

# Spletne tehnologije, UL, FRI (VSP)

## ST 4 – odjemalec, strežnik

doc.dr. Mira Trebar

# Vsebina

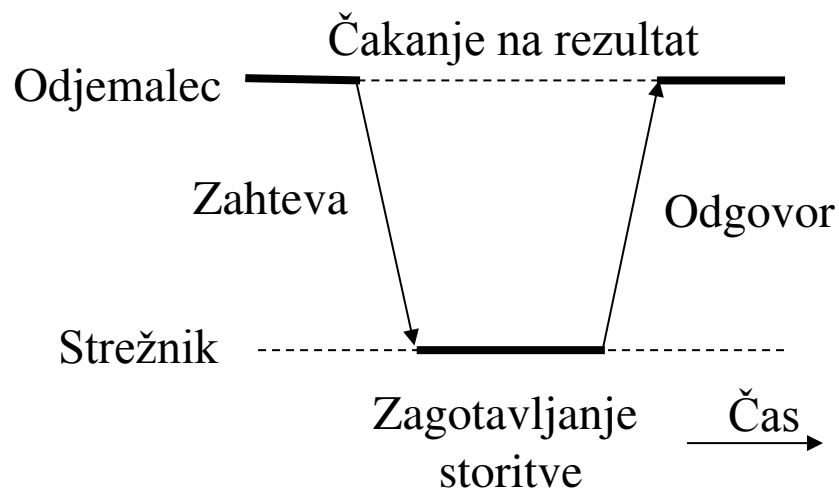
---

- ▶ Model Odjemalec –Strežnik (Client – Server)
  - ▶ HTTP protokol (HTTP protocol)
  - ▶ Spletni odjemalec (web browser)
  - ▶ Spletni strežnik (web server)
- 
- ▶ Literatura: J.C.Jackson, Web Technologies A computer science perspective, Pearson Prentice Hall, 2007
  - ▶ A.S.Tannenbaum, M.Steen, Distributed systems, principles and paradigms, Prentice-Hall, 2002
  - ▶ Splet: poiščite ustrezne vire.

