Programiranje korisničkih interfejsa:

Lekcija 6: Web serveri

zima 2019/2020

Branimir M. Trenkić

Uvod

- ***Web serveri*** omogućuju ***HTTP pristup*** ***kolekciji dokumenata*** i drugim informacijama
- ***Unutrašnja organizacija dokumenata*** – ***struktura stabla***, slično sistemu datoteka u računaru
- Omogućuju ***pristup***:
  - ***a)*** ***statičkim*** dokumentima,
  - ***b)*** ***dinamičkim*** sadržajima
    - Koriste ***različite protokole*** za slanje zahteva do posebnih ***softverskih aplikacija*** koje omogućuju pristup dinamičkim sadržajima

Uvod

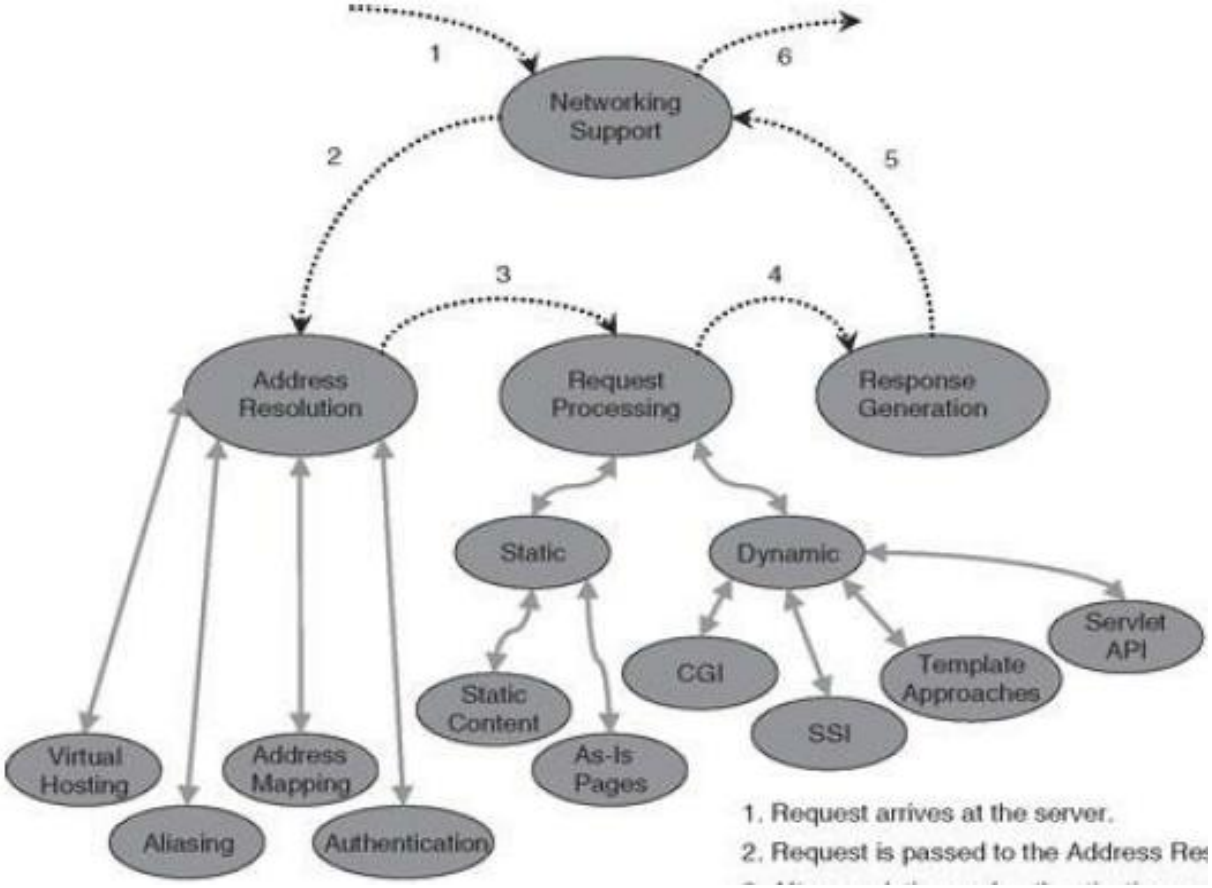
- Ovu ***temu počinjemo*** opisom ***procesa servisiranja zahteva statičkim dokumentima***
- Nakon toga, fokusiraćemo se na ***mehanizme*** koji ***omogućuju pristup dinamičkim sadržajima*** na serveru

Osnovna operacija

- ***Web serveri***
  a) ***primaju*** i ***interpretiraju*** ***HTTP zahteve***,
  b) ***lociraju*** i ***pristupaju*** zahtevanim ***resursima*** i
  c) ***kreiraju*** ***odgovore*** koji se potom ***šalju nazad*** do inicijatora zahteva
- ***Proces (I) interpretacije dolaznog zahteva i (II) kreiranja odlaznog odgovora*** - ***glavni proces*** koji se realizuje na web serveru
  - Predstavlja glavnu temu našeg današnjeg predavanja

Osnovna operacija

- ***Blok šema*** procesa (***1***) ***obrade*** dolaznog zahteva (***2***) ***kreiranja*** odgovora i (***3***) ***slanja*** odgovora nazad



1. Server prima zahtev
2. Zahtev se prosleđuje do **Modula za razrešavanje adrese**
3. Nakon razrešavanja i autentifikacije, zahtev se prosleđuje do **Modula za obradu zahteva**
4. Kontrola se ustupa **Modulu za generisanje odgovora**
5. Odgovor se prosleđuje **Mrežnom modulu**
6. Odgovor se šalje nazad do inicijatora zahteva

Osnovna operacija

Mrežni modul

- Odgovoran je za ***prijem zahteva*** i za ***slanje odgovora*** preko mreže
- Nakon prijema, ovaj modul prosleđuje zahtev do ***Modula za razrešavanje adrese*** koji je zadužen za ***analizu*** i ***“pre-procesiranje”*** zahteva

Osnovna operacija

Modul za razrešavanje adrese

- Zadužen je za ***analizu*** i ***“pre-procesiranje”*** zahteva
- Pre-procesiranje obuhvata:
  - ***Virtuelni hosting***
  - ***Preslikavanje adrese***
  - ***Autentifikaciju***

Osnovna operacija

Modul za razrešavanje adrese

- Virtuelni hosting:
- Web server može da ***pruža usluge za više domena***,
- Uloga ovog modula je ***određivanje ciljanog domena*** za prispeli zahtev
- Ova informacija se potom ***koristi*** kako bi se ***izabrali konfiguracioni parametri*** koji odgovaraju ciljanom domenu

Osnovna operacija

Modul za razrešavanje adrese

- Preslikavanje adrese:
- Na osnovu (***I***) ***URL-a*** i (***II***) selektovanih ***konfiguracionih parametara***:
  - utvrđuje se da li dati zahtev ukazuje na ***statički ili dinamički sadržaj*** i
  - ***razrešava se adresa***
    - ***Određivanje aktuelne lokacije*** traženog ***resursa*** u okviru serverovog sistema datoteka

Osnovna operacija

Modul za razrešavanje adrese

- Autentifikacija:
- Ako je ***traženi resurs zaštićen***, ***ispituju se autorizacioni kredencijali*** kako bi se utvrdilo da li je zahtev došao od autorizovanog korisnika

- Kada se pre-procesiranje završi - zahtev se šalje ***Modulu za obradu zahteva***

Osnovna operacija

Modulu za obradu zahteva

- ***Poziva pod-module*** koji ga ***opslužuju*** sa odgovarajućim statičkim ili dinamičkim sadržajem
- Kada selektovani pod-moduli kompletiraju obradu - šalje rezultat ***Modulu za generisanje odgovora***

