

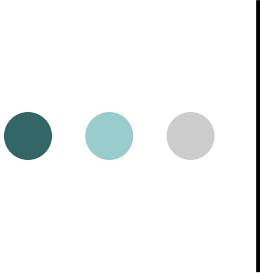


Programiranje korisničkih interfejsa: **Lekcija 2: Internet servisi(I)**

zima 2019/2020

Branimir M. Trenkić

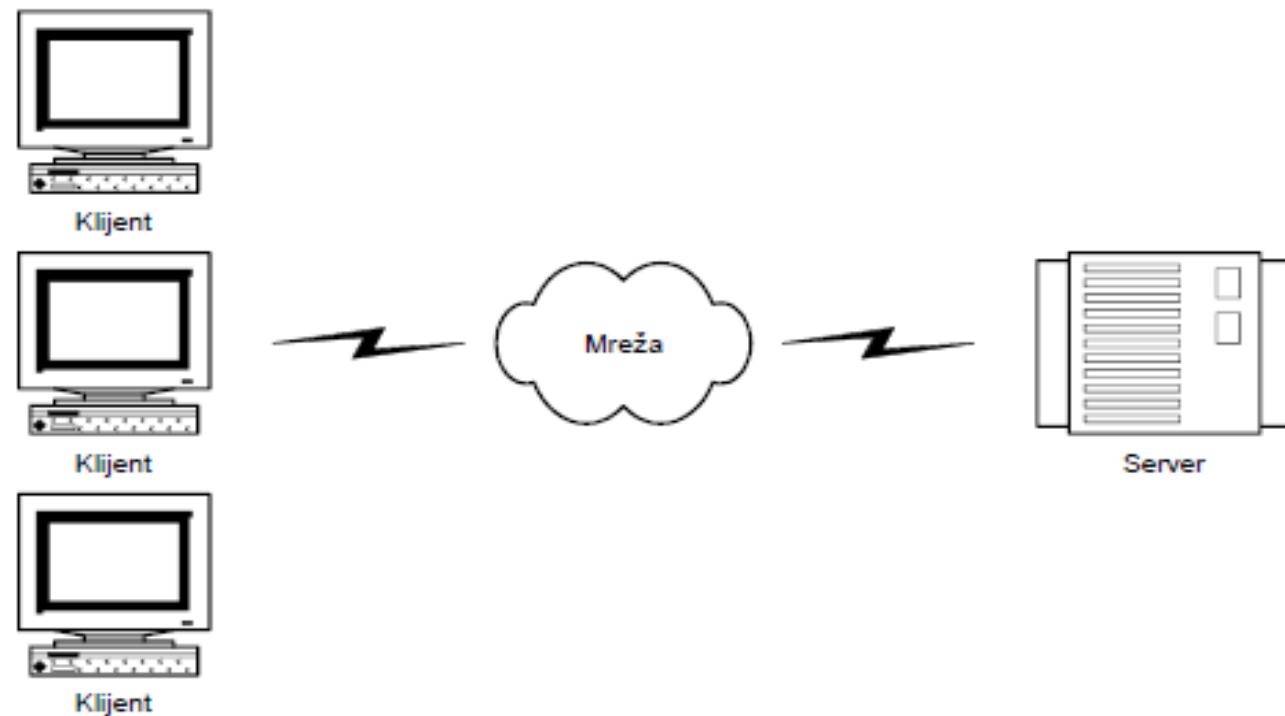
Fakultet za kompjuterske nauke, Univerzitet Megatrend



Klijent – server model

Klijent-server arhitektura

- U **klasičnim sistemima** za obradu podataka **po klijent/server modelu** mogu se uočiti tri klase komponenti: **server**, **klijenti** i **mreža**





Klijent-server arhitektura

- Namena **servera**:
 - **optimalno upravljanje zajedničkim resursima**, što su najčešće podaci,
 - ***upravljanje bazom podataka*** kojoj ***pristupa više korisnika***,
 - ***kontrola pristupa i bezbednosti*** podataka i
 - ***centralizovano obezbeđenje integriteta podataka*** za sve aplikacije



Klijent-server arhitektura

- **Klijent-aplikacije** vrše **upravljanje korisničkim interfejsom** i izvršavaju **deo logike aplikacije**
- **Računarska mreža** i komunikacioni softver omogućavaju **prenos podataka između klijenta i servera**



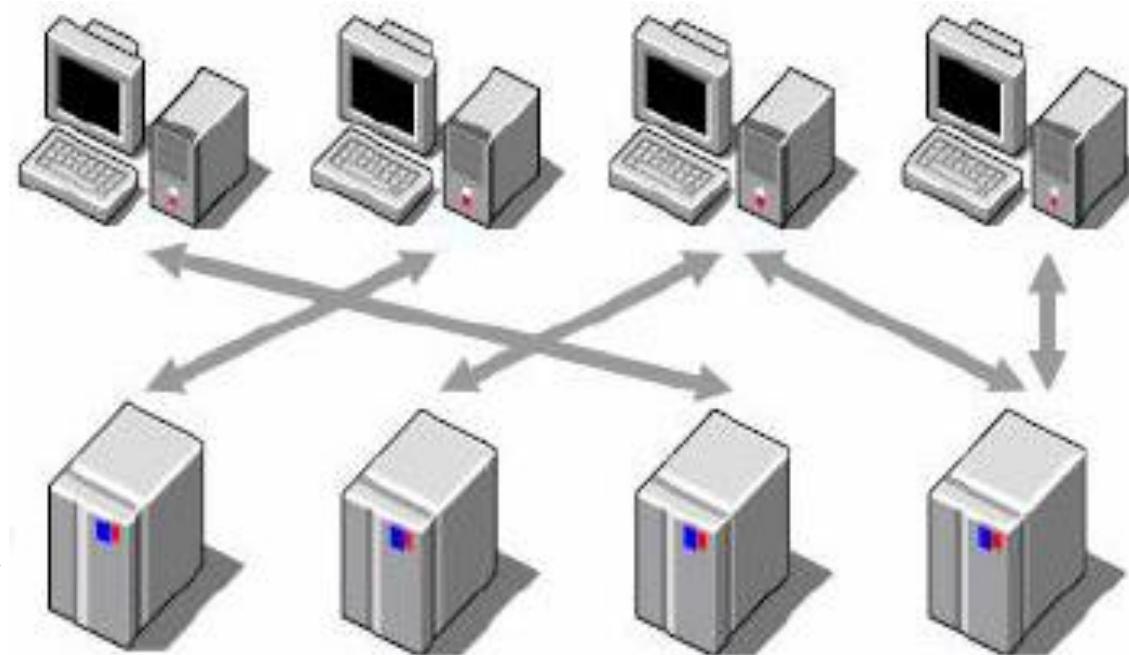
Dvoslojna arhitektura

- **Dvoslojna arhitektura** sastoji se od **3 komponente** distribuirane **u dva sloja – klijentskom i serverskom**
- **Komponente** su:
 - **Korisnički interfejs** – sesije, unos teksta, dijaloški prozori, prikaz na ekranu
 - **Upravljanje procesima** (*process management*) – generisanje, izvođenje i nadgledanje procesa i neophodnih resursa
 - **Upravljanje podacima** (*database management*) – servisi vezani za deljenje podataka i datoteka

Dvoslojna arhitektura

DVA SLOJA

korisnički interfejs +
deo upravljanja procesima



upravljanje podacima (DBM)
+ deo upravljanja procesima



Dvoslojna arhitektura

- Jedna od osnovnih ***karakteristika klijent/server sistema*** je ***distribuirana obrada podataka*** –
logika aplikacije je **podeljena** između klijenta i servera tako da obezbedi ***optimalno korišćenje resursa***



Dvoslojna arhitektura

- ***Klijentski deo aplikacije:***
 - ***prezentacija podataka*** i
 - provera ulaznih podataka
- ***Serverski deo aplikacije:***
 - ***rukovanje podacima***, u smislu njihovog fizičkog smeštaja i
 - kontrole pristupa

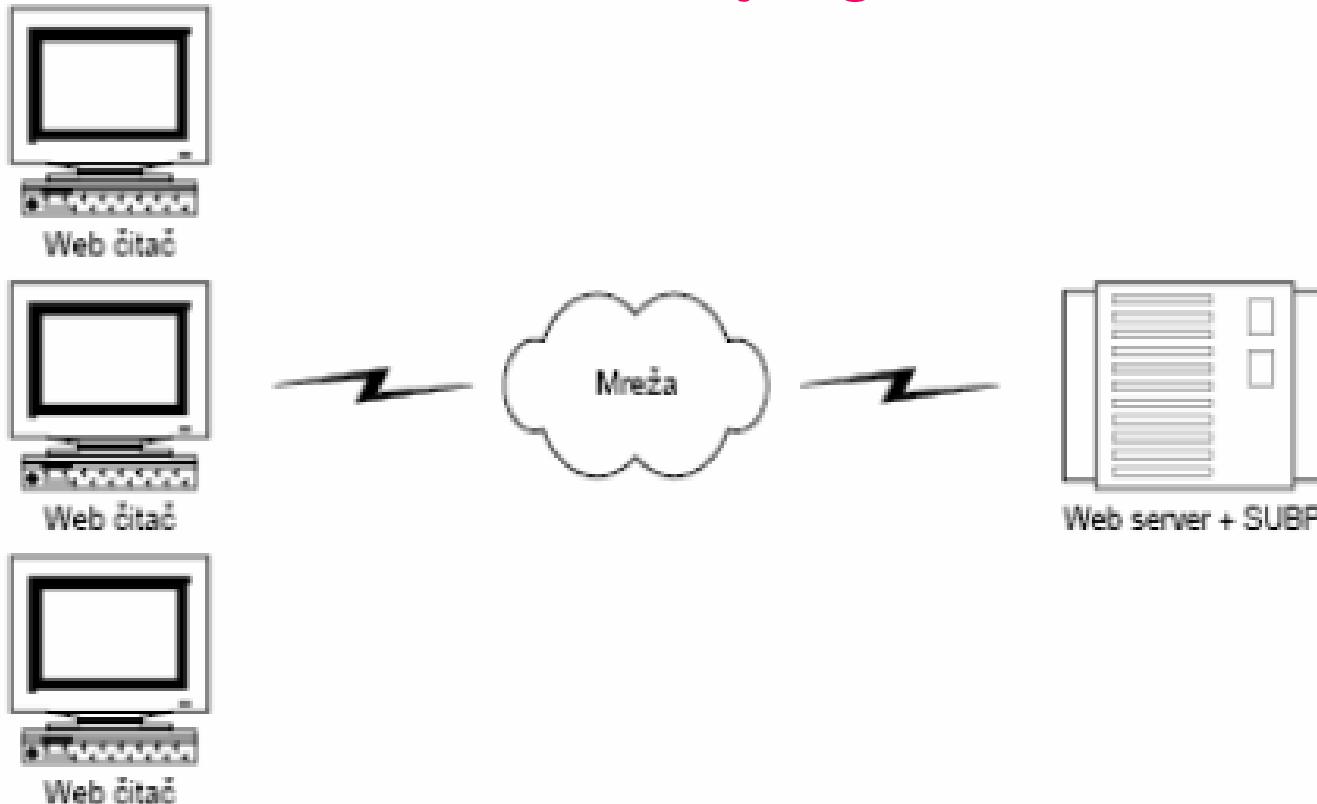


Dvoslojna arhitektura

- Neke od **prednosti** ovakvog modela su:
 - **centralizovano upravljanje** resursima sistema i
 - jednostavnije obezbeđivanje **sigurnosti podataka**
- **Osnovni problem** – **nedostatak skalabilnosti**
 - Pod skalabilnošću se podrazumeva osobina sistema da omogući efikasan rad velikom broju korisnika, i da dalje povećavanje broja korisnika ne izaziva drastičan pad performansi sistema

Dvoslojna arhitektura

Skica *dvoslojnog WEB sistema*



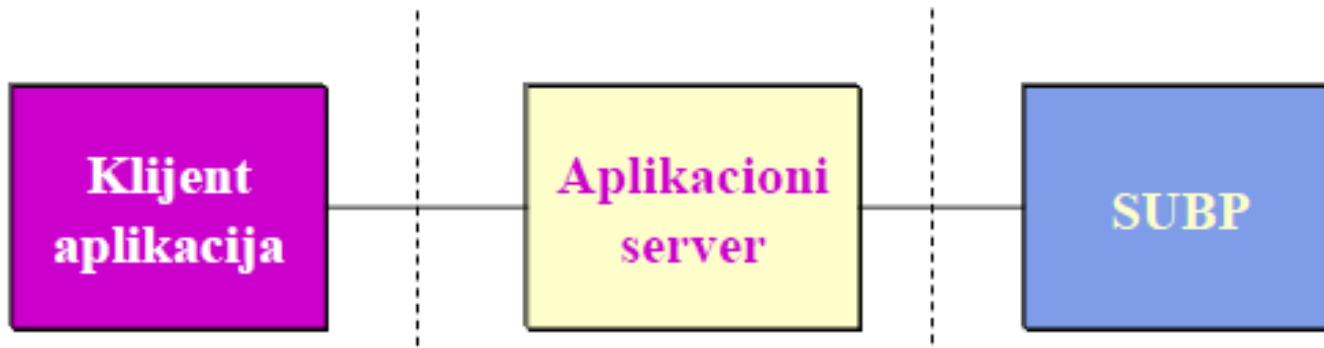


Troslojna arhitektura

- Klijent/server **sistemi sa troslojnom arhitekturom** (*three-tier architecture*) predstavljaju sisteme sa tri podsistema:
 1. **podsistem za interakciju sa korisnikom** (implementira funkcije korisničkog interfejsa);
 2. **podsistem za implementaciju osnovnih funkcija sistema** (implementira tzv. “poslovnu logiku”);
 3. **podsistem za rukovanje podacima**, pri čemu se pre svega misli na fizički smeštaj podataka (ovo je, zapravo, **sistem za upravljanje bazama podataka**)

Troslojna arhitektura

Elementi troslojne arhitekture klijent-server sistema



Ne postoji direktna veza između podistema za interakciju sa korisnikom i podistema za rukovanje podacima. Ovi podistemi se nazivaju i *slojevi*



Troslojna arhitektura

- ***U dvoslojnem modelu*** obrade podataka - **logika aplikacije je podeljena** između klijenta i servera,
- ***U troslojnem modelu*** ona se nalazi **koncentrisana u** tzv. ***aplikacionom serveru*** – čija je namena da izvršava programski kod koji implementira logiku aplikacije
- ***Klijentski deo aplikacija*** je namenjen **samo za implementaciju korisničkog interfejsa**



Troslojna arhitektura

- ***Funkcija sistema za upravljanje bazom podataka*** je isključivo ***fizičko rukovanje podacima***
 - U dvoslojnog modelu je, pored toga, izvršavao ***i deo logike aplikacije***



Prednosti troslojne arhitekture

- Troslojni koncept je doveo do **podele programskog koda na segmente** koji implementiraju tačno **određene funkcije sistema**
 - Tako organizovan sistem je **jednostavniji za održavanje**, jer je moguće nezavisno razvijati korisnički interfejs, i logiku aplikacije
 - Za potrebe fizičkog rukovanja podacima najčešće se koristi neki od **komercijalno dostupnih servera** za tu namenu



Prednosti troslojne arhitekture

- Troslojni koncept je doveo do podele programskog koda na segmente koji implementiraju tačno određene funkcije sistema
 - Troslojne arhitekture sistema podrazumevaju oslanjanje na **standarde** u odgovarajućim oblastima, zasnovane na Internet tehnologijama
 - Oslanjanje na standarde omogućava **integraciju sistema heterogenih** u pogledu korišćene hardverske i softverske opreme

Prednosti troslojne arhitekture

- Važna karakteristika troslojnih sistema je **skalabilnost**
 - a) **povećavanje** broja klijenata **je jednostavno**
 - b) povećavanje **propusne moći i brzine odziva** servera srednjeg sloja je moguće kroz **dodavanje novih serverskih mašina** uz korišćenje postojećih

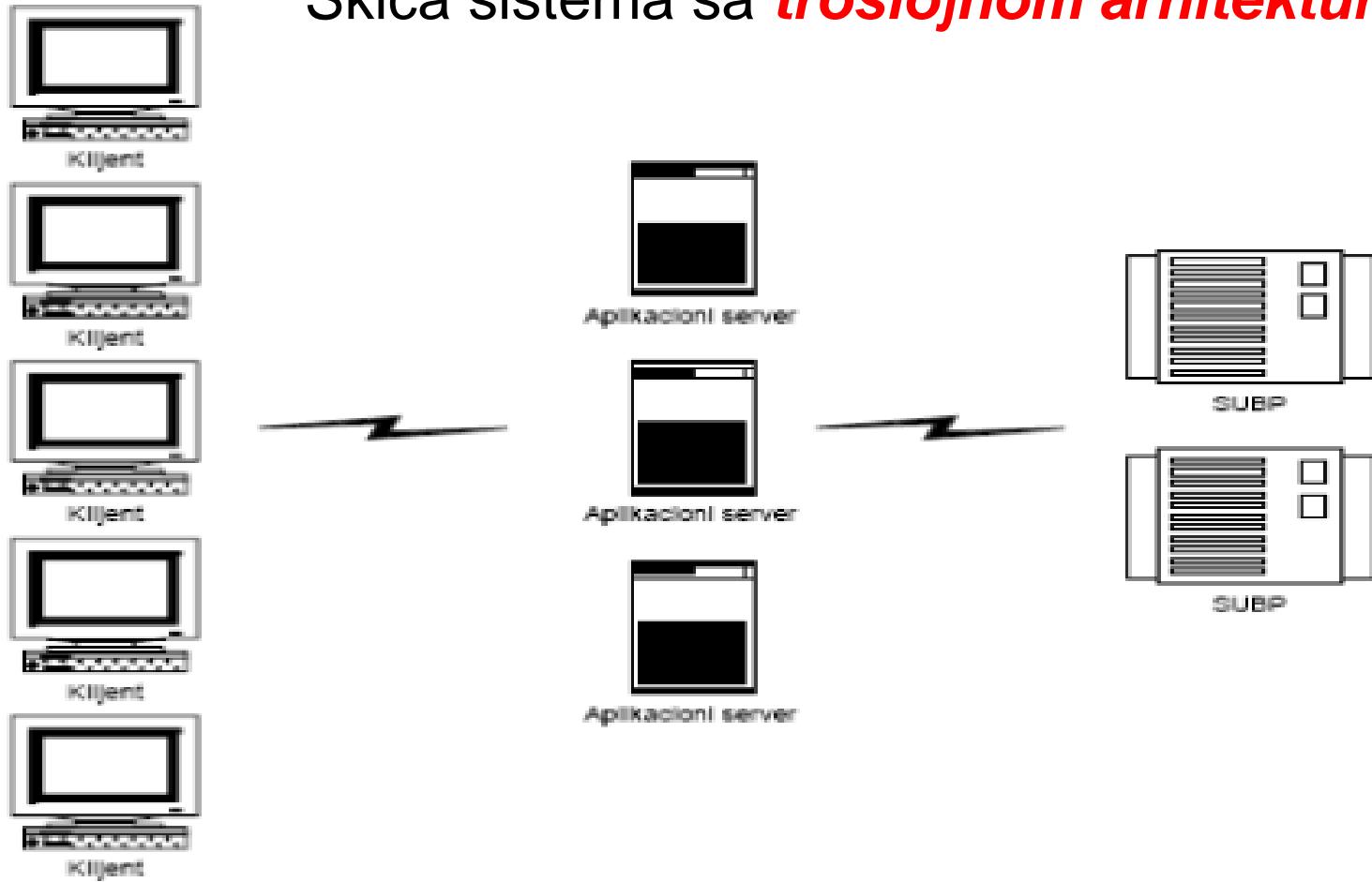


Prednosti troslojne arhitekture

- Sistem sa više servera karakteriše i povećana **pouzdanost i fleksibilnost**
- Logika aplikacije se može **menjati i u toku rada** sistema
- Moguće je efikasno vršiti **balansiranje opterećenja** serverskog podistema