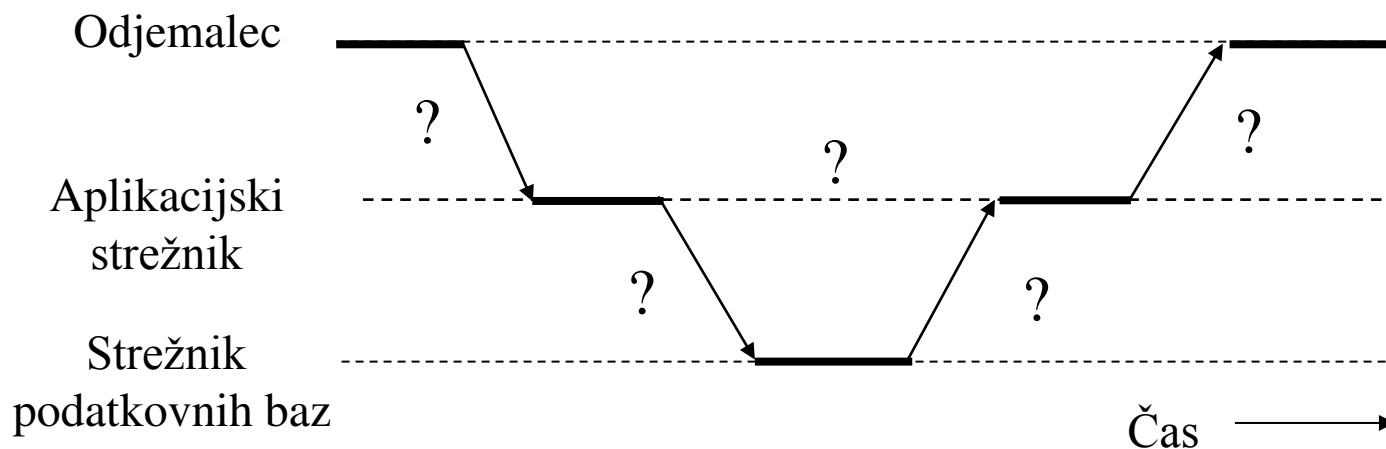
# Model odjemalec - strežnik

- ▶ ZAHTEVA – ODGOVOR

- ▶ Izmenjava sporočil

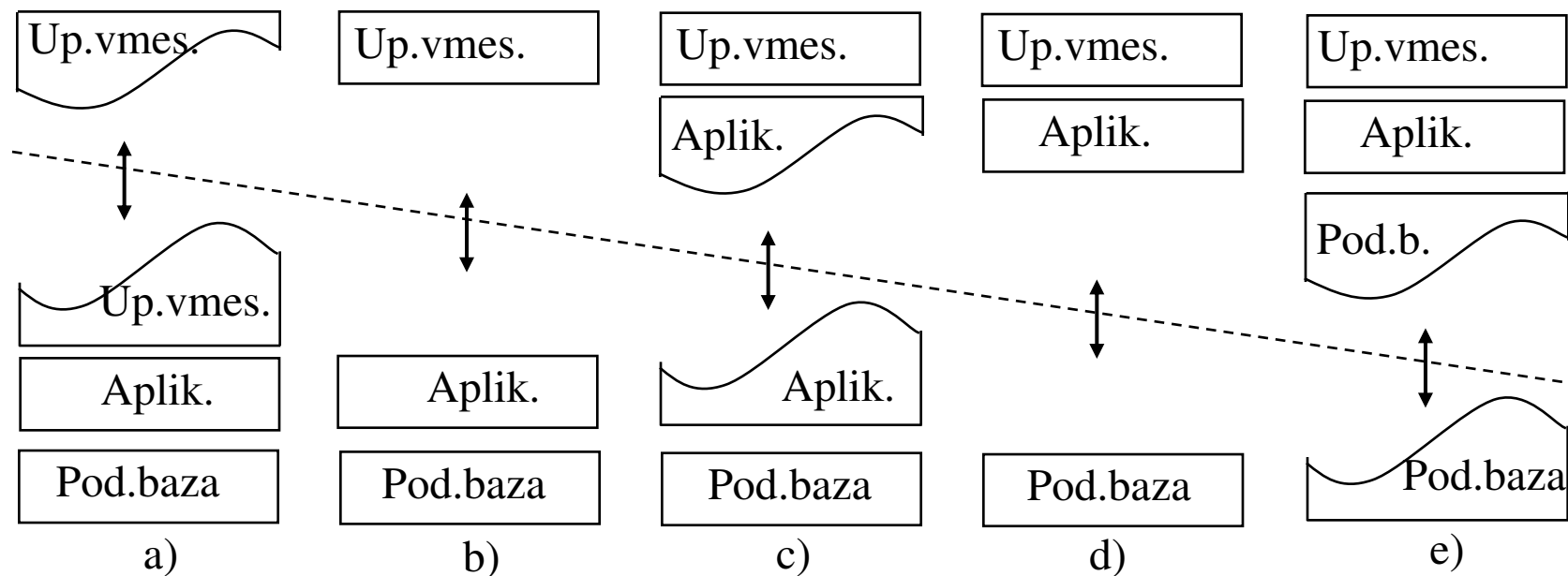


- ▶ Tri-vrstna arhitektura (3-tier architecture)



# Organizacija odjemalcev in strežnikov

- ▶ Aplikacija se izvaja na strežniku (a,b)
- ▶ Aplikacija se izvaja pri odjemalcu (d, e)
- ▶ Aplikacija se izvaja delno na strežniku in delno pri odjemalcu (c).



# Komunikacija

---

- ▶ Med-procesna komunikacija
- ▶ Najnižji nivo - pošiljanje sporočil v omrežju
- ▶ Izmenjava informacij - protokoli
  - proces A hoče komunicirati s procesom B
    - proces A zgradi sporočilo v svojem naslovnem prostoru
    - proces A izvede sistemski klic
    - operacijski sistem pošlje sporočilo preko omrežja procesu B
  - A in B se morata vnaprej dogovoriti o pomenu bitov, ki se pošiljajo
  - Koliko voltov naj bo uporabljenih za predstavitev bita 0, oziroma bita 1?
  - Kako bo sprejemnik vedel kateri bit je zadnji v sporočilu?
  - Kako bo ugotovil ali je bilo sporočilo poškodovano in kaj naj stori v takem primeru?
  - Kako dolga so števila, nizi in drugo in kako so predstavljeni?

- ▶ Protokol je zbirka pravil, ki

---

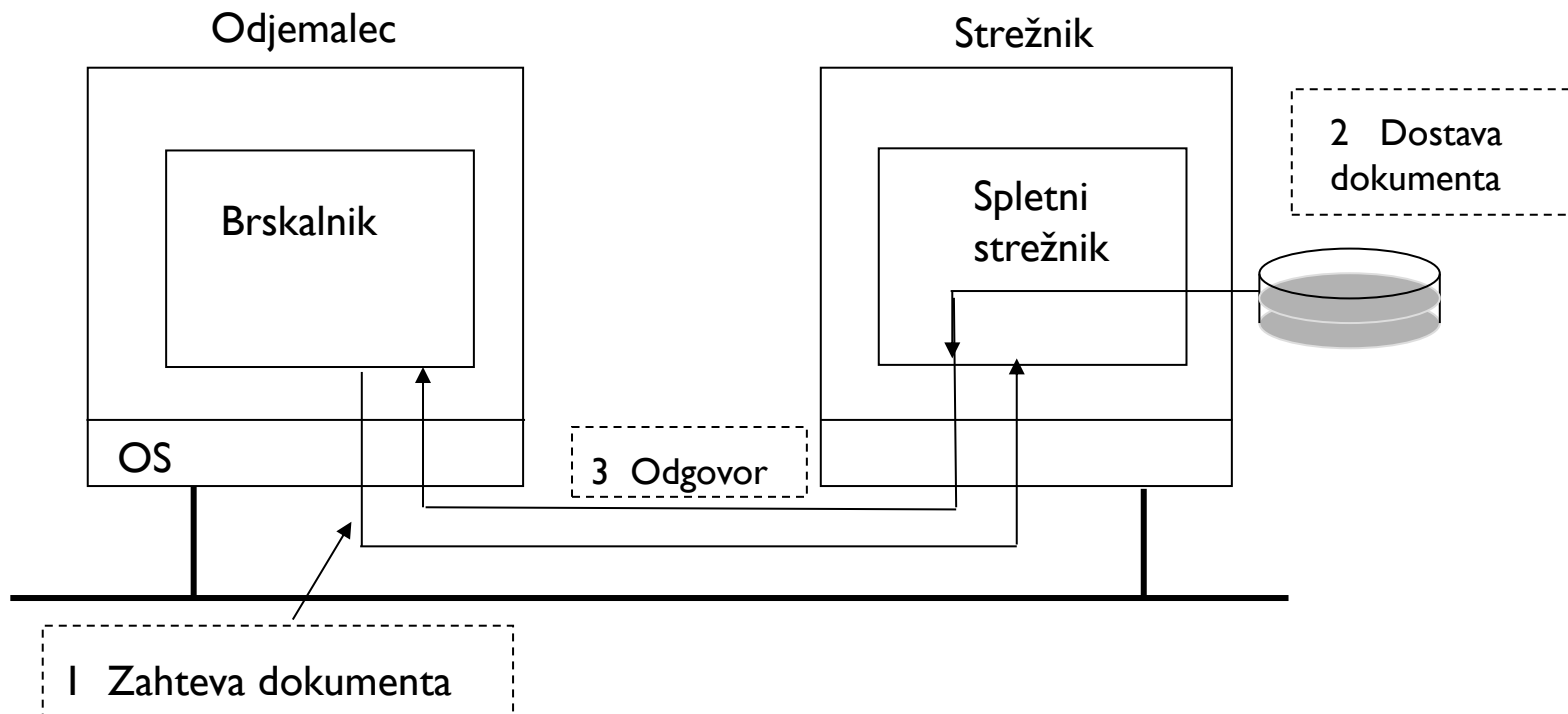
- ▶ določajo načine prenosa sporočil po računalniškem omrežju in
  - ▶ urejajo razumevanje sporočil.