Modulu za generisanje odgovora

- ***Kreira odgovor*** i prosleđuje ga ***Mrežnom modulu*** za slanje

Osnovna operacija – Važna napomena

- ***HTTP*** je ***stateless protokol***
- Jedine ***informacije koje se odnose na zahtev*** serveru – ***dostupne su u okviru*** samog zahteva
- Stanje može biti ***podržano od strane*** serverskog dela ***aplikacije*** u formi
  - informacije o sesiji ili
  - aplikacionom okruženju (npr. ***promenljive okruženja***)

  - ali ***reference*** koja ukazuju na te informacije

    moraju biti sadržane u samom zahtevu

Obrada HTTP zahteva

- ***Šta sve treba da se desi kako bi zahtev stigao do servera?*** (vidi Lekciju 5!)
- ***Razmotrimo seriju transakcija*** u kojima korisnik posećuje web- sajt koji se nalazi na adresi http://mysite.org/
- ***Proces započinje*** nalogom korisnika pretraživaču da ***pristupi stranici*** čiji je URL http://mysite.org/pages/simple-page.html

```
GET http://mysite.org/pages/simple-page.html HTTP/1.1
Host: mysite.org
User-Agent: Mozilla/4.75 [en] (WinNT; U)
```

Obrada HTTP zahteva

- Nakon prijema i prikaza stranice - korisnik uočava da ona ***sadrži linkove*** do druge ***dve stranice***, ***school.html*** i ***home.html***

```
<HTML>
<HEAD><TITLE>Simple Page</TITLE></HEAD>
<BODY>
<H2>My Links</H2>
<UL>
<LI><A href='school.html'>My school links</A></LI>
<LI><A href='home.html'>My home links</A></LI>
</UL>
</BODY>
</HTML>
```

- Neka ***korisnik sledi link*** do ***school.html***

Obrada HTTP zahteva

- Zahtev ne sadrži ***Connection: close*** zaglavlje
- ***Veza*** sa serverom ***će ostati otvorena*** tako da se može ***iskoristiti za slanje sledećih*** zahteva i prijem odgovora
- Međutim, to ***ne garantuje*** da će ***veza ostati otvorena*** i u trenutku slanje zahteva za ***school.html***

```
GET http://mysite.org/pages/school.html HTTP/1.1
Host: mysite.org
User-Agent: Mozilla/4.75 [en] (WinNT; U)
```

Obrada HTTP zahteva

- U tom trenutku, server, proxy ili čak i pretraživač mogu ***raskinuti vezu***
- ***Perzistentne veze*** su koncipirane da bi ***poboljšale performance*** ali se nikada ne treba oslanjati na logiku aplikacije

Obrada HTTP zahteva

- Ako je ***veza*** još uvek ***otvorena*** - ***pretraživač*** je ***koristi da pošalje zahtev*** za stranicu (***school.html***)
- ***Inače*** - ***pretraživač*** mora ***ponovo uspostaviti vezu***
- Zavisno od toga kako je konfigurisan pretraživač ***uspostavlja vezu***
  - direktno do server ili
  - preko proxy-ja
- Tako, pretraživač prima zahtevanu stranicu ***od proxy***-ja ***ili*** direktno ***od servera***

Obrada HTTP zahteva

- U slučaju ***perzistentnih veza***, ***server je odgovoran*** za
- ***uspostavljanje redova čekanja*** za zahteve i odgovore ***za svakog klijenta posebno***
- ***HTTP/1.1*** - u okviru jedne neprekidno otvorene veze, ***moguće je poslati više zahteva***
- Takođe, ***odgovori na te zahteve*** moraju biti ***poslati u istom redosledu*** kako su zahtevi pristigli (***FIFO***)

Obrada HTTP zahteva

- ***Rešenje***:- ***Server uspostavlja*** i ***dolazni*** i ***odlazni red čekanja*** za zahteve
- Kada se ***zahtev uputi na obradu*** - on se
  - ***isključuje*** iz dolaznog reda čekanja i
  - ***uključuje*** u odlazni red čekanja
- Kada se ***obrada završi***, zahtev
  - ***markira se*** da je spreman za slanje, ali
  - ***ostaje*** u odlaznom redu čekanja

Obrada HTTP zahteva

- Kada su ***svi prethodnici opsluženi*** –
  - zahtev se ***uklanja iz odlaznog reda*** čekanja a
  - ***pridruženi odgovor se šalje*** nazad ka pretraživaču ili direktno ili preko proxy-ja

Statički Web dokumenti

- Postoje ***dve kategorije statičkih*** web ***dokumenata*** (stranica):
- ***Stranica sa statičkim sadržajem***
  - statičke datoteke koje sadrže ***HTML stranice***, ***XML stranice***, ***plain tekst***, ***slike***,….
  - za njih se moraju ***kreirati HTTP odgovori***, uključujući i zaglavlja;
- ***As-in stranica***
  - statičke datoteke koje ***sadrže kompletne HTTP odgovore*** (uključujući zaglavlja)

Stranica sa statičkim sadržajem

- Server preslikava URL ***u*** lokaciju datoteke relativno u odnosu na koren (root) direktorijum za dokumenata u lokalnom sistemu datoteka
- U našem primeru, korisnik posećuje stranicu http://mysite.org/pages/school.html
- ***Putanja*** u tom URL-u: ***/pages/school.html***
- Ona se preslikava ***u konkretnu putanju do datoteke*** u lokalnom sistemu datoteka servera

Stranica sa statičkim sadržajem

- Na primer, ako je ***server konfigurisan*** tako da je ***koren (root) direktorijum za dokumenata*** - ***/www/doc***,
- tada je URL preslikan u ***/www/doc/pages/school.html*** u sistemu datoteka

Stranica sa statičkim sadržajem

- Za statičke stranice, ***server mora***:
  - ***preuzeti kopiju*** datoteke,
  - ***kreirati odgovor*** i
  - ***poslati ga*** nazad ka pretraživaču
    - U slučaju perzistentne veze - ***odgovor se uključuje u odlazni red čekanja*** pre slanja

Stranica sa statičkim sadržajem

Odgovor na zahtev sa slajda 15:

```
HTTP/1.1 200 OK
Date: Tue, 29 May 2001 23:15:29 GMT
Last-Modified: Mon, 28 May 2001 15:11:01 GMT
Content-type: text/html
Content-length: 193
Server: Apache/1.2.5

<HTML>
<HEAD><TITLE>School Page</TITLE></HEAD>
<BODY>
<H2>My Links</H2>
<UL>
<LI><A href="classes.html">My classes</A></LI>
<LI><A href="friends.html">My friends</A></LI>
</UL>
</BODY>
</HTML>
```

Prva linija odgovora sadrži ***statusni kod*** koji sumira rezultat operacije

Server kontroliše predstavljanje sadržaja odgovora na strani pretraživača kroz ***Content-Type zaglavlje*** ***setovanjem*** odgovarajućeg ***MIME tipa***

Stranica sa statičkim sadržajem

- ***Server kontroliše predstavljanje sadržaja odgovora na strani pretraživača*** kroz ***Content-Type zaglavlje*** - ***setovanjem*** odgovarajućeg ***MIME tipa***
- ***Način setovanja MIME tipa*** za statičke datoteke je ustanovljen ***kroz konfiguraciju servera***

Stranica sa statičkim sadržajem

- ***Najjednostavniji način*** – zasniva se na ***preslikavanju između ekstenzije datoteka i MIME tipova***
- ***Pretraživač ima istu mogućnost*** preslikavanja
- Ali, ***server*** je taj ko kroz ***Content-Type zaglavlje*** odgovora ***definiše preslikavanje***
- ***Ovo zaglavlje odlučuje*** o načinu kako će pretraživač prikazati telo odgovora

Stranica sa statičkim sadržajem

- Server može, takođe, da postavi i ***Content-Length zaglavlje***
- Ovo zaglavlje je ***opciono*** i može se ***izostaviti kod dinamičkog sadržaja*** zato što ***je teško odrediti dužinu*** odgovora pre nego što se završi process kreiranja
- Mada, ***ako je moguće*** predvideti dužinu odgovora, ***dobro je uključiti ovo zaglavlje***
- To omogućuje pretraživaču korektno izveštavanje o ***progresu preuzimanja*** sadržaja

