



Preddiplomski studij

Računarstvo

Komunikacijske mreže

7. Transportni sloj.

Transportni protokoli u Internetu:
TCP i UDP. (1. dio)

Ak.g. 2014./2015.



slobodno smijete:

- **dijeliti** — umnožavati, distribuirati i javnosti priopćavati djelo
- **remiksirati** — prerađivati djelo

pod sljedećim uvjetima:

- **imenovanje**. Morate priznati i označiti autorstvo djela na način kako je specificirao autor ili davatelj licence (ali ne način koji bi sugerirao da Vi ili Vaše korištenje njegova djela imate njegovu izravnu podršku).
- **nekomercijalno**. Ovo djelo ne smijete koristiti u komercijalne svrhe.
- **dijeli pod istim uvjetima**. Ako ovo djelo izmijenite, preoblikujete ili stvarate koristeći ga, preradu možete distribuirati samo pod licencom koja je ista ili slična ovoj.

U slučaju daljnjeg korištenja ili distribuiranja morate drugima jasno dati do znanja licencijske uvjete ovog djela. Najbolji način da to učinite je poveznicom na ovu internetsku stranicu.

Od svakog od gornjih uvjeta moguće je odstupiti, ako dobijete dopuštenje nositelja autorskog prava.

Ništa u ovoj licenci ne narušava ili ograničava autorova moralna prava.

Tekst licencije preuzet je s <http://creativecommons.org/>.

Zadaća transportnog sloja

- ◆ Transparentan prijenos transportnih jedinica podataka od izvora do odredišta
 - s kraja na kraj mreže (engl. *end-to-end*)
 - oslanja se na mrežni sloj, pruža (spojnu ili nespojnu) uslugu višem sloju (sjednički, prezentacijski, aplikacijski)

- ◆ prijenos uz zahtijevanu kvalitetu usluge:
 - pouzdana usluga - prijenos bez pogrešaka (dodatni mehanizmi za pouzdanost)
 - prijenos uz najmanje kašnjenje (minimalno procesiranje)

◆ Usluga transportnog sloja

◆ Funkcionalnost

- adresiranje
- multipleksiranje
- uspostava i raskid veze
- kontrola toka i privremena pohrana
- oporavak od prekida

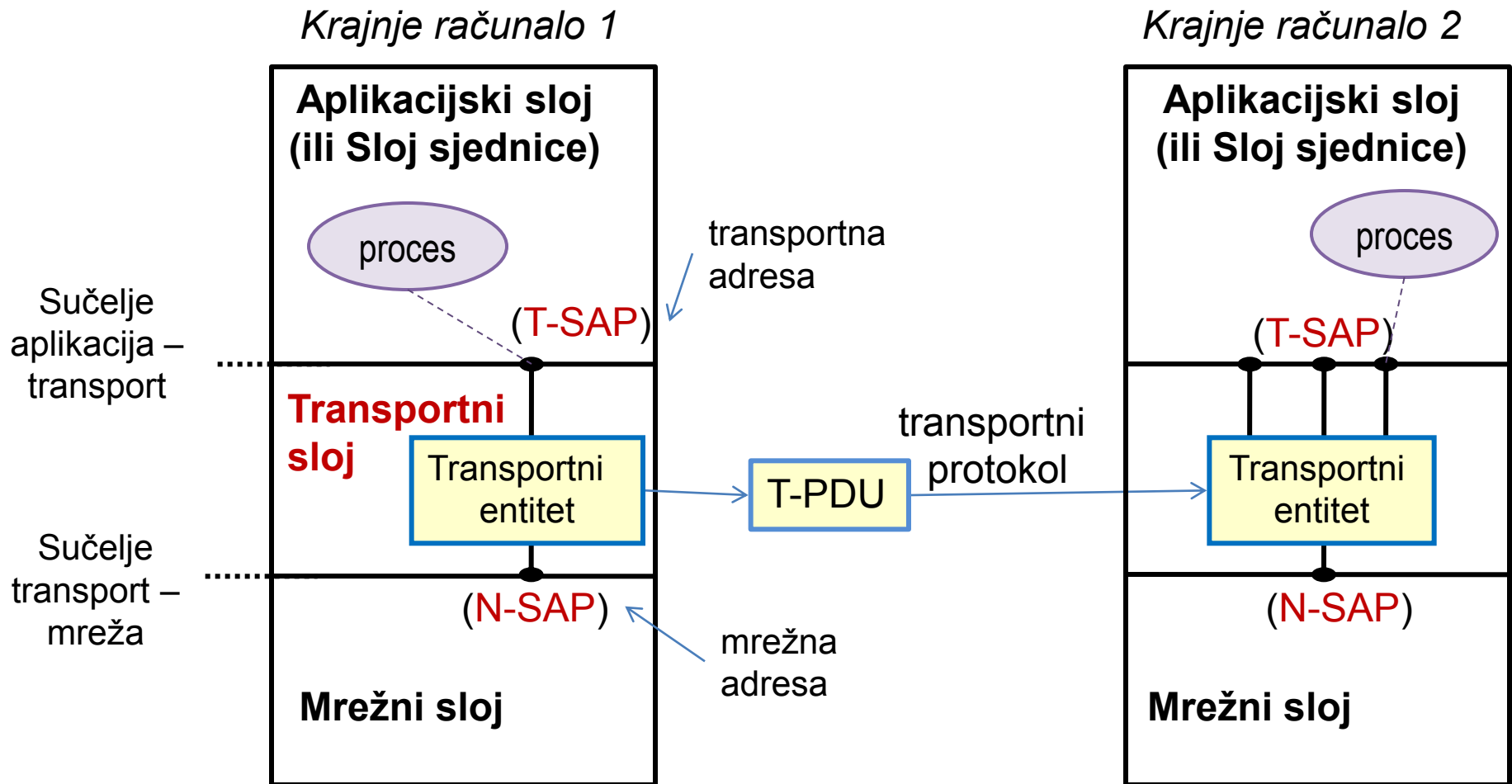
◆ Protokoli transportnog sloja u Internetu

- Transmission Control Protocol
- User Datagram Protocol

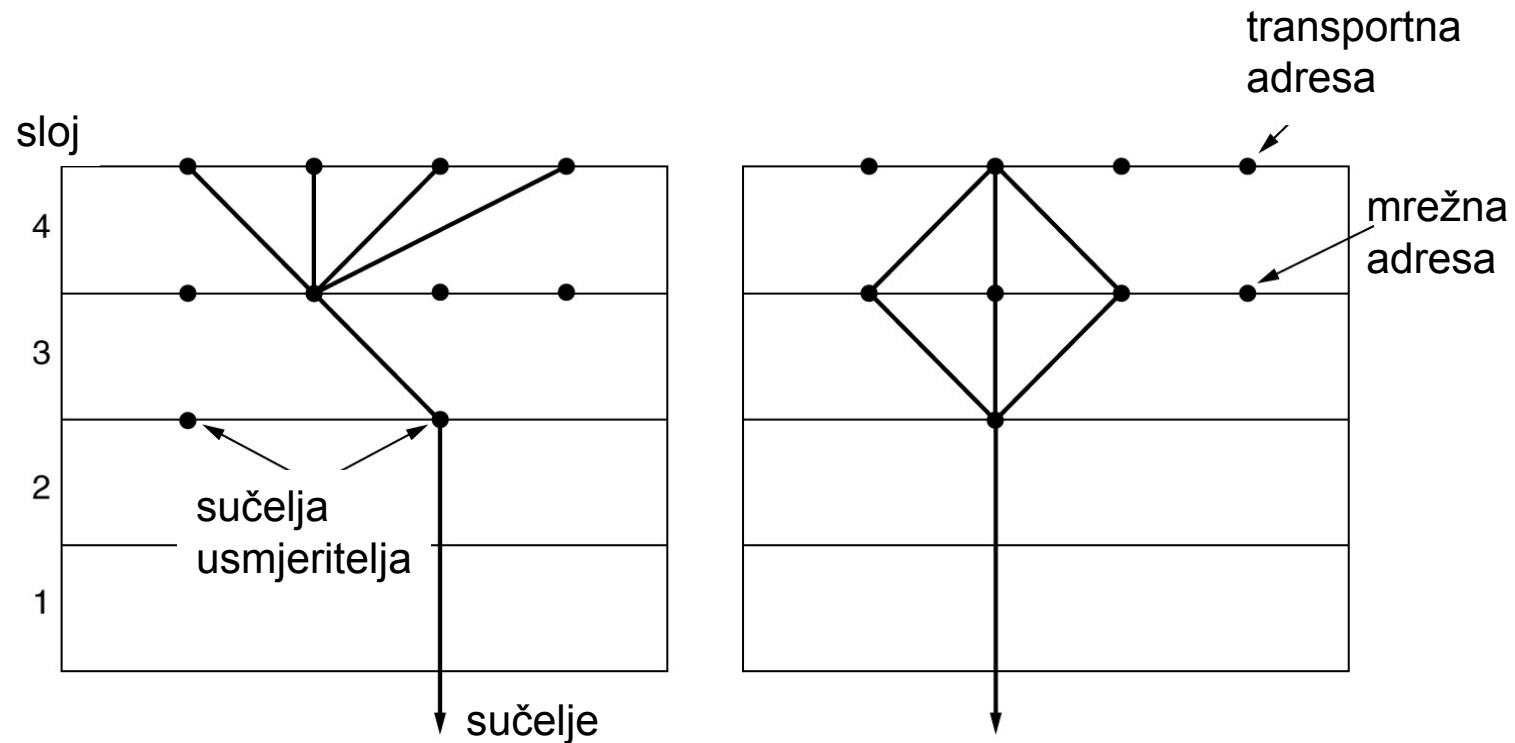
- ◆ svrha: omogućiti **logičko** povezivanje procesa na krajnjim računalima
- ◆ usluga može biti spojna i nespojna
- ◆ funkcije:
 - adresiranje (na razini transportnog sloja)
 - multipleksiranje
 - uspostava i raskid veze (za spojnu uslugu)
 - kontrola toka
 - oporavak od prekida komunikacije
- ◆ “kompenzacija nedostataka” mrežnog sloja
- ◆ izbor transportnog protokola ovisit će o parametrima kvalitete usluge koje zahtijeva aplikacija!