- ▶ Nivoji protokolov

- ▶ Fizični nivo (0,1)
  - ▶ Podatkovni nivo (grupiranje bitov, testni niz)
  - ▶ Omrežni nivo (IP- 'Internet protokol')
  - ▶ Transportni (TCP -'Transmission Control Protocol')
  - ▶ Sejni
  - ▶ Predstavitveni
  - ▶ Aplikacijski
    - ▶ FTP – 'File Transfer Protocol'
    - ▶ HTTP- 'Hypertext Transfer Protocol'

# Splet - organizacija

- ▶ Odjemalec ('Client') deluje s spletnim strežnikom ('Web server') preko aplikacije imenovane spletni pregledovalnik ali brskalnik ('Web Browser').



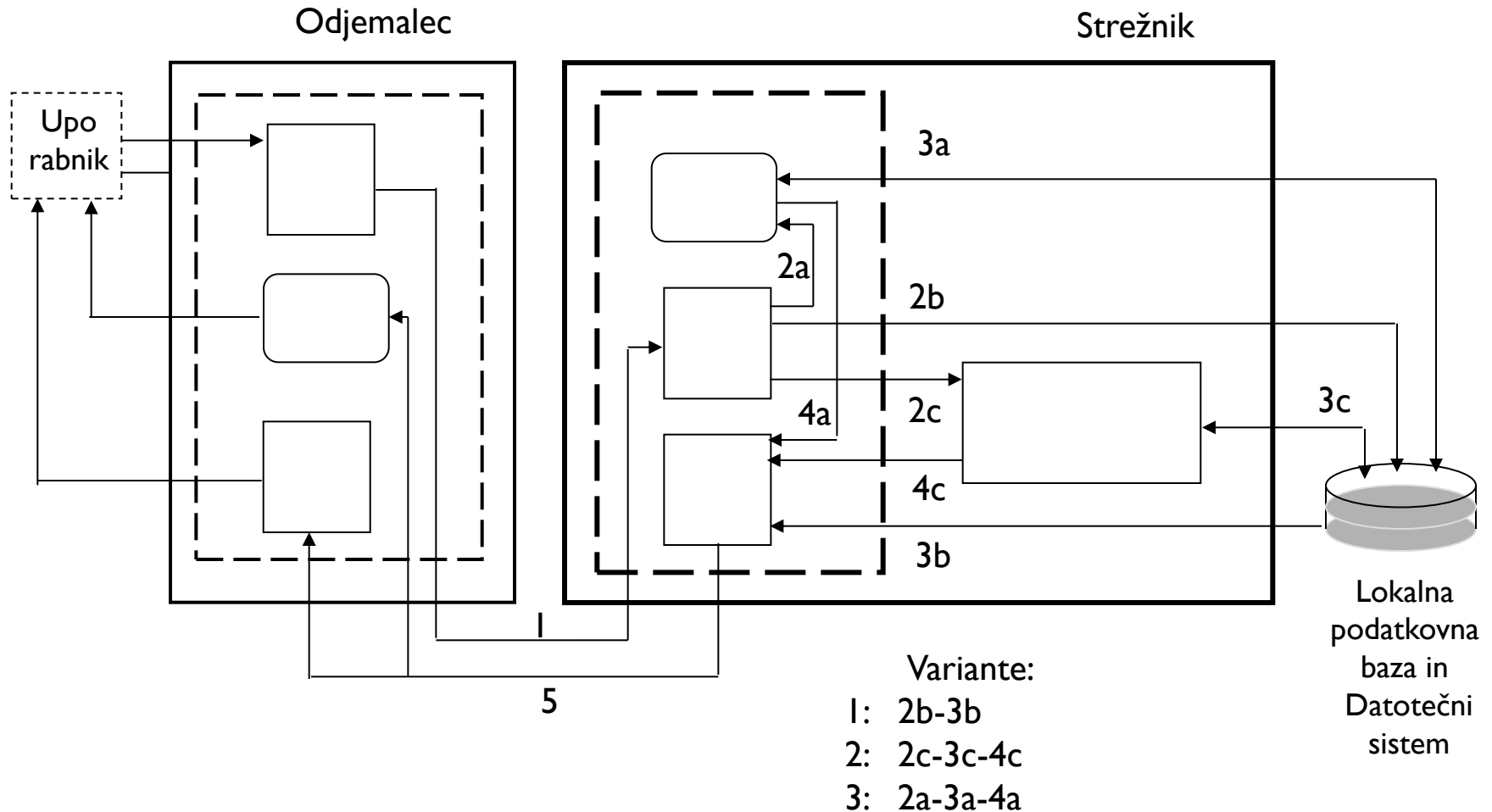
# Odjemalec (Brskalnik) - Spletni strežnik