Troslojna arhitektura

Skica sistema sa *troslojnom arhitekturom*



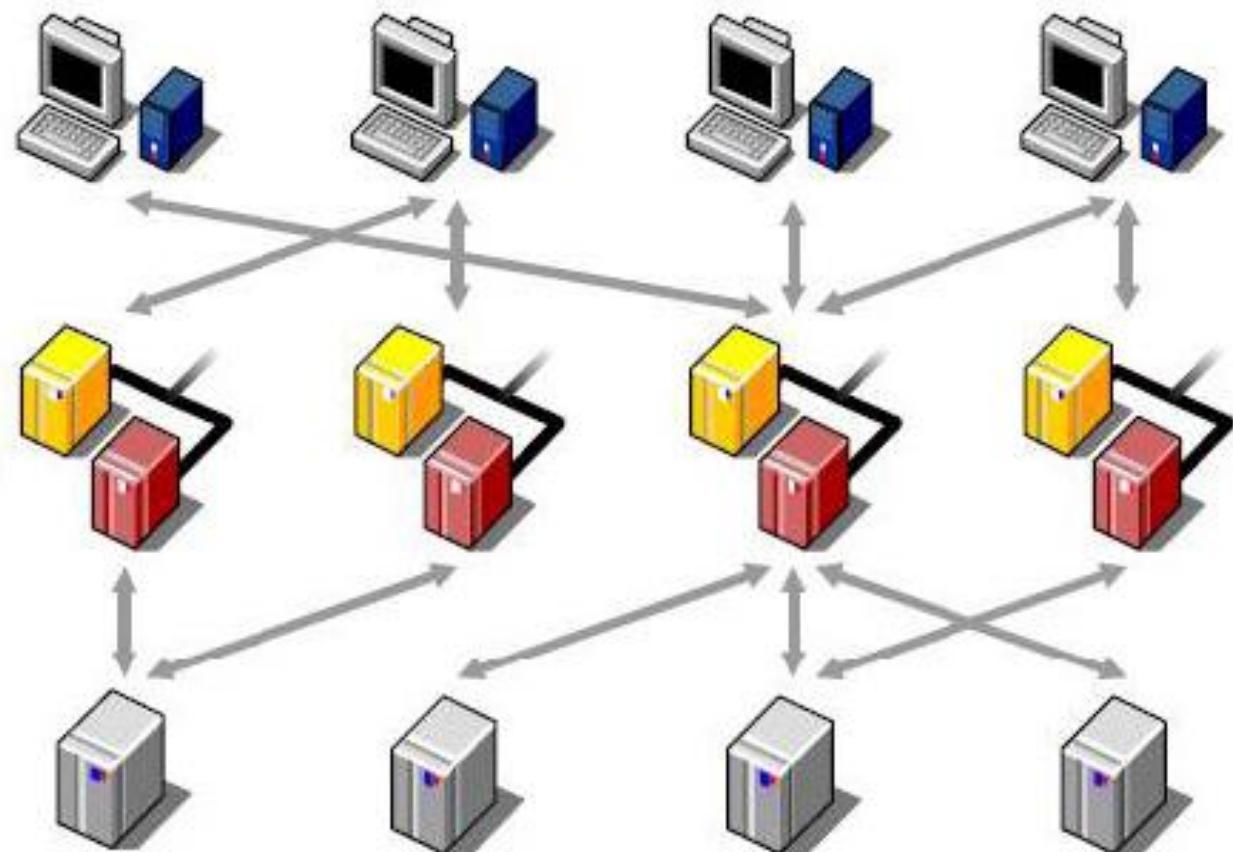
Troslojna arhitektura

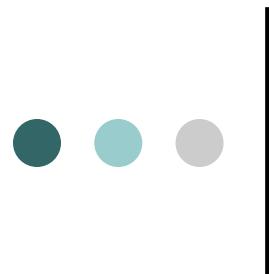
TRI SLOJA

korisnički interface

upravljanje procesima

upravljanje podacima
(DBM)

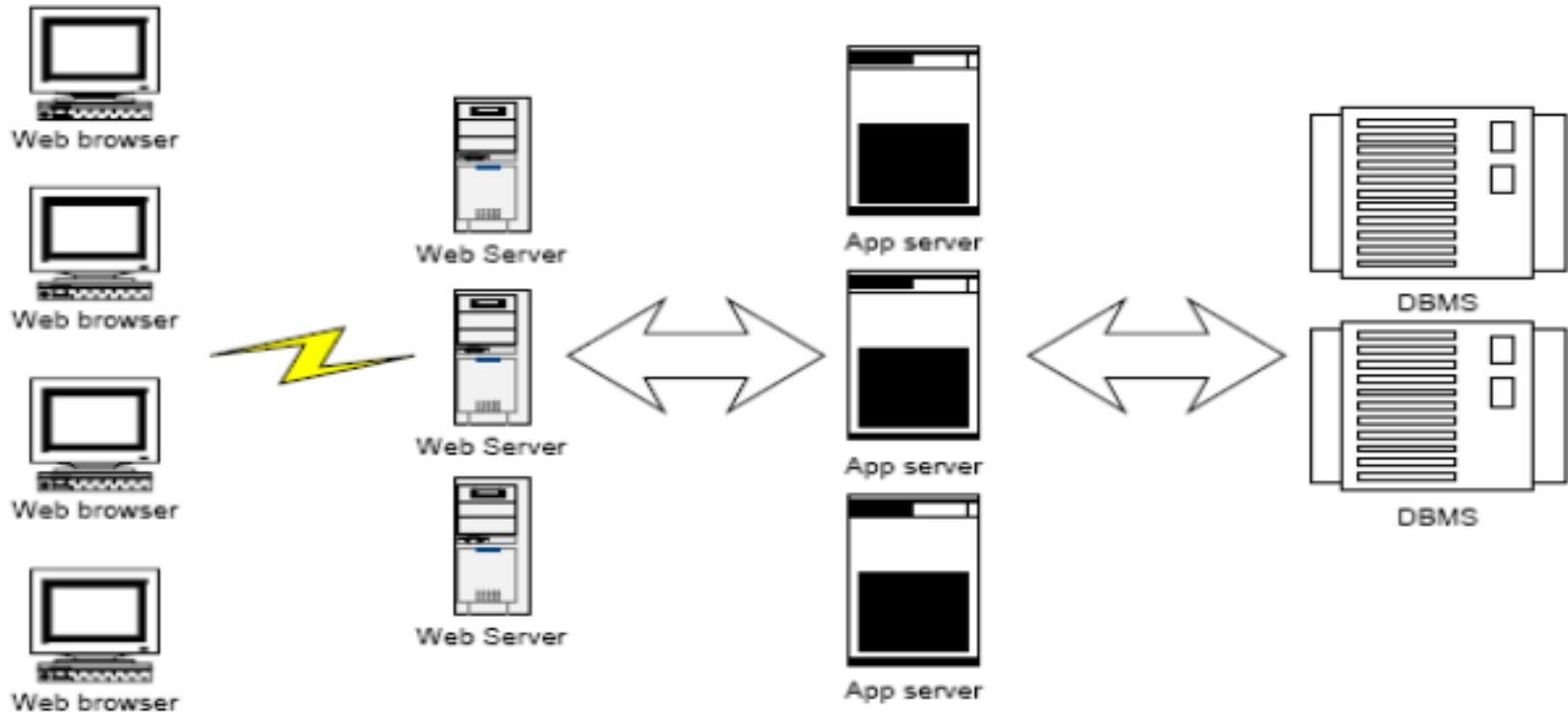




Arhitektura višeslojnih sistema

- **Daljim proširivanjem koncepta troslojnih sistema** dolazi se do pojma – **višeslojnih sistema** (*multitier architecture*)
- Vrši se **dalja podela na komponente u okviru srednjeg sloja** sa ciljem još većeg **povećanja skalabilnosti**, odnosno performansi

Arhitektura višeslojnih sistema



Srednji sloj je podeljen na dva sloja: jedan je namenjen za **opsluživanje Web klijenata**, a drugi sadrži komponente koje implementiraju **poslovnu logiku sistema**



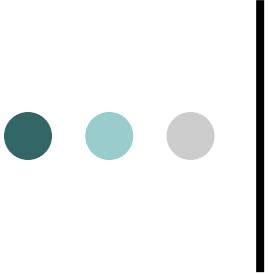
Adresiranje procesa (aplikacija)

- portovi



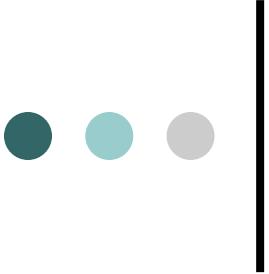
Transportni sloj

- **IP protokol** je odgovoran za komunikaciju ***između računara*** (tzv. **host-host komunikacija**) - isporučuje poruku ***od izvornog do odredišnog računara***
- Međutim, ovo je **nepotpuna isporuka**, jer često ***nije dovoljno*** samo ***isporučiti poruku*** odredišnom ***računaru***, već je treba i predati **odgovarajućem procesu** na odredišnom računaru koji će je prihvatiti i obraditi



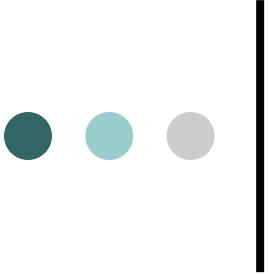
Transportni sloj

- Drugim rečima, ***konačno odredište poruke*** nije računar, već **aplikacioni proces** na odredišnom računaru (tzv. **proces-proces komunikacija**)
- Upravo ovaj poslednji korak u isporuci poruka predstavlja **odgovornost protokola transportnog sloja**, kao što su **UDP i TCP**



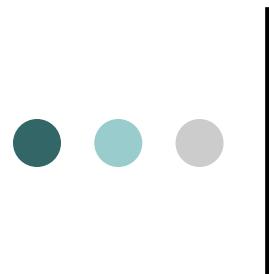
Portovi

- **Proces-proces komunikacija** se uobičajno ostvaruje **shodno klijent-server paradigmu**:
- Proces na lokalnom hostu (**klijent**), **traži uslugu** procesa na udaljenom računaru (**server**)
- Tipično, **oba procesa** (klijent i server) imaju **isto ime**
 - Da bi smo **od udaljene mašine** saznali **tekući datum** i **vreme**, potreban nam je **Daytime klijentski proces** pokrenut na lokalnom i **Daytime serverskom proces** pokrenut na udaljenom hostu



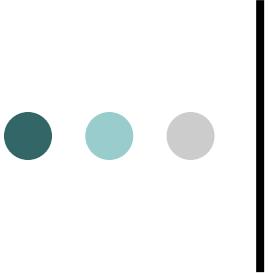
Portovi

- Savremeni ***operativni sistemi*** podržavaju ***više-korisnička*** (*multiuser*) i ***više-programska*** (*multiprogramming*) ***okruženja***
- Na udaljenom računaru mogu se ***u isto vreme*** izvršavati ***više serverskih procesa***, slično kao što se na lokalnom računaru mogu izvršavati ***više klijentskih procesa***



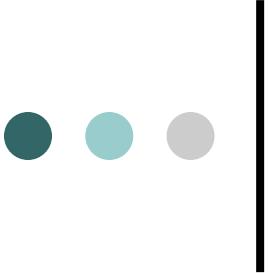
Portovi

- Da bi se uspostavila ***komunikacija između*** odgovarajućeg para ***procesa nepohodno je definisati (identifikovati)***:
 - Lokalni host
 - Lokalni proces
 - Udaljeni host
 - Udaljeni proces



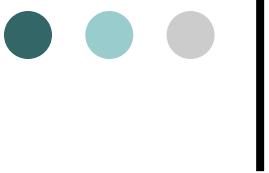
Portovi

- Lokalni i udaljeni ***host*** su identifikovani ***IP adresama***
- Da bi se ***identifikovali procesi*** - neophodan je ***još jedan identifikator***, koji se zove ***adresa servisa ili broj porta*** (ili samo port)
- Kod ***TCP/IP***, brojevi portova su ***celi brojevi iz opsega 0 – 65.535*** (opseg 16-bitnih brojeva)



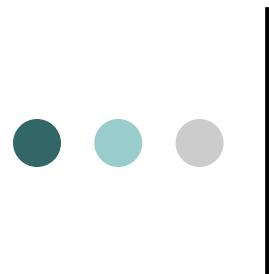
Portovi

- Za *identifikaciju klijentskog procesa* koristi se broj porta koji se zove **efemerni broj porta**
- Reč efemerni znači *"kratko-živući"* i koristi se zato što je život klijenta po pravilu kratak
- Efemerni brojevi porta su **veći od 1023**



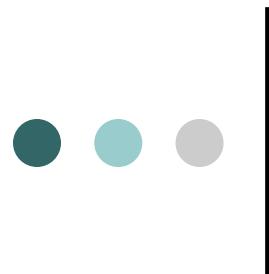
Portovi

- Tipično, *klijentskom procesu* se prilikom pokretanja **dodeljuje proizvoljno izabran jedan od neiskorišćenih** privremenih ***portova na klijentskom računaru***, kojeg on koristi za svolu identifikaciju prilikom obraćanja serverskom procesu
- Kada ***klijentski proces završi*** sa radom, **njegov broj porta se oslobođa** i može biti dodeljen nekom drugom klijentskom procesu



Portovi

- Server se takođe identificuje brojem porta
- Međutim, ovaj broj **ne može biti proizvoljno izabran**
- Ako bi serverski računar dodeljivao portove svojim serverskim procesima na slučajan način, *klijentski procesi* koji se izvršavaju na klijentskim računarima **ne bi znali preko kog porta da zatraže uslugu** udaljenog serverskog procesa

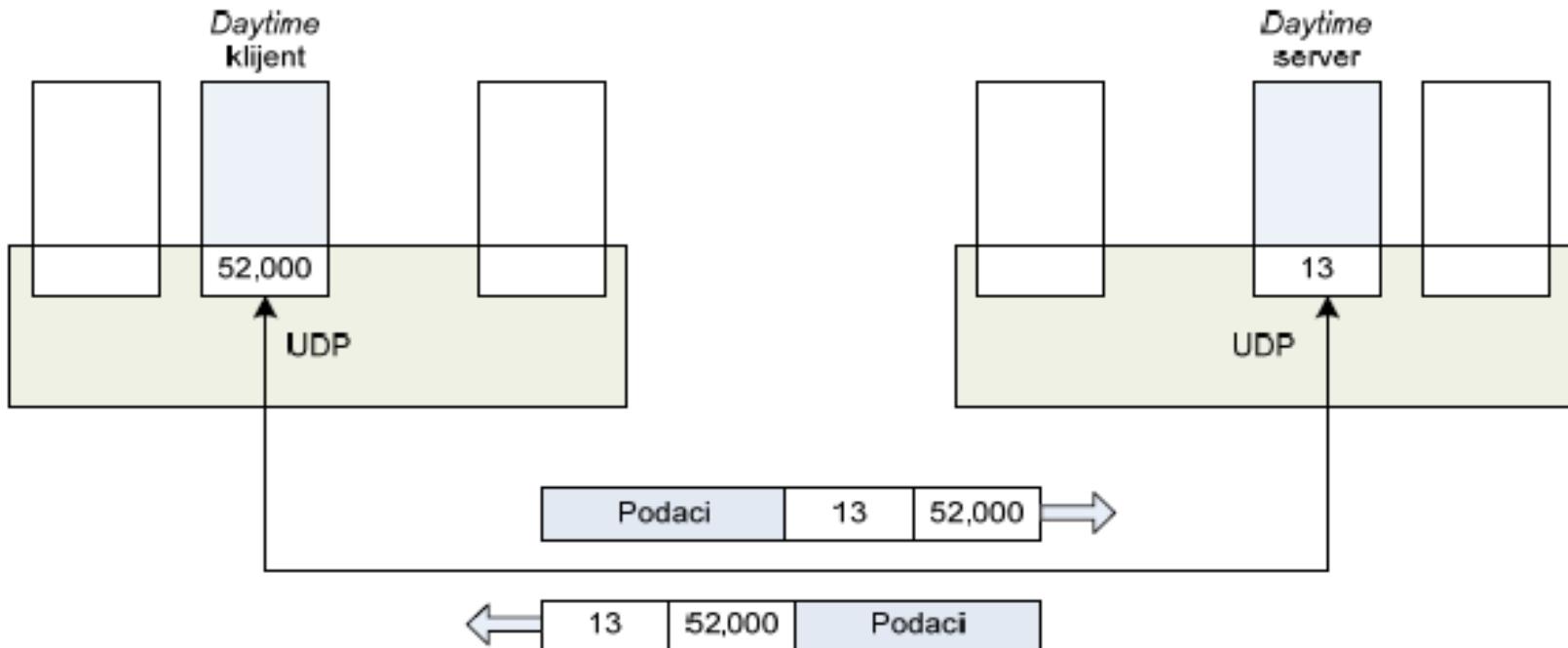


Portovi

- Zato se kod **TCP/IP za standardizovane servise** koriste univerzalni, tzv. **dobro-poznati** brojevi porta
- Svi **klijentski procesi** znaju **dobro-poznati port** odgovarajućeg serverskog procesa

Portovi

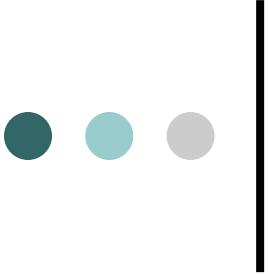
- Na primer, dok se za identifikaciju **Daytime klijentskog procesa** može koristiti **broj porta 52,000**, **Daytime serverski proces** mora da koristi **broj porta 13**