Logičko povezivanje procesa

logička veza – procesi koji komuniciraju ponašaju se kao da su izravno spojeni



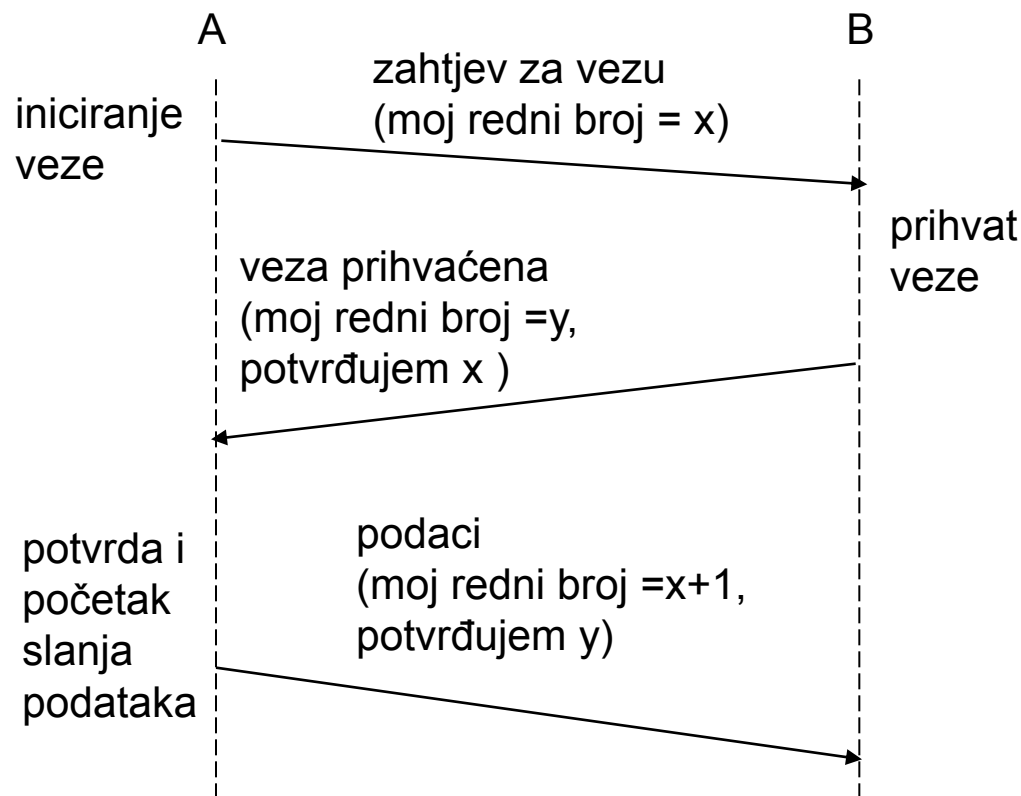
- ◆ na sučelju transporta i mreže: mrežna pristupna točka usluzi – N-SAP (Network-Service Access Point)
 - adresa mrežnog sučelja
 - u internetskom modelu: IP adresa
- ◆ na sučelju transporta i aplikacije: transportna pristupna točka usluzi – T-SAP (Transport-Service Access Point)
 - adresa transportnog entiteta
 - u internetskom modelu: vrata (engl. *port*)
- krajnje točke logičke veze:
(IP adresa izvora, vrata na izvoru) --- (IP adresa odredišta, vrata na odredištu)
 - u internetskom modelu: priključnica (engl. *socket*)
- ◆ multipleksiranje: način preslikavanja T-SAP:N-SAP
 - odozgo, n:1
 - odozdo, 1:m



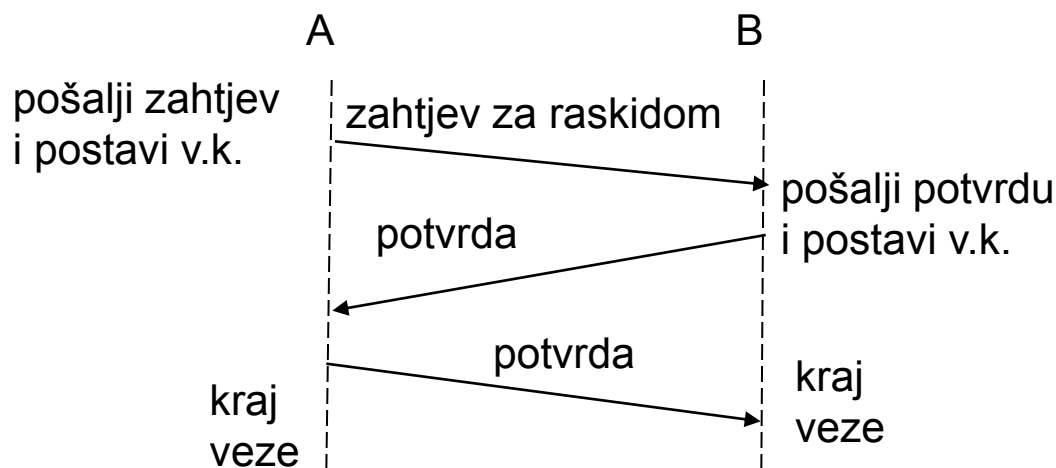
- ◆ multipleksiranje odozgo:
više procesa komunicira preko iste
mrežne adrese
- npr: TCP, UDP

- ◆ multipleksiranje odozdo:
proces otvara više mrežnih veza i
šalje podatke naizmjenice po njima
- npr.: mTCP i SCTP

- ◆ kontrolne poruke: zahtjevi i odgovori u zadanom (dogovorenom) obliku
- ◆ mehanizmi:
 - ◆ numeracija poruka
 - ◆ (pozitivne) potvrde
 - ◆ negativne potvrde
 - ◆ vremenska kontrola
 - ◆ klizeći prozor
- ◆ vremenska kontrola (v.k.) nužna za slučajeve ispada ili gubitka zahtjeva/ potvrde



- ♦ različiti scenariji urednog zatvaranja veze
- ♦ primjer:



- ♦ nužna vremenska kontrola (v.k.) za slučajeve ispada ili gubitka zahtjeva/ potvrde

- ◆ kontrola toka – sličan problem kao u sloju podatkovne poveznice:
 - usklađivanje pošiljaateljeve brzine slanja i brzine primanja/obrade na strani primatelja
 - najčešći model kontrole toka – klizeći prozor!