Stranica sa statičkim sadržajem

- Već smo diskutovali o HTTP podršci za keširanje
- Upoznali smo se sa ***Last_Modified zaglavljem*** i njegovoj ***upotrebi u logici pretraživača***
  - u formiranju zahteva
  - korišćenju lokalno keširanog sadržaja
- Iako ***Last_Modified*** zaglavlje ***nije obavezno***, ***od servera*** se očekuje da ***odredi vreme poslednje promene*** zahtevanog sadržaja i koristiti ga u postavljanju ovog zaglavlja

as-is stranice

- Omogućuje ***isključenje serverove logike u formiranju zaglavlja*** odgovora i statusnog koda
- ***Razlozi*** za to:
  - ***Testiranje*** pretraživača ili proxy-ja,
  - Potreba za ***brzo*** (***bez detaljne analize***) postavljanje određene Content-Type vrednosti za neke stranice
  - ***Lak*** i pogodan ***način*** regularne ***promene cilja redirekcije*** za određene sadržaje
- Postoji ***način rešavanja*** svih ***ovih situacija*** – ***koristi as-is stranice***

as-is stranice

- ***Ideja***:
- ***Stranice*** (***datoteke***) ***sadrže kompletni odgovor*** - i kao takve ih server šalje nazad – ***bez dodavanja*** statusnog koda ili zaglavlja
- ***Ako je u datoteci smešten odgovor***, koja je od strane server ***prepoznata kao as-in datoteka***, ***to će garantovati*** da kada se zahteva ta datoteka, server vraća njen sadržaj kao odgovor bez promene sadržaja

as-is stranice

- ***Mehanizam za prepoznavanje*** as-is datoteka je vrlo ***sličan*** mehanizmu preslikavanja ***statičkog sadržaja u MIME tipove***
- Čest slučaj je da se server konfiguriše tako da preslikava ***.asis*** ***ekstenziju*** datoteke ***u as-is obradu***
- ***Razlika*** u odnosu ***na statički sadržaj*** je, da server ***umesto da formira Content-Type zaglavlje***, oslanja se na Content-Type postavljenom u as-is datoteci

as-is stranice

- as-is mehanizam možemo posmatrati kao ***način upravljanja serverovim odgovorima*** kroz ***manuelno*** kreiranje i modifikovanje odgovora
- Upotreba reči “manuelno” je adekvatno – kad god ***želimo da promenimo odgovor***, moramo ***editovati as-is datoteku***
- Pogodan za jednostavne scenarije ali ***ne pruža mogućnosti implementacije*** čak i osnovne ***logike obrade***

Dinamički web dokumenti

- ***U početnom periodu*** razvoja Web-a ***celokupan sadržaj dostupan na Web-u bio je statički***
- Međutim, ***u novije vreme***, ***sadržaj*** koji se može preuzeti ***sa Web-a postaje u sve većoj meri dinamički***
  - ***Ne postoji*** u unapred definisanom obliku, već se ***kreira na zahtev***

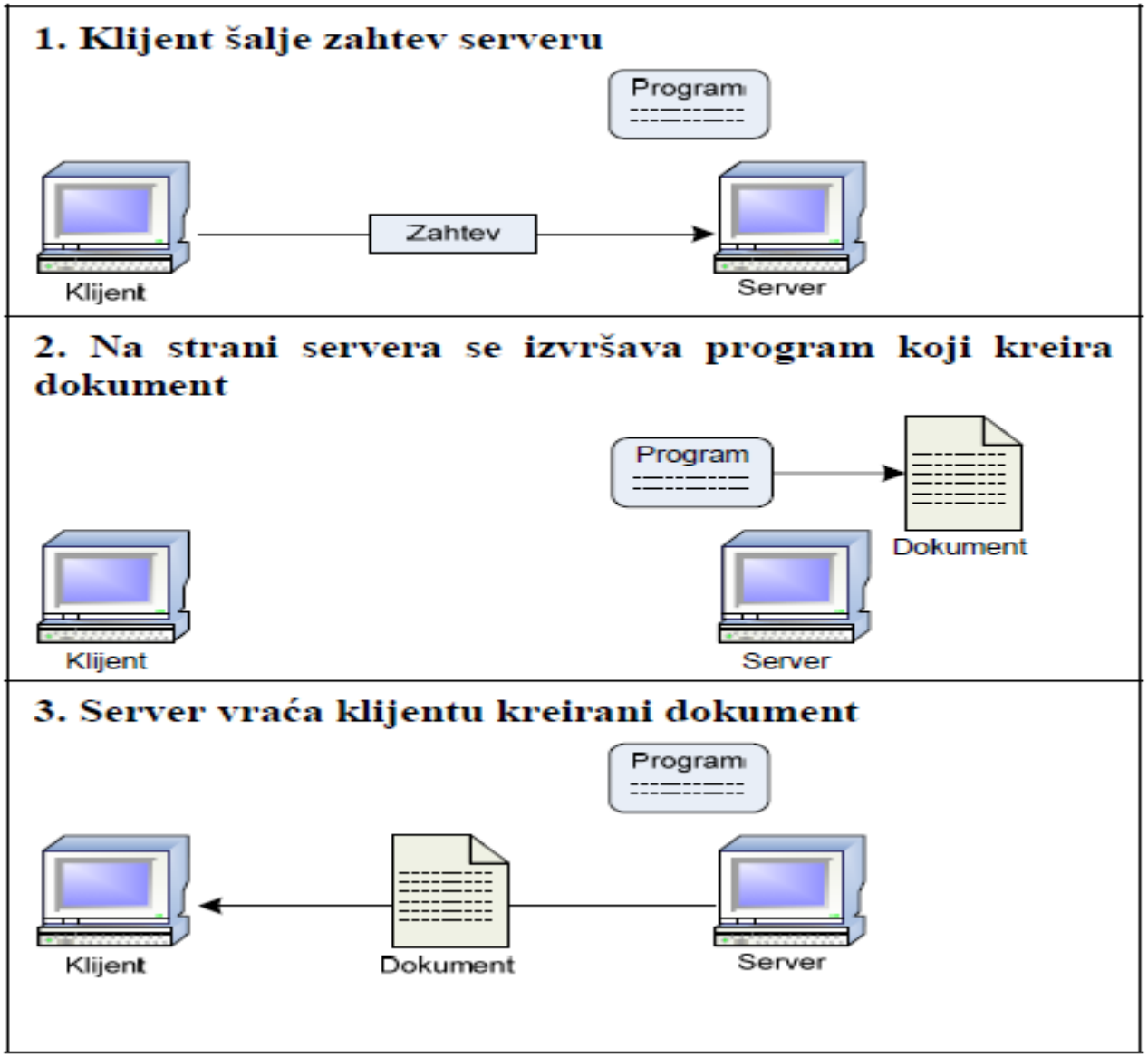
Dinamički web dokumenti

- ***Generisanje*** sadržaja Web stranica ***može se obaviti***:
  - na strani servera - ***Dinamički web dokumenti***
  - na strani klijenta - ***Aktivni web dokumenti***

Dinamički web dokumenti

- U prvom slučaju, tzv. *dinamički web dokumenti*, ***web server***, ***po prijemu zahteva***,
  - ***pokreće*** odgovarajući ***aplikacioni program koji kreira dinamički dokument***, a
  - ***klijentu vraća izlaz programa***
- Kao odgovor ***na svaki zahtev klijenta***, na strani severa se ***kreira nova verzija HTML dokumenta***

Dinamički web dokumenti



Dinamički web dokumenti

- Na primer, klijent može da zahteva od servera da izvrši program koji čita sistemsko vreme serverskog računara
- Program konvertuje informaciju o vremenu u tekst,
- Formatira tekst pomoću HTML tagova i
- Generisani HTML dokument predaje Web serveru koji ga vraća nazad klijentu

