Tietoliikenne II (2 ov)

Syksy 2001

Liisa Marttinen

Kurssikirja:

Kurose & Ross, Computer Networking

Lisämateriaalia: Aiheeseen liittyvät RFC:t

Tietoliikenne II

Täydennystä Tietoliikenne I -kurssin asioihin

- perusteellisemmin
- laajemmin
- ‘teoreettisemmin’

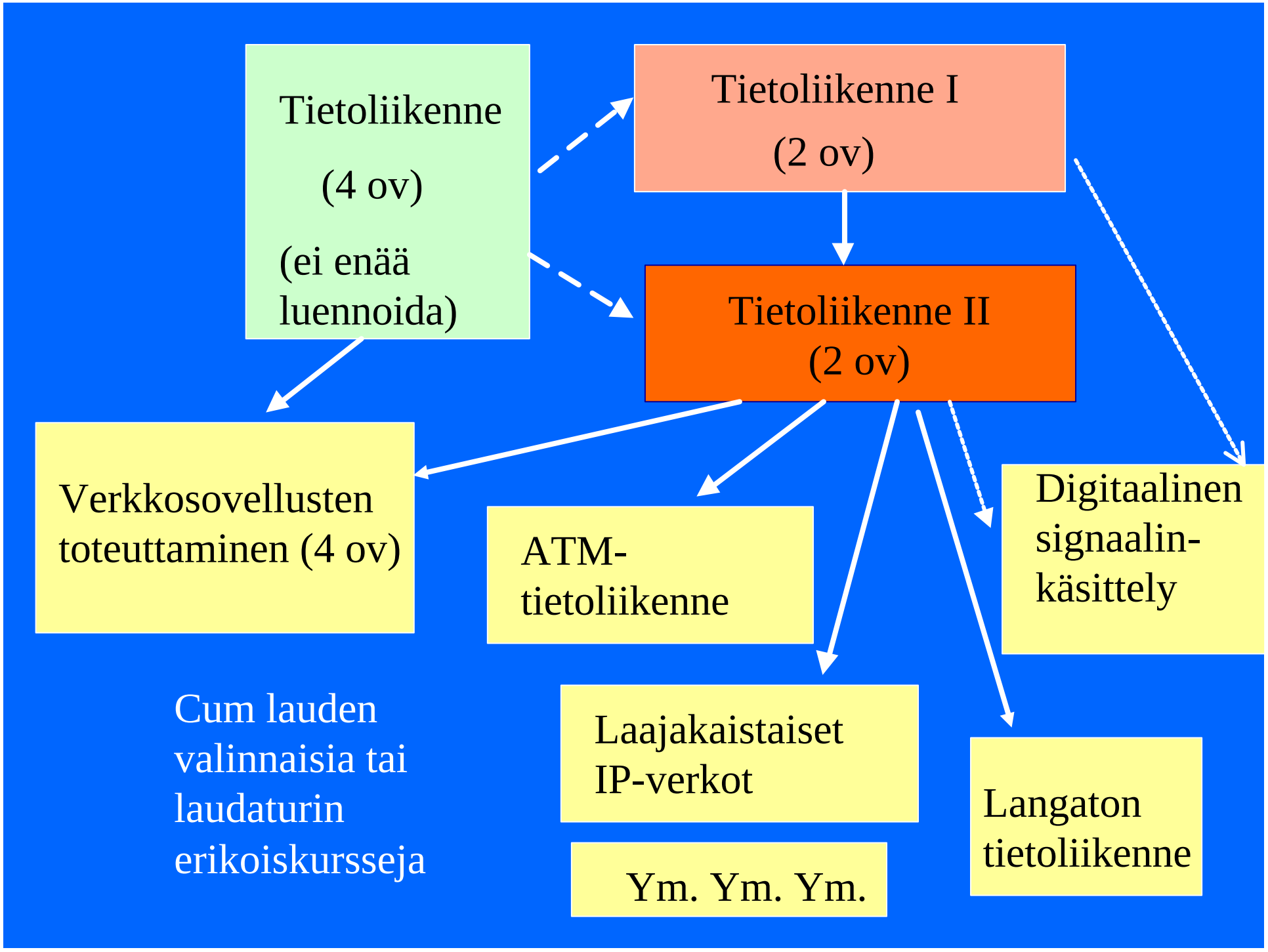
- perus-, linkki- ja MAC-kerros
- reititys, IPv6
- TCP: suorituskyky ja uudet piirteet
- DNS, ..

Alustava sisällysluettelo

- TCP:n suorituskyky
  - optiot
  - uudet piirteet ruuhkanvalvonnassa
- IPv6, IPsec?
- ICMP
- Reititys
  - OSPF, BGP, monilähetysreititys, mobiilireititys
- muita verkkoja: atm, FDDI, ...