- ◆ privremena pohrana – zbog nepouzidane dostave i moguće promjene redosljeda datagrama
 - treba privremeno pohranjivati T-PDU
 - pohrana i prilikom slanja, i prilikom primanja!
 - rješenje: usklađivanje veličine klizećeg prozora i dinamičko rukovanje memorijskim spremnikom

- ◆ prekidi se događaju - moguća mjesta kvara:
 - krajnji uređaji
 - usmjeritelji

- ◆ kvar na usmjeritelju i krajnje točke sa očuvanim stanjem transportne veze – jednostavan oporavak

- ◆ ako je kvar na krajnjem uređaju - što se događa s transportnom vezom nakon ponovnog pokretanja?
 - teži problem, stanje nije sačuvano
 - općeniti zaključak: nemoguće je sasvim prikriti kvar od viših slojeva, ali vrijedi - prekid na sloju N može ispraviti sloj N+1 pod pretpostavkom da krajnje točke “znaju gdje su stale”

Izbor transportnog protokola ovisi o parametrima kvalitete usluge (pouzdanost, kašnjenje, itd.) koje zahtijeva aplikacija!

Primjeri aplikacija – zahtjevnost /oštrina kriterija

Aplikacija	Pouzdanost	Kašnjenje	Kolebanje kašnjenja	Širina pojasa
Elektronička pošta	Visoki	Niski	Niski	Niski
Transfer datoteka	Visoki	Niski	Niski	Srednji
Pristup Webu	Visoki	Srednji	Niski	Srednji
Rad na daljinu	Visoki	Srednji	Srednji	Niski
Audio na zahtjev	Niski	Niski	Visoki	Srednji
Video na zahtjev	Niski	Niski	Visoki	Visoki
Telefonija	Niski	Visoki	Visoki	Niski/Srednji
Videokonferencija	Niski	Visoki	Visoki	Visoki

- ◆ dvosmjerna komunikacija
 - sposobnost istovremenog slanja i primanja

- ◆ pouzdanost transporta
 - detekcija gubitka paketa i eventualna reakcija

- ◆ transfer poruka ili niza okteta
 - dvije mogućnosti tretiranja podataka: kao blokovi/poruke, ili kao niz okteta

- ◆ očuvanje redoslijeda podataka
 - rekonstrukcija izvornog redoslijeda poruka ili okteta na odredištu za slučaj narušavanja redoslijeda pri prolasku kroz mrežu

- ◆ kontrola toka
 - usklađivanje brzina slanja i primanja podataka između krajnjih točaka (procesa)

- ◆ Usluga transportnog sloja
- ◆ Funkcionalnost
 - adresiranje
 - multipleksiranje
 - uspostava i raskid veze
 - kontrola toka i privremena pohrana
 - oporavak od prekida
- ◆ Protokoli transportnog sloja u Internetu
 - Transmission Control Protocol
 - User Datagram Protocol

4 Aplikacijski sloj
3 Transportni sloj
2 Mrežni/internetski sloj
1

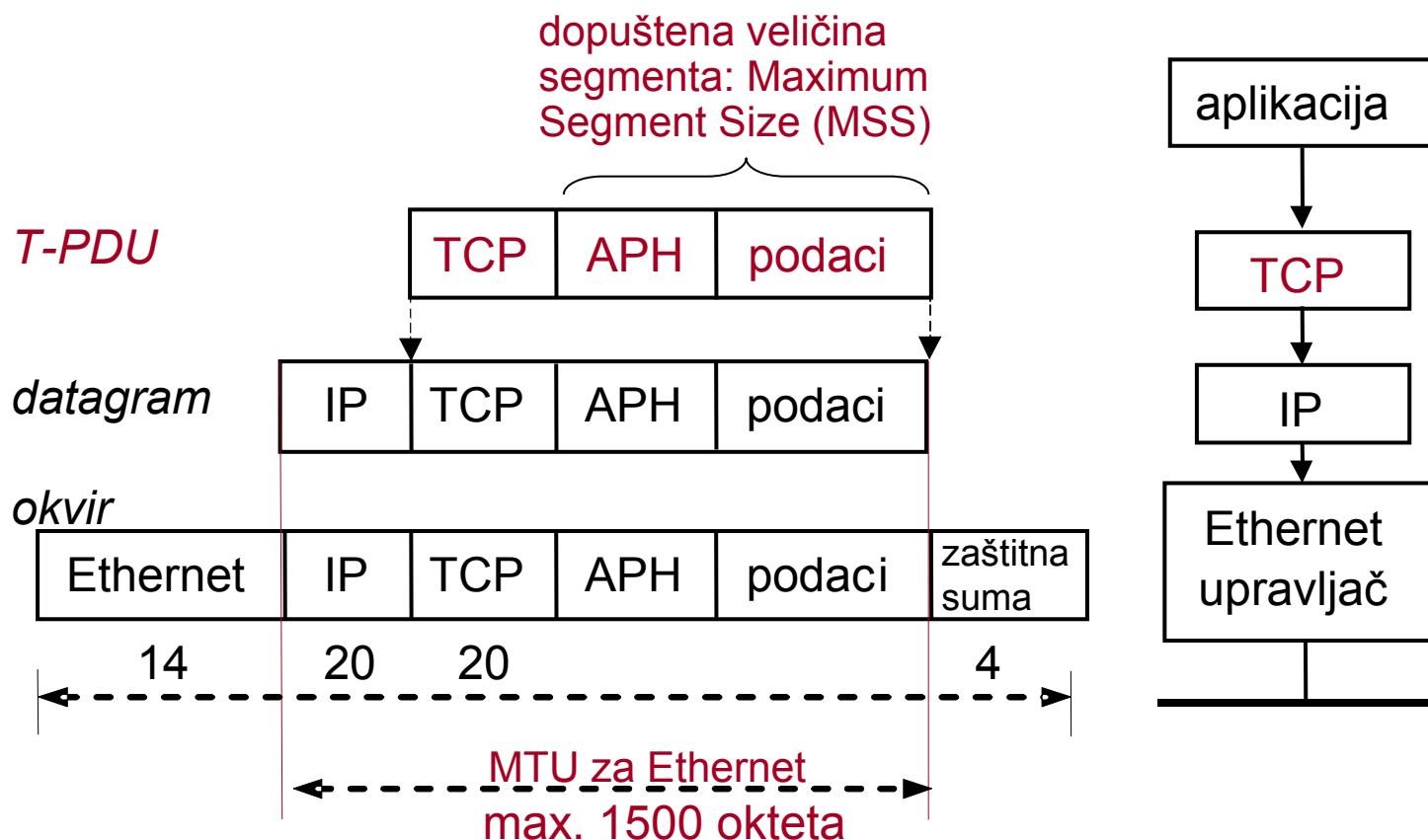
- ◆ transmisijski kontrolni protokol (*Transmission Control Protocol*, **TCP**)
 - pouzdana transportna usluga: prijenos niza okteta bez pogrešaka, uz isporuku potpune informacije u nepromijenjenom redoslijedu
- ◆ Korisnički datagramski protokol (*User Datagram Protocol*, **UDP**)
 - jednostavna transportna usluga: prijenos uz najmanje moguće kašnjenje informacije

- ◆ TCP je spojno-orijentirani, pouzdani internetski protokol transportnog sloja
 - TCP pruža spojnu uslugu transporta struje okteta povrh nespojnog IP-a
 - uspostavlja logičku vezu između procesa na krajnjim računalima
 - osigurava pouzdan transport s kraja na kraj pomoću mehanizama potvrde i retransmisije, uz očuvani redoslijed struje okteta i upravljanje transportnom vezom.
 - logička veza između procesa definirana je parom 16-bitnih transportnih adresa, koje se u internetskoj terminologiji nazivaju vrata (engl. *port*).
 - TCP PDU naziva se (TCP) segment.