Dinamički web dokumenti

- ***Originalni mehanizmi*** za generisanje dinamičkih sadržaja web stranica su:
  - ***CGI*** (*Common Gateway Interface*) i
  - ***SSI*** (*Server-Side Includes*)
- ***Današnji web serveri*** koriste ***sofisticiranije i efikasnije mehanizme*** za generisanje dinamičkih sadržaja
- ***Poznavanje kako CGI i SSI*** mehanizmi ***funkcionišu*** je ***vrlo bitno*** za razumevanje savremenih pristupa

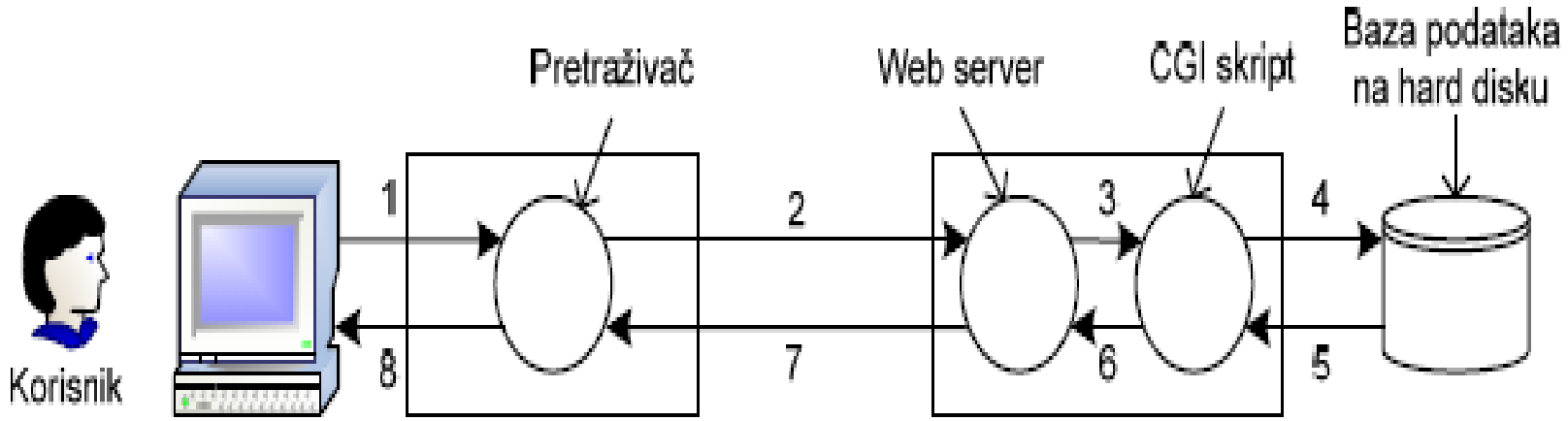
CGI (*Common Gateway Interface*)

- ***CGI mehanizam***:

  "*kada zahtev za izvršavanje CGI skripte stigne na server, **"kreira" će se novi "proces"** u okviru koga će se određena **aplikacija** (pozadinski program, konkretna **CGI skripta**) **izvršiti**, opskrbljujući taj program, **kroz interfejs**, sa nizom parametara*"

CGI (*Common Gateway Interface*)

- ***Obrada HTML forme***



Korisnik popunjava formu (**1**); Forma se vraća serveru (**2**); i predaje CGI-u (**3**); CGI postavlja upit bazi podataka (**4**); Iz baze se preuzimaju traženi podaci (**5**); **CGI kreira HTML stranicu** (**6**); Stranica se vraća klijentu (**7**); i prikazuje u pretraživaču (**8**).

CGI (*Common Gateway Interface*)

- ***Srce CGI*** specifikacije je ***standardizovani interfejs***
- Omogućava ***web serveru da ˝komunicira˝*** sa pozadinskim ***programima***
- Po pravilu, pozadinski programi su ***skripte*** napisane ***u jeziku Perl*** (***CGI skripte***)
- Interfejs je ***ustrojen skupom varijabli okruženja*** koje sve CGI aplikacije znaju i mogu im pristupiti

CGI (*Common Gateway Interface*)

- ***Uloga web servera*** je da ***podatka iz HTTP zahteva*** postavi ***u varijable okruženja***
- ***Podaci*** korišćeni ***za popunjavanje*** tih varijabli dolaze iz:
  - ***prve linije*** HTTP zahteva,
  - parametara povezivanja,
  - ***URL***-a,
  - HTTP ***zaglavlja***,
  - ....

CGI (*Common Gateway Interface*)

- ***Svako HTTP zaglavlje*** se preslikava u ***jednu varijablu okruženja***
- ***Ime varijable*** se dobija:
  - ***konvertovanjem malih slova u*** imenu zaglavlja u ***velika slova***,
  - ***zamenom*** crtice (minusa) sa donjom crtom i
  - ***dodavanjem*** HTTP_ kao prefiksa
- Na primer, ***vrednost User-Agent zaglavlja*** se smešta u varijablu okruženja ***HTTP_USER_AGENT***

CGI (*Common Gateway Interface*)

- Server ***uključuje u interfejs*** ***i osnovni mehanizam komunikacije Perl programa*** – (I) ***standardni ulaz*** i (II) ***standardni izlaz***
  - ***Telo HTTP zahteva*** server upućuje ***na standardni ulaz*** Perl (CGI) programa
  - Vrši ***redirekciju standardnog izlaza*** ka ***Modulu za obradu zahteva*** (generisanje odgovora)

CGI (*Common Gateway Interface*)

- ***Sledi primer*** konkretizacija jedne jednostavne ***HTML stranice koja sadrži formu***
  - Omogućuje korisnicima ***da unesu svoje ime*** i ***poštanski broj***
  - Atribut ***action*** u okviru taga <form> ***upućuje na*** serversku ***aplikaciju koja obrađuje*** informacije iz forme

```
<HTML>
<HEAD><TITLE>Simple Form</TITLE></HEAD>
<BODY>
<H2>Simple Form</H2>
<FORM action="http://mysite.org/cgi-bin/zip.cgi" method="post">
Zip Code: <INPUT size="5" name="zip">
Name: <INPUT size="30" name="name">
<INPUT type="submit" value="set zip">
</FORM>
<BODY>
</HTML>
```

CGI (*Common Gateway Interface*)

- Iz koje ***sledi*** konkretni ***HTTP zahtev***:

```
POST http://mysite.org/cgi-bin/zip.cgi HTTP/1.1
Host: mysite.org
User-Agent: Mozilla/4.75 [en] (WinNT; U)
Content-Length: 26
Content-Type: application/www-form-urlencoded
Remote-Address: 127.0.0.1
Remote-Host: demo-portable

zip=08540&name=Leon+Shklar
```

SSI (*Server Side Includes*)

- ***SSI*** specifikacija je ***stara*** skoro ***koliko i CGI*** specifikacija
- Obezbeđuje ***mahanizam za uključivanje***:
  - ***Sadržaja*** pomoćnih ***datoteka*** u HTML stranicu
  - ***Rezultata*** izvršavanja ***CGI skripti*** ***u HTML stranicu***

SSI (*Server Side Includes*)

- Klasična ***CGI funkcija***:
  - Kao ***izlaz*** ima ***HTML oznake***;
  - Ako želimo ***promeniti HTML***, ***moramo*** se vratiti nazad i ***promeniti kôd*** funkcije
  - ***To nije poželjno*** jer ***komplikuje održavanje*** programa

SSI (*Server Side Includes*)

- ***Rešenje*** se sastoji u:
- kreiranju ***delomično popunjene HTML stranice*** ili obrazca,
- praznine se ***popunjavaju rezultatima izvršavanja CGI skripte*** ili drugim operacijama na strani servera
- SSI nije zamena za CGI - to je ***jednostavan način*** da se doda dinamički sadržaj stranicama