Sisällysluettelo jatkuu

- Linkkikerroksen ja peruskerroksen asioita
  - HDLC, PPP, SONET, ..
  - Tiedonsiirron teoreettinen perusta (Shannon, Nyquist, ..)
- Internetin palvelun laatu (QoS)
  - integroidut palvelut
  - eriytyneet palvelut
- ???


Tietoliikenne
(4 ov)
(ei enää luennoida)
Tietoliikenne I
(2 ov)
Tietoliikenne II
(2 ov)
Verkkosovellusten toteuttaminen (4 ov)
ATM-tietoliikenne
Digitaalinen signaalin-käsittely
Laajakaistaiset IP-verkot
Langaton tietoliikenne
Ym. Ym. Ym.
Cum lauden valinnaisia tai laudaturin erikoiskursseja

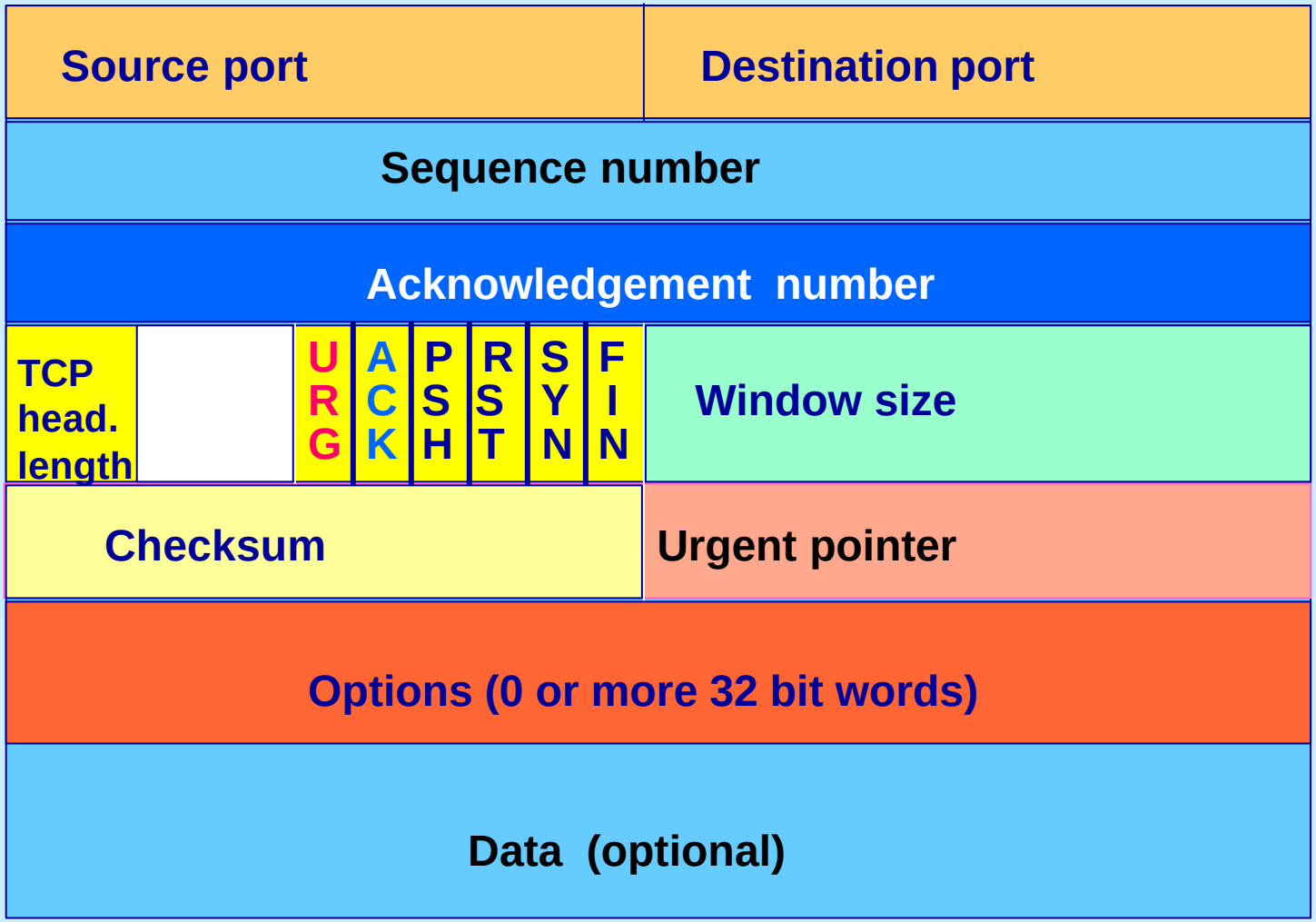
Suoritus

- kurssikoe maks. 50 p, min 20 p
  - ma 17.12. klo 9-13 sali 1 päärakennus
- kurssiaktiivisuus maks. 20 p
  - traditionaaliset harjoitukset => maks. 10 p
  - miniesseet (1-5 sivua) ja -esitelmät (10-15 min), keskusteluaktiivisuus yms. => maks. 10 p
- 30 p => 1-, 51 => 3
- tai sitten loppukokeella maks. 60 p
  - tammi- tai helmikuussa 2002