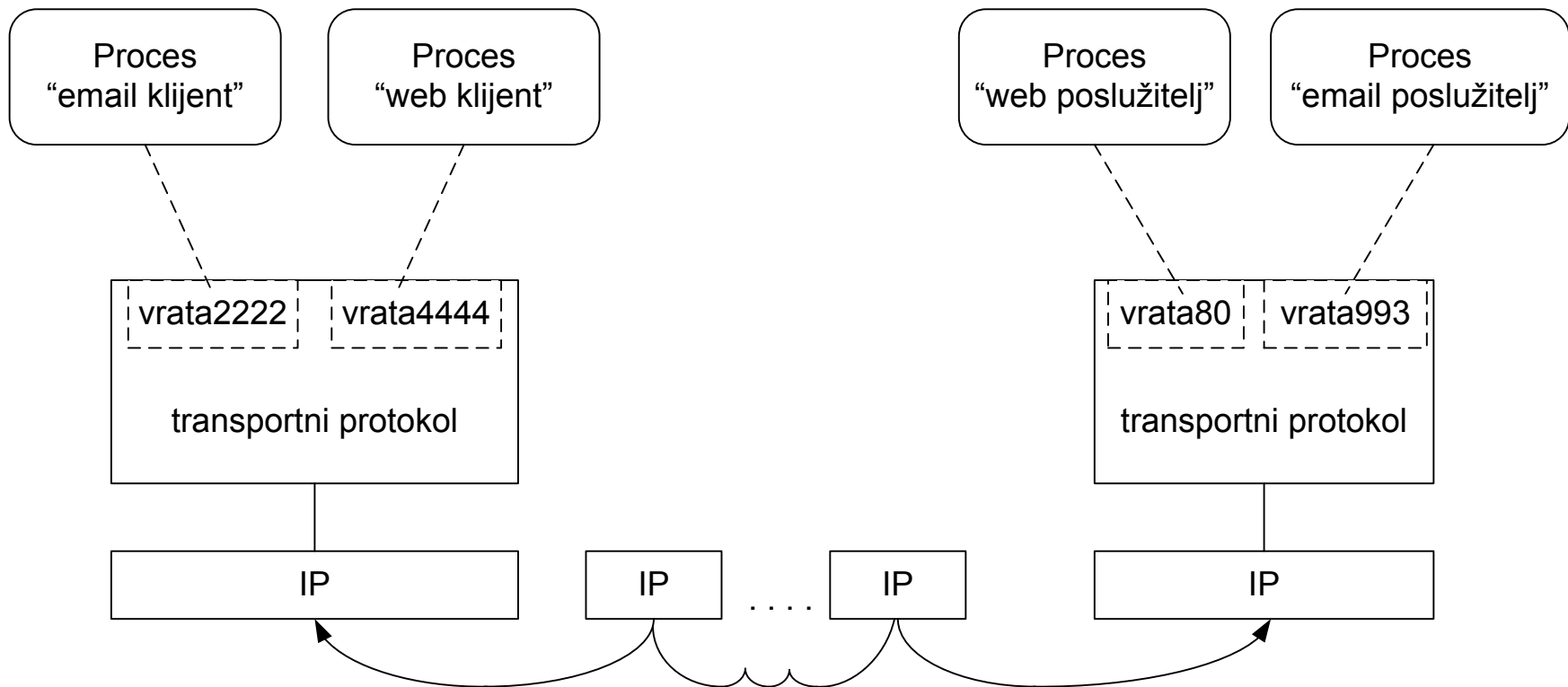
◆ osnovne funkcije:

- osnovni transport podataka
- adresiranje i multipleksiranje
- pouzdanost
- upravljanje logičkom vezom
- upravljanje tokom
- prioritet i sigurnost (logičke veze, ne podataka!)

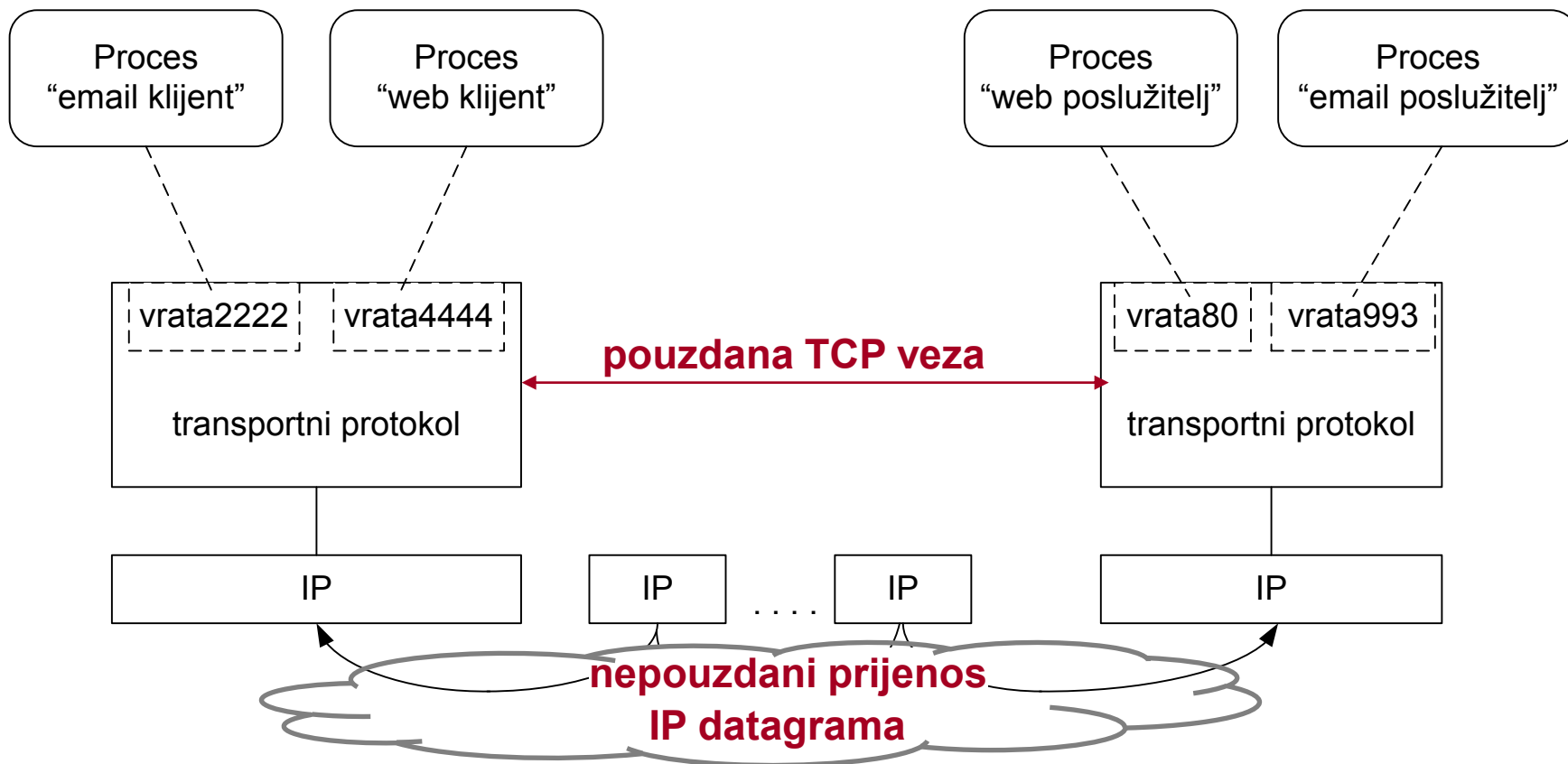
- dvosmjerni transport kontinuiranog niza podataka, pakiranjem okteta podataka u *segmente*, koje potom predaje protokolu mrežnog sloja



- ◆ raščlanjivanje tokova podataka koji pripadaju različitim procesima
- ◆ Krajnje točke komunikacije:
<IP adresa izvora, vrata na izvoru, IP adresa odredišta, vrata na odredištu>



- ◆ pouzdana TCP veza preko nepouzdanog datagramskog mrežnog sloja



■ mehanizmi koji osiguravaju pouzdanost:

- detekcija pogrešaka
- retransmisija
- kumulativna potvrda
- vremenska kontrola
- polja u zaglavlju koja služe za oznake segmenta u nizu i oznake potvrda

■ osnovna ideja:

- svaki segment je **numeriran** →
- primatelj potvrđuje primljene segmente
- potvrda je kumulativna – potvrđuje sve oktete do onog na kojeg se potvrda odnosi
- pošiljatelj postavlja vremensku kontrolu prilikom slanja segmenta
- ako do isteka vremenske kontrole ne primi potvrdu, pošiljatelj smatra segment izgubljenim

važno! numeriraju se segmenti, ali taj broj NIJE redni broj segmenta, nego redni broj **okteta** koji je **prvi u promatranom segmentu!**

Numeracija segmenata - primjer

Primjer:

podaci aplikacijskog sloja: 6027 okteta , duljina podatkovnog polja = 1460 okteta