---

- ▶ **Brskalnik - programska aplikacija**
  - ▶ Komunicira s spletnim strežnikom – HTTP  
(‘Hyper-text Transfer Protocol’) protokol
  - ▶ Naslavlja spletne strani (URL-‘Uniform Resource Locator’)
- ▶ **Spletni strežnik - program**
  - ▶ Sprejema HTTP zahteve odjemalcev
  - ▶ Zagotavlja HTTP odgovore
  - ▶ Preverja verodostojnost (uporabniško ime, geslo)
  - ▶ Upravlja statične in dinamične vsebine
  - ▶ Podpira HTTPS povezave (varnost)

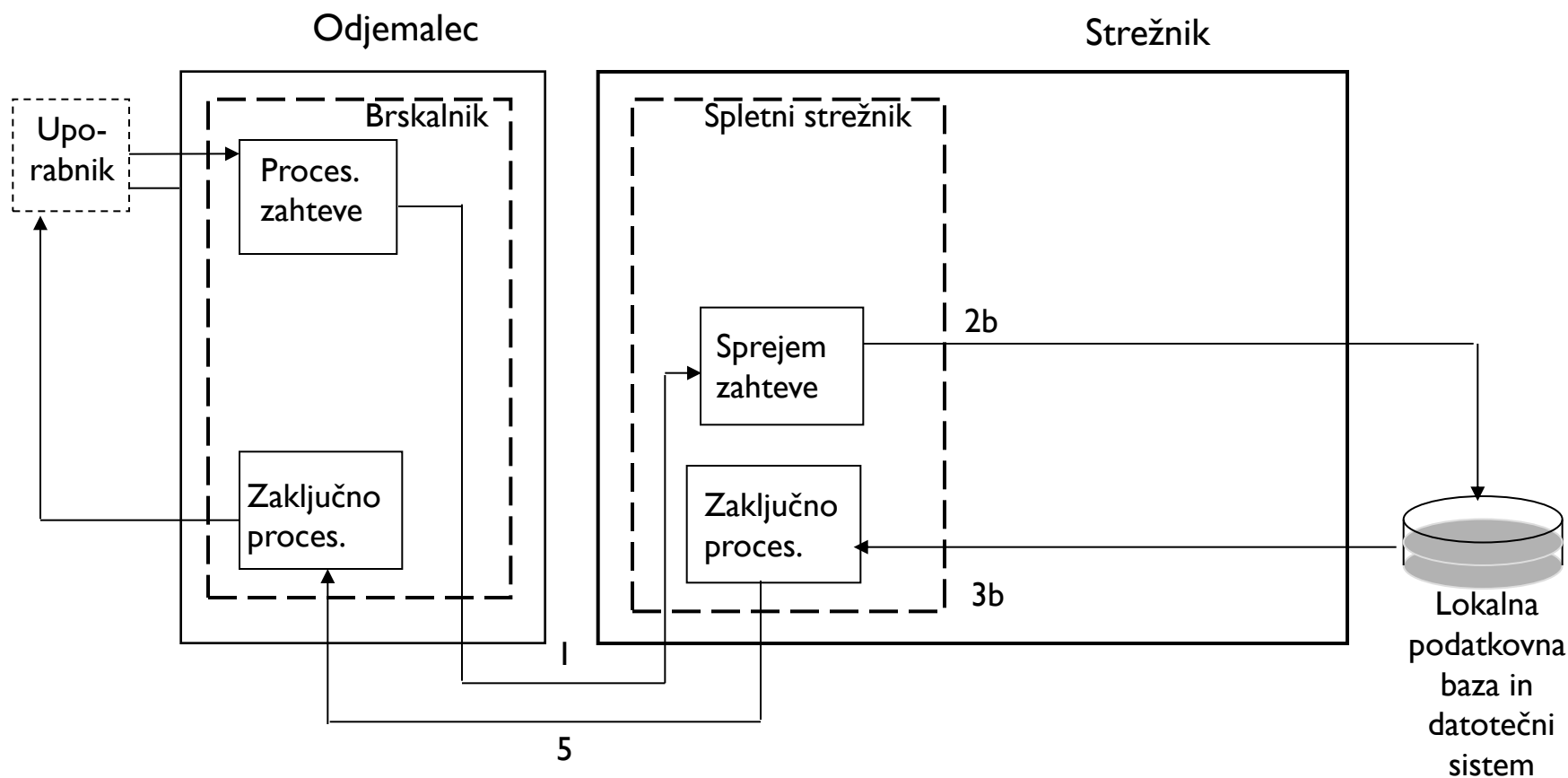


# Splet - Arhitekturni pregled



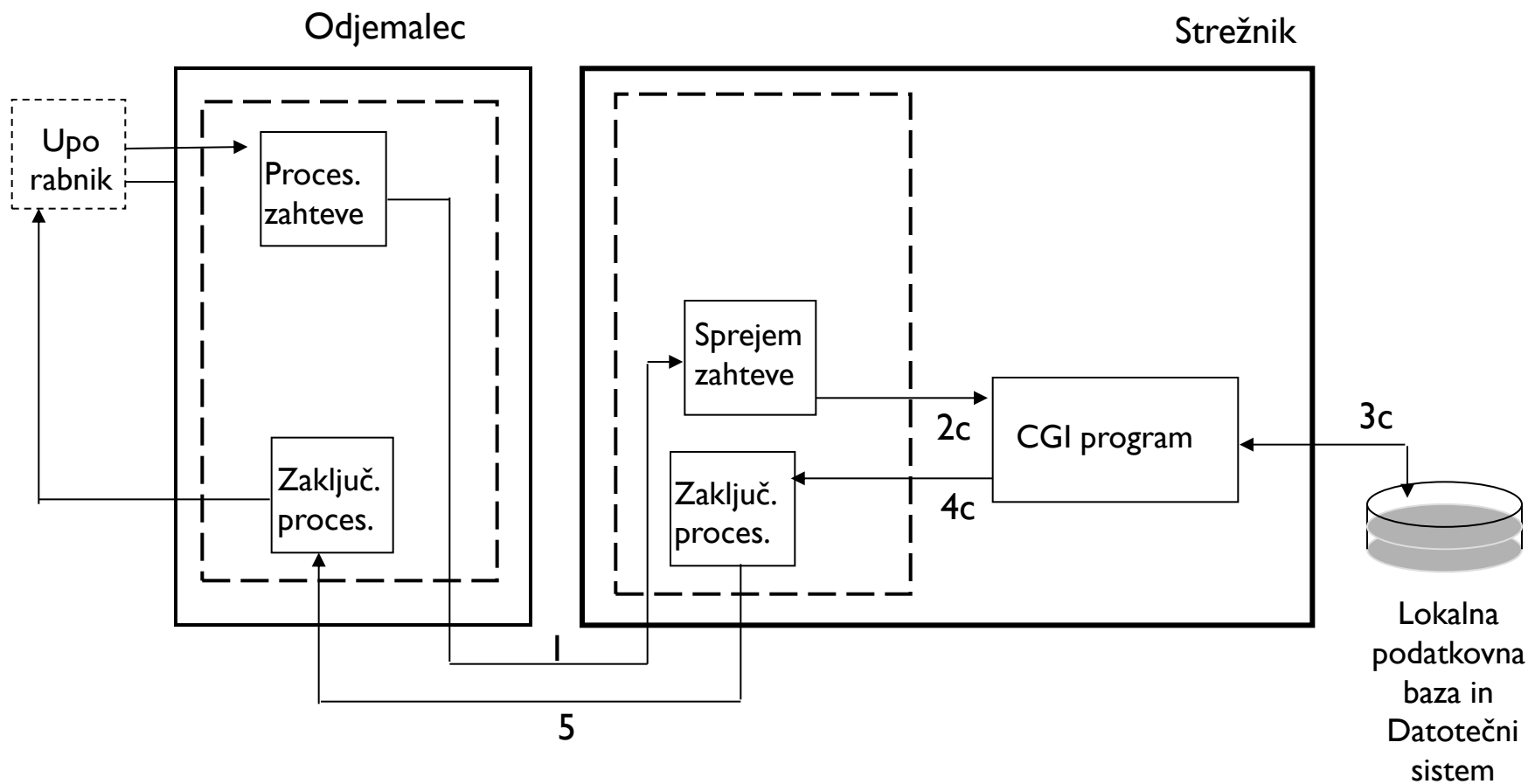
# I.) Dokument je direktno dostavljen iz lokalnega datotečnega sistema

Varianta 1: 1 - 2b - 3b - 5



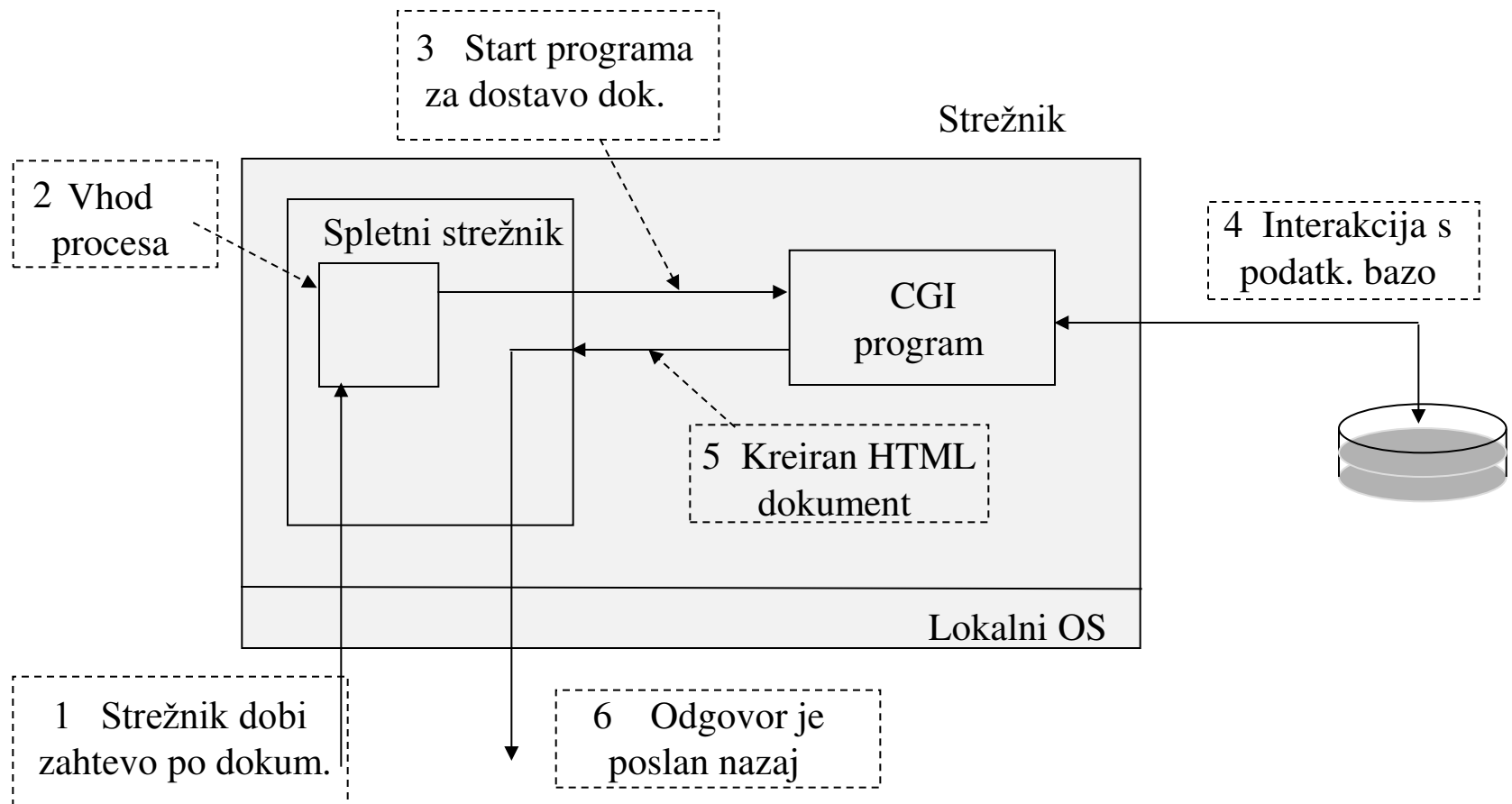
## 2) CGI program ustvari dokument s podatki iz lokalne podatkovne baze