1. TCP

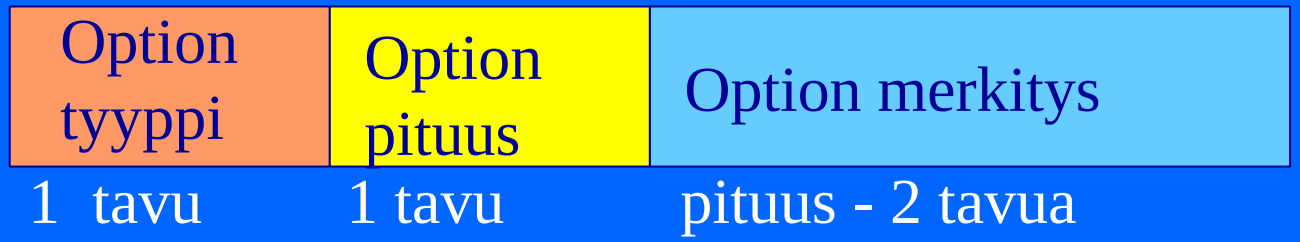
- TCP:n peruspiirteiden toiminta tarkemmin
  - osin harjoitustehtävissä
- TCP:n lisäpiirteitä
  - Window scaling
  - time stamping
  - SACK (Selective Acknowledgement)
  - RED (Random Early Detection)
  - ECN (Explicit Congestion Notification)

TCP-otsakkeen kentät



TCP-optiot

- Optio-kenttä
  - erilaisia valinnaisia piirteitä varten
  - Option pituus on 40 tavua
    - » TCP header length -kenttä = 4 bittiä kertoo otsakkeen pituuden 32 bitin sanoina => 15*4 tavua = 60 tavua
    - » 60 tavua -20 tavua vakio-otsaketta => enintään 40 tavua optioita varten



Optioita:

- MSS (Maximum Segment Size)
  - käytetään ilmoittamaan vastaanottajan yhteydellä hyväksymä suurin segmentin koko
    - » eri suuntiin voi olla eri koko
    - » voi olla suurempi tai pienempi kuin oletus MSS
      - MTU (Maximum Transfer Unit) = suurin yhdessä verkon kehyksessä kulkeva datamäärä
      - eri verkoissa eri kokoja, mutta minimi MTU= 576 B
      - MSS = MTU - IP-otsake - TCP-otsake
      - oletus MSS = 576 - 20 - 20 = 536
      - otsakkeet voivat olla suurempia!

- MSS ilmoitetaan yhteyttä muodostettaessa eli SYN-segmenteissä
  - kumpikin osapuoli voi ilmoittaa oman MSS-arvonsa
    - » jos ei ilmoita , niin suostuu vastaanottamaan minkä tahansa kokoisia segmenttejä.

2	4	maksimi segmentin koko

Muita ehdotettuja optioita (RFC 1323):

- ikkunaskaalaus (window scaling factor)
  - kasvattaa TCP-otsakkeen 16-bitin ikkunan koon 32-bitin ikkunan kooksi

3	3	skaalaus

- aikaleimaus (timestamp)
  - segmentin aikaleima palautetaan kuittauksessa

8	10	segmentin aikaleima	kuitatun segmentin aikaleiman kaiutus

Erilaisia suorituskykyongelmia

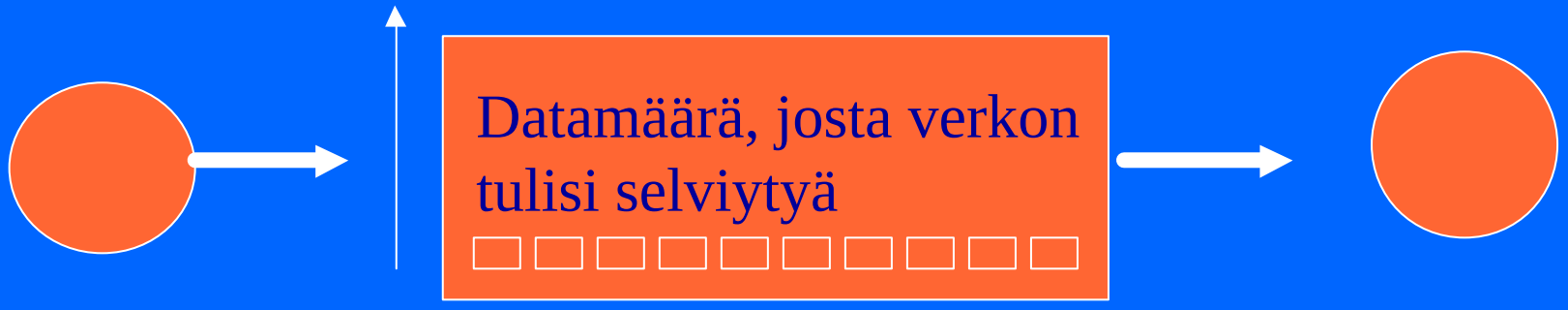
- TCP-protokolla käytössä hyvin erilaisissa ympäristöissä
  - » pitkän viipeen satelliittiyhteyksillä
  - » erittäin nopeilla yhteyksillä
  - » langattomilla yhteyksillä
- => suorituskykyongelmia
  - – otsakkeen kentät liian pieniä
    - » **ikkunankoko 16 bittiä => 65536 tavua**
      - rajoittaa lähetysnopeutta mm.satelliittiyhteyksillä
    - » järjestysnumero 32 bittiä
      - rajoittaa lähetysnopeutta erittäin nopeilla yhteyksillä