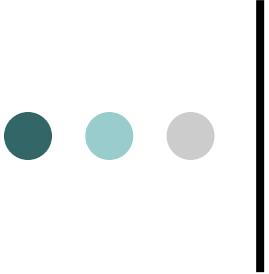
Opsezi portova

- Organizacija ICANN podelila je brojeve portova na tri opsega: ***dobro-poznati***, ***registrovani*** i ***dinamički*** (ili privatni)
- ***Dobro-poznati portovi*** - portovi iz opsega ***0 – 1.023***, koje dodeljuje (***definiše njihovu namenu***) organizacija ICANN



Opsezi portova

- ***Registrirani protovi*** - portovi iz opsega ***1.024 – 49.151***, koji ***nisu dodeljeni*** (organizacija ICANN nije definisala njihovu namenu)
- Neke ***druge organizacije*** ili ***firme mogu ih registrirati*** kod ICANN organizacije da bi se predupredila dupliranja



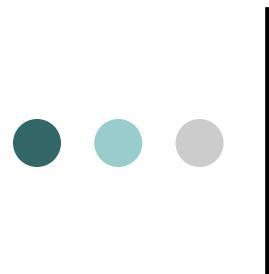
Opsezi portova

- **Dinamički portovi** - portovi iz opsega **49.152 – 65.535**, koji nisu **ni dodeljeni niti registrovani**
- Oni mogu biti korišćeni **kao privremeni** ili privatni brojevi portova
- Organizacija ICANN-a preporučuje da se **efemerni brojevi** porta biraju iz ovog opsega
- Međutim, kod mnogih sistema ova preporuka **nije ispoštovana**

Adrese soketa

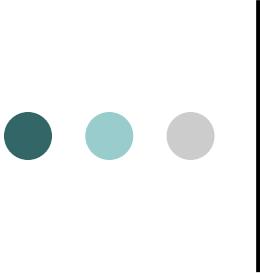
- Kombinacija *IP adresе* i *broja porta* se naziva **adresom soketa**
- **Klijentska adresa soketa jednoznačno definiše** klijentski proces
- **Serverska adresa soketa jednoznačno definiše** serverski proces





Adrese soketa

- Da bi se koristile *usluge transportnog sloja* neophodan je par klijentska/serverska **adresa soketa**
- Ova ukupno četiri podatka su **deo IP zaglavija** i UDP (odnosno **TCP**) **zaglavija**
- IP zaglavije sadrži IP adrese, dok UDP (TCP) zaglavije sadži brojeve portova

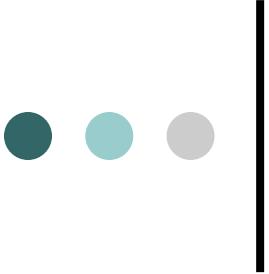


Internet servisi



FTP

- **Prenos datoteka** (*file transfer*) predstavlja jedan od klasičnih **servisa Interneta** i datira još od **ranih 1970-tih**
- Prenos datoteka se vrši **između klijentskog računara i serverskog računara u oba smera** (mogu se preuzimati i postavljati datoteke na server)
- Ovaj servis danas obično koristi **za postavljanje datoteka na WEB servere** kao i za preuzimanje velikih binarnih datoteka



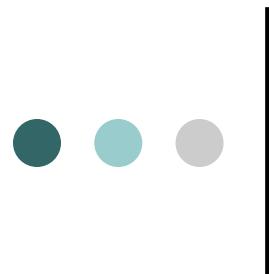
FTP

- Za **prenos datoteka** koristi se **FTP protokol** koji **koristi TCP na portu 20 i 21**, zatim **SCP i SFTP protokoli bazirani na SSH** koji nude **enkripciju** pri prenosu datoteka
- Za prenos datoteka, **na klijentskim računarima** se obično koriste programi poput:
 - **ftp** (komandni program koji **direktno implementira FTP protokol**),
 - **scp** (komandni program koji kopira datoteke uz **korišćenje enkripcije**)



FTP

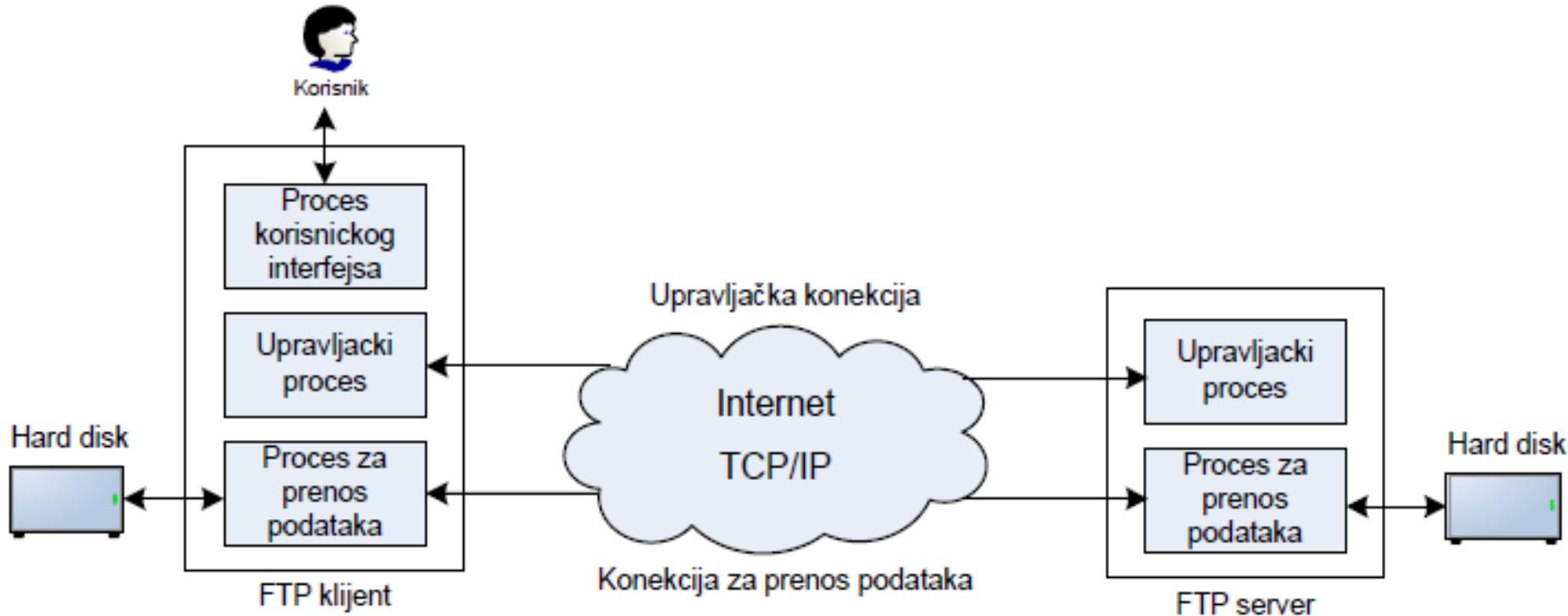
- *FTP se razlikuje od drugih klijent-server aplikacija* po tome što **ne uspostavlja samo jednu već dve TCP konekcije** između hostova (klijenta i srevera)
- Jedna konekcija se koristi **za prenos podataka**, a druga za **prenos upravljačkih informacija** (komande i odzivi)



FTP

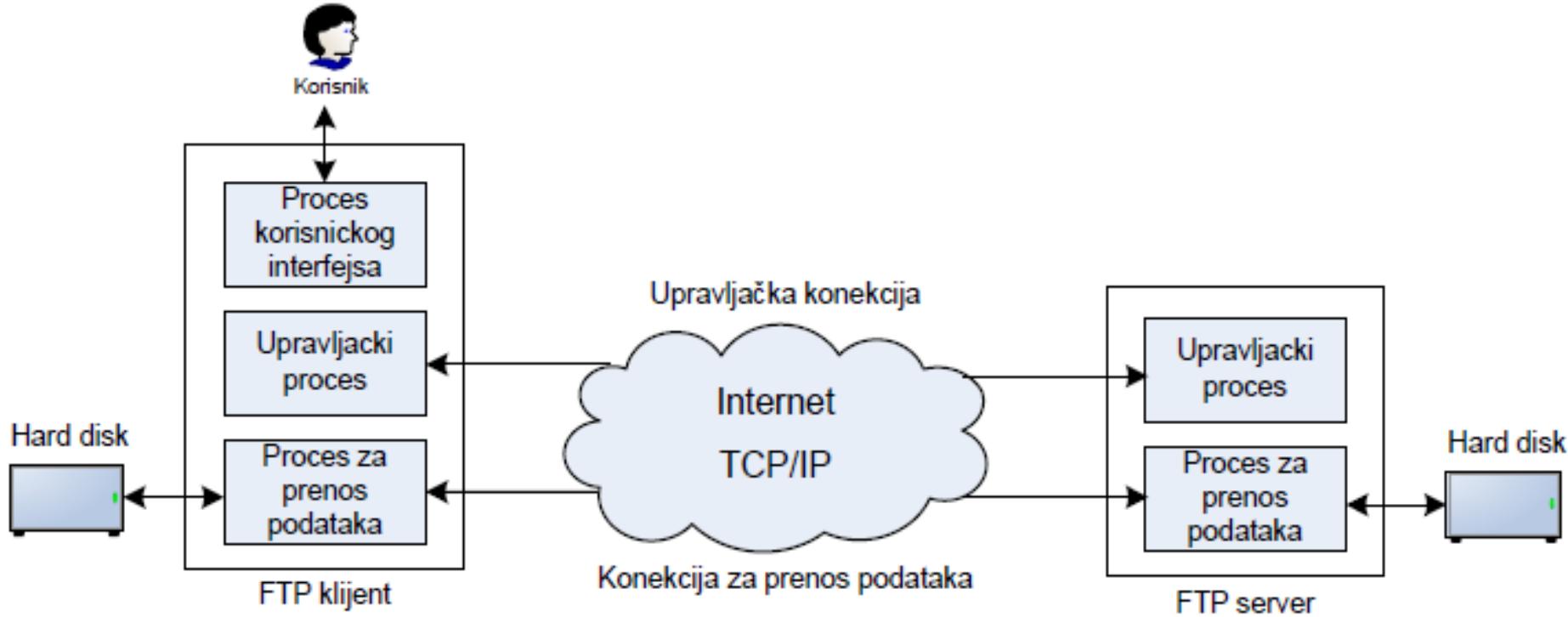
- **FTP klijent** sadrži ***tri komponente***:
 1. korisnički interfejs,
 2. ***proces za upravljanje*** klijentom i
 3. ***proces za prenos*** podataka
- **Server** ima ***dve komponente***:
 1. ***proces za upravljanje*** serverom i
 2. ***proces za prenos*** podataka

FTP



FTP zahteva dve TCP konekcije. Dobro-poznati port 21 se koristi za upravljačku konekciju, a port 20 za konekciju za prenos podataka
Upravljačka konekcija je otvorena za sve vreme trajanja sesije
Konekcija za prenos podataka se otvara, a onda i zatvara za prenos svakog pojedinačnog fajla

FTP

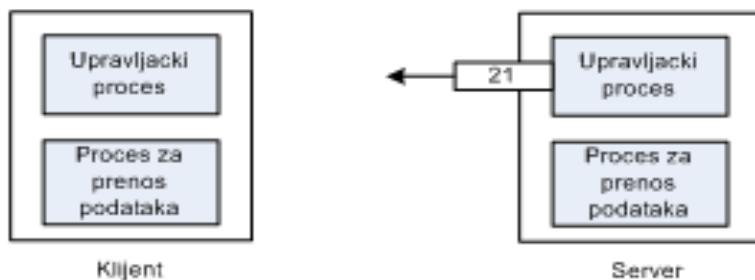


Upravljačka konekcija koristi jednostavna **pravila komunikacije**: klijent šalje **komandu**, a server vraća **odziv**.

Konekcija za prenos podataka zahteva **složenija pravila komunikacije**, s obzirom na (1) **brojne tipove fajlova** i (2) specifične **načine prenosa** podatak. FTP koristi TCP.

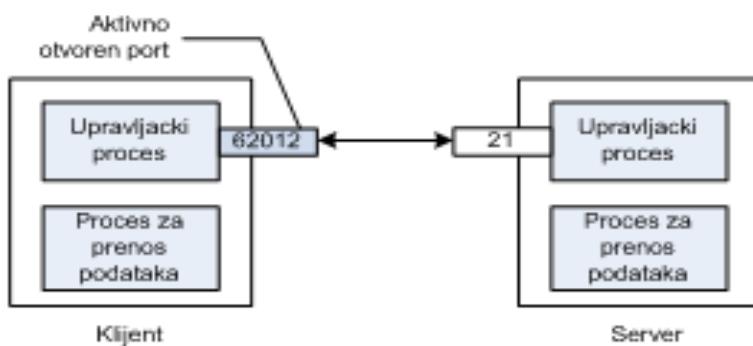
FTP - konekcije

Otvaranje *upravljačke* konekcije:



(1)

Server je izvršio pasivno otvaranje dobro-poznatog porta 21 i čeka na klijenta.



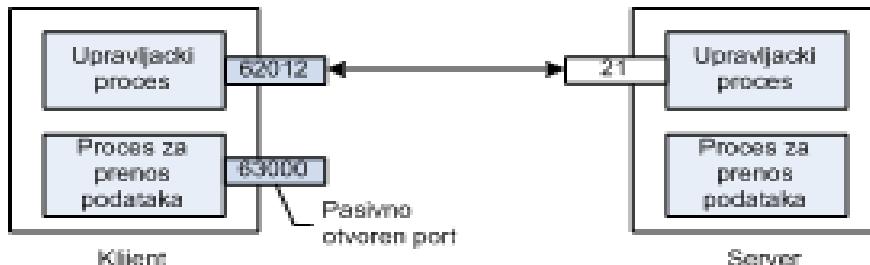
(2)

Klijent koristi dinamički prort i inicira aktivno otvaranje konekcije sa serverom.

Konekcija ostaje otvorena za **sve vreme trajanja FTP sesije**.
Korisnik unosi komande i od servera dobija odzive.

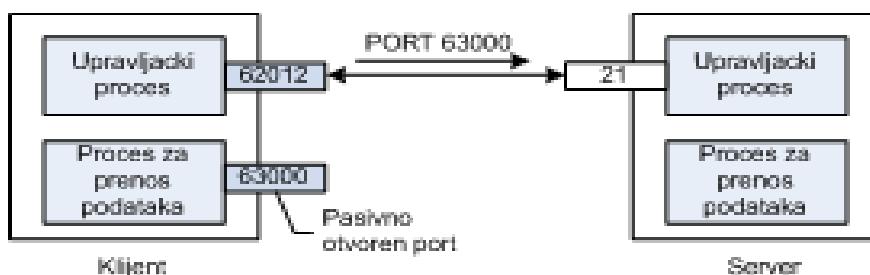
FTP - konekcije

Otvaranje **konekcije za prenos podataka**:



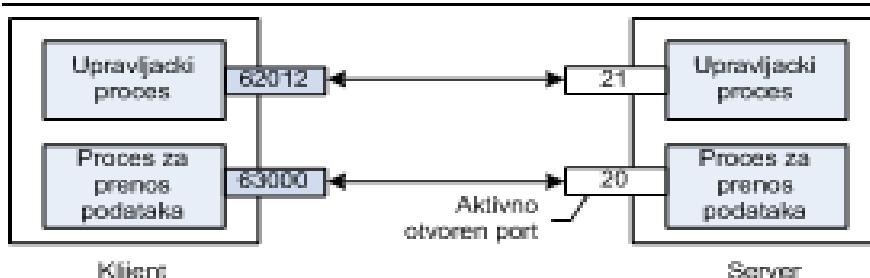
(1)

Klijent, a ne server, izvršava pasivno
otvaranje dinamičkog porta.



(2)

Klijent šalje broj dinamičkog porta
serveru, korišćenjem komande
PORT (prenosi se preko
upravljačke konekcije)



(3)

Server izvršava aktivno otvaranje
konekcije sa klijentom koristeći
dobro-poznati port 20 za sebe i
primljeni dinamički broj porta za
klijenta.

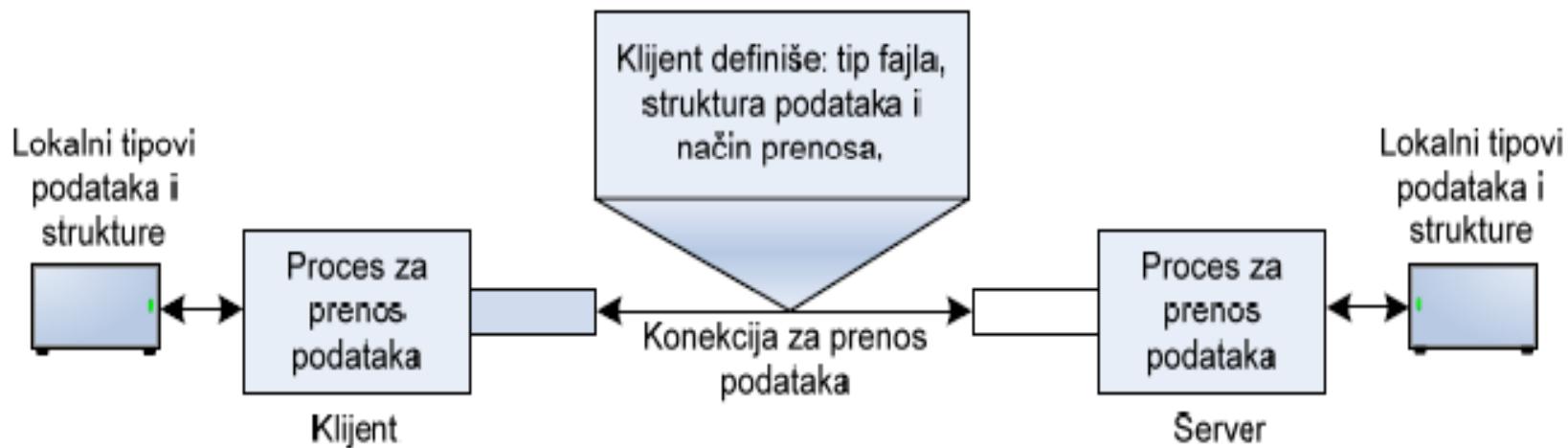
FTP - Komunikacija

Za komunikaciju preko upravljačke konekcije kod FTP-a se koristi NVT ASCII skup karaktera. Komunikacija se ostvaruje putem **komandi i odziva**. Ovakav način komunikacije (**polu dupleks**) je pogodan za upravljačku konekciju zato što se preko konekcije, u bilo kom trenutku, prenosi **najviše jedna komanda (ili odziv)**.