SSI (*Server Side Includes*)

- ***SSI makro-naredbe*** imati sledeći oblik:

```
<!--#command attr1="value1" attr2="value2" -->
```

- ***Sintaksa*** je osmišljena tako ***da se SSI naredbe postavljaju unutar HTML komentara***
- Na taj način je osigurano da se ***neobrađene naredbe ignorišu*** kada se stranica analizira od strane pretraživača

SSI (*Server Side Includes*)

- Validne ***komande***:
- ***config*** - za ***kontrolu parsiranja*** (raščlanjivanja),
- ***echo*** - za ***prikaz vrednosti varijable*** okruženja,
- ***include*** - za ***umetanje*** dodatnih ***datoteka***,
- ***fsize*** - za izbacivanje ***informacije o veličini datoteke*** i
- ***exec*** - za ***izvršenje programa na strani servera***

SSI (*Server Side Includes*)

- ***Uobičajeno korištenje exec komande*** je da se ***pozove CGI skripta*** kao

```
exec cgi="http://mysite.org/cgi-bin/zip.cgi"
```

- Može se koristiti i za ***pokretanje drugih programa na strani servera***

```
<HTML>
<HEAD><TITLE>hello</TITLE></HEAD>
<BODY>
<!--#exec cgi http://mysite.org/cgi-bin/zip-ssi.cgi -->
</BODY>
</HTML>
```

SSI (*Server Side Includes*)

- Kao što znamo - ***serveri ne analiziraju statičke stranice***
- ***Pretraživači*** su odgovorni za njihovo ***parsiranje*** i ***slanje dodatnih zahteva*** za dodatne ugrađene objekte
- ***U slučaju SSI stranice*** – ***to nije slučaj*** – server ne može pronaći i izvršiti SSI naredbe ako ne izvrši parsiranje sadržaja stranice
- Datotekama koje sadrže SSI naredbe dodeljuje se ***drugačija ekstenzija – .shtml*** kako bi se ukazalo na potrebu posebne obrade od strane server

SSI (*Server Side Includes*)

- ***SSI mehanizam*** omogućuje ***jednostavan i praktičan način*** za ***dodavanje dinamičkog sadržaja*** postojećim stranicama ***bez potrebe generisanja cele stranice***
- Ništa ne dolazi besplatno
- ***Cena praktičnosti*** u korištenju SSI je
  - ***dodatno opterećenje*** na strani servera i
  - ***sigurnosni problemi***

Nakon CGI i SSI

- ***CGI*** - jednostavan mehanizam za implementaciju ***portabilnosti*** (prenosivosti) ***serverskih aplikacija***
- ***Koristi se naširoko*** diljem weba
- Međutim, ***postoji niz problema*** povezanih sa CGI obradom - ***glavni nedostatak su performance***
- ***SSI*** - ima ***slične nedostatke*** kada njihova naredba obrade koristi CGI pod plaštom HTML komentara

Nakon CGI i SSI

- ***Dodatna analiza SSI stranica*** od strane server, ***dodatno pogoršava performance*** sistema
- Ono što je najvažnije, ***SSI može predstavljati ozbiljan sigurnosni rizik***, pogotovo kada server nije pažljivo konfigurisan od strane administratora
- Imajući to na umu, ***niz drugih pristupa*** za obradu dinamičkog sadržaja se pojavilo u web server okruženju:
  - ***Ugrađeni API*** (ISAPI i *Apache Server API*)
  - ***FastCGI***
  - ***Obrada pomoću šablona***

Obrada pomoću šablona

- ***Aktuelni pristup*** za generisanje dinamičkog sadržaja - ***podrazumeva korišćenje tzv. šablona*** (*templates*)
- Šabloni su u suštini ***obične HTML datoteke*** sa ***dodatnim "tagovima"*** koji opisuju načine umetanja dinamički generisanog sadržaja iz nekog eksternog izvora
  - ***HTML*** - obezbeđuje ***opšti izgled stranice***
  - ***Dodati tagovi*** - postavljeni ***sa ciljem*** odgovarajućeg ***prikaza sadržaja*** u isporučenoj stranici

Obrada pomoću šablona

- ***Među najpopularnijim pristupima*** obrade pomoću šablona su
  - ***PHP*** (open-source proizvod),
  - ***ColdFusion*** (Adobe proizvod) i
  - ***Active Server Page*** (ASP, Microsoft proizvod)

Obrada pomoću šablona

- ***SSI*** se može smatrati ***pretečom pristupa obrade pomoću šablona***
- ***SSI naredbe*** mogu da izvrše ***jednostavne poslove***, kao što su ***umetanje***
  - sadržaja eksternih datoteka i
  - izlaza CGI programa

  unutar stranice
- Napredni pristupi ***obrade pomoću šablona*** obezbeđuju ***funkcionalnosti koje su mnogo sofisticiranije***

Obrada pomoću šablona

- To su ***funkcionalnosti*** koje srećemo ***u većini programskih i skript jezika***:
  - ***Podnošenje upita*** bazi podataka
  - ***Iterativna obrada*** (analogno ponavljanjima “for-each” petlje)
  - ***Uslovna obrada*** (analogno “if” naredbama)

Obrada pomoću šablona

- ***Da bi PHP mogao da se koristi*** u ovakve svrhe, ***server mora da razume PHP jezik***
- Server očekuje da Web stranice koje sadrže PHP ***imaju ekstenziju .php*** umesto .html ili .htm

Obrada pomoću šablona

- Na slici je prikazana ***HTML stranica sa ugrađenim*** jednostavnim ***PHP skriptom***

```
<html>
<body>
<h2>Ovo je sve sta znam o tebi:</h2>
<?php echo $HTTP_USER_AGENT ?>
</body>
</html>
```

Osim standardnog HTML-a -stranica **sadrži još i PHP skript** unutar taga **<?php . . . ?>**
Efekat ovog skripta je **generisane Web stranice** koja **sadrži opšte podatke** o klijentu koji je zatražio stranicu

Obrada pomoću šablona

- ***Pretraživači***, ***uz zahtev*** upućen serveru obično ***šalju i neke dodatne informacije*** o sebi
  - tip pretraživača, jezik, operativni sistem, tip klijentskog računara, ...
- One se ***u PHP skript prenose*** kao vrednost ***promenljive HTTP_USER_AGENT***
- Pretpostavimo da je ***ovaj skript smešten*** u datoteci ***test.php*** koji se nalazi u ***WWW direktorijumu*** kompanije ***ABCD***

Obrada pomoću šablona

- Kada se korisnik obrati URL-u ***www.abcd.com/test.php***, server kompanije ABCD:
  - ***otvara fajl*** *test.php*,
  - ***pronalazi*** u njemu ***PHP skript***,
  - ***zamenjuje ga*** vrednošću promenljive ***HTTP_USER_AGENT*** iz pristiglog zahteva i
  - tako modifikovanu ***stranicu vraća klijentu***

Obrada pomoću šablona

- ***PHP*** je naročito ***pogodan za obradu formi***. Razmotrimo ***primer HTML stranice***:

```
<html>
<body>
<form action="action.php" method="post">
<p>Unesi svoje ime: <input type="text" name="name">
  </p>
<p>Koliko imas godina: <input type="text" name="age">
  </p>
<input type="submit">
</form>
</body>
</html>
```

Obrada pomoću šablona

- ***Jedina specifičnost ove stranice*** je ***sadržaj prve linije*** koja ukazuje da radi obrade podataka unetih u formu i vraćenih serveru treba koristiti fajl ***action.php***
- Stranica prikazuje ***dva tekstualna polja***
  1. predviđeno ***za unos imena*** (***name***)
  2. ***za unos godine rođenja*** (***age***) korisnika
- Nakon što je korisnik popunio oba polja, ***klikom na dugme Submit***, ***putem HTML zahteva*** se uneti ***podaci se šalju serveru***