Kaistan ja kiertoviiveen tulo (bandwidth * delay product)

- TCP:n suorituskyky riippuu siirtonopeudesta (kaista, bandwidth) ja kiertoviiveestä (RTD, round-trip delay, 1 ms - 100 s)
- tulo siirtonopeus * kiertoviive kertoo sen datamäärän, joka TCP:n täytyy pystyä käsittelemään, jotta lähettäjä ja vastaanottaja voisivat toimia täydellä vauhdilla
  - paljonko kuittaamatonta dataa verkon täytyy pystyä välittämään
- Ongelmia syntyy, jos tulo on hyvin suuri!

Ideaalitilanteessa ‘lähetysputki’ on koko ajan täynnä!

Lähetysnopeus (bps)



RTD (s)

aika lähetyksen alusta
kuittauksen saapumiseen

Ongelmia aiheuttavat LFN-verkot (long, fat pipe network):
pitkä viive ja suuri lähetysnopeus, esim. satelliittiyhteydet
ja nopeat runkolinjat => tulo > 1 Mb

Ongelmia LFN-verkoissa

- Ikkunan koko -kenttä liian pieni => ‘putki’ ei täyty
  - ikkunan skaalaus -optio
- pakettien katoaminen => hidas aloitus eli ‘putki’ joudutaan tyjentämään
  - parannukset ruuhkanhallintakäytäntöön
    - » Fast Retransmit, Fast Recovery, SACK, …
    - » uudelleenlähetysajastimen tarkempi ajastus
      - Timestamp -optio

Ikkunan skaalaus (Window scale factor)

- ikkunakoko = 16 bittiä => 65536 tavua
  - » vuonvalvonnassa
  - » kertoo vastaanottajan ikkunan = kuinka monta tavua voi lähettää ennenkuin täytyy jäädä odottamaan kuittausta
- jos käytössä ikkunan skaalaus -optio, ikkunakentän arvo kerrotaan 2**F, jossa F on skaalausoption arvo.
  - » Suurin F:n arvo on 14.
- käytetään vain yhteyden aloituspyynnössä

Miksi uudelleenlähetysajastimen arvo on tärkeä!

- Ruuhkan oikea havaitseminen riippuu uudelleenlähetysajastimen ‘oikeasta’ arvosta.
  - Liian suuri arvo => alkavaa ruuhkaa ei huomata ajoissa => verkkoa ylikuormitetaan => syntyy ruuhkatilanne => resurssien hukkakäyttöä
  - Liian pieni arvo => luullaan ruuhkaksi, vaikka ei olekaan => hidastetaan turhaan lähetystä => resurssien hukkakäyttöä

Uudelleenlähetysajastimen arvo

- mitataan paketin kiertoviive M ja viiveen poikkeama odotetusta eli |RTT-M|
  - RTT = aRTT + (1-a)M
  - D = bD + (1-b)|RTT-M|
  - ajastimen arvo = RTT + 4D
- Ongelmia aiheuttavat uudelleenlähetykset
  - Mikä sanomista kuitataan?
  - Karn: uudelleenlähetyksiä ei oteta mukaan, vaan ajastimen arvo kaksinkertaistetaan aina uudelleenlähetyksessä, kunnes saadaan onnistuneesti kuittaus.
- Useat toteutukset mittaavat vain yhden paketin ikkunasta!
  - Ongelmia, jos ikkuna suuri.

Aikaleima (timestamp)

- Kaksi eri optiota
  - Timestamp Value
    - lähtevissä segmenteissä,
  - Timestamp Echo Reply
    - kuittauksessa
    - sama kuin kuitatun segmentin Timestamp-arvo
- Voidaan käyttää missä tahansa datasegmentissä
- => Voidaan laskea kiertoviive jokaiselle segmentille, myös uudelleenlähetyksille.