FTP - Komunikacija

Konekcija za podatke ima različitu namenu i **ostvaruje se na drugačiji način** u odnosu na komunikaciju preko upravljačke konekcije. Preko konekcije za podatke se prenose fajlovi. Za **svaki fajl koji želi da prenese, klijent mora definisati tri atributa komunikacije:** (I) **tip fajla,** (II) **strukturu podataka** i (III) **način prenosa**

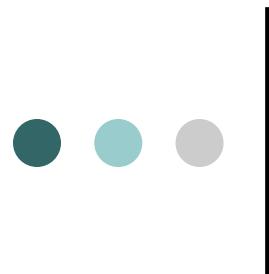




FTP - Komunikacija

Tip fajla

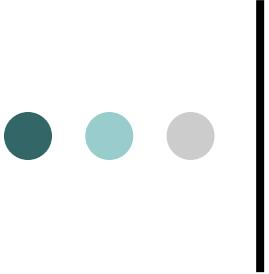
- Preko konekcije za podatke **FTP može prenosit sledeće tipove fajlova:**
- **ASCII.** Ovo je podrazumevani **format prenosa tekstualnih fajlova.** Svaki karakter se kodira u **NVT ASCII kôdu**
 - **Predajnik** transformiše fajl iz svoje sopstvene reprezentacije **u NVT ASCII karaktere**
 - S druge strane, **prijemnik** transformiše **NVT ASCII karaktere u** fajl kodiran svojim kôdom



FTP - Komunikacija

Tip fajla

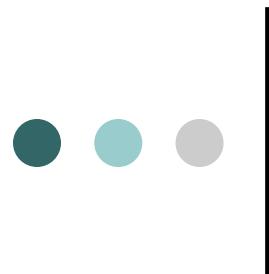
- **EBCDIC**. Ako jedna ili obe strane u komunikaciji koriste EBCDIC kôd, *tekstualni fajl* se može prenositi *korišćenjem EBCDIC kodiranja*
- **Image**. Ovo je *podrazumevani format* za prenos *binarnih fajlova*
 - *Fajl se prenosi kao kontinualni tok bitova bez bilo kakve *interpretacije ili kodiranja**
 - Uglavnom se koristi za prenos binarnih fajlova kao što su *izvršni programi*.



FTP - Komunikacija

Struktura podataka

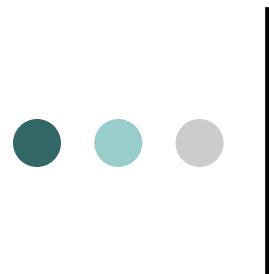
- FTP može prenositi fajlove preko konekcije za podatke korišćenjem jedne od sledeće ***tri interpretacije strukture podatka u fajlu:***
 - **File** (podrazumevana opcija). ***Podaci u fajlu nisu strukturirani***, tj. sadrži kontinualni tok bajtova
 - **Record**. Fajl je **podeljen na zapise**. Primenljivo **samo** na **tekstualne** fajlove.
 - **Page**. Fajl je **podeljen na stranice**. Svaka stranica ima **broj i zaglavje**. Stranicama se može pristupati bilo sekvensijalno bilo proizvoljno



FTP - Komunikacija

Način prenosa

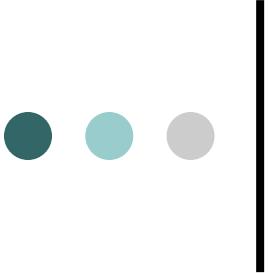
- Postoje ***tri načina prenosa fajlova*** preko konekcije za podatke:
 1. ***Stream***
 2. ***Block***
 3. ***Compressed***



FTP - Komunikacija

Način prenosa

- Postoje ***tri načina prenosa fajlova*** preko konekcije za podatke:
- **Stream**. Ovo je ***podrazumevani način prenosa***
- FTP isporučuje TCP-u podatke ***u vidu kontinualnog toka podataka***
 - **TCP** je odgovoran za podelu podataka ***na segmente*** odgovarajuće veličine



FTP - Komunikacija

Način prenosa

- **Stream.**
 - Ako su podaci koji se prenose prosti **tok bajtova** (struktura tipa **File**), **marker** za kraj fajla (**EOF-End-Of-File**) nije potreban. U ovom slučaju, **kraj fajla konicidira sa zatvaranjem konekcije** od strane predajnika
 - Ako su podaci **podeljeni na zapise** (struktura tipa **Record**), **svakom zapisu** se **pridodaje** 1-bajtni specijalni karakter *end-of-record* (**EOR**), dok se kraj celokupnog fajla označava karakterom **EOF**

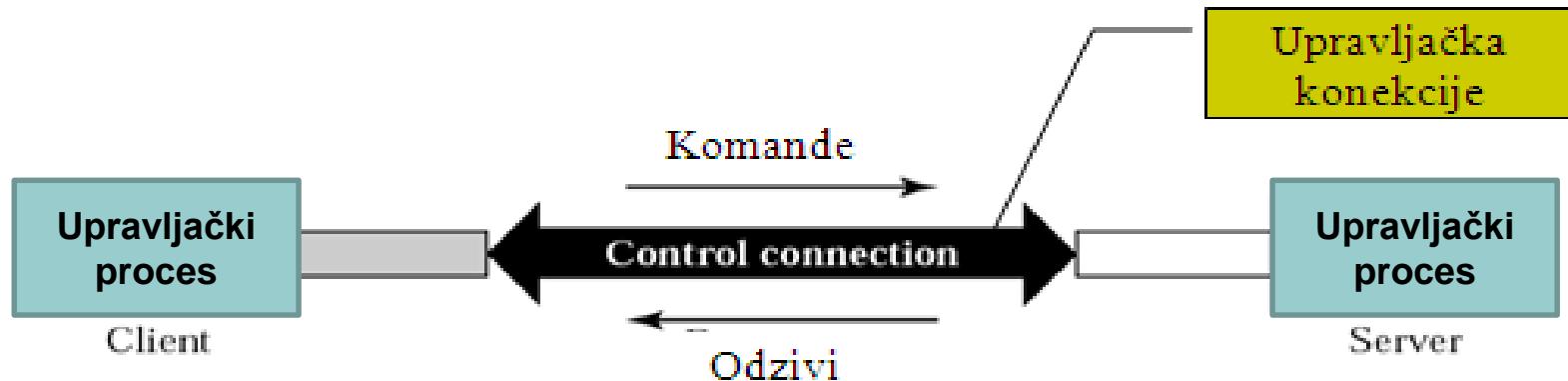


FTP - Komunikacija

Način prenosa

- **Block.** FTP može isporučivati TCP-u podatke *u blokovima*. U ovom slučaju, **svakom bloku** prethodi **3-bajtno zaglavlje**
 - *Prvi* bajt je bajt za **opis bloka** (block descriptor),
 - **Sledeća dva** bajta definišu **veličinu bloka** u bajtovima
- **Compressed.** Ako je **fajl isuviše veliki**, podaci su mogu **komprimovati**, kako bi se smanjila količina podataka koju treba preneti preko mreže.

FTP – Obrada komandi



Komande za pristup:

Komanda	Argument(i)	Opis
USER	User id	Korisničko ime
PASS	User password	Korisnička lozinka
ACCT	Nalog koji se menja	Podaci o nalogu
REIN		Ponovna inicijalizacija
QUIT		Logout (odjavljivanje)
ABOR		Poništavanje prethodne komande

FTP – Obrada komandi

- **Komande za upravljanje fajlovima.** Komande iz ove grupe omogućavaju korisniku da [pristupa fajl sistemu na udaljenom računaru](#)
 - Komande **omogućavaju korisniku** da se **kreće** po strukturi direktorijuma, **kreira** nove direktorijume, **briše** fajlove i slično

Koman.	Argument(i)	Opis
CWD	Ime direktorijuma	Promena radnog direktorijuma
CDUP		Prelazak na roditeljski direktorijum
DELE	Ime fajla	Brisanje fajla
LIST	Ime direktorijuma	Listanje poddirektorijuma ili fajlova
NLIST	Ime direktorijuma	Listanje imena direktorijuma ili fajlova
MKD	Ime direktorijuma	Kreiranje novog direktorijuma
PWD		Prikazivanje imena radnog direktorijuma
RMD	Ime direktorijuma	Brisanje direktorijuma
RNFR	Ime fajla (staro ime fajla)	Identificuje fajla čije se ime menja
RNTO	Ime fajla (novo ime fajla)	Promena imena fajla
SMNT	Ime fajl-sistema	Priklučivanje fajl-sistema.

FTP – Obrada komandi

- **Komande za formatiranje podataka.** Ove komande omogućavaju **korisniku da definiše strukturu** podataka, **tip fajla** i **način prenosa**
- Definisani atributi se potom koriste prilikom prenosa fajla

Komanda	Argument(i)	Opis
TYPE	A (ASCII), E (EBCDIC), I (Image)	Definisanje tipa fajla
STRU	F (File), R (Record), P (Page)	Definisanje strukture fajla
MODE	S (Stream), B (Block), C (Compressed)	Definisanje načina prenosa



FTP – Obrada komandi

- **Komande za definisanje porta.** Ovim komandama se definiše **broj prota** konekcije podataka **na strani klijenta**
- Postoje **dva načina** kako se to može uraditi
- **Prvi način** koristi **komandu PORT**. Klijent bira dinamički port, ***pasivno otvara izabrani port***, a serveru, putem upravljačke konekcije šalje ***komandu PORT sa brojem izabranog porta.*** Server koristi broj porta da bi uspostavio konekciju za podatke prema klijentu



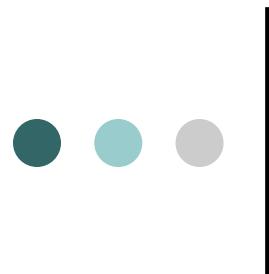
FTP – Obrada komandi

- **Komande za definisanje porta.**
- Drugi način koristi komandu PASV. Slanjem ove komande klijent *traži od servera da izabere dinamički port* i *izvrši pasivno otvaranje* izabranog porta
- Nakon što otvori port, server *u odzivu* vraća klijentu *broj izabranog porta*
- Konačno, **klijent**, koji je sada aktivna strana konekcije, **uspostavlja konekciju** za prenos podataka sa servorom, na dobijenom portu

FTP – Obrada komandi

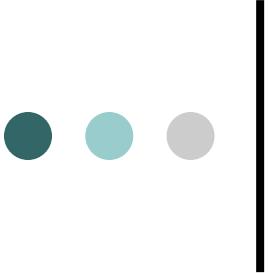
- ***Komande za prenos fajlova.***

Kom.	Argument(i)	Opis
RETR	Ime fajla(ova)	Preuzimanje fajla. Fajl se prenosi sa servera na klijenta.
STOR	Ime fajla(ova)	Smeštanje fajla. Fajl (ili fajlovi) se prenosi sa klijenta na server
APPE	Ime fajla(ova)	Slično STOR s tom razlikom da ako fajl postoji, podaci se dodaju fajlu
STOU	Ime fajla(ova)	Isto kao STOR, s tim da se očekuje da ime fajla bude jedinstveno u direktorijumu. Ako fajl sa datim imenom postoji, postojeći fajl neće biti prebrisan.
ALLO	Ime fajla(ova)	Rezerviše prostor za smeštanje fajla na serveru.
REST	Ime fajla(ova)	Pozicionira marker fajla na navedenu poziciju u fajlu.
STAT	Ime fajla(ova)	Vraća status fajla.



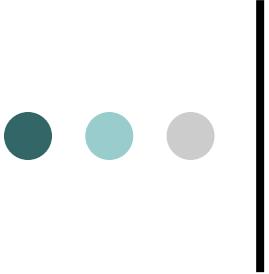
FTP – Odzivi

- **Svaka FTP komanda** generše **barem jedan odziv**
- Odziv se sastoji **iz dva dela**:
 - *trocifarski broj* i
 - *prateći tekst*
- **Numerički deo** definiše **potrebne parametre** ili dodatne informacije
- **Tri cifre odziva** predstavićemo u obliku **xyz**.



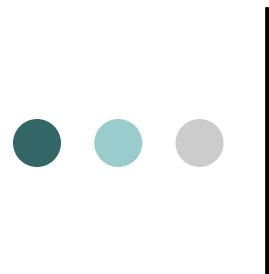
FTP – Odzivi

- **Prva cirfra**. Prva cifra odziva definiše ***status komande***:
 - **1yz** (**pozitivan** preliminarni odziv). **Akcija** zahtevana komandom **je startovana**. Kada završi akciju, **server će poslati još jedan odziv**.
 - **2yz** (konačni **pozitivan** odziv). **Akcija je završena**. Server je spreman da prihvati sledeću komandu.
 - **3yz** (**pozitivan** među-odziv). **Komanda je prihvaćena**, ali su **neophodne dodatne informacije**



FTP – Odzivi

- **Prva cirfra.** Prva cifra odziva definiše **status komande:**
 - **4yz** (prolazni **negativan** odziv). **Akcija nije startovana**, jer **server trenutno nije u mogućnosti da je izvrši**. Ista komanda može biti poslata kasnije.
 - **5yz** (konačni **negativni** odziv). **Komanda nije prihvaćena** i ne bi trebalo pokušavati ponovo.



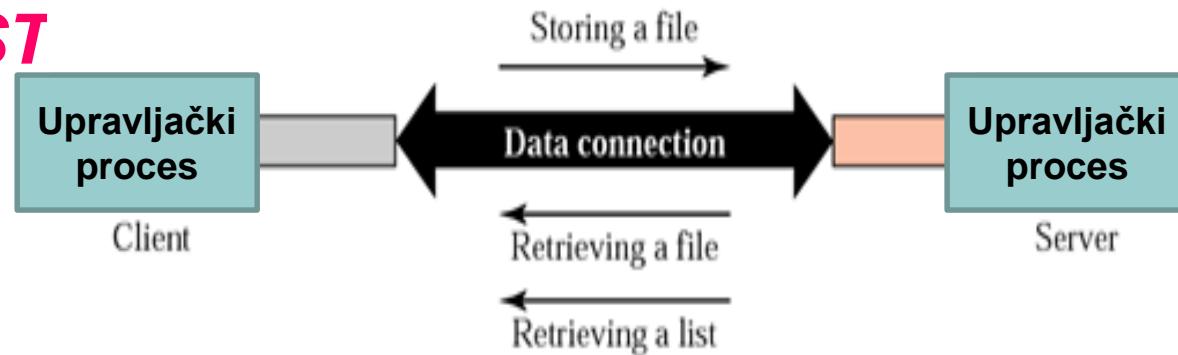
FTP – Odzivi

- **Druga cifra**. Druga cifra odziva takođe definiše ***status komande***:
 - **x0z** - odziv se odnosi na sintaksu komande
 - **x1z** - odziv je informativne prirode
 - **x2z** - odziv se odnosi na konekciju
 - **x3z** - odziv se odnosi na autorizaciju korisnika
 - **x4z** - značenje nije definisano
 - **x5z** - odziv se odnosi na fajl sistem
- **Treća cifra**. Treća cifra ukazuje na dodatne informacije

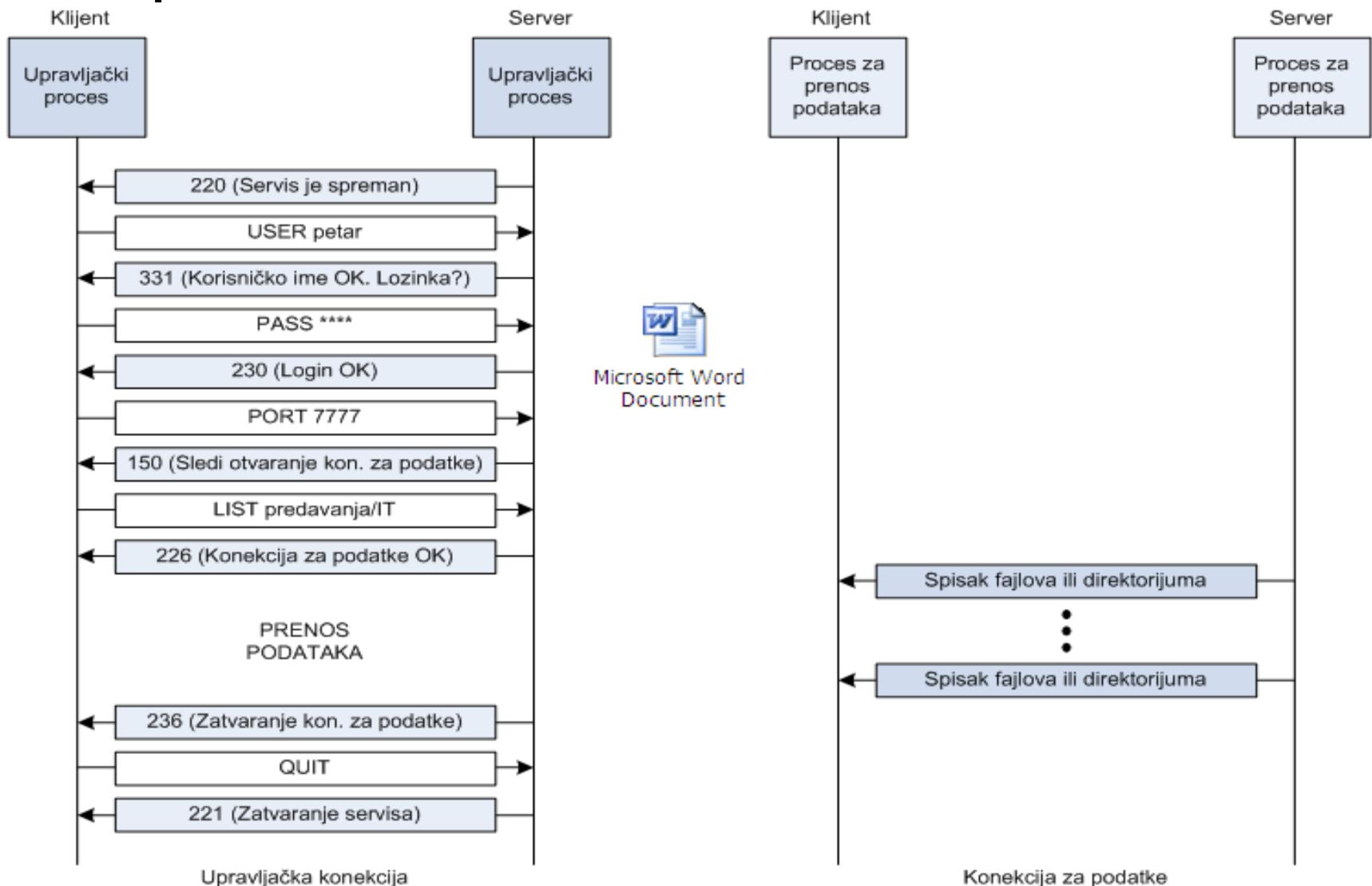
FTP – Prenos fajlova

Tri opcije:

1. Fajl se kopira sa serverskog na klijentski računar.
Ova operacija se pod kontrolom komande **RETR** i zove se **preuzimanje fajla**.
2. Fajl se kopira sa klijentskog na serverski računar.
Ova operacija je pod kontrolom komande **STOR** i zove se **smeštanje fajla**.
3. Server šalje klijentu **spisak direktorijuma** ili imena fajlova. Ova operacija je pod kontrolom komande **LIST**



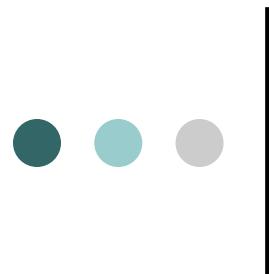
FTP – Primer FTP sesije





Elektronska pošta

- ***Elektronska pošta*** (e-mail) je sigurno jedan od ***najpopularnijih i najstarijih*** Internet servisa
- U današnje vreme, elektronska pošta ima tendenciju da skoro ***u potpunosti zameni klasičnu poštu***
- ***Na početku*** ere Interneta, poruke koje su slate elektronskom poštom bile su ***kratke i isključivo tekstualne***
- ***Danas*** je elektronska pošta ***mnogo složenija*** i omogućava prenos ne samo teksta, već ***i audio i video*** zapisa



Elektronska pošta

- Slanje i primanje pošte **korisnik** obično obavlja **preko klijenta** instaliranog **na** svom **računaru**
 - **Najpoznatiji klijenti** za elektronsku poštu danas su Microsoft Office Outlook, Microsoft Outlook Express, Apple Mail, Mozilla Thunderbird, Lotus Notes, Eudora, mapine, elm, . . .
- Sve više na značaju dobijaju **i klijenti za mobilne uređaje** u kojima prednjači iPhone/iPod Touch



Elektronska pošta

- **Značajan obim** elektronske pošte **se odvija** preko **javnih servisa** elektronske pošte **vezanih za WEB** koji **ne zahtevaju korišćenje posebnog klijenta** elektronske pošte
- Rad sa elektronskom poštou se obavlja **korišćenjem WEB aplikacija**
- **Najznačajniji servisi** ovog tipa su *Yahoo! Mail, Microsoft Hotmail, Google Gmail, itd.*



Arhitektura elektronske pošte

- **Protokoli koji se koriste** u okviru elektronske pošte su **SMTP**, **POP3** i **IMAP**
 - Svi oni **koriste TCP** i to na portovima **25**, **110** i **143**
- **Arhitekturu e-mail sistema** istražićemo kroz opis **četiri tipična scenarija**
- Počećemo sa najjednostavnijom, a završiti sa najsloženijom i u isto vreme najčešćom situacijom



Arhitektura elektronske pošte

- **Prvi scenario.** *Pošiljalac i primalac e-mail-a su korisnici (ili aplikacioni programi) na istom sistemu*
- Administrator je *kreirao jedno* poštansko sanduče (**mailbox**) **za svakog korisnika**
- Sve poruke koje prima neki korisnik, smeštene su u njegovom poštanskom sandučetu
- **Mailbox** je *fajl na lokalnom hard disku sa ograničenim pravom pristupa*
 - *Mailbox-u može pristupati jedino njegov vlasnik*



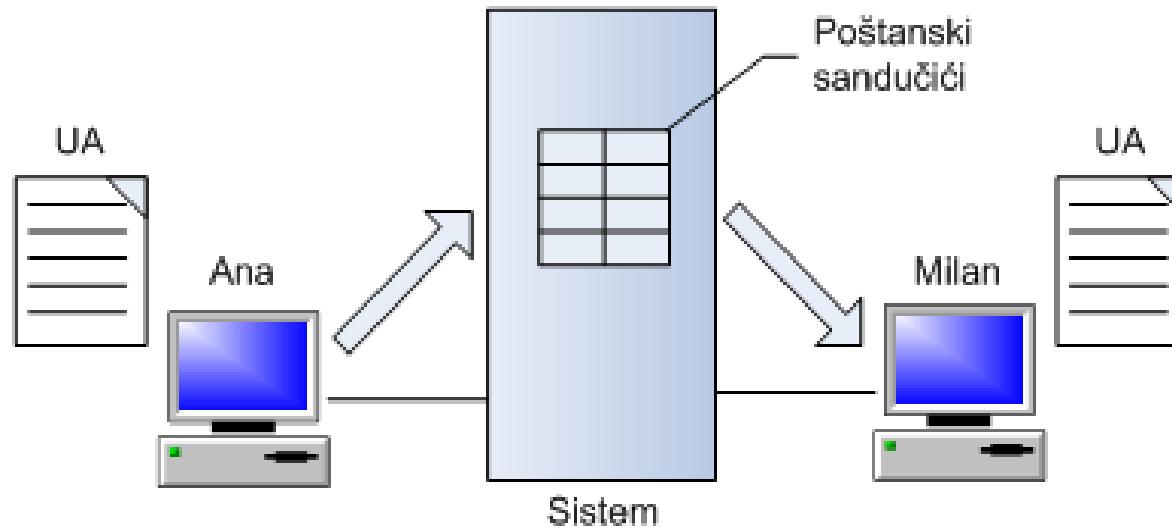
Arhitektura elektronske pošte

- ***Prvi scenario***
- Prepostavimo da korisnik **Ana** želi da **pošalje** poruku korisniku **Milanu**
- **Ana** pokreće program **korisnički agent** (**UA** - User Agent) pomoću koga **priprema** (**piše**) **poruku** i **smešta** je u Milanovo poštansko sadnuče
 - **Poruka sadrži** **mailbox adrese** (tj. **imena** odgovarajućih **fajlova**) pošiljaoca i primaoca
- Takođe pomoću **korisničkog agenta** **Milan** može preuzeti⁷⁶ i pročitati poruku, kada je njemu zgodno

Arhitektura elektronske pošte

○ Prvi scenario

Pošiljalac i primalac e-mail-a su korisnici (ili aplikacioni programi) na istom sistemu - dovoljna su dva korisnička agenta



UA - Korisnički agent



Arhitektura elektronske pošte

- **Drugi scenario.** Pošiljalac i primalac e-mail-a su korisnici (ili aplikacioni programi) **na dva različita sistema**
- Poruka mora biti **preneta kroz Internet**, a pored **korisničkog agenta**, sada je neophodan i **agent za prenos poruka** (*MTA - Message Transfer Agent*)



Arhitektura elektronske pošte

- **Drugi scenario.** Ani je neophodan **korisnički agent** da bi poruku prosledila **lokalnom** sistemu, tzv. **mail serveru**
- **Mail server** koristi red čekanja za smeštanje poruka koje još uvek nisu poslate
- Milanu je takođe neophodan **korisnički agent**, kako bi mogao da **preuzme poruke** iz poštanskog sandučeta **u njegovom lokalnom sistemu**
- Međutim, **poruka treba biti preneta kroz Internet**, od Aninog do Milanovog sistema



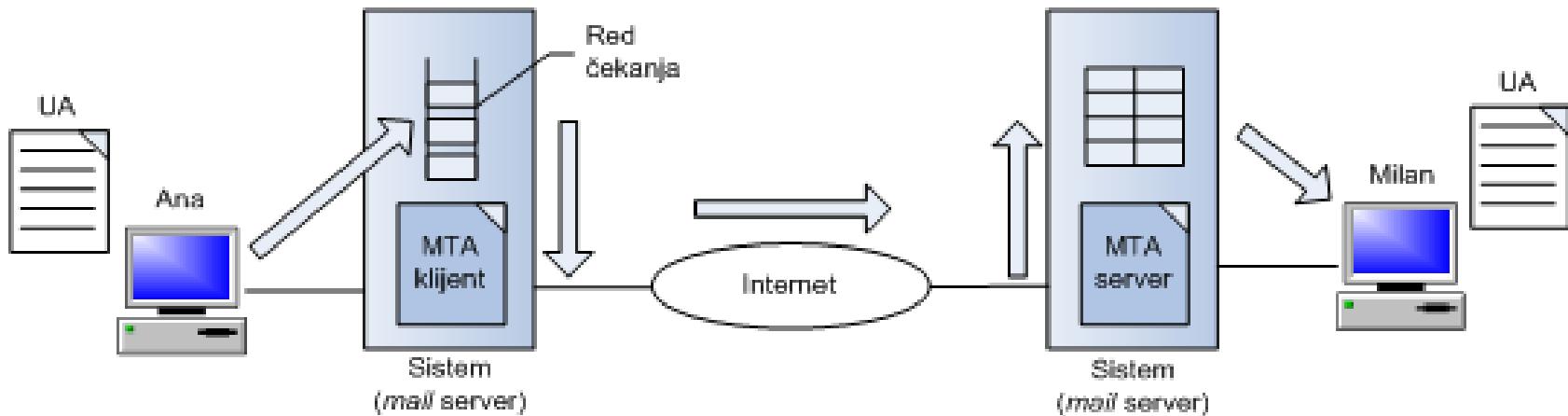
Arhitektura elektronske pošte

- **Drugi scenario.**
- Ovaj zadatak obavljaju **dva agenta za prenos poruka**: *jedan* ima **ulogu klijenta**, a **drugi servera**
- Poput većine klijent-server programa na Internetu, **server radi sve vreme** zato što unapred **ne zna kada će mu se neki klijent obratiti**
- S druge strane, **klijenta može pokrenuti sistem** onda **kada u redu čekanja ima poruka** koje treba poslati

Arhitektura elektronske pošte

○ Drugi scenario.

Pošiljalac i primalac e-mail-a su korisnici (ili aplikacioni programi) na dva različita sistema. Potrebna su dva korisnička agenta (UA) i par agenata za prenos prouka MTA (klijent i server)



MTA - Agent za prenos poruka



Arhitektura elektronske pošte

- **Treći scenario.** *Milan* je, kao i u prethodnom scenariju, ***direktno povezan na svoj sistem***
- Međutim, to nije slučaj sa *Anom*, koja je ***sa svojim sistemom povezana putem dial-up, ADSL ili kablovskog modema,*** ili je ***povezana na LAN* u organizaciji ***koja koristi jedan mail server*** za sve korisnike (svi korisnici moraju da šalju svoje mejlove ovom *mail serveru*)**



Arhitektura elektronske pošte

- **Treći scenario**. Ana je i dalje neophodan **korisnički agent** za pripremu poruke
- Sada je **specifično** to što ona treba da **pošalje poruku mail serveru kroz LAN**
- To se postiže **parom (klijent-server) agenata za prenos poruka** (MTA)
- Uvek **kada Ana ima poruku za slanje**, ona poziva **korisničkog agenta**, koji dalje poziva **MTA klijenta**
- **MTA klijent** uspostavlja vezu sa **MTA serverom** koji radi sve vreme



Arhitektura elektronske pošte

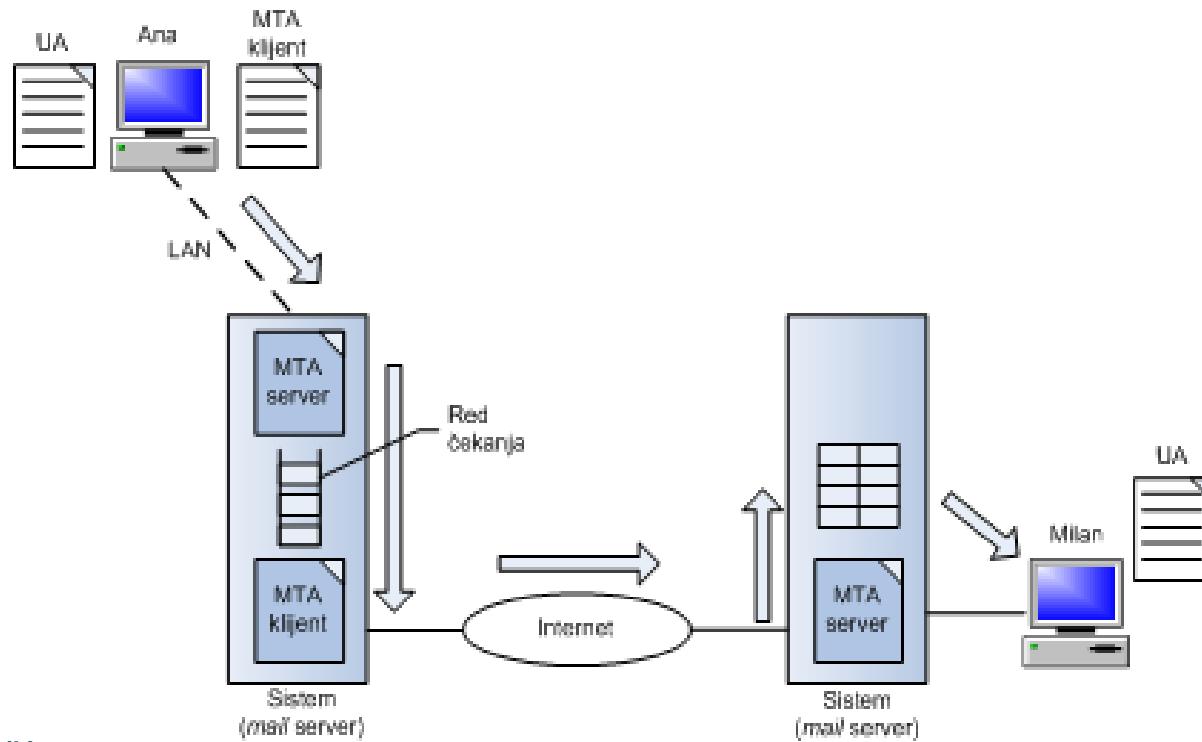
- **Treći scenario.** **Sistem na Aninoj strani** sve primljene poruke **čuva u redu čekanja**, a **koristi MTA klijenta** za **slanje poruka sistemu** na Milanovoj strani
- **Milanov sistem prima poruku** i **smešta je** u Milanovo poštansko sanduče
- Milan koristi **korisnički agent** za preuzimanje poruke iz sandučeta i čitanje poruke
- Uočimo da su u ovom slučaju **potrebna dva para MTA klijent/server programa**

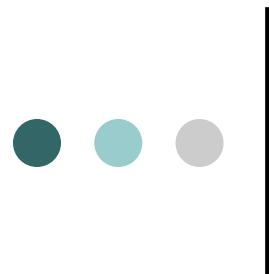
Arhitektura elektronske pošte

○ Treći scenario.

Pošiljalac nije direktno povezan sa svojim sistemom, već preko LAN-a.

Potrebna su dva korisnička agenta i dva para agentata za prenos prouka (klijent i server)





Arhitektura elektronske pošte

- **Četvrti scenario.** (**najčešći scenario**), **Milan** je takođe **povezan sa svojim mail serverom putem LAN-a**
- Pošto **Milan nema mogućnost direktnog pristupa mail serveru**, potreban je mehanizam koji će mu omogućiti **da preuzme primljene poruke** iz svog poštanskog sandučeta
- Ovaj zadatak **obavlja još jedan par klijent-server agenata**, tzv. **agenti za preuzimanje poruka (MAA - Message Access Agent)**



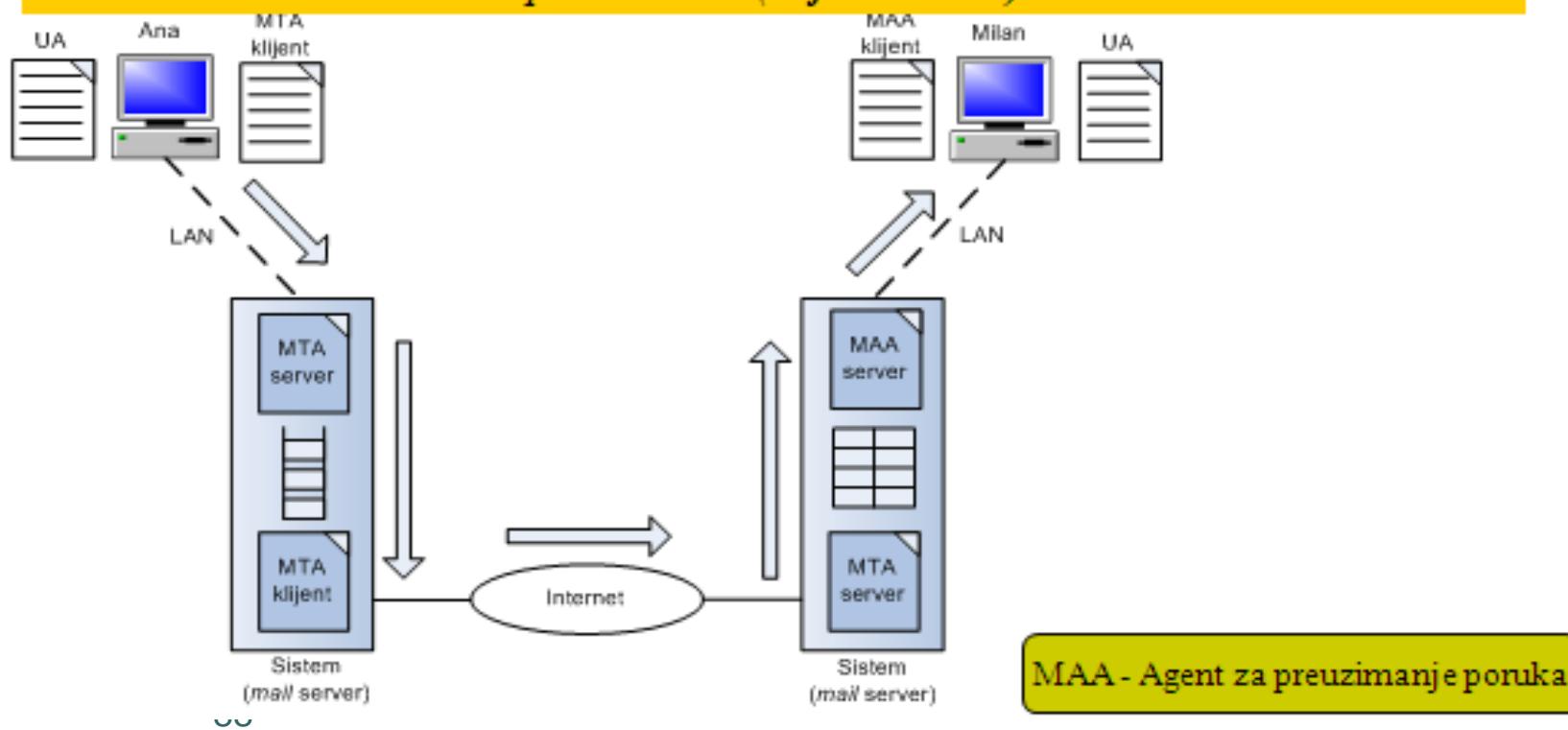
Arhitektura elektronske pošte

- **Četvrti scenario.** Za preuzimanje svojih poruka, Milan koristi ***MAA klijenta***
- ***Klijent šalje zahtev MAA serveru***, koji neprekidno radi, ***zahtevajući prenos poruka iz Milanovog mailbox-a***

Arhitektura elektronske pošte

○ Četvrti scenario.

