierasagentiksi**
 - haluaa kaapata koneen liikenteen
 - tehokas autentikointi estää
 - autentikoinnin laajennusosa
 - MN -->FA, MN --> HA, FA --> HA
- **vanhojen rekisteröintipyyntöjen lähettäminen kotiagentille**
 - kotiagentille väärä osoite => koneelle ei voi lähettää kotiosoitteella
 - pyyntöihin aikaleimat

Agentin löytäminen verkosta

- **Agentit ilmoittelevat itsestään säännöllisin välein**
 - ilmoituksissa
 - reitittimen osoite
 - rekisteröinnin kesto
 - joukko lippuja: toimiiko vieras- vai kotiagenttina, onko kiireinen, millaista kapselointi kykenee käyttämään
 - vierasosoitteita, vähintään yksi
- **ilmoitusten avulla kone havaitsee siirtyneensä toiseen verkkoon**
 - agentin osoite vaihtuu => uudelleenrekisteröinti
 - kotiverkossa, kun saa ilmoituksia omalta kotiagenttiltaan
 - peruutettava rekisteröinti

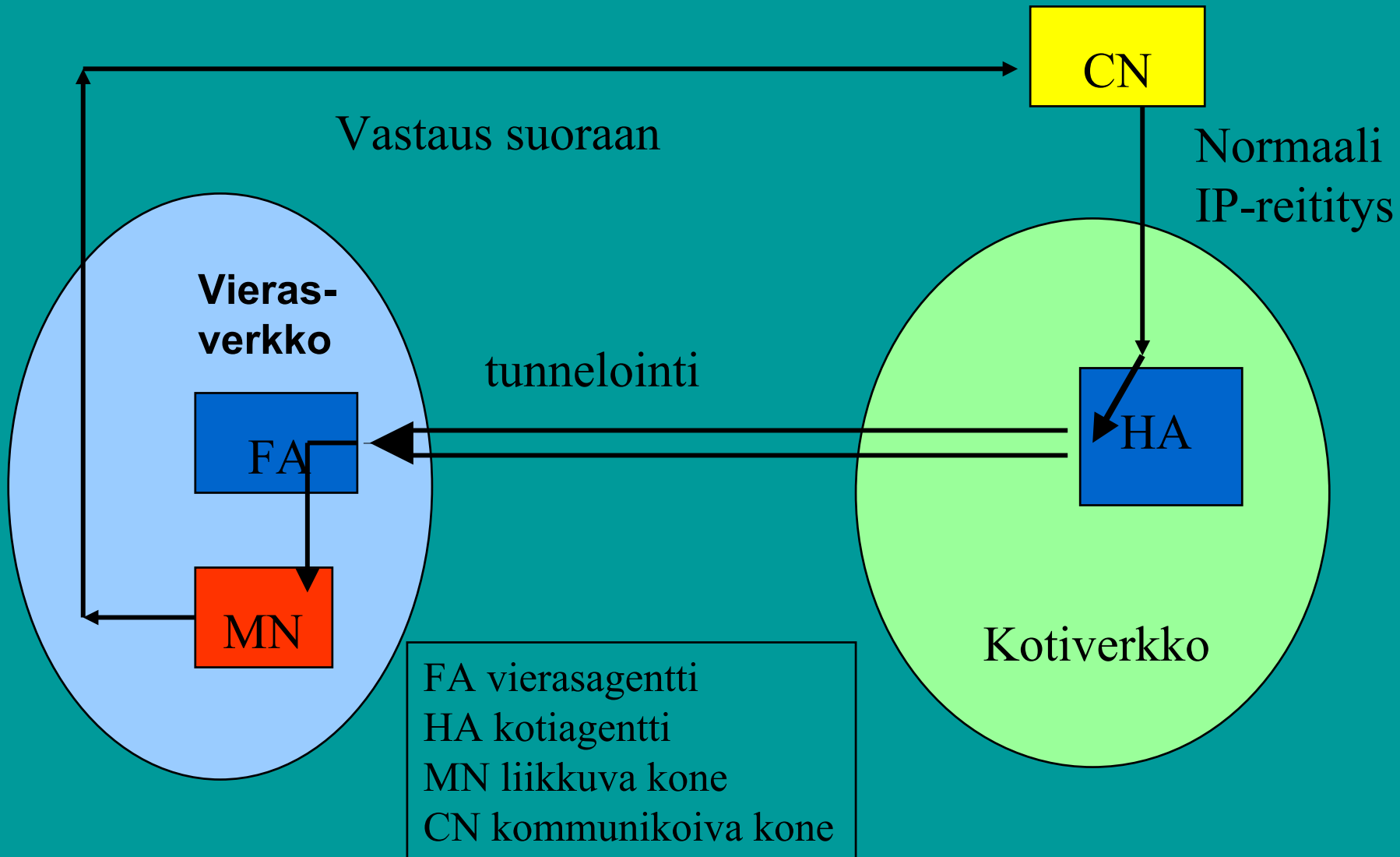
-
-



- **Liikkuva kone kysyy itse agenttia**

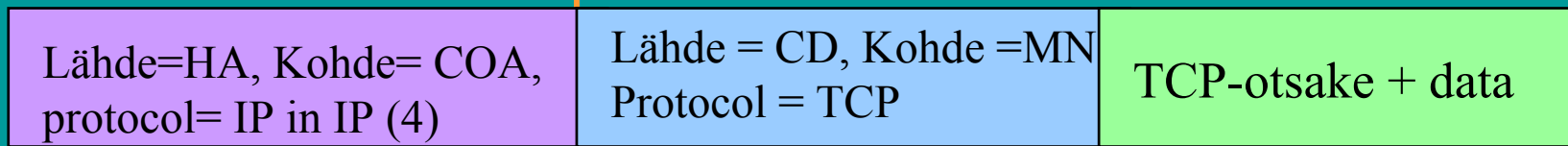
- huomaa liikkuneensa toiseen verkkoon, kun alkaa saada sanomia toisella taajuudella
- lähettää verkkoon kyselypyynnön, johon agentti vastaa ilmoituksella suoraan kyselevälle koneelle

Sanoman reititys vieraassa verkossa olevalle koneelle



Kotiagentti välittäjänä

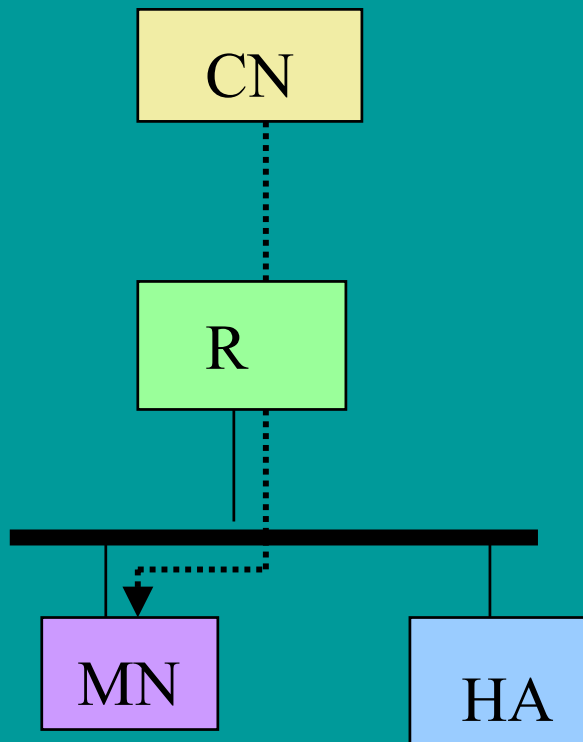
- **Kun paketti lähetetään liikkuvalla,**
 - se ohjautuu IP-osoitteen perusteella kotiverkkoon.
 - Kotiagentti ottaa paketin itselleen. Se tietää vastaanottajan nykyisen sijainnin ja ohjaa paketin sinne.
 - Käytetään IP-tunnelointia
 - uusi osoite COA on usein FA:n valvoma osoite