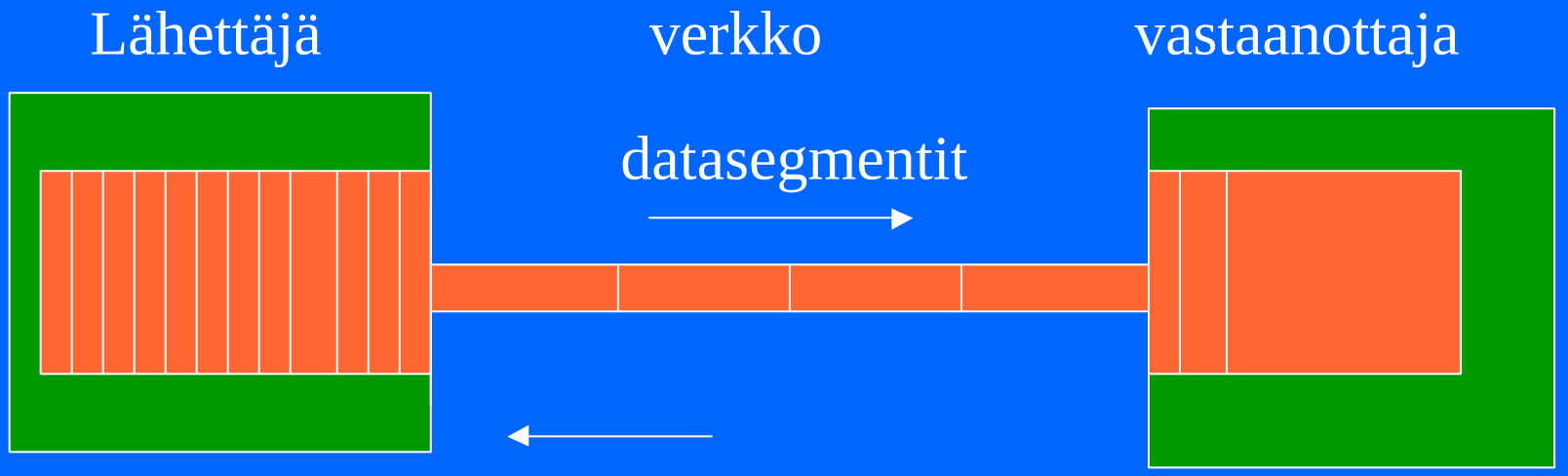
RTTM (Round-Trip Time Measurement)

- lähetettävään sanomaan liitetään aikaleima-optioon aikaleima
  - aikaleimakello, joka tikittää riittävän nopeasti
- sama aikaleima palautetaan sanoman kuittauksessa
- ongelmatilanteita:
  - viivästyneet kuittaukset: aikaisin kuittaamaton
    - TCP:n ei tarvitse kuitata jokaista segmenttiä
  - puuttuva segmentti: viimeisin hyväksytty
  - puuttuvan segmentin saapuminen: viimeisin puuttuva

Viivästetty ACK (Delayed ACK)

- Ei tarvitse välttämättä kuitata jokaista segmenttiä
  - kuitenkin kuitattava ainakin joka toinen ja viive saa olla korkeintaan 500 ms,
    - usein noin 200ms
- Hyöty: kuittaus kulkee datan mukana
  - samalla kertaa ikkunan muotos, kuittaus ja kaiutus
- Haitta: kiertoviiveen laskeminen, pakettien kellotus

TCP-ruuhkanvalvonta



ACK

toimii lähetyksen tahdistajana

putkestä poistunut dataa, joten voidaan lähettää sama määrä lisää

TCP self-clocking

- TCP tahdistaa itse oman lähetyksensä ACK:ien avulla
  - nopeutta voi rajoittaa
    - » verkko
      - ruuhkan takia syytä vielä pienentää lähetysnopeutta
    - » vastaanottaja
      - lähetysnopeus ok
  - lähettäjä ei voi tietää kumpi

Ruuhkanvalvonta on hankalaa!

- Sitä varten on koko ajan kehitetty yhä parempia menetelmiä
  - uudelleenlähetysajastimen arvo
    - » RTT:n varianssin arviointi
    - » Karnin algoritmi
    - » exponential retransmission timer backoff
  - lähetysikkunan hallinta
    - » slow start
    - » congestion avoidance
    - » fast retransmit
    - » fast recovery

Lähetettynä voi olla vain rajallinen määrä kuittaamatonta dataa (‘Flight size’)