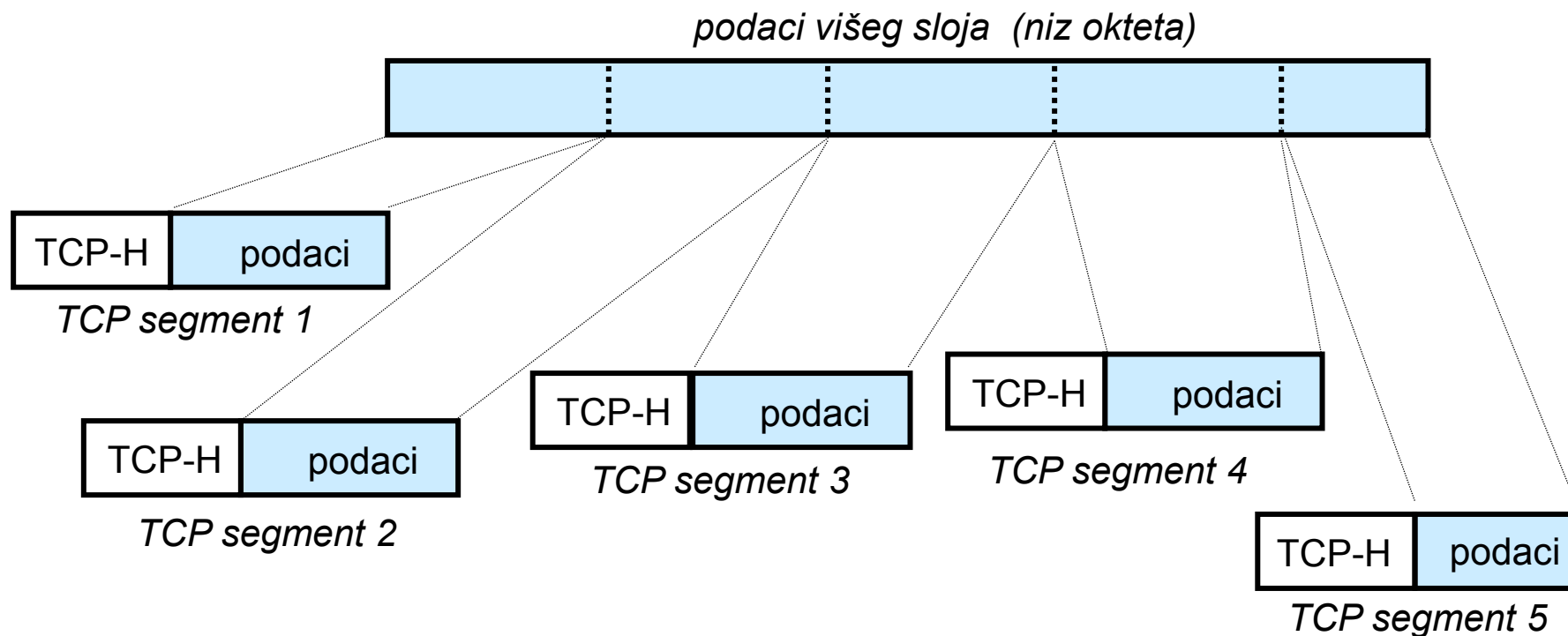
segment 1: okteti 0-1459 (duljina 1460) -> redni broj segmenta = x (početni broj)

segment 2: okteti 1460-2919 (duljina 1460) -> redni broj segmenta = x+1460

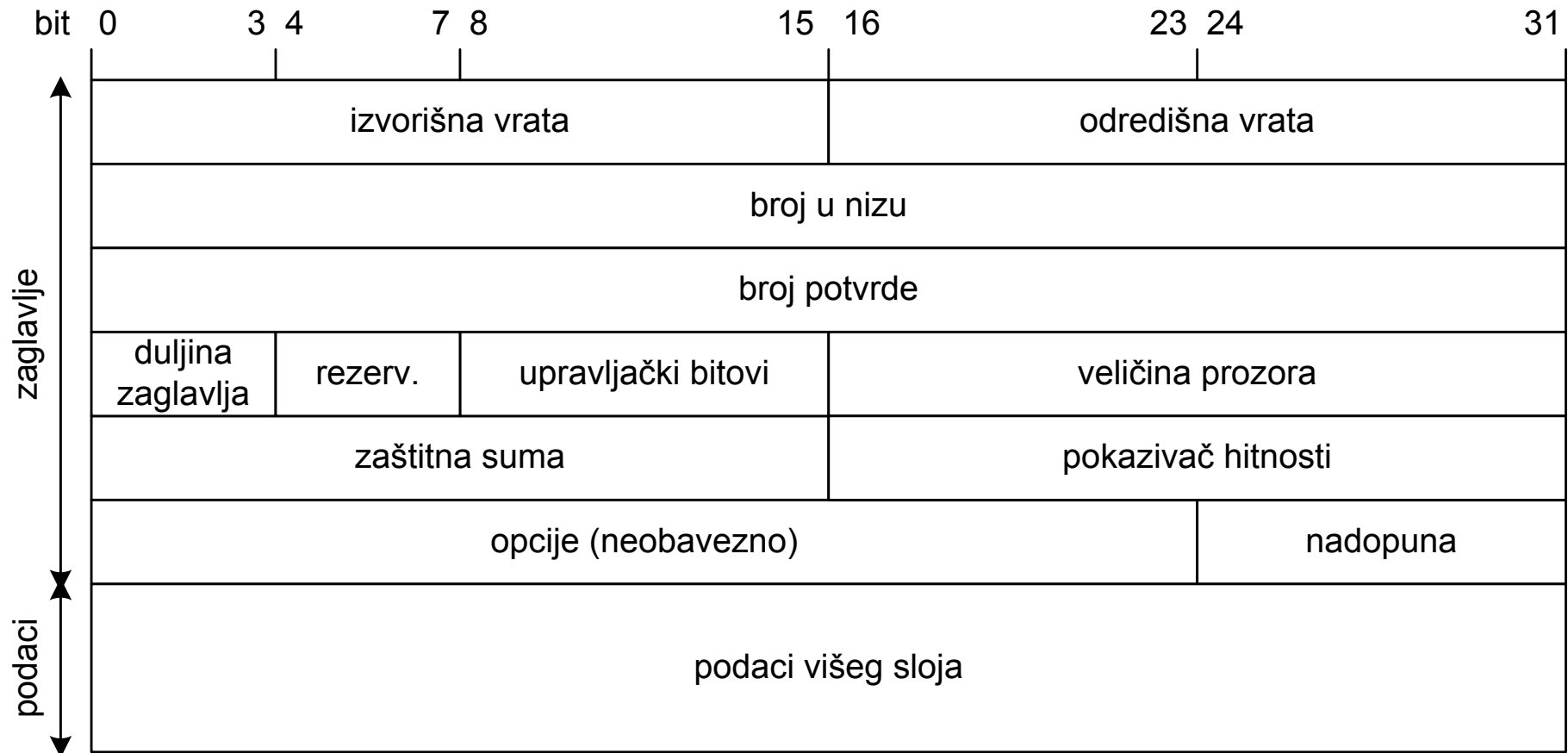
segment 3: okteti 2920-4379 (duljina 1460) -> redni broj segmenta = x+2920

segment 4: okteti 4380-5839 (duljina 1460) -> redni broj segmenta = x+4380

segment 5: okteti 5840-6186 (duljina 367) -> redni broj segmenta = x+5840



Struktura TCP-segmenta



upravljački bitovi

U	A	P	R	S	F
R	C	S	S	Y	I
G	K	H	T	N	N

- ◆ uspostava i raskid veze:
 - ◆ SYN (od engl. *synchronize*)
 - ◆ FIN (od engl. *finish*)
- ◆ potvrda:
 - ◆ ACK (od engl. *acknowledgement*)
- ◆ prioritetni podaci:
 - ◆ URG (od engl. *urgent*)
 - ◆ PSH (od engl. *push*)
- ◆ poništavanje veze:
 - ◆ RST (od engl. *reset*)