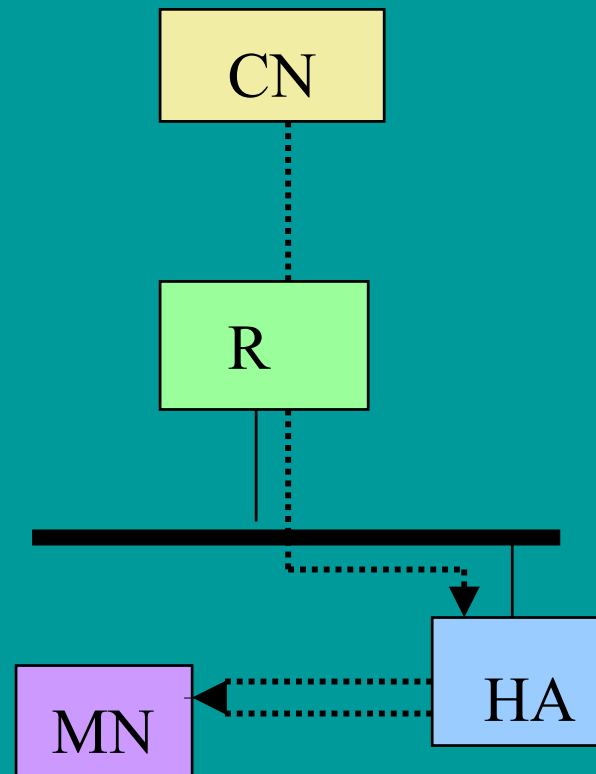
Uusi IP-otsake

alkuperäinen IP-paketti

Toiminta eetteriverkossa



Normaali reititys

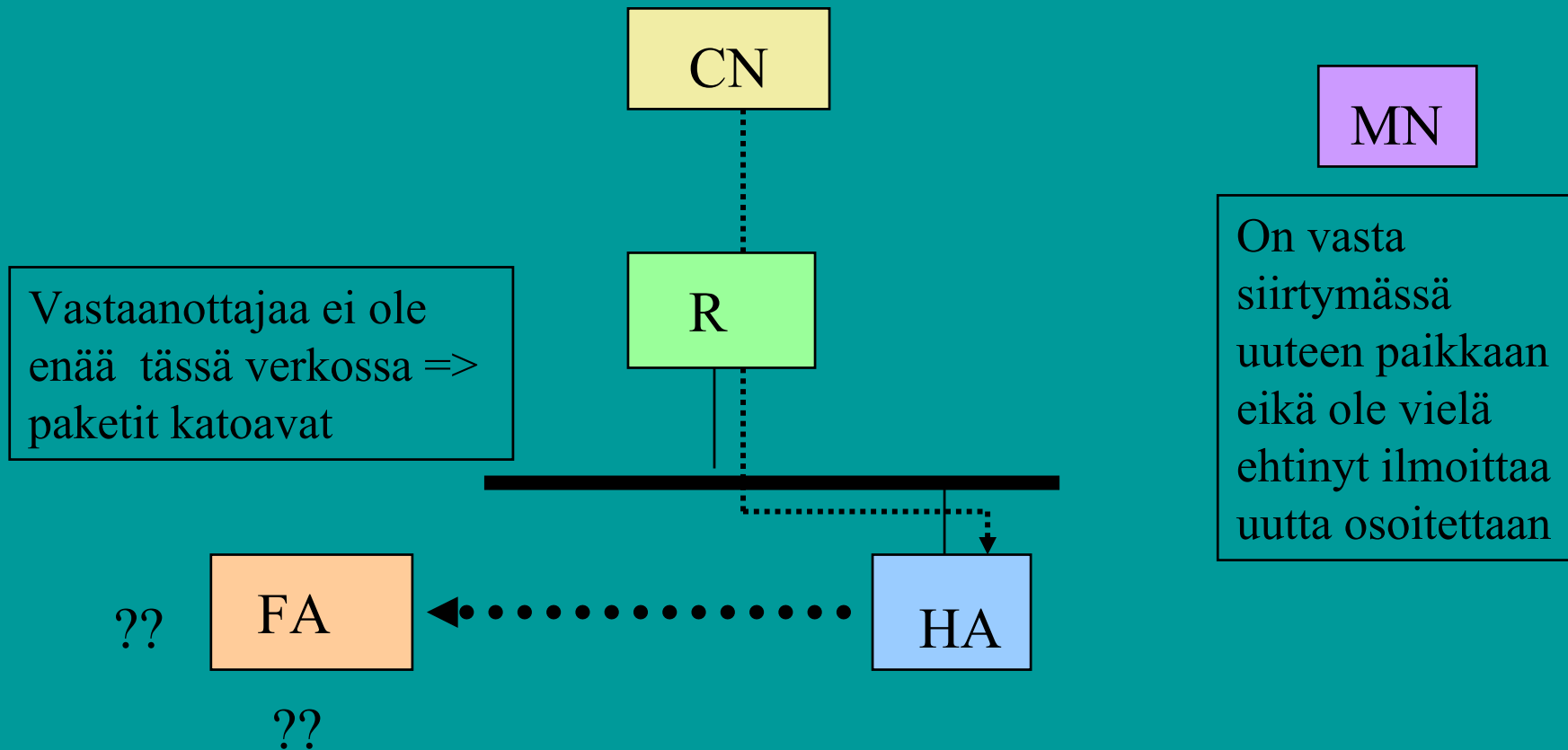


Kotiagentti tunneloi
sanoman MN:lle

Kun joku lähettää liikkuvalla paketin

- se tulee ensin reitittimelle
- reititin kysyy vastaanottajan LAN-osoitetta ARP:illa
- jos liikkuva on kotiverkossaan, se vastaa ja ilmoittaa oman koneosoitteensa