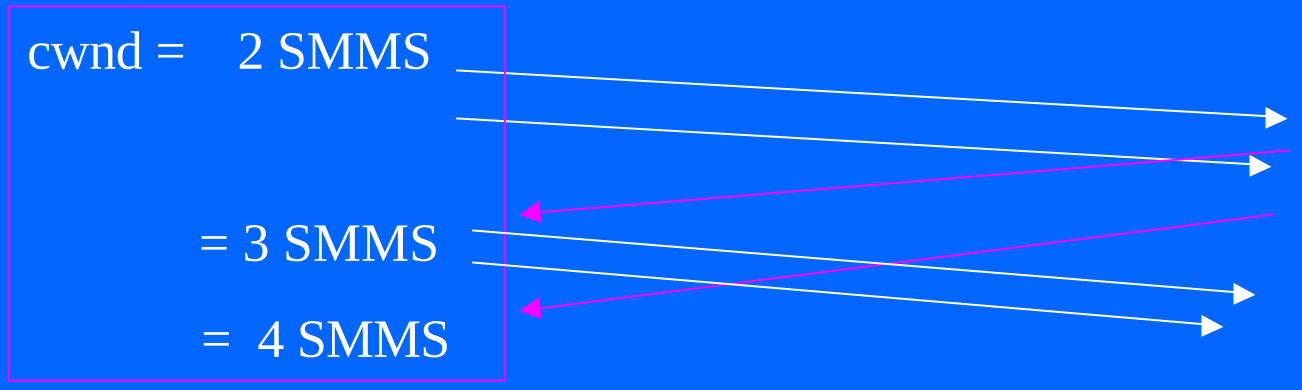
- vastaanottoikkuna (receiver window, **rwnd**)
  - vastaanottaja ilmoittaa lähettämiensä segmenttien ikkunakentässä
  - vastaanottaja voi vapaasti kasvattaa tai pienentää
  - vuonvalvontaa varten
- ruuhkaikkuna (congestion window, **cwnd)**
  - lähettäjä saa korkeintaan lähettää verkkoon , jotta verkko ei tukkeutuisi
  - ruuhkanhallintaa varten
- **min(rwnd, cwnd)** rajoittaa lähettämistä

Ruuhkaikkunan arvo eri tilanteissa

- initial window (IW)
  - » ruuhkaikkunan arvo heti kolminkertaisen kättelyn jälkeen
    - korkeintaan kaksi segmenttiä tai 2* suurin määrä tavuja, jonka lähettäjä voi kerralla lähettää (SMSS)
- loss window (LW)
  - » ikkunan arvo, kun TCP on havainnut , uudelleenlähetysajastimen lauettua, segmentin kadonneeksi
- restart window (RW)
  - » kun lähetys käynnistetään uudelleen joutilaana olon jälkeen

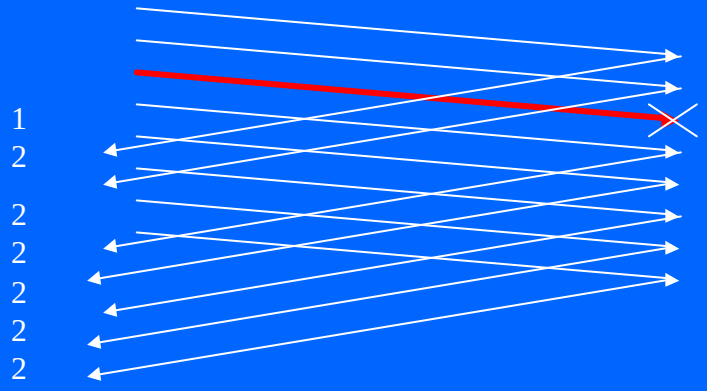
Slow start

- Hitaan aloituksen aikana
  - Ruuhkaikkunaa cwnd kasvatetaan korkeintaan maksimilähetysmäärällä (SMSS) jokaista uutta dataa kuittaavaa ACKia kohden



Hidas aloitus

- Aina yhteyden alussa
- kun kuittausta ei tule ajoissa (paketti kadonnut!)