Obrada pomoću šablona

```
POST http://www.abcd.com/test.php HTTP/1.1
Host: abcd.com
User-Agent: Mozilla/4.75 [en] {WinNT; U}
Content-Length: 23
Content-Type: application/www-form-urlencoded
Remote-Address: 127.0.0.1
Remote-Host: demo-portable

name=Petar+Peric&age=24
```

- (pod pretpostavkom da je korisnik u polja *name* i *age* upisao Petar Peric i 24)

Obrada pomoću šablona

- Server preuzima sadržaje polje *name* i *age*, a zatim otvara i prolazi kroz fajl ***action.php*** i izvršava svaki PHP skript na koji naiđe

```
<html>
<body>
<h1> Odgovor:</h1>
Zdravo <?php echo $name;?>.
Predvidjanje: sledece godine imaces
 <?php echo $age + 1;?> godina
</body>
</html>
```

Obrada pomoću šablona

- ***HTML fajl koji se vraća klijentu*** imaće oblik:

```
<html>
<body>
<h1> Odgovor:</h1>
Zdravo Petar Peric.
Predvidjanje: sledece godine imaces 25 godina
</body>
</html>
```

Obrada pomoću šablona

- ***PHP*** je ***moćan*** programski jezik, koji se uz to i ***jednostavno koristi***
- Orijentisan na ***spregu između Web servera i servera baze podataka***
- PHP poseduje ***promenljive***, ***stringove***, ***polja*** i većinu ***upravljačkih struktura*** koje srećemo u C-u
- PHP je ***open source*** i dostupan je ***za slobodno korišćenje***
- Posebno je projektovan za rad sa ***Apache web serverom*** (koji je takođe *open source*)

Obrada pomoću šablona

- ***Treća tehnika*** za dinamičko kreiranje Web stranica je ***JSP (JavaServer Pages)***
- JSP je ***slična PHP-u***, ***sa razlikom*** što se ***dinamički deo stranice*** piše ***u programskom jeziku Java***, umesto u PHP-u
- Stranice koje koriste ovu tehniku obično ***imaju ekstenziju .jsp***

Obrada pomoću šablona

- ***Četvrta tehnika*** za dinamičko kreiranje Web stranica je ***ASP (Active Server Pages)***
- ***Microsoft-ova verzija*** PHP-a i JSP-a
- Za generisane dinamičkog sadržaja kod ASP se koristi ***skript jezik Visual Basic Script*** (ili VB skript)
- Stranice koje koriste ASP, obično ***imaju ekstenziju .asp***

Aktivni web dokumenti

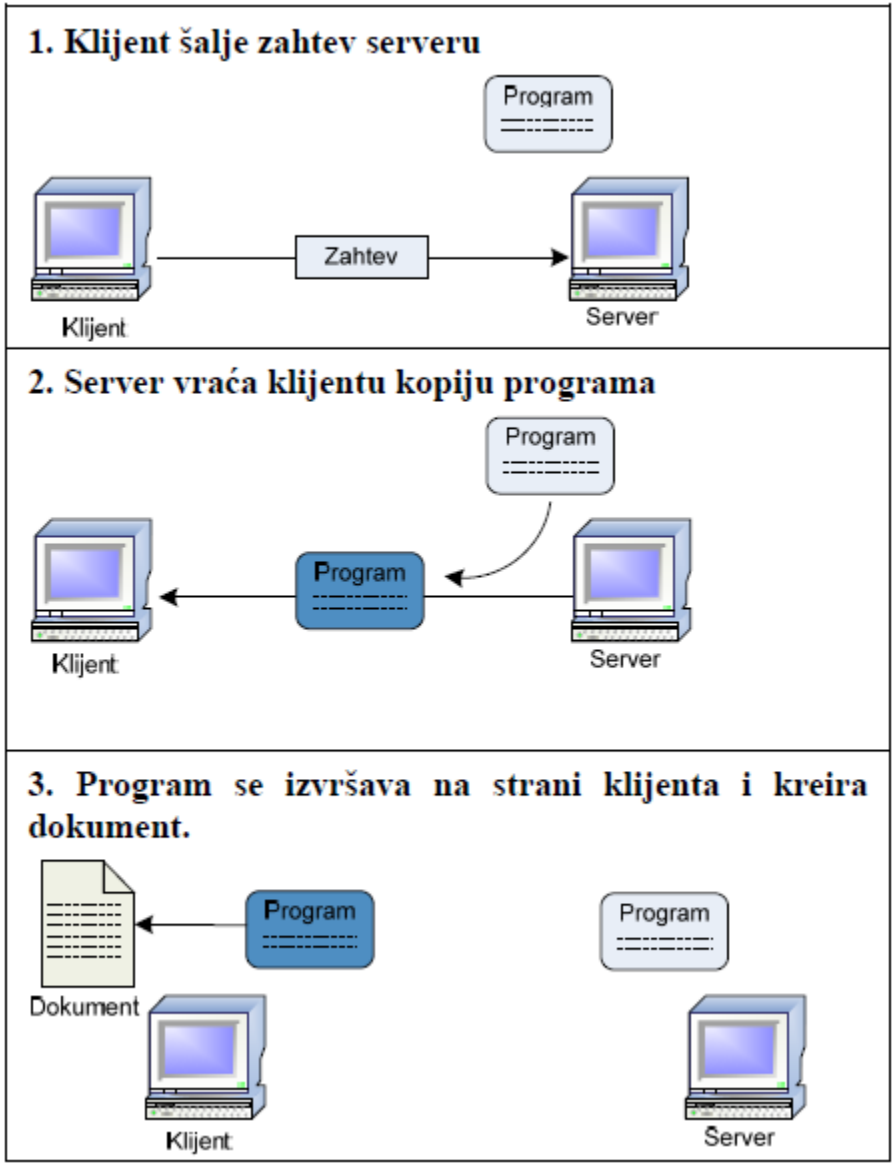
- ***Aktivni web dokumenti*** su ***web stranice*** koje sadrže:
  - ***statički sadržaj*** (HTML, slike i sl.) ***i***
  - ***program*** koji se nakon učitavanja stranice ***izvršava*** na strani klijenta (***u pretraživaču***)
- ***Primeri*** aktivnih web dokumenata:
  - web stranica koja sadrži ***animiranu grafiku*** ili
  - web stranica koja sadrži ***interaktivni grafički interfejs*** (***html forme***)

Aktivni web dokumenti

- ***HTML*** reguliše samo ***prikaz sadržaja stranice*** na ekranu pretraživača – za ***navedene primere neophodan je poseban program*** koji će
  - ***kreirati animacije*** ili
  - ***omogućiti*** neku specifičnu ***interakciju sa korisnikom***
- Takav jedan ***program se mora izvršavati na računaru klijenta*** - tamo gde se odvija interakcija sa korisnikom

Aktivni web dokumenti

Uvek kada klijent zahteva aktivni dokument od servera, ***server šalje kopiju dokumenta sa sadržanim program*** koji se po učitavanju izvršava u pretraživaču



Aktivni web dokumenti

- ***Skript jezici*** kao što su CGI, PHP, JSP i ASP ***rešavaju problem obrade formi*** i interakcije sa bazama podataka na serveru
  - prihvataju informacije unete u formu,
  - pretraže informacije u bazi podataka i
  - generišu HTML stranicu sa rezultatima obrade

Ali,

- ***Skript jezici nisu u stanju*** da reaguje na klik mišem ili ***da direktno interaguje sa korisnikom*** koji koristi pretraživač

Aktivni web dokumenti

- Za tu namenu, ***neophodan je program koji će se izvršavati u samom pretraživaču***, na računaru klijenta umesto na računaru servera
- ***Web stranica*** sa pridruženim programom koji se izvršava na strani klijenta ***naziva se aktivnim dokumentom***

Aktivni web dokumenti

- Postoje ***dva načina za kreiranje*** aktivnih dokumenata.