Kada su i pošiljalac i primalac povezani sa mail serverom preko LAN-a, potrebna su dva korisnička agenta, dva para MTA (klijent i server) i jedan par agentata za preuzimanje poruka MAA (klijent i server)





Arhitektura elektronske pošte

- **Četvrti scenario**. Potreban je *još jedan par* klijent-server programa, *MAA*, koji se *koristi za preuzimanje poruka iz poštanskog sandučeta*
- Ovu operaciju ne može da obavlja MTA, zato što je **MTA program *push tipa*** (uvek samo šalje poruke severu)
- ***Preuzimanje poruka*** je operacija ***pull tipa*** (poruke se uzimaju od servera)

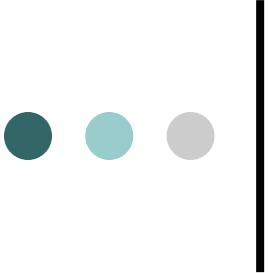


Korisnički agent

- **Prva komponenta** sistema elektronske pošte je **korisnički agent**
- Njegova namena je **da korisniku olakša slanje i primanje poruka**

Servisi korisničkog agenta

- **Korisnički agent** je softverski paket (program kao npr. *Microsoft Outlook*) za kompoziciju (**kreiranje**) poruka, **čitanje** poruka, **odgovaranje** na i **prosleđivanje** poruka
- Korisnički agent se takođe **brine o poštanskom sandučetu**
90



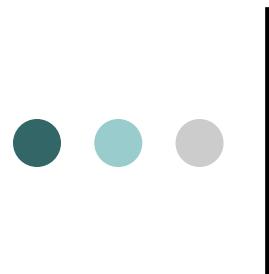
Korisnički agent

- Kreiranje poruka.
- Korisnički agent pomaže korisniku da **kreira poruku** koju želi da pošalje
- Većina korisničkih agenata **na ekranu monitora prikazuje šablon** kojeg **korisnik popunjava**
 - adresom primaoca,
 - temom (*subject*) i
 - sadržajem poruke



Korisnički agent

- Čitanje poruka.
- Drugi zadatak korisničkog agenta je da **omogući čitanje primljenih poruka**
- Kada pozve korisničkog agenta, korisnik **prvo proverava da li** u sadučetu primljenih poruka (**Inbox**) **ima novih poruka**
 - Većina korisničkih agenata **prikazuje spisak poruka prisutnih u Inbox-u**, gde za svaku poruku daje podatak o **veličini poruke, pošiljaocu poruke, status poruke** (indikaciju da li je poruka nova, već pročitana ali neodgovorena, pročitana i odgovorena) i **opciono temu (subject)** poruke



Korisnički agent

- Odgovaranje na poruku.
- Pošto je pročitao poruku, korisnik može da iskoristi korisničkog agenta da bi **odgovorio na poruku**
- Korisnički agent omogućava korisniku da **odgovor uputi samo pošiljaocu prvobitne poruke** ili **svima kojima je prvobitna poruka bila upućena**
- **Poruka odgovora** obično, osim novog teksta, **sadrži i tekst iz prvobitne poruke**



Korisnički agent

- Prosleđivanje poruka.
- Odgovor na poruku podrazumeva slanje poruke pošiljaocu ili svim primaocima primljene poruke.
- ***Prosleđivanje znači slanje primljene poruke, u neizmenjenom obliku, nekom trećem licu***



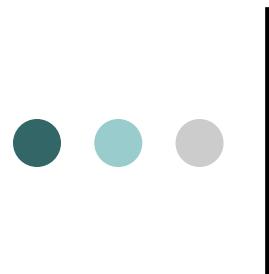
Korisnički agent

- **Mailbox-ovi.**
- Korisnički agent **obično kreira dva mailbox-a:**
 - za primljene poruke (**inbox**) i
 - poslate poruke (**outbox**)
- **Svaki mailbox je fajl** u specifičnom formatu
kojeg kreira i o **kome se stara** **korisnički agent**
- **Inbox** čuva primljene poruke sve dok ih korisnik ne obriše
- **Outbox** čuva sve poruke koje je korisnik poslao, sve dok ih korisnik ne obriše



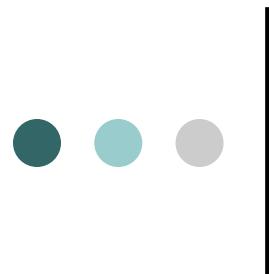
Korisnički agent - tipovi

- Postoje **dva tipa** korisničkih agenata:
- Komandni**
 - poziva se **sa komandne linije**, a zatim korisnik unosi **tekstualne komande** kojima agentu nalaže da izvrši željenu operaciju (npr. r ili R). *mail, elm,...*
- Grafički**
 - obezbeđuju **grafičke interfejse za interakciju** sa korisnikom. *Microsoft's Outlook, Eudora, Netscape*



Format elektronskog pisma

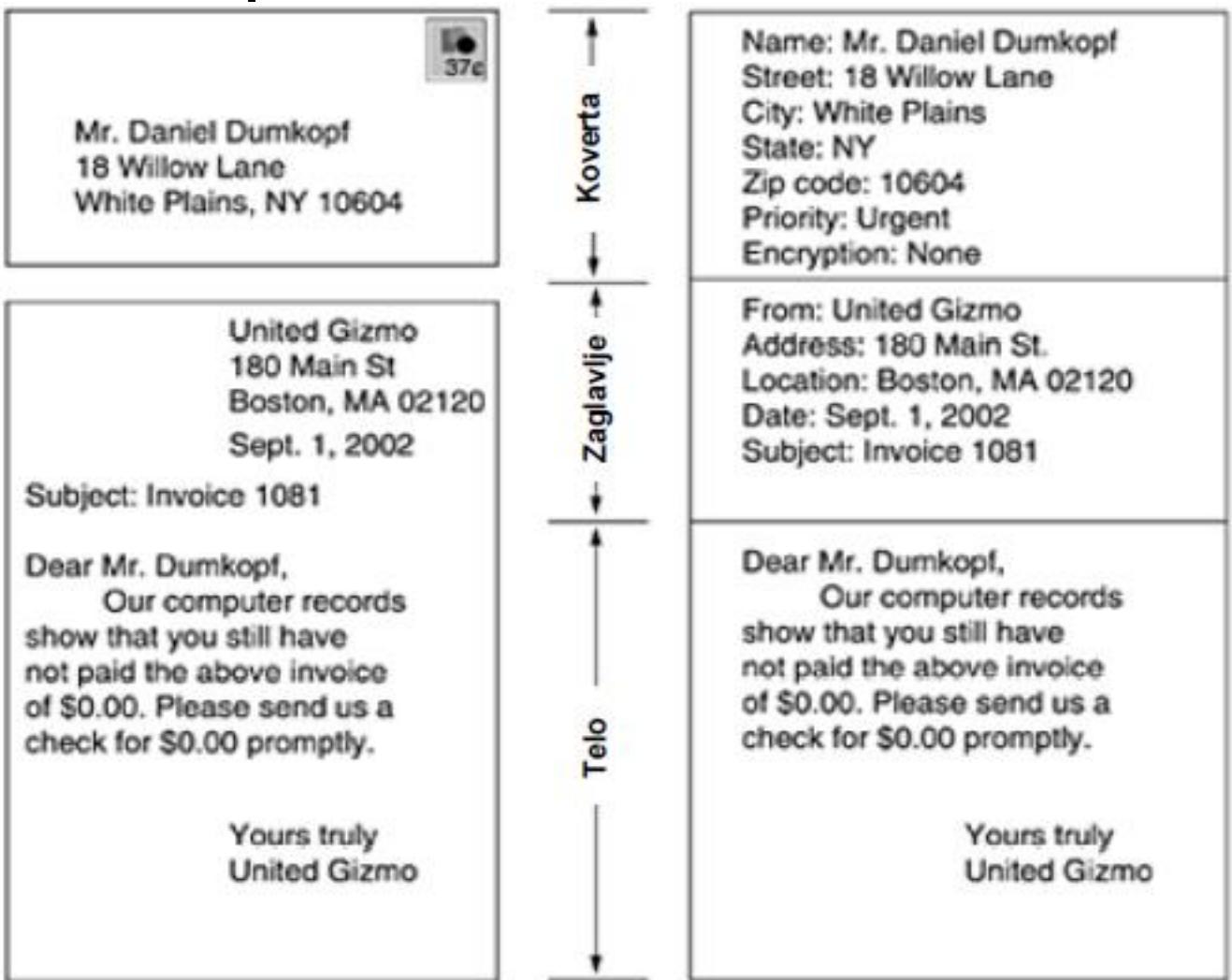
- Posredstvom korisničkog agenta, korisnik kreira **elektronsko pismo (e-mail)**
- Po strukturi ***nalikuje poštanskom pismu*** i
- Sastoji se iz (I) ***koverte*** i (II) ***poruke***
- ***Koverta*** sadrži obično
 - i. ***adresu pošiljaoca***,
 - ii. ***adresu primaoca*** i eventualno
 - iii. neke ***druge informacije***



Format elektronskog pisma

- **Poruka** se sastoji iz
- **Zaglavlja**
 - Zaglavljе poruke **definiše pošiljaoca, primaoca, temu poruke i neke druge podatke**
- **Tela**
 - Telo poruke **sadrži tekst** koji će pročitati primalac pisma

Format elektronskog pisma



Koverta
**koristi MTA za
isporuku poruke**

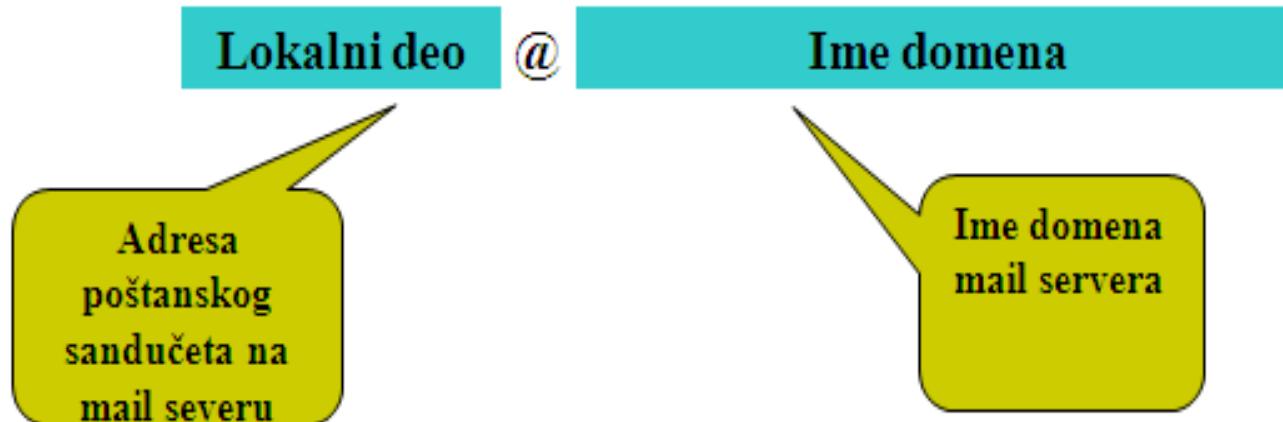
informacije za
korisničkog agenta

Poruka

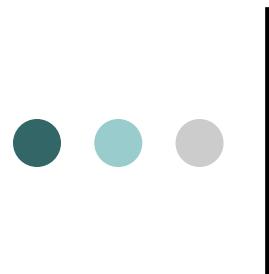
namenjeno **primaocu**
(čoveku)

e-mail adresa

- Da bi isporučio poruku, sistem elektronske pošte mora da poznaje adresu primaoca (tj. **e-mail adresu**)
- e-mail adresa** se sastoji **iz dva dela**:

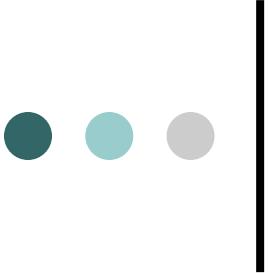


lokalni deo i ime_domena razdvojena znakom @



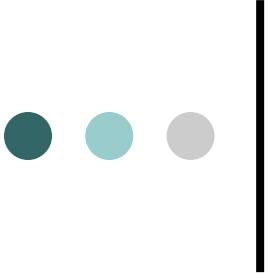
e-mail adresa

- **Lokalni deo** je *ime* jednog specijalnog fajla, koji se zove **korisničko poštansko sanduče** (*user mailbox*)
 - Tu se **čuvaju sve poruke** upućene konkretnom korisniku, dok ih on ne preuzme, **direktno ili posredstvom MAA agenta**
- **Drugi deo adrese** ukazuje na domen odredišta
 - Organizacije obično **izaberu jedan (ili više) hostova** koji će **prihvatići sve e-mail poruke** upućene na njihov domen



MIME

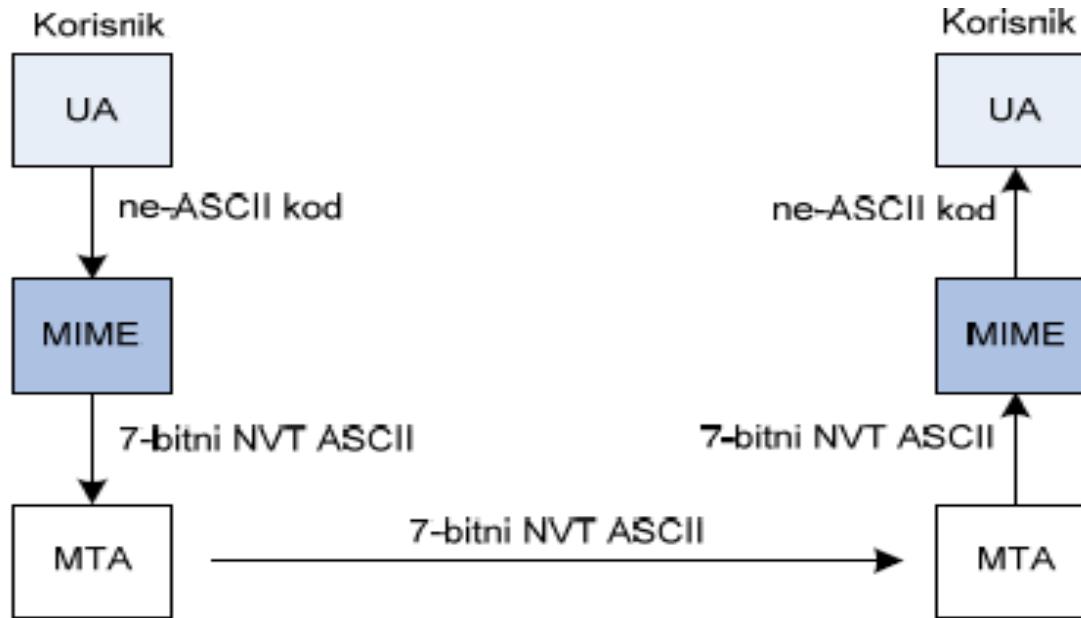
- Sistem elektronske pošte ima **jednostavnu strukturu**
- Međutim, **jednostavnost ima svoju cenu**: mogu se prenosi **isključivo tekstualne poruke** u **NVT** 7-bitnom **ASCII** formatu
 - Otuda potiču sledeća ograničenja: **nije moguće prenositi poruke** u jezicima koji **koriste alfabete sa apostrofima** (srpski, nemački, francuski); **ne-latinične alfabete** (ćirilica), poruke u jezicima koji **ne koriste alfabet** (kineski i japanski), a ni poruke sa **ne-tekstualnim sadržajem** (slike, audio, video)



MIME

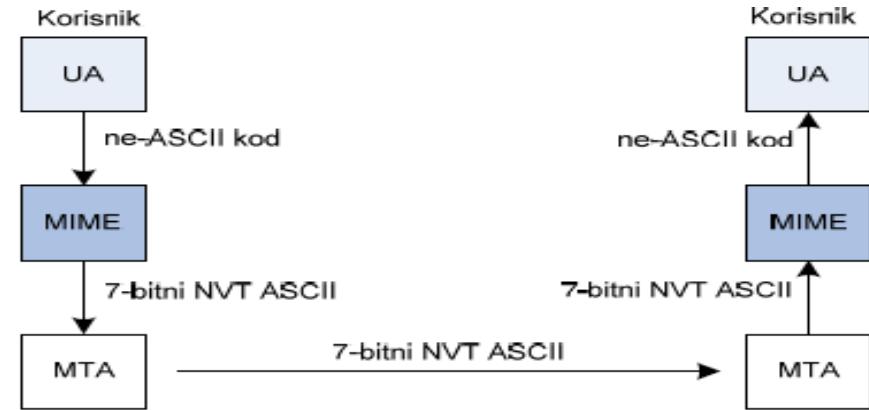
- **MIME** (*Multipurpose Internet Mail Extensions – višenamenska proširenja Internet pošte*) je dodatni protokol koje **omogućava prenos ne-ASCII sadržaja** u e-mail poruci
- **Na strani pošiljaoca**, MIME *transformiše ne-ASCII podatke u ASCII podatke*
 - **MTA klijent** isporučuje ih na identičan način **kao standardne, tekstualne poruke**
- **MTA server**, na strani primaoca, prihvata ASCII podatke i *prosleđuje ih MIME konvertoru* da ih prevede **u prvobitni oblik**

MIME



- **MIME** можемо замислiti као *skup funkcija za konverziju ne-ASCII podataka u ASCII i obrnuto*

MIME



- Sa podrškom za MIME, e-mail poruka može sadržati:
 - **više objekata** u jednom telu,
 - tekst **bez ograničena** u **dužini linija**
 - bez ograničena u **ukupnoj dužini**
 - skup **karaktera različit od ASCII**,
 - različite fontove,
 - binarne fajlove i
 - **m^{ultimedijalne} poruke** (slike, audio i video)

MIME zaglavlja

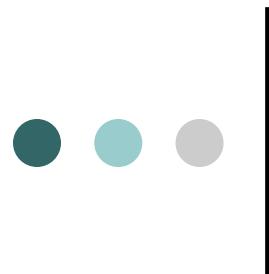
- ***MIME uvodi pet novih zaglavlja*** koji se mogu **pridodati originalnom zaglavljju** e-mail poruke kako bi se **definisali parametri transformacije**

Zaglavje	Značenje
MIME-Version:	Definiše verziju MIME standarda
Content-Description:	String koji opisuje sadržaj poruke
Content-Id:	Jedinstveni identifikator poruke
Content-Transfer-Encoding	Način kodiranja tela poruke
Content-Type:	Tip i format sadržaja poruke



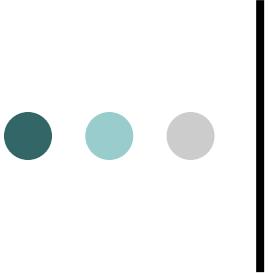
MIME zaglavlja

- Content-Type (**tip sadržaja**). Ovo zaglavje **definiše tip podataka** u telu poruke
- Standardom je definisano **nekoliko tipova**, od kojih **svaki ima jedan ili više podtipova**
- **Tip i podtip** su **razdvojeni kosom crtom**, npr.:
Content-Type: video/mpeg



MIME tipovi i podtipovi

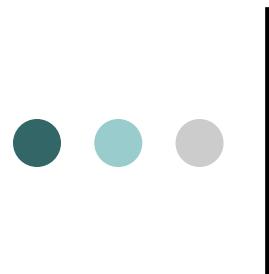
Tip	Podtip	Opis
Text	Plane	Neformatirani tekst
	Enriched	Tekst sa jednostavnim formatiranjem
	HTML	HTML format
Image	Gif	Slike u GIF formatu
	Jpeg	Slike u JPEG formatu
Audio	Basic	Zvuk
Video	Mpeg	Film u MPEG formatu



MIME tipovi i podtipovi

- **Text.** Koristi se za ***osnovni 7-bitni ASCII tekst.***
- Kombinacija **text/plane** je za obične, ***neformatirane poruke*** koje se mogu prikazati u obliku u kom su primljene, ***bez dodatnog procesiranja*** od strane MIME
- Podtip **text/enriched** znači da se u tekstu može koristiti jednostavan jezik za formatiranje teksta
 - Npr. boldirani delovi teksta, definiše poravnanje linija i sl.