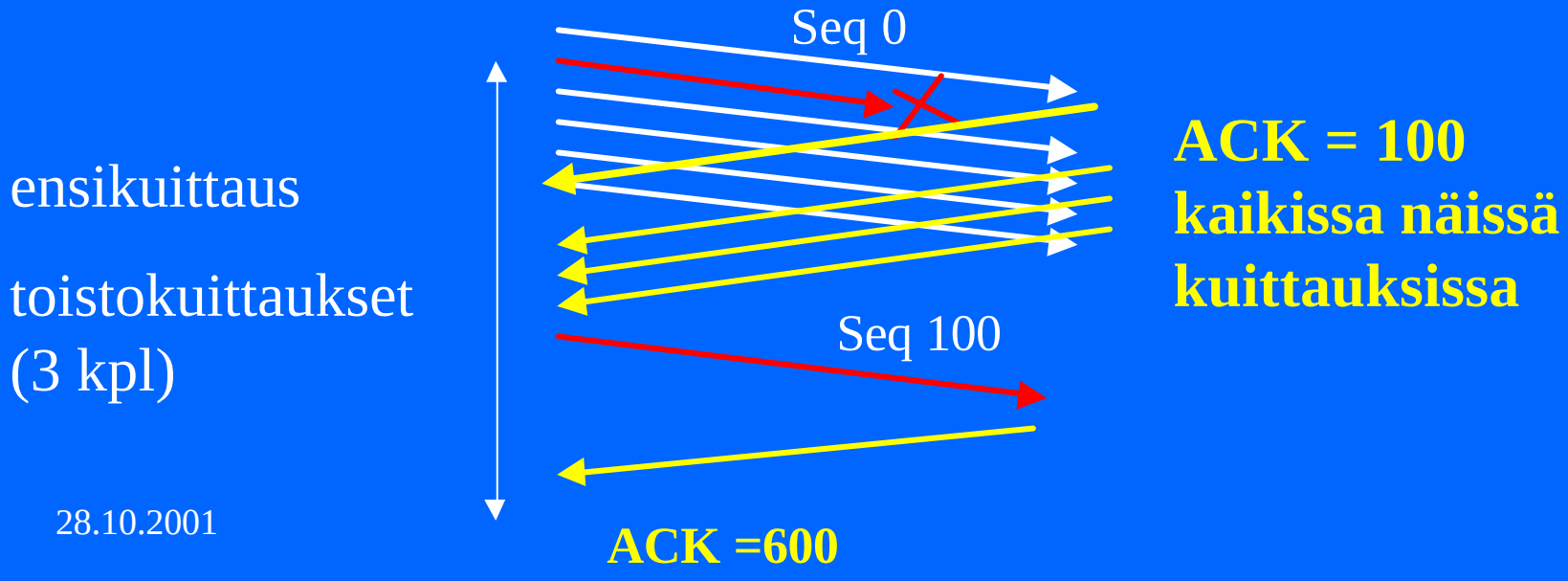
Ajastin laukeaa noin 400 ms kuluttua, jonka jälkee aloitetaan hidas aloitus!

“Duplicate Ack”

- ensikuittaus (first-time ACK)
  - segmentin ensimmäinen kuittaus
  - tähän saakka kaikki on kunnossa
- toistokuittaus (duplicate ACK)
  - vastaanottaja kuittaa viimeksi saatua hyväksyttyä segmenttiä aina kun saa virheellisen tai väärässä järjestyksessä tulevan segmentin
  - NAKin korvike, jolla ilmoitetaan ongelmista lähettäjälle

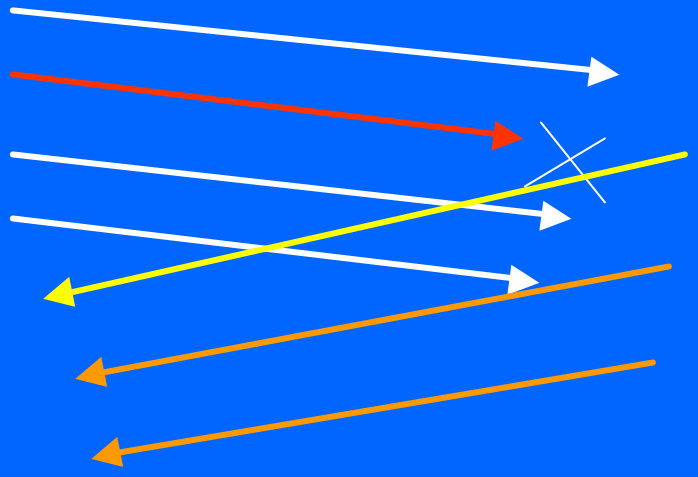
Nopea uudelleenlähetys (Fast retransmit)

- Kun lähettäjä vastaanottaa 3 toistokuittausta samalle segmentille, se lähettää heti puuttuvan segmentin uudestaan
  - eikä odota segmentin ajastimen laukeamista



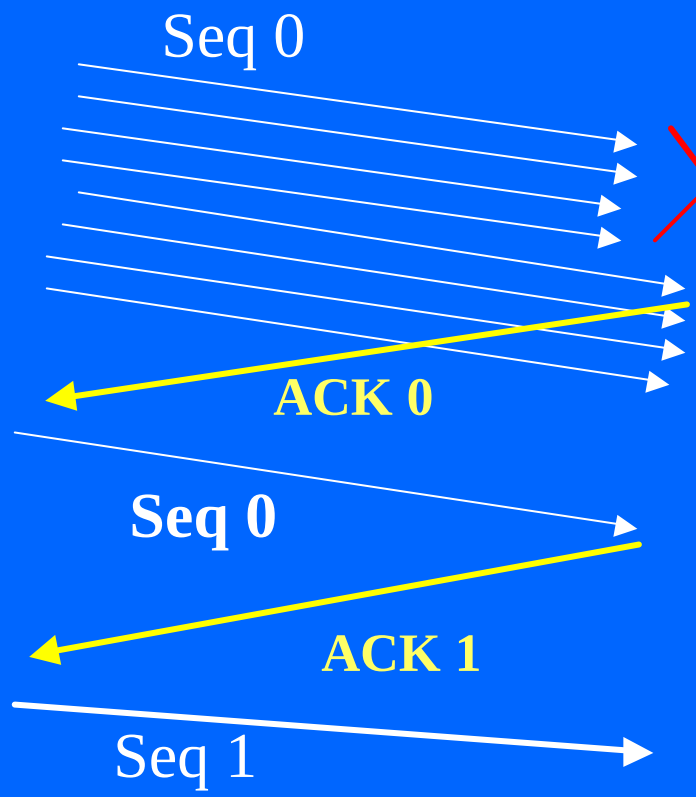
Ongelma 1: Ei saada kolmea toistokuittausta