I) Podrazumeva da se program, u obliku ***binarnog kôda***, ***čuva na serveru***, a da ***u HTML stranici postoji tag*** u kome je navedeno ***ime fajla koji sadrži program***

- Kada HTML interpretator u pretraživaču naiđe na ovakav tag, on ***najpre preuzima fajl programa od servera***, a ***zatim ga izvršava***
- Programi ovog tipa najčešće se pišu u programskom jeziku ***Java***

Aktivni web dokumenti

- Postoje ***dva načina za kreiranje*** aktivnih dokumenata.

II) Zasnovan je na ***skrip jeziku*** koji se, slično PHP-u, ***ugrađuje u sam HTML***

  - Za ***razliku od PHP skripta*** koji se ***izdvaja*** iz HTML stranice ***i izvršava na serverskom računaru*** - skript namenjen klijetu iz HTML-a ***izdvaja pretraživač i izvršava ga*** uz pomoć odgovarajućeg interpretatora
  - Najpoznatiji skript jezik za ovu namenu je ***JavaScript***

Java

- ***Java*** je ***objektno-orijentisan jezik*** zasnovan na C++
- Za razliku od C++, i većine drugih viših programskih jezika - ***kompajlirani programi pisani u Javi su portabilni (prenosivi)***
- ***Mogu se izvršavati na bilo kom računaru***, nezavisno od tipa procesora i operativnog sistema
- U terminologiji Jave, ***kompajlirani program se naziva bajtkôdom*** (*bytecode*)

Java

Bajtkôd

- ***ne sadrži*** mašinske instrukcije za neki konkretan procesor,
- ***sadrži*** instrukcije koje se izvršavaju u interpretatoru za Java bajtkôd, tj. u tzv. ***Java virtuelnoj mašini***

- Računar na kome je instalirana Java virtuelna mašina u stanju je da izvršava programe pisane u Javi
- U osnovi, ***bajtkôd*** predstavlja ***međukôd***, koji je po složenosti ***između*** Java ***izvornog kôda*** i ***mašinskog kôda***

Java

- ***Virtuelna mašina izvršava bajtkôd*** tako što

1) u bajtkôdu ***identifikuje pojedinačne komande*** (tzv. metode) i
2) ***poziva odgovarajuće funkcije*** pisane ***u mašinskom jeziku*** za ciljnu mašinu

Java

- Java programi namenjeni za Web ***nazivaju se apletima***
- ***Apleti*** se čuvaju na Web serveru ***u fajlovima sa ekstenzijom .class***
- Aplet se ***uključuje u HTML pomoću*** odgovarajućeg ***taga***
- Tag predstavlja ***instrukciju HTML interpretatoru*** da od Web servera
  - ***zatraži*** .*class* fajl, naveden u tagu, i
  - ***prosledi ga virtuelnoj mašini*** gde će aplet biti izvršen

Java

- ***Primer jednostavnog apleta*** je aplet koji reprodukuje audio fajl, kao pozadinsku muziku, za vreme dok je u pretraživaču prikazana stranica
- ***Pretpostavimo*** da je aplet:
- uključen ***u Web stranicu*** na serveru sa URL-om ***http://www.viser.edu.rs/***
- smešten u fajlu ***bgsound.class***,
  - Na URL-u ***http:// www.viser.edu.rs/java-apps***

Java

Tag koji uključuje aplet u HTML je sledećeg oblika:

```
<OBJECT CODEBASE=”http://www.viser.edu.rs/java-apps”
CLASSID = ˝java:bgsound.class˝
DATA = ˝bgsound.data˝
CODETYPE = ˝audio/MP3˝></OBJECT>
```

Tag sadrži ***više atributa***, sledećeg značenja. Vrednost atributa ***CLASSID*** definiše ***ime fajla*** u kome je ***smešten aplet***. ***CODEBASE*** definiše ***server i putanju*** na kojoj se fajl nalazi. ***Ime fajla***, „*bgsound.data*” u kome se ***čuvaju podaci koje aplet treba da procesira*** navedeno je u atributu ***DATA***. Ovaj ***fajl sadrži podatke tipa*** audio/MP3, što je navedeno u atributu ***CODETYPE***.

Java

- ***Nakon učitavanja***, ***aplet*** i prateći ***audio fajl*** se ***predaju virtuelnoj mašini***;
  - ***Aplet se startuje*** i
  - ***Sadržaj audio fajla***, u dekodiranom obliku, ***šalje u audio karticu*** gde se obavlja reprodukcija
- Na sličan način, ***aplet može sadržati grafičku animaciju*** ili video
  - U tom slučaju, kao atributi taga OBJECT navode se ***dimenzije pravougaone oblasti na ekranu pretraživača***, gde će animacija/video biti prikazani

Java

- ***Java apleti*** se mogu razumeti i ***kao fleksibilna zamena za pomoćne aplikacije i plug_in-ove***
- ***Na primer***, pretpostavimo da ***Web stranica sadrži sliku u*** nekom novom grafičkom ***formatu*** za koji ***na klijentskom računaru ne postoji podrška*** u vidu *plug-in*-a ili pomoćne aplikacije
- Zbog toga ***slika ne može biti prikazana*** na ekranu pretraživača

Java

- Međutim, ***ako se za prikazivanje slike koristi aplet***, ***dekoder za novi format može biti smešten u samom apletu*** koji se ***automatski preuzima sa servera*** prilikom učitavanja stranice u pretraživač

JavaScript

- ***Slična funkcionalnost*** (kao sa Java apletima) može se dobiti i ***korišćenjem JavaScript jezika***
- ***JavaScript*** skript jezik koji se u izvornom obliku ***ugrađuje u HTML stranicu***
  - Različito od ***Java apleta***
  - Slično ***PHP***-u

JavaScript

- ***JavaScript*** program se smešta u poseban HTML ***tag***, ***<script>***

```
<head>
.........
<script language="javascript" type="text/javascript">
.........
</script>
</head>
```

- Kada ***HTML interpreter*** naiđe na tag <*script*> on ***poziva*** ***JavaScript interpreter*** koji ***izvršava skript***

JavaScript

- ***JavaScript*** kao jezik ***nema direktne veze*** sa Javom (uprkos sličnom imenu)
- ***JavaScript*** sadrži programske konstrukcije ***veoma visokog nivoa*** (slično ostalim skript jezicima)
- Na primer, ***u jednoj liniji JavaScript programa*** moguće je:
  - prikazati dijalog za unos teksta,
  - čekati da tekst bude unesen i
  - smestiti uneti tekst u neku promenljivu

JavaScript

- Zahvaljujući ovim mogućnostima - ***JavaScript*** je ***pogodan za*** jednostavno projektovanje ***interaktivnih Web stranica***
- Takođe, ***JavaScript poseduje*** mnoge ***osobine viših programskih jezika***, kao što su:
  - tipovi podataka
  - aritmetički operatori
  - logički operatori
  - petlje (*for()*, *while()*)
  - funkcije
  - itd.

JavaScript

Primer programa ***u JavaScript jeziku***

"Neka stranica sadrži formu za unos imena i godina korisnika, a po unosu traženih podataka predviđa koliko će godina osoba imati sledeće godine"

- Slično primeru sa slajda 66 (primenom PHP-a)
- Sekcija ***<body>*** je gotovo ***identična*** kao u primeru za PHP
- ***Glavna razlika*** je u
  - ***deklaraciji Submit*** dugmeta i
  - ***naredbe dodele sadržane*** u ovoj ***deklaraciji***

JavaScript

Primer programa ***u JavaScript jeziku***

Naredba dodele

- ukazuje pretraživaču da kao odgovor na klik dugmeta (događaj ***onclick***)
  - treba izvršiti JavaScript funkciju (recimo, *response*) i
  - kao parametar preneti joj formu

Dinamički vs. Aktivni dokument

- Iz navedenog primera možemo ***zaključiti***:
  - Sa tačke gledišta krajnjeg korisnika - ***rezultat oba primera je isti***
  - ***Bitno je razumeti*** da se ***načini obrade*** ova dva primera ***suštinski razlikuju***

Dinamički vs. Aktivni dokument

U primeru koji koristi PHP (vidi slajdove 66-70!)