Varianta 2:  $1 - 2c - 3c - 4c - 5$



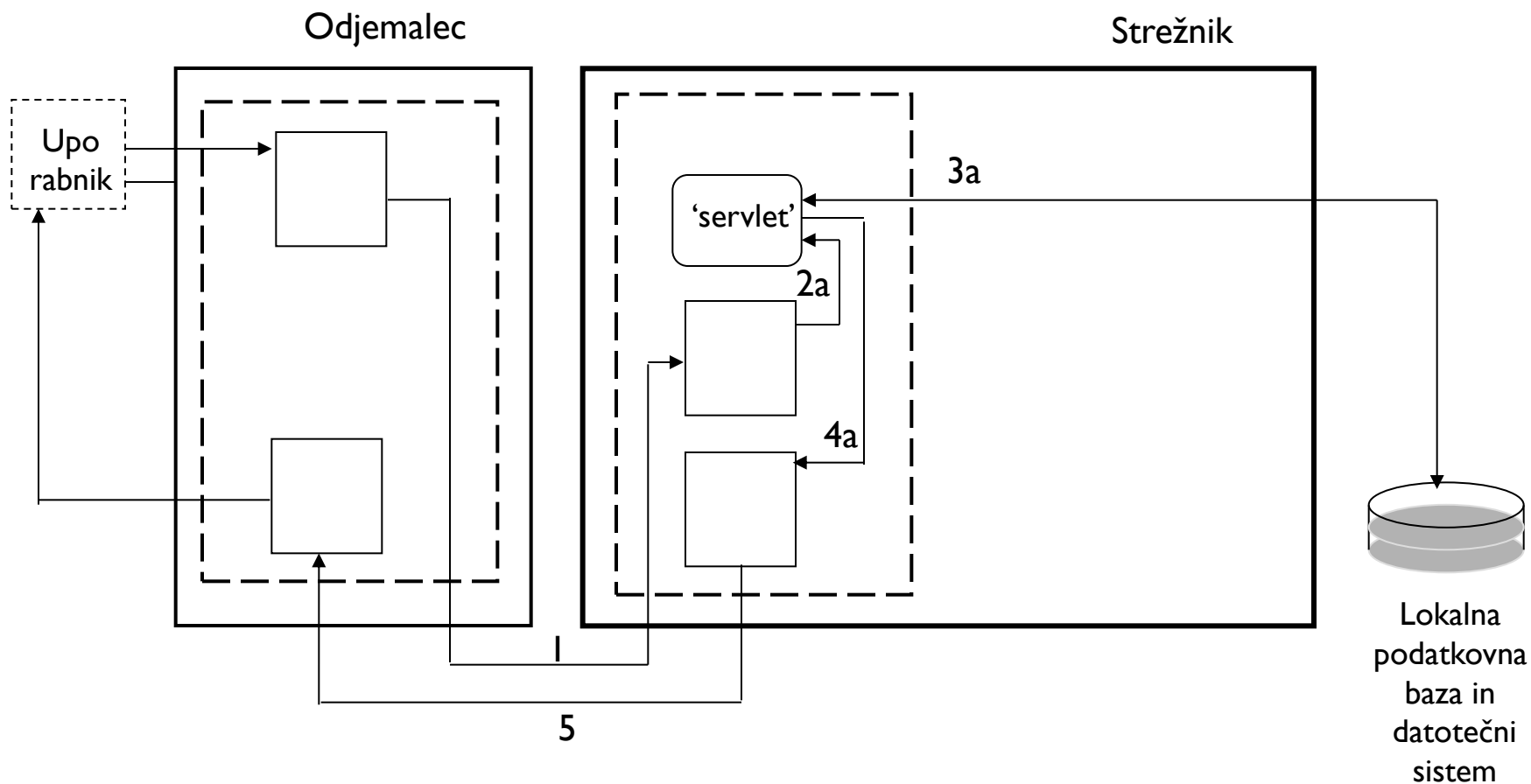
## Uporaba CGI programa na strani strežnika

CGI – ‘Common Gateway Interface’ - standarden način s katerim spletni strežnik lahko izvaja program z uporabnikovimi podatki na vhodu.