Danas je **<bold>** lep **</bold>** dan bez **<italic>** snega **</italic>**

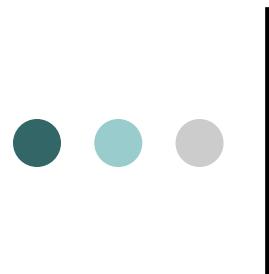


MIME tipovi i podtipovi

- **Text.** Koristi se za ***osnovni 7-bitni ASCII tekst.***
- Kada je Web postao popularan, dodat je novi podtip, **text/html** koji omogućava da se **e-mejlor
šalju Web stranice**

MIME tipovi i podtipovi

Tip	Podtip	Opis
Application	Octet-stream	sekvenca bajtova
	Postscript	dokument za štampanje u PostScript formatu
Message	Rfc822	MIME RFC 822 poruka
	Partial	Poruka je podeljena na delove
	External-body	Poruka se mora preneti nezavisno
Multipart	Mixed	Nezavisni delovi definisanom redosledu
	Alternative	Više poruka različitog formata
	Parallel	Delovi moraju biti prikazani istovremeno
	Digest	Svaki deo je jedna kompletan RFC 822 poruka



MIME tipovi i podtipovi

- **Application**. Ovaj tip se odnosi na **formate** koji **zahtevaju eksterno procesiranje** koje nije predviđeno nekim drugim tipom
 - Podtip **octet-stream** predstavlja proizvoljnu (neinterpretiranu) sekvencu bajtova (binarni fajl)
 - Po prijemu takvog sadržaja, korisnički agent obično predloži korisniku da dobijeni **sadržaj kopira u fajl**, a **naknadno procesiranje** se prepušta korisniku
 - Podtip **postscript** se odnosi na jezik PostScript
 - Široko se koristi za **opis stranica predviđenih za štampanje**. Mnogi **štampači imaju ugrađene interpretatore** za PostScript



MIME tipovi i podtipovi

- **Message**. Ovaj tip omogućava da *jedna poruka bude u potpunosti sadržana u nekoj drugoj*
- Podtip rfc822 se odnosi na slučaj kada *telo poruke sadži kompletnu klasičnu e-mail poruku* (zajedno sa njenim zaglavljem). Ova šema je korisna za prosleđivanje e-mail poruka
- Podtip partial se koristi ako je prvobitna *velika poruka razbijena na više e-mail poruka*. Svaka takva e-mail *poruka - fragment* sadži zaglavljje *message/partial* sa još tri dodatna parametra: *id*, *number* i *total*



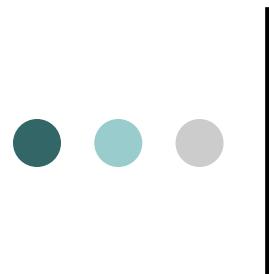
MIME tipovi i podtipovi

- **Message.**
- ***id*** definiše originalnu poruku i sadržano je ***u svim fragmentima***
- ***number*** (broj) definiše redni broj fragmenta,
- ***total*** definiše ukupni broj fragmenata
- ***MIME na odredištu*** prikuplja fragmente i ***rekonstruiše prvočitnu poruku***

MIME tipovi i podtipovi

- Podtip **external-body** se može koristi **za izrazito velike poruke** (npr. film)
- **Umesto** da se u e-mail poruku uključi **MPEG fajl**, **daje se FTP adresa** sa koje korisnički agent može da pribavi fajl
- **Tri prateća parametra** definišu ime fajla, njegovu lokaciju i način pristupa, na primer:

Content-Type: message/external-body;
name="IT.pdf";
site=es.viser.edu.rs/it;
access-type="ftp";



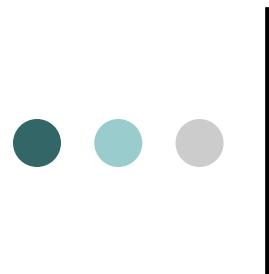
MIME tipovi i podtipovi

- **Multipart**. Telo poruke može sadržati **više jasno razdvojenih delova**
- Svrha ***multipart zaglavlja*** je da **definiše granice** između ovih delova
- Poseban ***parametar zaglavlja multipart*** definiše **granični string**



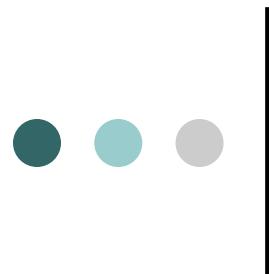
MIME tipovi i podtipovi

- **Multipart.**
- U telu poruke, ***svakom delu poruke prethodi linija koja sadrži granični string*** kome predhode ***dva znaka minus*** “--”
- ***Telo poruke se završava*** linijom koja sadrži ***graničnog stringa*** omeđen ***sa obe strane sa po dva znaka minus***



MIME tipovi i podtipovi

- **Multipart**. Tip *multipart* podržava **četiri podtipa**:
(I) ***mixed***, (II) ***parallel***, (III) ***digest*** i
(IV) ***alternative***
- **Podtip mixed** omogućava da ***svaki deo*** bude ***različitog tipa***
- Tip tog dela (tj. njegov **Content-Type**) naveden je **odmah nakon granične linje**



MIME tipovi i podtipovi

- **Multipart.**
- Sledeći primer pokazuje **poruku iz više delova mixed tipa:**

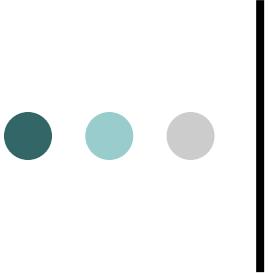
Content-Type: multipart/mixed;boundary=xxxx;

--xxxx

Content-Type:text/planer;

--xxxx

Content-Type:image/gif;



MIME tipovi i podtipovi

- **Multipart.**
- **Podtip parallel** se koristi kada **svi delovi moraju biti prikazani istovremeno**
- Na primer, **filmovi** često imaju razdvojene **audio i video kanale**, koje treba **istovremeno reprodukovati**
- **Podtip digest** se koristi u slučajevima kada je u e-mail poruku **upakovan veći broj drugih e-mail poruka**



MIME tipovi i podtipovi

- **Multipart.**
- **Podtip alternative** omogućava da *isti sadržaj bude uključen u poruku više puta*, ali u *različitim oblicima*
 - Na primer, **poruka može biti poslata u tri oblika**: kao običan **ASCII tekst**, kao **enriched tekst** i kao **PostScript**. **Korisnički agent** će *najpre pokušati* da poruku prikaže kao PostScript, ako je moguće. Druga mogućnost je *enriched*, a ako ni to nije u stanju, prikazaće ASCII tekst

Multipart poruka

Primer odgovara jednoj multimedijalnoj poruci

Content-Type: multipart/alternative; boundary=qwerqcewfqweffqe

--qwerqcewfqweffqe

Content-Type: text/enriched;

Happy birthday to you

Happy birthday to you

Happy birthday dear Tanja

Happy birthday to you

--qwerqcewfqweffqe

Content-Type: message/external-body;

access-type="ftp";

site="music.com"

name="birthday.snd"

Content-type:audio&basic

Content-transfer-encoding: base64

--qwerqcewfqweffqe--

Zaglavje
poruke

Telo
poruke

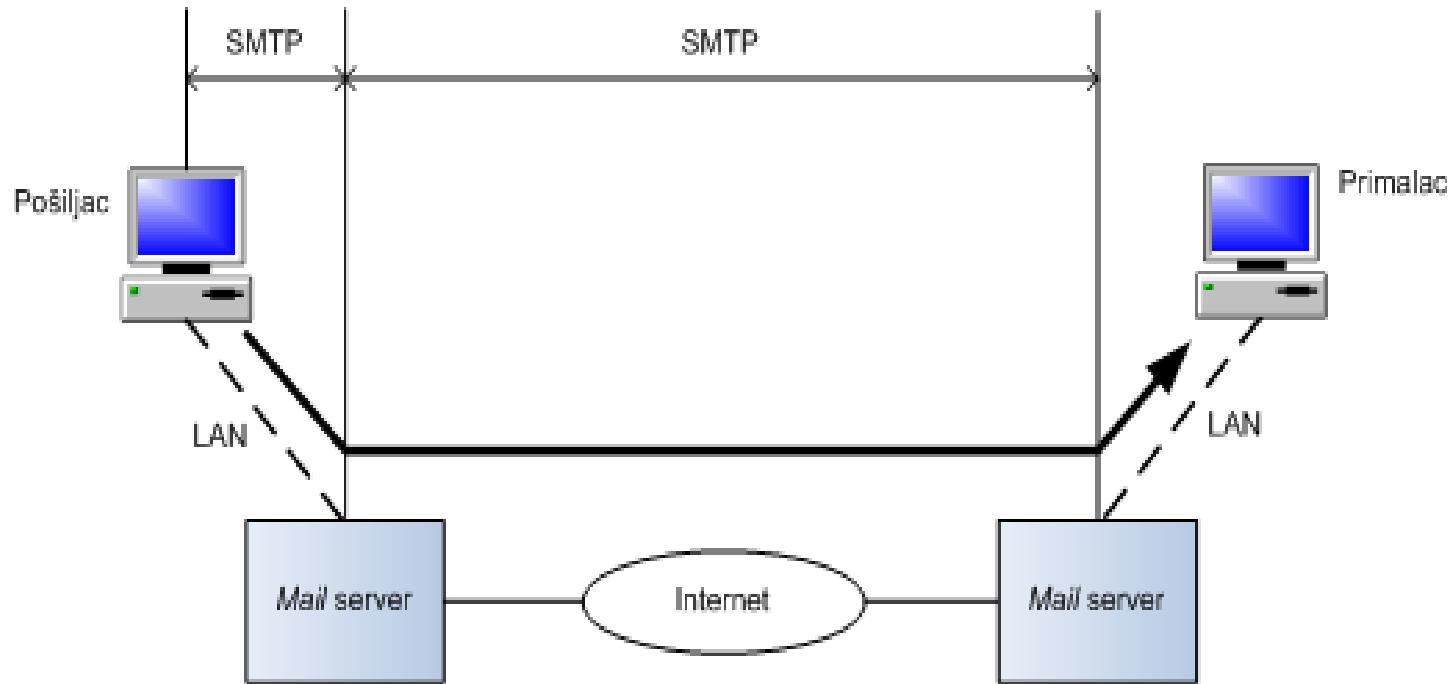


Agent za prenos poruka: SMTP

- **Prenos e-mail poruka** obavlja **agent za prenos poruka (MTA)**
- **MTA klijent** šalje, a **MTA server** prima poruku
- **SMTP** (*Simple Mail Transfer Protocol* - jednostavan protokol za prenos poruka) je formalni **Internet protokol** koje **realizuje MTA klijenta i servera**
- Za prenos e-mail prouka, u opštem slučaju (četvrti scenario), neophodna su dva para MTA klijent-server programa

Agent za prenos poruka: SMTP

domen SMTP protokola za četvrti scenario:



SMTP se **koristi dva puta**. Između **pošiljaoca poruke i njegovog mail servera**, kao i između **mail servera pošiljaoca i mail servera primaoca**¹²⁴

Agent za prenos poruka: SMTP

- SMTP **definiše komande i odzive** koji se razmenjuju *između MTA klijenta i servera*
- SMTP **koristi komande za prenos poruke** između MTA klijenta i servera





Agent za prenos poruka: SMTP

- **Komande**. Komande se šalju od klijenta ka serveru
- Format komande je jednostavan - komanda **počinje ključnom rečju** nakon koje **slede nula ili više argumenata**

Ključna_reč : argument(i)

- SMTP definiše **14 komandi**



Agent za prenos poruka: SMTP

Ključna reč	Argumenti
HELO	Ime hosta pošiljaoca
MAIL FROM	Pošiljalac poruke
RCPT TO	Primalac poruke
DATA	Telo poruke
QUIT	
RSET	
VRFY	Ime primaoca koje treba verifikovati
NOOP	
TURN	
EXPN	<i>Mailing lista koju treba proširiti</i>
HELP	Ime komande
SEND FROM	Primalac poruke
SMOL FROM	Primalac poruke
SMAL FROM	Primalac poruke

Prvih 5 su **obavezne**

Sledeće tri se
povremeno koriste

Poslednjih šest se
retko koriste



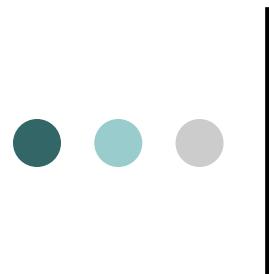
Agent za prenos poruka: SMTP

- **Odzivi.** Prenose ***od servera ka klijentu.*** Odziv je ***tro-cifarski numerički kôd.*** Posle kôda, u istoj liniji, ***može da sledi tekstualna informacija***
- Značenja ***prve cifre*** su:
 - ***2yz (konačni pozitivan odziv).*** Akcija je završena. Server je spremam da prihvati sledeću komandu
 - ***3yz (pozitivan među-odziv).*** Komanda je prihvaćena, ali su neophodne dodatne informacije



Agent za prenos poruka: SMTP

- **Odzivi.**
- Značenja prve cifre su:
 - **4yz (prolazni negativan odziv).** Akcija nije startovana, jer server trenutno nije u mogućnosti da je izvrši. Ista komanda može biti poslata kasnije.
 - **5yz (permanentni negativan odziv).** Komanda nije prihvaćena i ne bi trebalo pokušavati ponovo

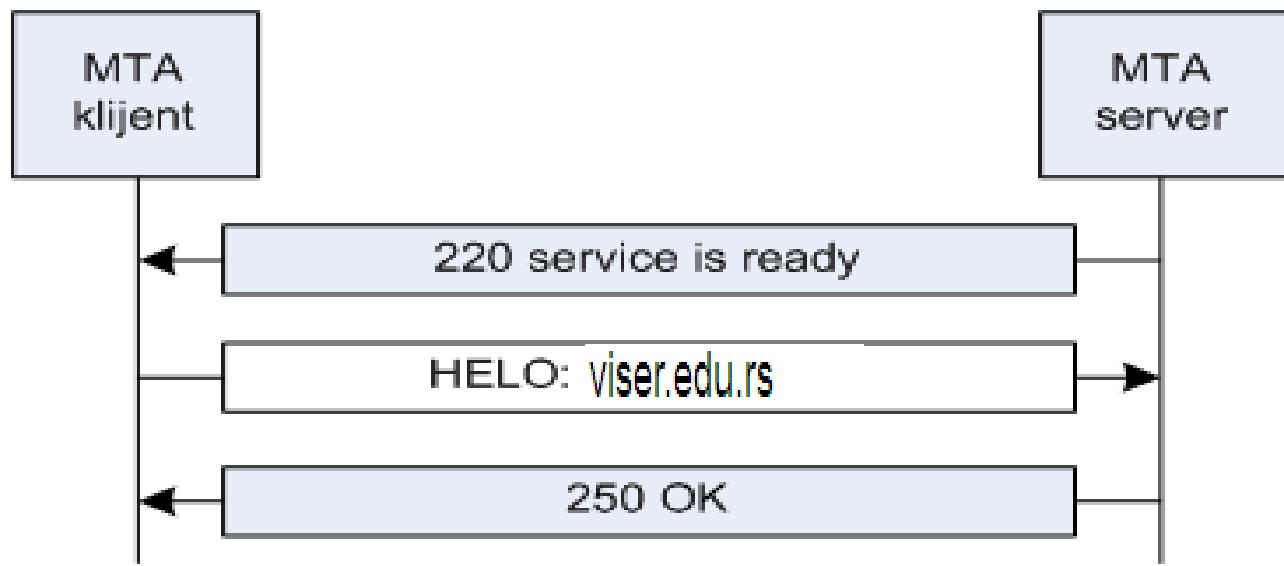


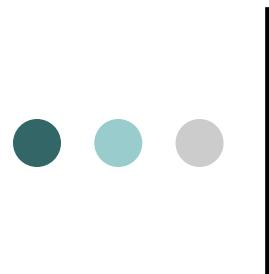
SMTP – faze prenosa poruke

- Proces prenosa e-mail poruke ostvaruje se u tri faze:
 - *uspostavljanje veze*,
 - *prenos poruke* i
 - *raskidanje veze*

SMTP – faze prenosa poruke

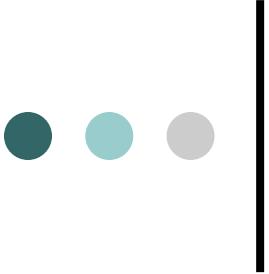
- **Uspostavljanje veze.** SMTP **server započinje** fazu uspostavljanja veze, **nakon što klijent uspostavi TCP konekciju** sa serverom **na dobropoznatom portu 25**. Faza sadrži sledeća ***tri koraka***