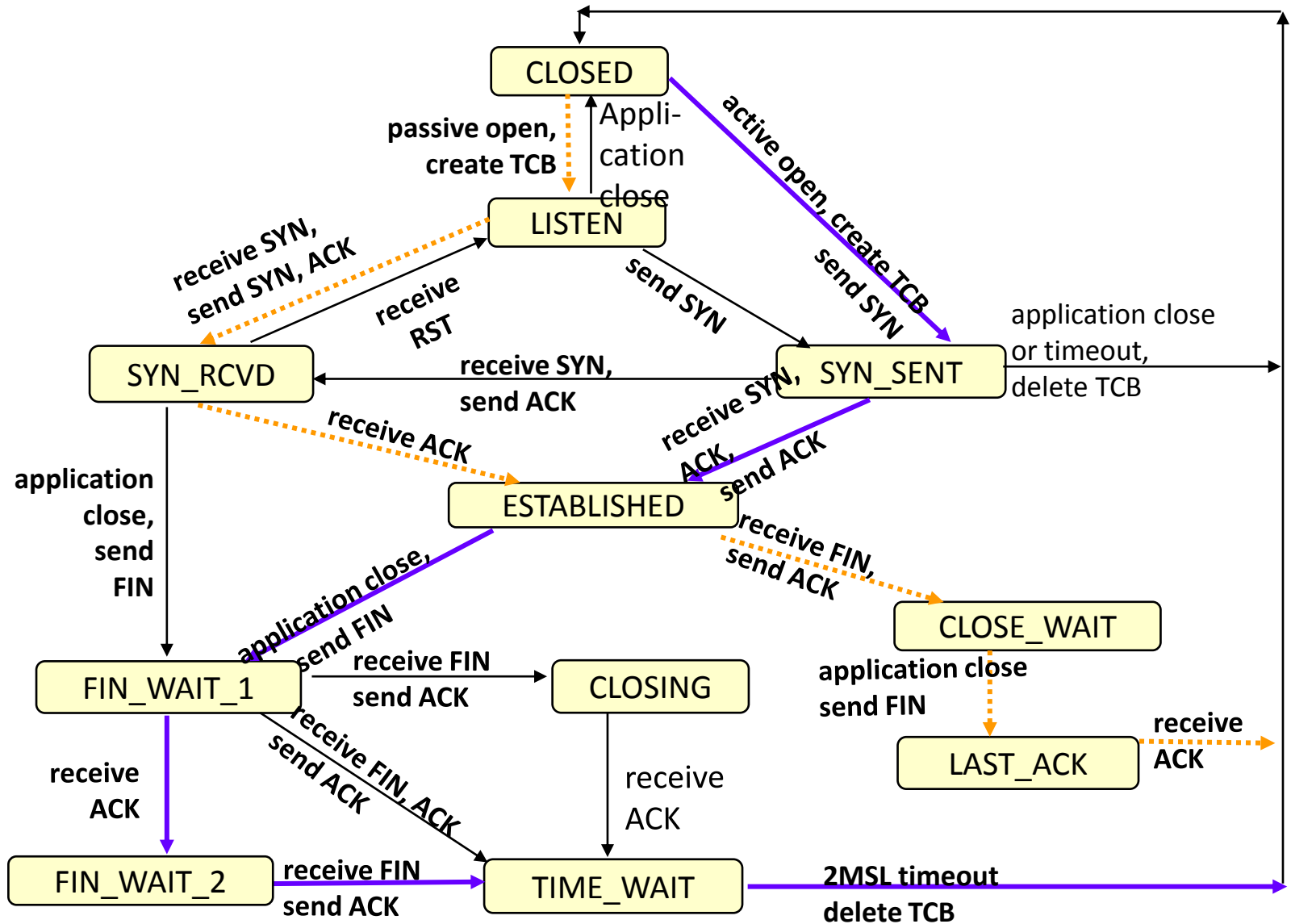
- ◆ Da bi ostvario svoje funkcije, TCP “pamti” niz parametara vezanih uz svaki par komunicirajućih procesa, odn. uz stanje logičke veze
 - ◆ ponašanje se može opisati automatom stanja
 - ◆ automat stanja opisuje jedan kraj veze, odn. jedan TCP entitet (instancu TCP automata stanja na strani klijenta i li na strani poslužitelja) – uočimo da oni ne moraju biti u istom stanju!
 - ◆ stanje – karakterizira ga skup internih varijabli procesa
 - ◆ prijelaz – karakteriziraju ga ulazni događaj koji izaziva promjenu stanja i izlazni signal ili poruka prema drugoj krajnjoj točki

◆ Stanja

- Listen
 - Syn_Rcvd
 - Syn_Sent
 - Established
 - Close_wait
 - Last_ack
 - Fin_wait1
 - Fin_wait2
 - Time_wait
 - Close_wait
- uspostavljanje veze
- veza uspostavljena
- čekanje na završetak
pasivno zatvaranje (poslužitelj)
- čekanje na završetak
aktivno zatvaranje (klijent)

◆ Prijelazi – TCP segmenti (upravljačke poruke ili podaci)

Automat stanja - TCP

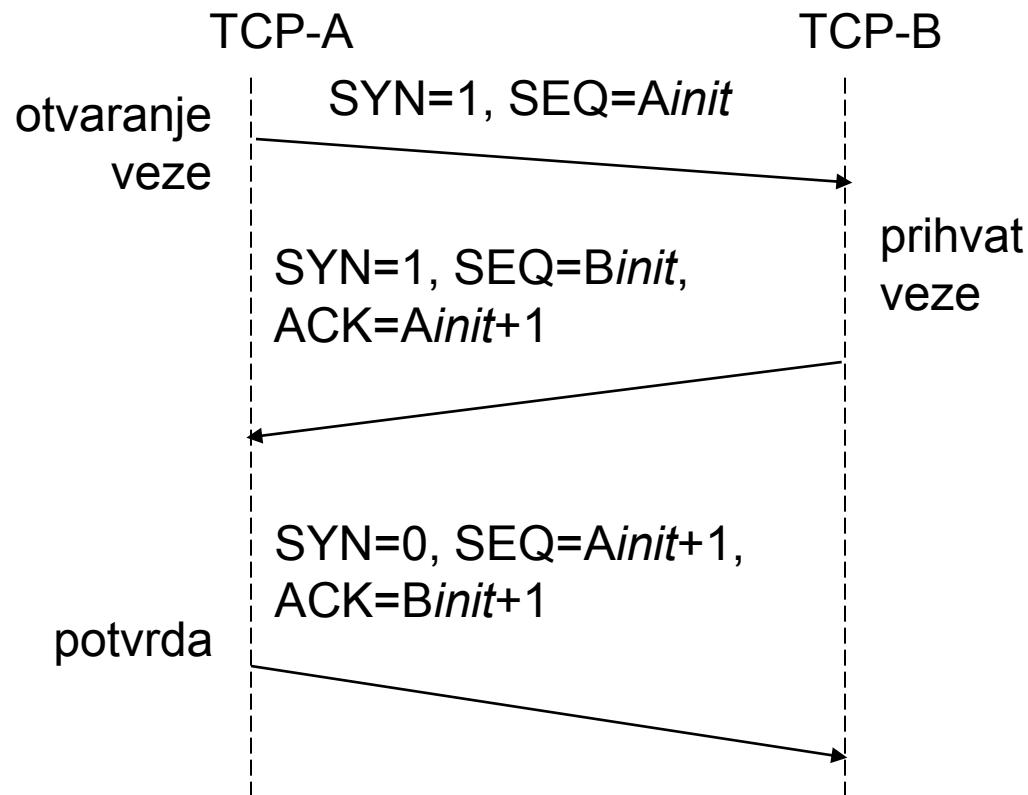


◆ Procedura uspostave veze:

- ◆ svaka od strana mora poslati tzv. **SYN segment** u kojem objavljuje parametre bitne za vezu, npr.:
 - početni broj za numeraciju okteta
 - veličinu prozora
 - izvorišna i odredišna vrata
 - (može biti i drugih parametara, u opcijama)
- strana koja primi SYN segment mora poslati potvrdu (ACK) da ga je primila