- Nakon klika na dugme *Submit*, ***pretraživač izdvaja informacije iz forme*** i u vidu stringa ih ***šalje nazad serveru*** koji je poslao stranicu
- ***Server prepoznaje*** ime ***PHP*** fajla i ***izvršava ga***
- ***PHP*** skript ***kreira novu HTML stranicu*** koja se vraća pretraživaču i prikazuje

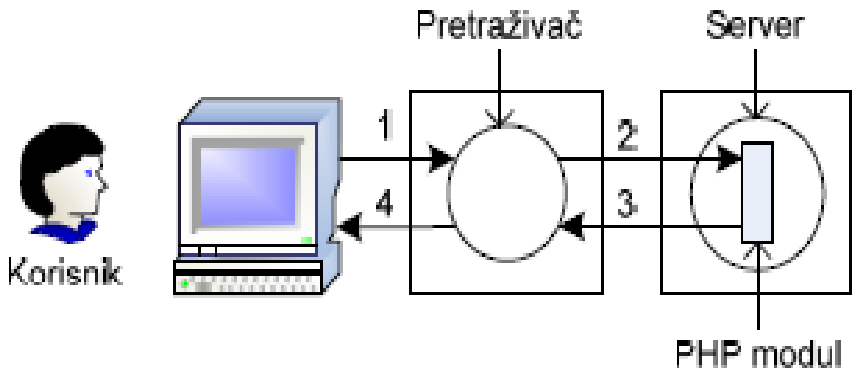
Dinamički vs. Aktivni dokument

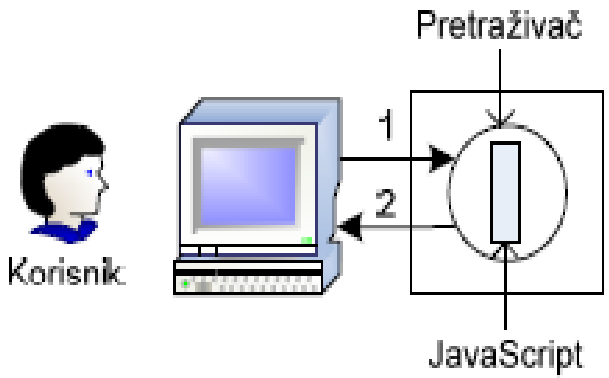
U primeru koji koristi JavaScript

- Nakon klika na dugme *Submit,* ***pretraživač interpretira JavaScript funkciju*** sadržanu u samoj stranici
- ***Ne postoji*** novi ***kontakt sa serverom***, već se sve obavlja ***lokalno, unutar pretraživača***
- Zbog toga se ***rezultat*** obrade ***pojavljuju gotovo trenutno***, za razliku od primera sa PHP gde postoji izvesno kašnjenje (dok rezultujući HTML fajl ne stigne nazad do pretraživača)

Dinamički vs. Aktivni dokument

- ***Na slici je ilustrovana razlika*** između procesiranja skripti na ***strani klijenta*** i ***strani servera***. U oba slučaja, ***korak 1*** počinje nakon što je forma učitana u pretraživač. ***Korak 2*** iniciran je ***klikom na*** dugme ***Submit***, a nakon toga sledi ***procesiranje forme***, koje u ova dva slučaja ***teče na različite načine***



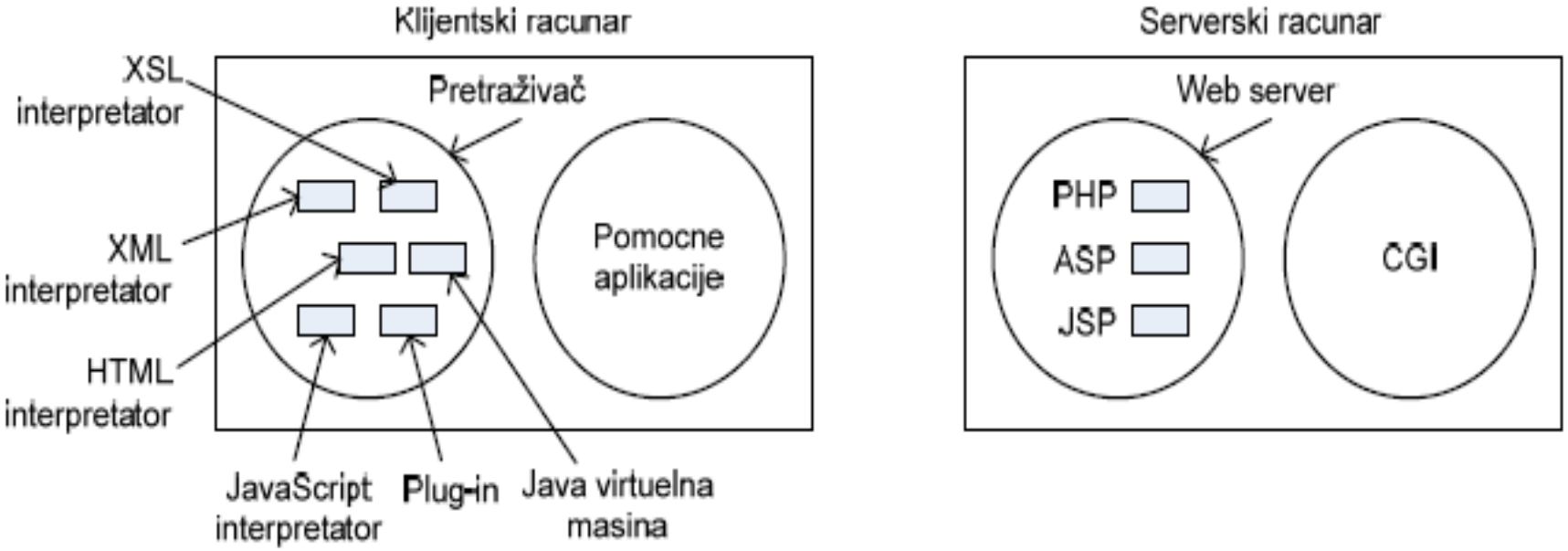


Dinamički vs. Aktivni dokument

- ***Namena*** ova dva skrip jezika je ***potpuno različita***
- ***PHP*** (i srodni skript jezici) prevashodno se koriste ***za interakciju sa udaljenom bazom podataka***
- ***JavaScript*** se koristi za ***interakciju sa korisnikom*** na strani klijentskog računara
- Moguće je, i uobičajeno, da ***Web stranica sadrži oba skript jezika***, sa raspodeljenim zadacima

Dinamički vs. Aktivni dokument

- ***Slika ilustruje različite tehnike*** za kreiranje ***dinamičkih*** i ***aktivnih*** Web stranica, o kojima je bilo reči u ovom predavanju



Dinamički vs. Aktivni dokument

- ***Dinamičke Web stranice***
- Generišu se ***na strani servera***,
- ***Uz pomoć*** različitih skript jezika (Perl, ***PHP***, JSP ili ASP),
- ***Klijent dobija standardnu HTML stranicu*** koju prosto treba da prikaže

Dinamički vs. Aktivni dokument

- ***Aktivne Web stranice***
- ***Sadrže***
  - ***ugrađene skriptove*** (pisane u ***JavaSctipt*** jeziku)
  - ***hiperveze*** na kompletne programe (u vidu ***Java apleta*** ili ActiveX kontrola)
- ***Nakon učitavanja*** oni ***se izvršavaju na klijentskom računaru***,
- Omogućava kreiranje interaktivnih korisničkih interfejsa, kao i složena procesiranja grafičkih, audio ili video sadržaja