- muuten kotiagentti vastaa omalla osoitteellaan ja saa paketin
- kotiagentti lähettää tunneloinnilla vierasosoitteeseen (yleensä vierasagentin oma osoite)
- vierasagentti kysyy ARP:lla vierailijan LAN-osoitetta ('koneosoitetta')
- Ja lähettää sanoman vierailevalle koneelle.

Ongelma:

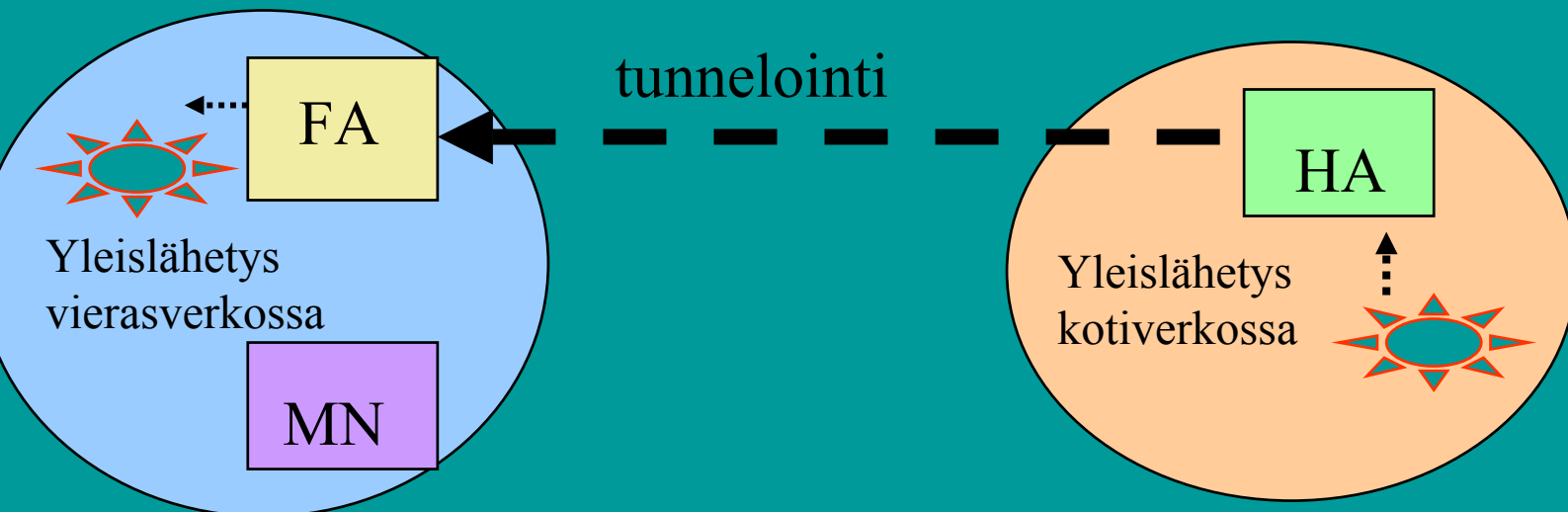


Entä jos vierasverkossa ei ole FA:ta?