Uspostava TCP-veze

- prilikom uspostave veze, inicijaliziraju se parametri komunikacije



- nakon uspješne uspostave veze, slijedi razmjena podataka aplikacijskih procesa

Transport niza podatkovnih okteta - primjer

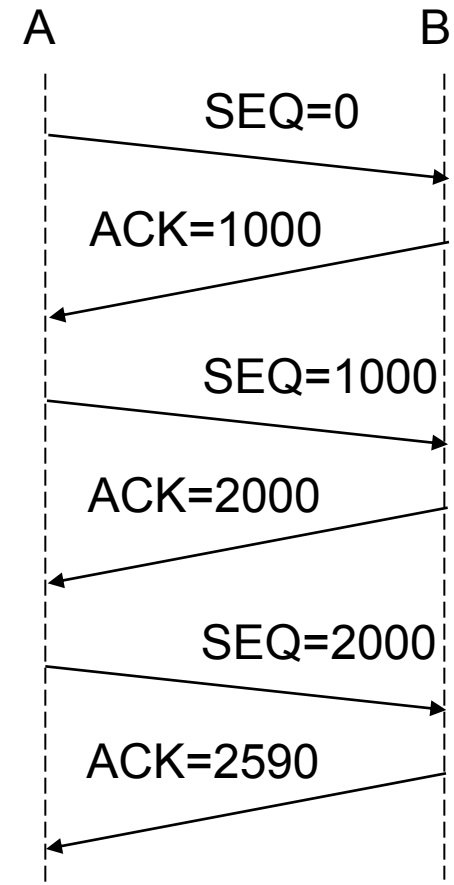
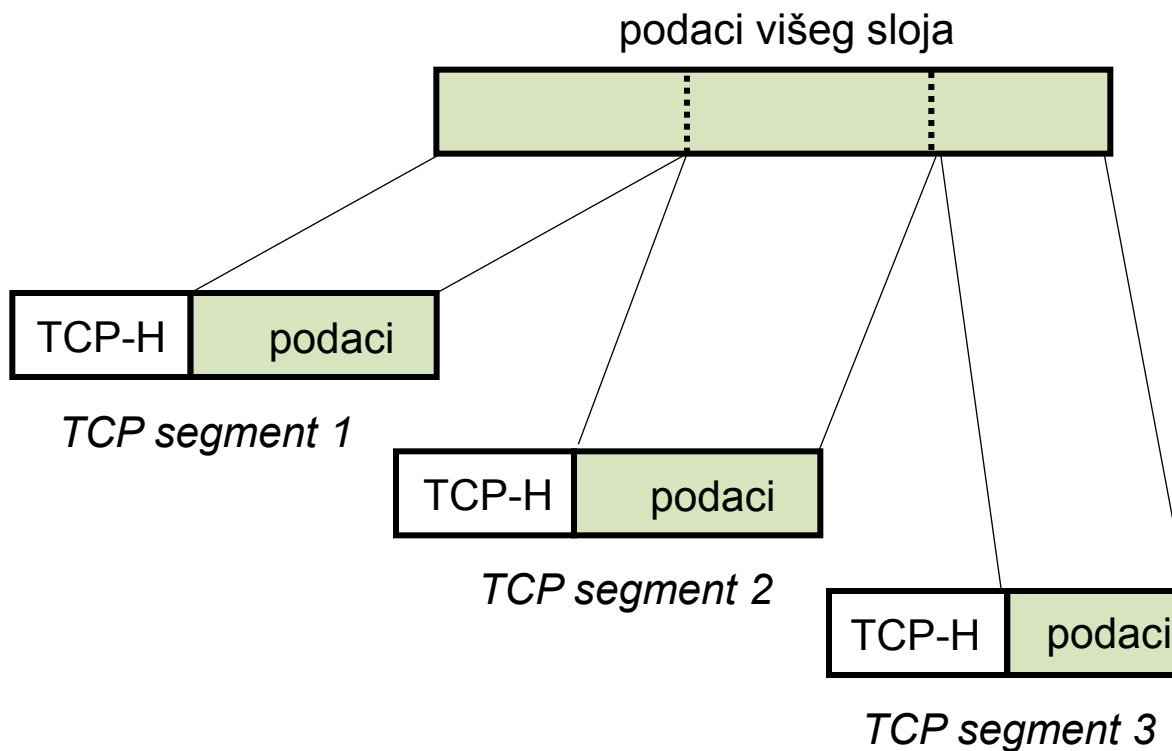
podaci 2590 okteta

MSS = 1000 okteta

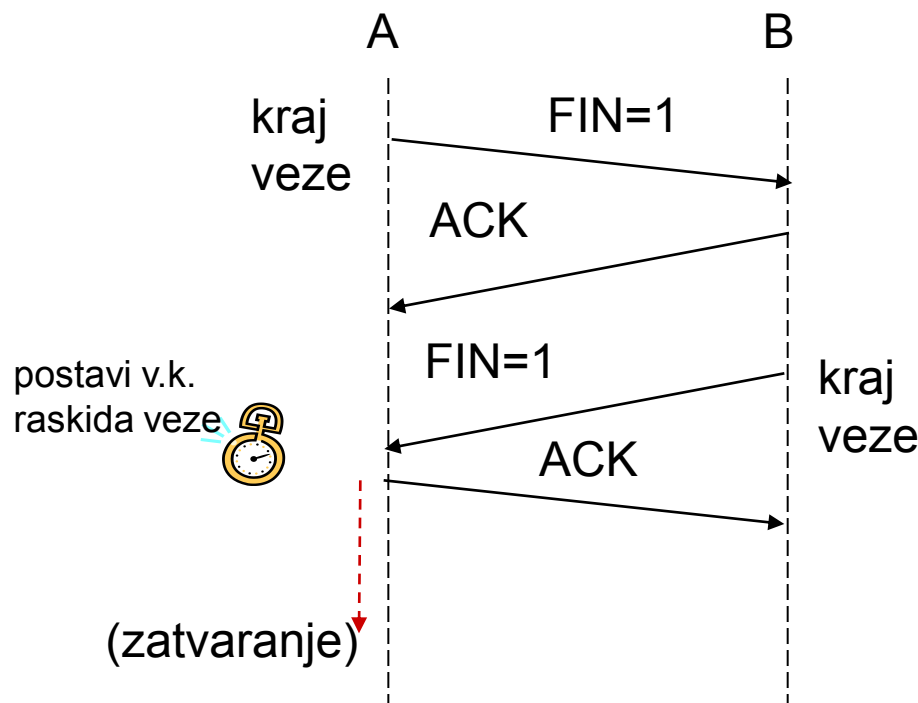
segment 1: okteti 0-999 -> redni broj = 0

segment 2: okteti 1000-1999 -> redni broj = 1000

segment 3: okteti 2000-2589 -> redni broj = 2000



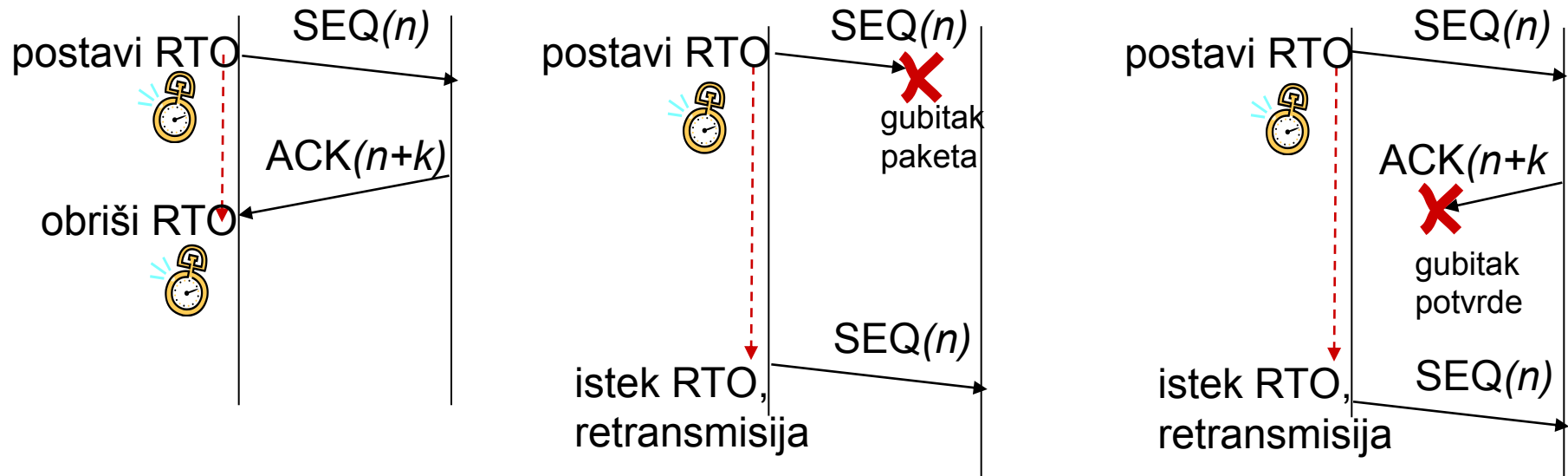
- kad aplikacije nemaju više podataka za poslati, započinju “dogovor” o raskidu veze



- ◆ Segmenti se u mreži mogu izgubiti iz više razloga
 - greške u prijenosu koje se ne daju ispraviti
 - usmjerivači ispuštaju pakete pri preopterećenosti linkova
 - petlje u usmjeravanju i TTL mehanizam

- ◆ TCP garantira da će poduzeti sve mjere da nadomjesti oktete izgubljene u mreži
 - ponavljanje slanja nakon određenog vremenskog perioda (*retransmission timeout, RTO*)
 - eksponencijalno produljivanje vremenskog intervala za ponavljanje slanja (*exponential backoff*)