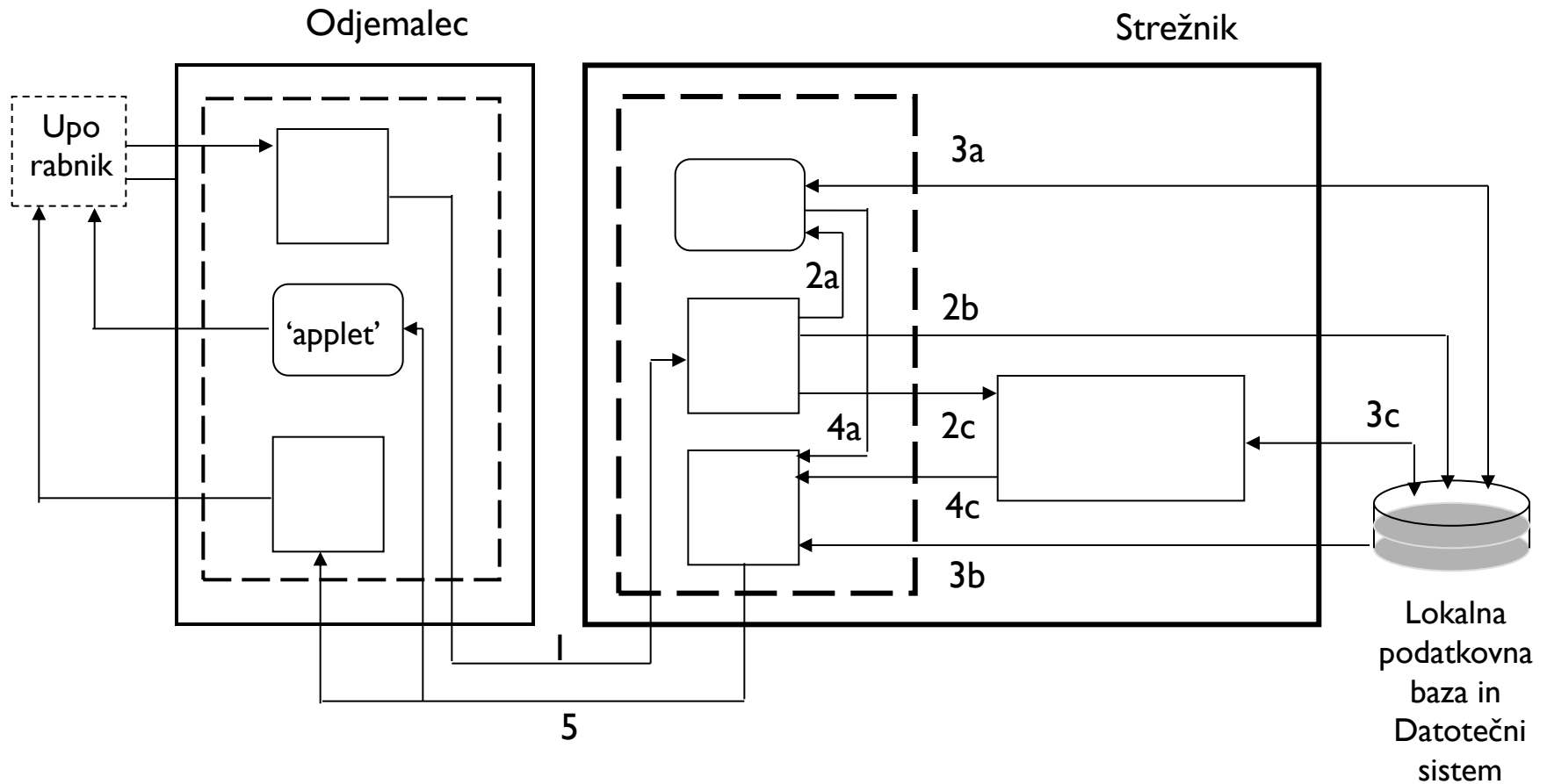


### 3.) Zahteva je poslana programu na strežniku 'servlet'

Varianta 3: 1 – 2a – 3a – 4a - 5



#### 4.) Brskalnik izvaja aplikacijo imenovano 'applet'



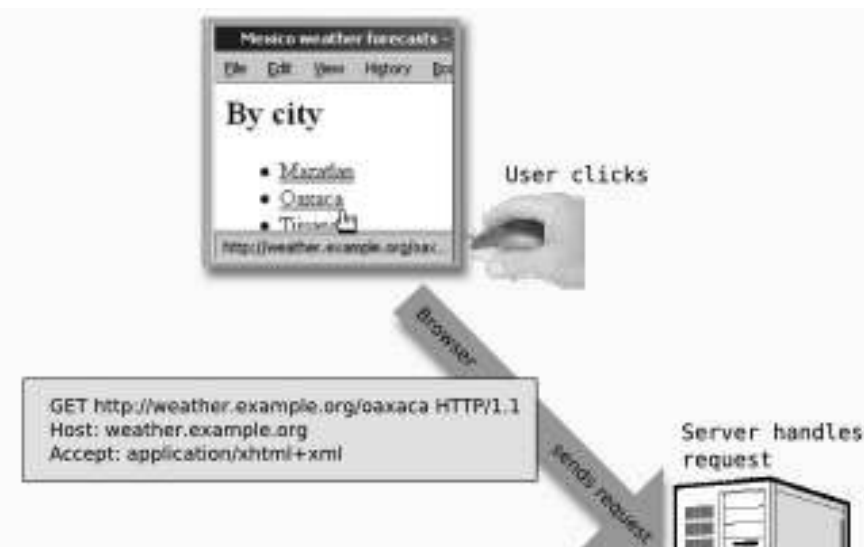
# HTTP - Hypertext Transport Protocol

- ▶ Komunikacijski protokol
- ▶ Spletni odjemalci in spletni strežniki
- ▶ Model zahteva-odgovor
- ▶ Postopek:
  - V oknu brskalnika kliknemo <http://www.w3schools.com/>
  - Brskalnik ustvari **sporočilo z zahtevo HTTP**
  - Uporabi DNS (Domain Name System) za pridobitev naslova IP
  - Vspostavi povezavo TCP s strežnikom (naslov IP)
  - Pošlje sporočilo HTTP preko povezave TCP
  - Od strežnika sprejme nazaj **sporočilo z odgovorom HTTP**
  - Prikaže se vsebina v brskalniku