SMTP – faze prenosa poruke

- **Prenos poruke**. Nakon što je između klijenta i servera ***uspostavljena veza***, ***moguće je preneti jednu poruku*** između jednog pošiljaoca i jednog ili više primaoca. Ova faza ***sadrži osam koraka***.
- ***Koraci 3 i 4 se ponavljaju*** ako postoji ***više od jednog primaoca***



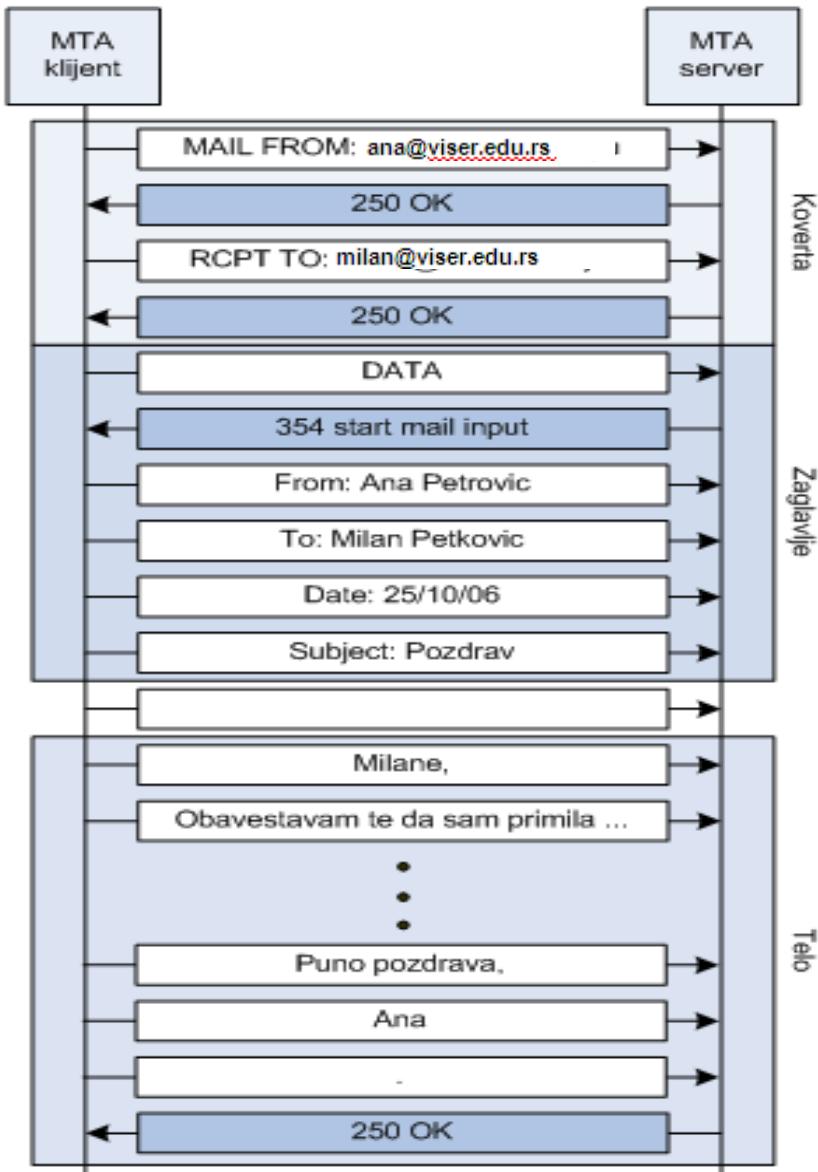
SMTP – faze prenosa poruke

○ **Prenos poruke.**

1. Klijent šalje komandu MAIL FROM kojom predstavlja pošiljaoca poruke. Komanda sadrži e-mail adresu pošiljaoca. Ovaj korak je potreban kako bi se serveru dostavila povratna e-mail adresa na koju će moći, u slučaju neke greške, da pošalje poruku sa obaveštenjem o problemu u isporuci poruke.
2. Server odgovara kôdom 250 (OK).
3. Klijent šalje komandu RCPT TO koja sadrži e-mail adresu primaoca.
4. Server odgovara kôdom 250 (OK).
5. Klijent šalje komandu DATA, kojim inicijalizuje prenos poruke.
6. Server odgovara kôdom 354 (*start mail input* - počni sa slanjem poruke).
7. U narednim linijama, klijent šalje sadržaj poruke. Svaka linija se završava kombinacijom specijalnih karaktera CR LF. Poruka se završava linijom koja sadrži samo jednu tačku.
8. Server odgovara kodom 250 (OK).

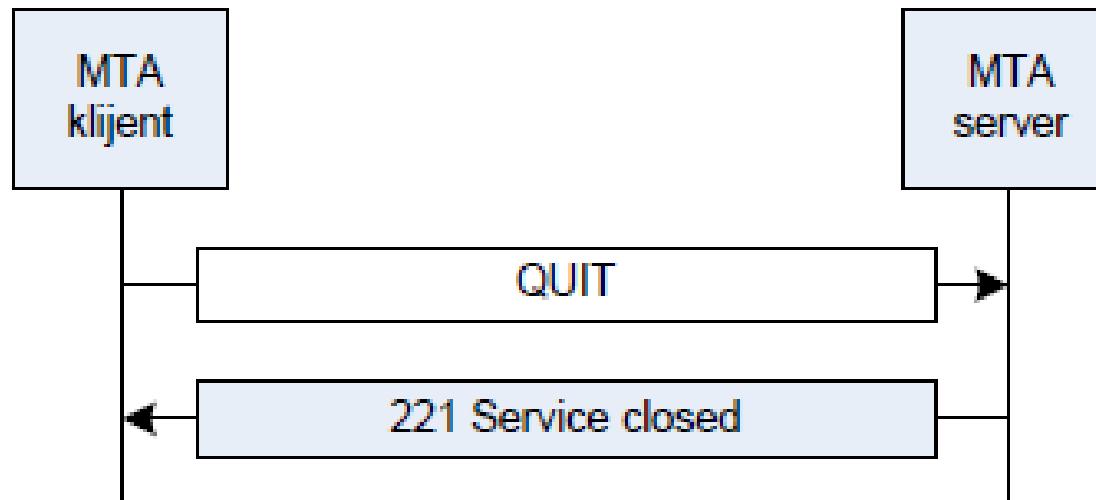
SMTP – faze prenosa poruke

○ *Prenos poruke.*



SMTP – faze prenosa poruke

- **Raskidanje veze**. Nakon što je poruka uspešno preneta, **klijent raskida vezu sa serverom**
- Ova faza uključuje **dva koraka**:



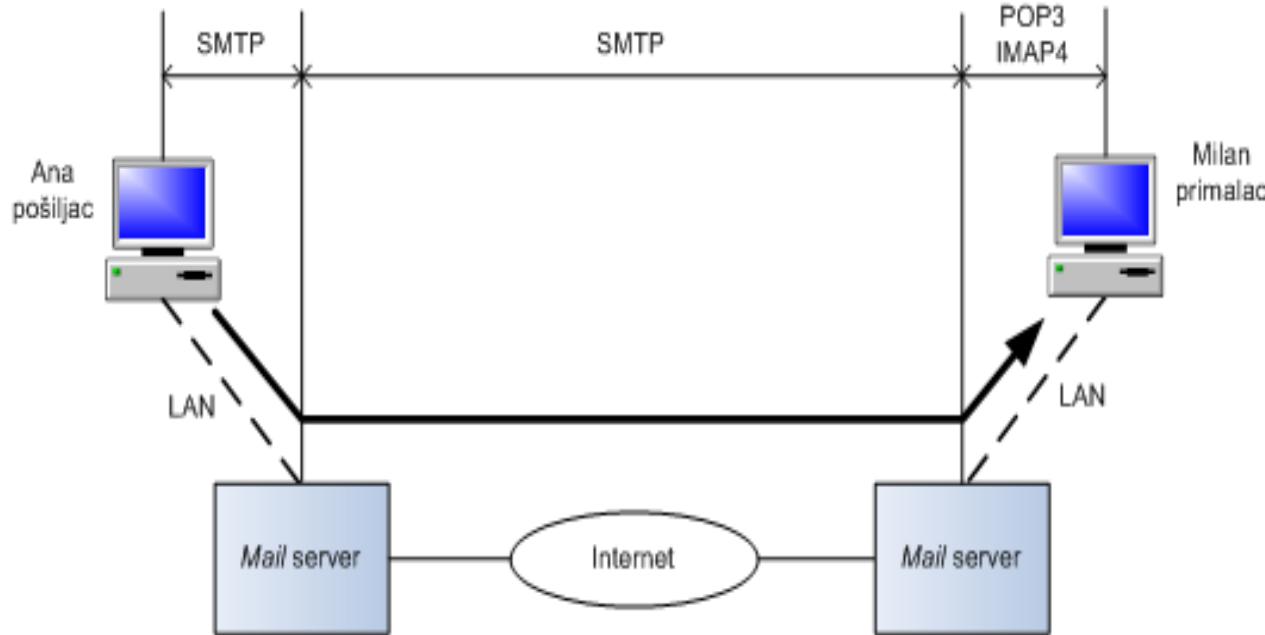


Agent za preuzimanje poruka

- **Isporuka** e-mail poruke od pošiljaoca do primaoca, u najopštijem slučaju, uključuje **tri faze**
 - **klijentski računar** šalje poruku **svom mail serveru**
 - **klijentski računar** isporučuje poruku **mail serveru primaoca**
 - **primalac** preuzima poruku od **svog mail servera**
- SMTP se koristi u prvoj i drugoj, ali ne i u trećoj fazi

Agent za preuzimanje poruka

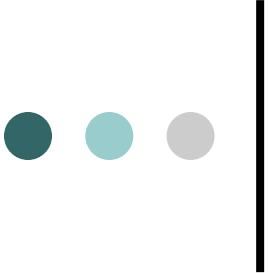
- **SMTP** je protokol ***push tipa*** koji omogućava klijentu **da pošalje** (preda) poruku severu
- Međutim, ***u trećoj fazi*** neophodan je protokol ***pull tipa***: klijent **preuzima** (traži) poruku od servera. U trećoj fazi koristi se ***agent za prisup porukama (MAA)***





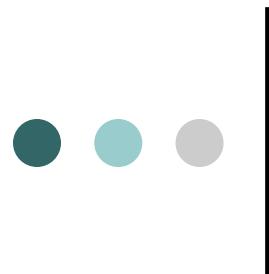
Agent za preuzimanje poruka

- Danas su u upotrebi **dva MAA protokola**:
- **POP3** (*Post Office Protocol, version 3* - Poštanski protokol, verzije 3) i
- **IMAP4** (*Internet Mail Access Protocol, version 4* - Internet protokol za pristup e-mail-ovima, verzije 4)



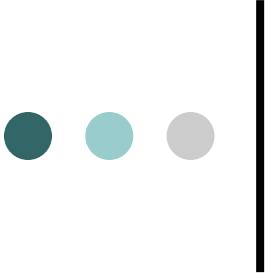
POP3

- POP3 je **jednostavan** protokol za **preuzimanje poruka ograničene funkcionalnosti**
- **POP3 klijent** je instaliran **na računaru primaoca**, a **POP3 server** na **mail serveru**
- **POP3 počinje sa radom** kada **korisnik startuje program za email-ove**, sa namerom **da preuzme pristigle e-mail poruke** iz svog poštanskog sandučeta na **mail serveru**
- Program za e-mail-ove se obraća ***POP3 klijentu*** koji **uspostavlja TCP konekciju** na portu **110** sa POP3 serverom



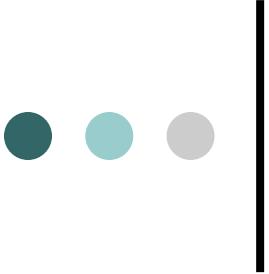
POP3

- *Po uspostavljanju konekcije, POP3 obavlja sledeća tri koraka:*
 - Autorizacija
 - Transakcije
 - Ažuriranje



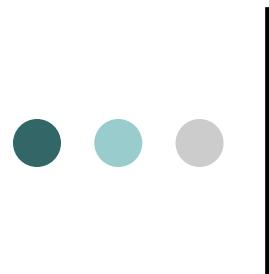
POP3

- **Autorizacija** se odnosi na ***proveru prava pristupa*** korisnika *mailbox-u* na mail serveru, što uključuje ***slanje korisničkog imena i lozinke***
- **Transakcija** omogućava korisniku ***da pregleda sadržaj i preuzme*** e-mail poruke, jednu po jednu, iz svog *mailbox-a*
- POP3 podržava ***dva režima rada:***
 - režim ***brisanja*** i
 - režim ***čuvanja*** e-mail-ova



POP3

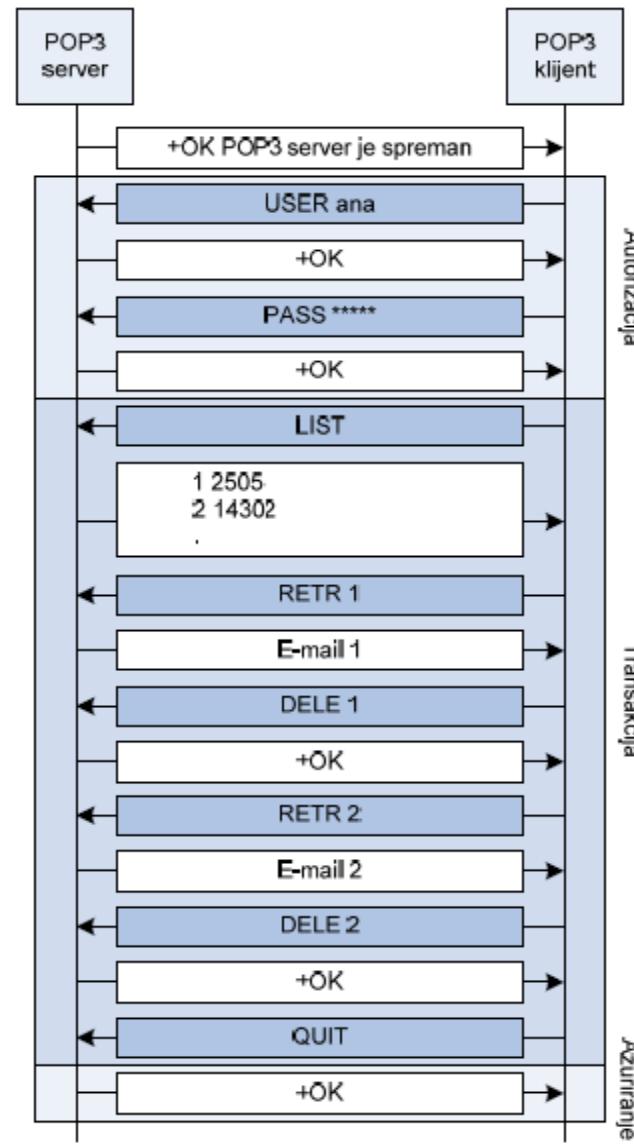
- **U režimu brisanja**, nakon preuzimanja, e-mail poruka se **označava za brisanje**
- U fazi **ažuriranja**, e-mail-ovi **označeni za brisanje se odstranjuju** iz *mailbox-a*
- **U režimu čuvanja**, preuzeti e-mail-ovi **ostaju u korisnikovom mailbox-u**
- **Režim brisanja** se obično koristi kada korisnik radi na svom stalnom računaru na kome čuva i organizuje sve svoje e-mail poruke



POP3

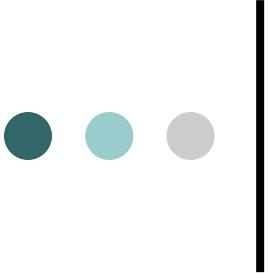
- **Režim čuvanja** se obično koristi kada korisnik pristupa svojim **e-mail-ovima sa nekog drugo računara**, kojeg **privremeno koristi**
 - Korisnik je preuzeo i pročitao e-mail, ali umesto da bude odstranjen, *e-mail* je ostao u *mailbox-u* na *mail serveru*, tako da kasnije ponovo može biti preuzet i eventualno prenet na korisnikov stalni računar

POP3 - Primer



POP3 - Primer

Nakon što klijent uspostavi TCP konekciju sa serverom na portu 110, server se prvi javlja slanjem ASCII poruke. Ova poruka obično sadrži +OK, na početku i neki komentar u nastavku. U toku faze autorizacije, klijent šalje svoje korisničko ime, a onda i lozinku (komande USER i PASS). Nakon uspešne autorizacije, klijent prelistava sadržaj svog *mailbox*-a pomoću komande LIST. Server vraća spisak poruka (svaka poruka u jednoj liniji) zajedno sa veličinama poruka. Spisak se završava tačkom. Klijent, koji sada poseduje spisak, preuzima poruke komandom RETR navodeći redni broj poruke kao parametar. U datom primeru, korisnik je obe preuzete poruke označio za brisanje (komandom DELE). Po što je preneo sve svoje poruke, klijent završava fazu transakcije komandom QUIT. Po priјemu ove komande server ulazi u fazu ažuriranja, kada iz korisnikovog *mailboxa* briše sve poruke označene za brisanje. Nakon završenog brisanja poruka, server vraća odziv (+OK) i zatvara TCP konekciju.



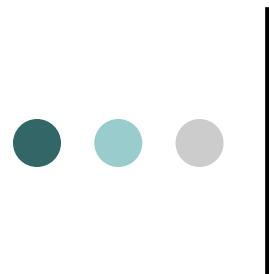
IMAP4

- *POP3 ima **nekoliko nedostataka***
- POP3 *ne omogućava korisnicima da organizuju svoje e-mail poruke* na mail serveru
- POP3 ne pruža mogućnost korisnicima da *delimično preuzmu e-mail* pre nego što odluče da li će ga preuzeti u celosti ili obrisati
- Protokol *IMAP4* je **sličan** protokolu *POP3*, ali ima **više funkcija**



IMAP4 – dodatne mogućnosti

- Korisnik može **proveriti zaglavje e-mail-a** pre nego što ga preuzme
- Korisnik može **pretraživati e-mail tražeći u njemu zadati string** karaktera pre nego što ga preuzme
- Korisnik može **delimično preuzeti e-mail**
 - Ova mogućnost je korisna, ako se za vezu sa *mail serverom* koristi *dila-up* modem, a u *e-mail* je prisutan multimedijalni sadržaj koji zahteva daleko veću propusnost radi preuzimanja



IMAP4 – dodatne mogućnosti

- Korisnik može ***da kreira, briše i preimenuje mailbox-ove na mail serveru.***
- Korisnik može da ***kreira hijerarhijski organizovane mailbox-ove***