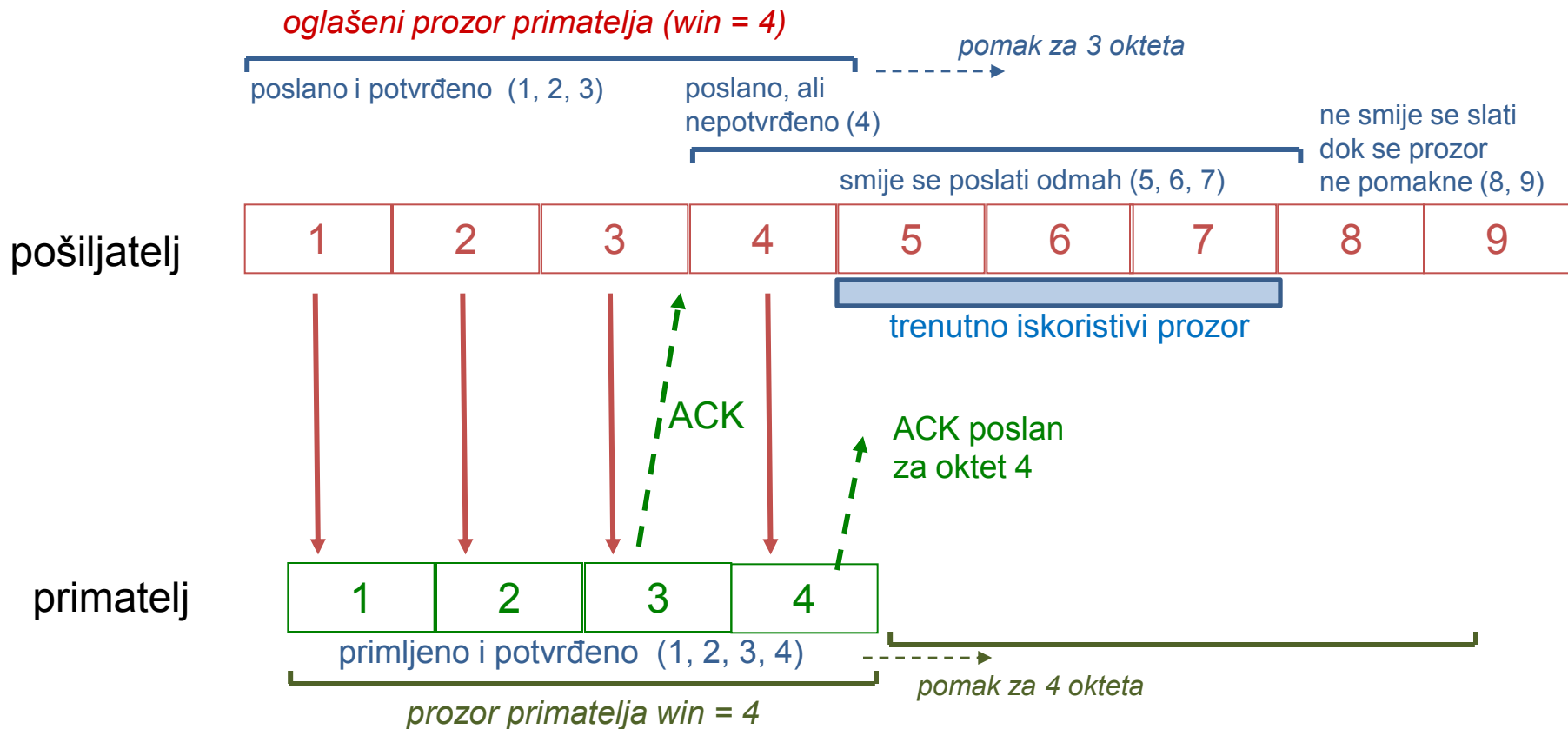
- ◆ TCP pošiljatelj postavlja RTO prilikom slanja segmenta
- ◆ ako potvrda za segment ne stigne do trenutka isteka RTO, pošiljatelj smatra segment izgubljenim i šalje ga ponovno



(promatramo segment koji počinje na oktetu n i ima duljinu k okteta)

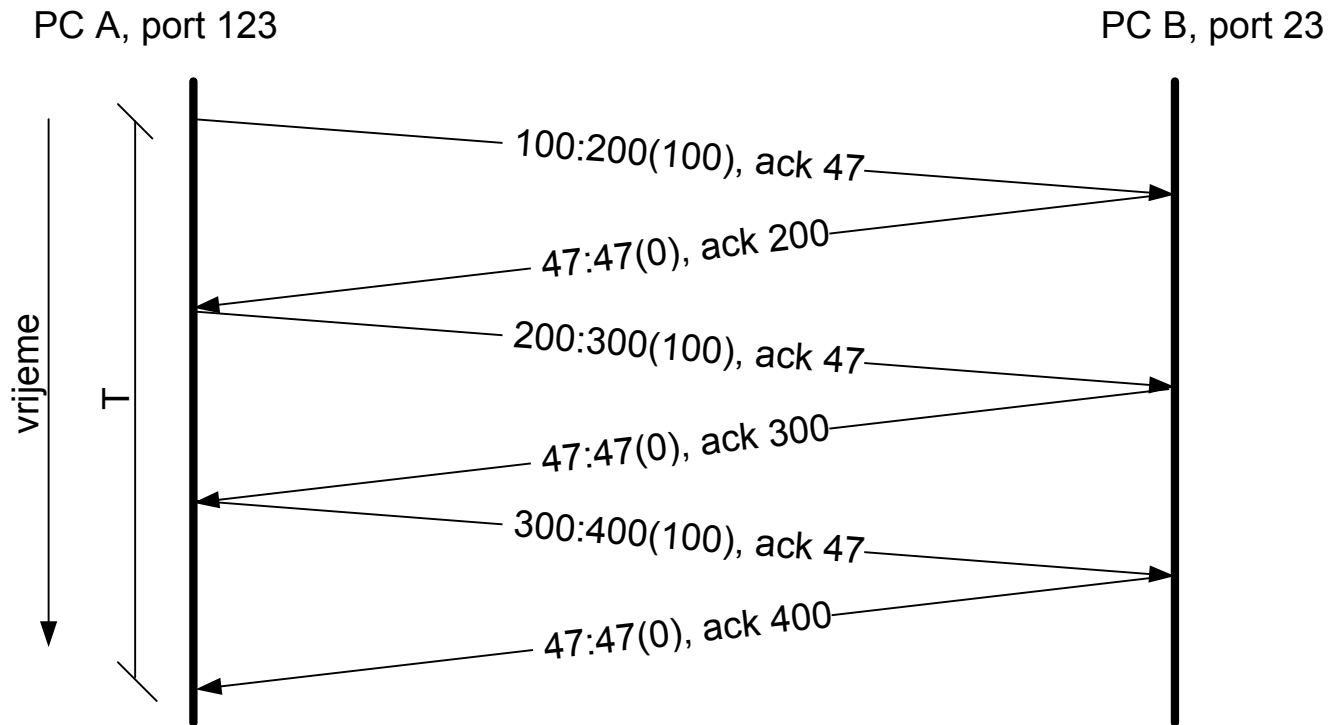
- ◆ Usklađivanje brzine slanja i primanja
 - što ako pošiljalac šalje brže nego što primatelj može obrađivati?
- ◆ TCP koristi mehanizam **klizećeg prozora**
 - prozor je broj okteta koje pošiljalac u danom trenutku smije poslati a da ne čeka potvrdu bilo kojeg od tih okteta
 - prozor objavljuju obje strane i to u sklopu potvrda koje šalju
- ◆ Primjer
 - pretpostavimo da je pošiljalac primio potvrdu za oktet $N-1$ i u potvrdi je objavljen prozor W
 - pošiljalac smije poslati sve oktete do $N+W$ bez ikakve daljnje potvrde
 - međutim, ne smije poslati oktete od $N+W$ nadalje sve dok mu se barem jedan dio okteta na “donjoj granici” prije $N+W$ ne potvrdi

- ◆ prilagodba brzine slanja prema brzini primanja primatelja i stanju u mreži
- ◆ **klizeći prozor** - veličinom prozora je definiran max. broj nepotvrđenih okteta koje pošiljalatelj može poslati odjednom



◆ Primjer:

Oznake: od:do(duljina), ack “oktet koji se potvrđuje”



Je li moguće prenijeti segmente sa slike u još kraćem vremenu?