- Jos ruuhkaikkuna on hyvin pieni,
  - ei voi tulla kolmea toistokuittausta, jos ruuhkaikkuna sallii vain kolme kuittaamatonta lähetystä



Ongelma 2: Virheryöppy tuhoaa monta segmenttiä

- Kun useita segmenttejä katoaa ‘samasta ikkunasta’



Segmentit hävivät

Joka kierroksella pystytään uudelleenlähettämään vain yksi kadonnut segmentti!!

Entä jos suuri kiertoviive

Limited Transmit

- RFC 3042: **Enhansing TCP's Loss Recovery Using Limited Transmit.**

  M. Allman, H. Balakrishnan, S. Floyd. January 2001

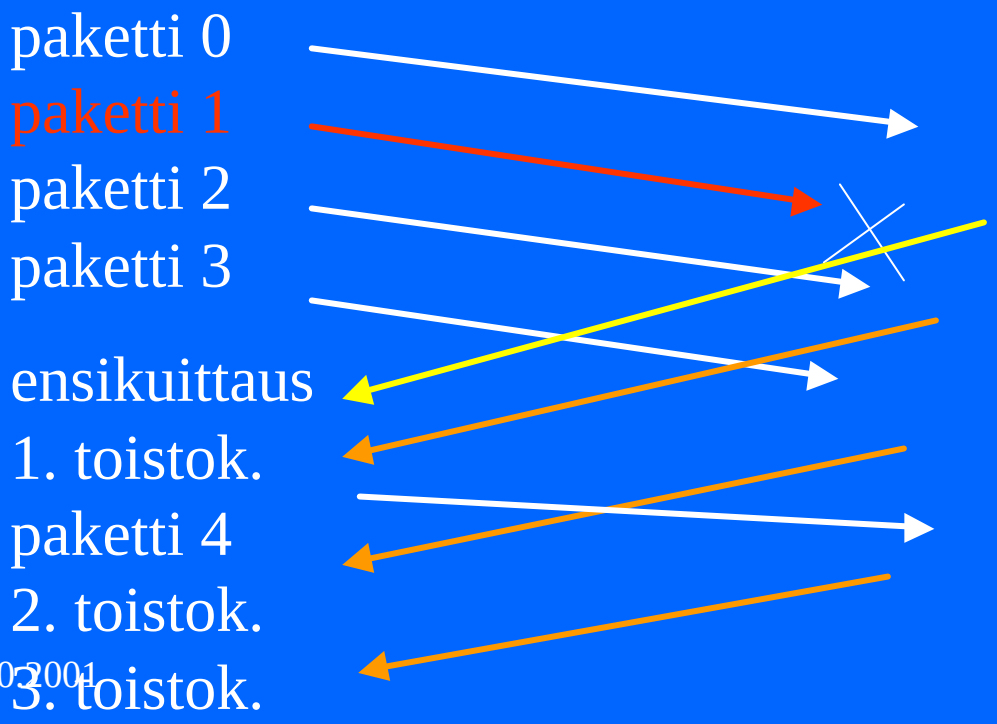
  (Status: PROPOSED STANDARD)

- Lähettäjä ei saa kolmea toistokuittausta =>
  - odotettava aina ajastimen laukeamista ja
  - suoritettava hidas aloitus
  - => hidastaa usein turhaan lähettämistä

Ratkaisu:

- Kun lähettäjä saa toistokuittauksen, se saa aina lähettää yhden **uuden paketin** verkkoon
  - » kuittaus kertoo, että verkosta poistettu paketti, joten verkkoon siis mahtuu!
- Kun saman paketin toistokuittauksia tulee kolme, niin suoritetaan nopea uudeelleenlähetys ja nopea toipuminen (fast recovery)

- Vaikka ruuhkaikkuna on pieni, niin rajoitetulla lähetyksellä saadaan tarvittaessa syntymään kolme toistokuittausta



Miksi lähetetään uusi paketti?

- Miksi ei heti ensimmäisen toistokuittauksen jälkeen lähetä uudestaan sitä jo lähetettyä kuittaamatonta pakettia?

- Koska ei vielä olla varmoja siitä, että paketti on todella kadonnut.
  - Se voi olla vain viivästynyt
  - tai paketit ovat matkalla joutuneet väärään järjestykseen
- => näin vältetään turhia uudelleenlähetyksiä