- MN saa tilapäisen IP-osoitteen verkkoon PPP- tai DHCP-protokollalla
- käyttäen tätä osoitetta COA-osoitteena se voi itse toimia omana FA:na
- Ongelmia:
 - tunnelointi lisää yleisarasitetta (ylim. IP-otsake) ja viimeinen linkki on hidas radiolinkki
 - liikkuvat tarvitsevat paljon tilapäisiä IP-osoitteita => osoitteet voivat loppua
 - poistuva kone ei aina ilmoita lähdöstään
 - kun kone poistuu, FA katoaa ja matkalla olevat paketit varmasti katoavat

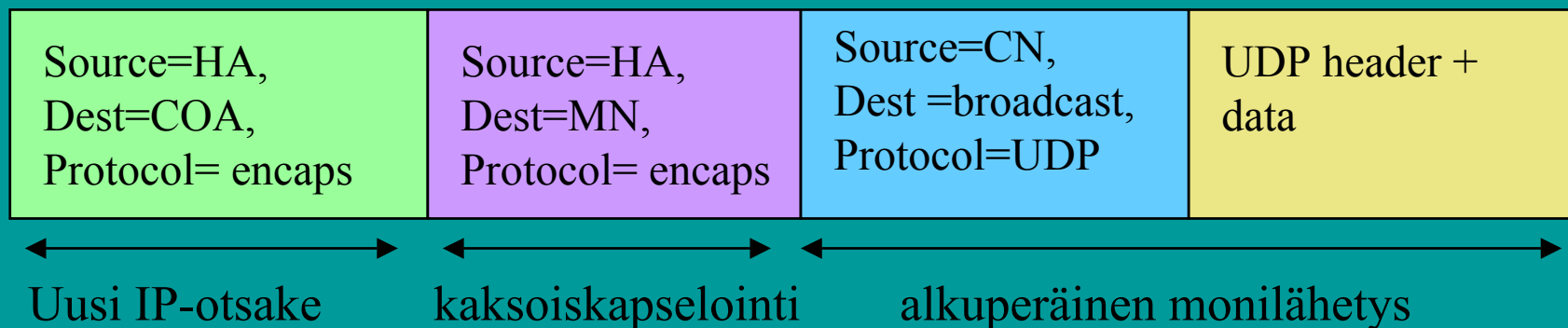
Monilähetys vierasverkossa olevalle koneelle

- **HA ohjaa yksitellen kaikki lähetykset**
 - tehotonta, voi aiheuttaa turhaa kuormitusta
 - voi aiheuttaa turhaa monilähetystä
 - tai sitten sanomaa ei toimiteta MN:lle



Monilähetys vierasverkossa olevalle koneelle

- **Jos käytössä verkon FA:aa, niin käytetään kaksoiskapselointia:**
 - HA tietää rekisteröintitiedoista, käyttääkö MN FA:ta vai toimiiko itse oman FA:naan



Järkevämpää tosin olisi rekisteröityä monilähetysryhmään uudelleen vierasverkossa!!

Mobile IPv6

- **Osoitteita riittää**
 - vierailijat tarvitsevat IP-osoitteita vierailunsa aikana
- **Ei tarvita erityistä vierasagenttia**
 - MN toimii yleensä itse omana vierasagenttinaan
 - neighbor discovery
 - stateless address autoconfiguration
- **Paremmat turvallisuuspiirteet**
 - mm. estämään väärennetyt osoitemuutokset ja toistohyökkäykset (replay attack)

- **Reitin optimointi**
 - turvalaajennoksen avulla MN voi ilmoittaa oman uuden osoitteensa suoraan lähettäjäille
 - binding updates/acks/requests
 - kolmioreititystä ei tarvitse turvattomuuden takia käyttää
 - Route Optimization
- **Tehokkaampi kotiagentin löytäminen**
 - Dynamic Home Agent Discovery
 - lähetetään Binding Update-sanoma kotiagenteille anycast-osoitteella, jolloin vain yksi ehkä useasta kotiagentista vastaa.
 - Kotiagentin osoite voi olla muuttunut poissaolon aikana
- **lähdereititysotsakkeen avulla saadaan tehokas kapselointi**
 - two-hop source route
- **Filteroivien palomuurien läpäisy**