# Sporočilo z zahtevo HTTP

1. Začetna vrstica sporočila (ang. Start line)
  - 1.1. Metoda zahteve HTTP (ang. Request method)
  - 1.2. Zahteva – URI, ki je del spletnega naslova (ang. Request –URI)
  - 1.3. Verzija protokola HTTP (HTTP/1.1)
2. Glava sporočila (ang. Header field(s)) – ima eno ali več polj
3. Prazna vrstica (ang. Blank line)
4. Telo sporočila (ang Message body) - opcijsko



<http://www.ltg.ed.ac.uk/~ht/WhatAreURIs/>



## 1.1 Metode zahteve HTTP

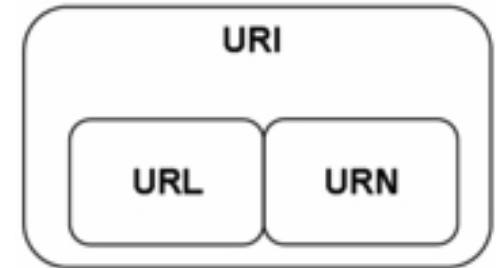
---

- ▶ GET - dostava vira (dokumenta), ki je določen z Zahteva-URI
- ▶ POST - zagotovi podatke, ki bodo dodani dokumentu ali zbirki dokumentov
- ▶ HEAD - enako kot metoda GET, vendar v odgovoru strežnik vrne samo naslov, ne vrne pa vira podanega z Zahteva-URI.
- ▶ OPTIONS – vrne seznam metod HTTP, ki so lahko uporabljene za dostop do vira, ki je podan z Zahteva-URI.
- ▶ PUT – shrani telo sporočila na strežnik.
- ▶ DELETE - brisanje dokumenta.
- ▶ TRACE – vrne kopijo celotnega sporočila zahteve HTTP, vključno z začetno vrstico, glavo in telo , ki ga prejme od strežnika – namenjeno za testiranje.
- ▶ CONNECT – za ustvarjanje določenih tipov varnih povezav
- ▶ **Pomembno:**  
Obvezen zapis metod z velikimi črkami!

## 1.2. Zahteva-URI

---

- ▶ URI (Uniform Resource Identifier)
- ▶ Sestava URI:
  - shema, ki je pred (:
  - drugi del, ki je odvisen od sheme (http, ftp, ...)



- ▶ URL (Uniform Resource Locator) – tip URI, ki določa lokacijo vira na spletu.

Primer:

`http://www.cs.vu.nl/home/steen/mbox`

- ▶ URN (Uniform Resource Name) – enolično ime vira (globalna lokacijsko neodvisna referenca dokumenta).

Sestava: urn: identifikator imenskega prostora:string

Primer:

knjiga “War and Peace“ ima kodo ISBN (International Standard Book Number) 0-1404-4417-3

(urn: ISBN: 0-1404-4417-3)

# URL (Uniform Resource Locator)

---

Shema/ Protokol	Ime gostitelja	Vrata (?)	Pot do dokumenta
--------------------	----------------	-----------	------------------

[http://www.fri.uni-lj.si/si/izobrazevanje/prva\\_stopnja/](http://www.fri.uni-lj.si/si/izobrazevanje/prva_stopnja/)

- ▶ Kombinacija IP naslova s številko vrat ('port')  
(<http://130.37.24.11:80/home/steen/mbox>)
- ▶ Uporaba DNS ('Domain Name Service') imena  
(<http://www.cs.vu.nl/home/steen/mbox>)
- ▶ Kombinacija DNS imena s številko vrat ('port')  
(<http://www.cs.vu.nl:80/home/steen/mbox>)
- ▶ Dodatne možnosti v Zahtevi - URI:  
? in # - povpraševanje (query string):  
    <https://www.google.si/search?q=HTTP#GET>  
uporabnik sprašuje po virih, ki vsebujejo HTTP, ko je ta vezan s subjektom GET

## 2. Glava sporočila (zahteva HTTP)

---

- ▶ **Host** – določa del naslova URL (strežnik in vrata)
- ▶ **Accept** – je uporabljen za določanje posebnih tipov MIME (Multipurpose Internet Mail Extensions), ki so sprejemljivi v odgovoru
- ▶ **Accept-language** - določa jezik za odgovor
- ▶ **Accept-encoding** - gzip, deflate – določa tip kodiranja (stiskanje, ..)
- ▶ **Accept-charset** - določa prednosti za izbiro znakov
- ▶ **User-Agent** – niz, ki določa brskalnik ali drug program
- ▶ **Connection** - določa ali želi odjemalec ohraniti odprto povezavo TCP tudi potem, ko je bil poslan odgovor.
- ▶ **Keep-Alive** – čas (sek), ko naj bo povezava TCP odprta
- ▶ **Content-Type** – Tip dokumenta (MIME), ki se nahaja v telesu sporočila
- ▶ **Content-Length** – število bajtov podatkov v telesu sporočila

# Tipi dokumentov - MIME

---

## ► MIME (Multipurpose Internet Mail Extensions):

- text/html – HTML dokument
- image/gif – slika v GIF formatu

Type	Subtype	Opis
Text	Plain	Neformatiran tekst
	HTML	Tekst z vključenimi HTML 'markup' komandami
	XML	Tekst z vključenimi XML 'markup' komandami
Image	GIF	Negibna slika v GIF formatu
	JPEG	Negibna slika v JPEG formatu
Audio	Basic	Audio, 8-bit PCM z 8000 Hz vzorčenjem
	Tone	Poseben 'audible tone'
Video	MPEG	Film v MPEG formatu
	Pointer	Predstavitev naprave pointer za predstavitev
Application	Octet-stream	Neprekinjeno zaporedje bajtov
	Postscript	Dokument za tisk v Postscript formatu
	PDF	Dokument za tisk v PDF formatu
Multipart		Večkratne entitete, vsaka ima svojo glavo in telo.

# Sporočilo z odgovorom HTTP

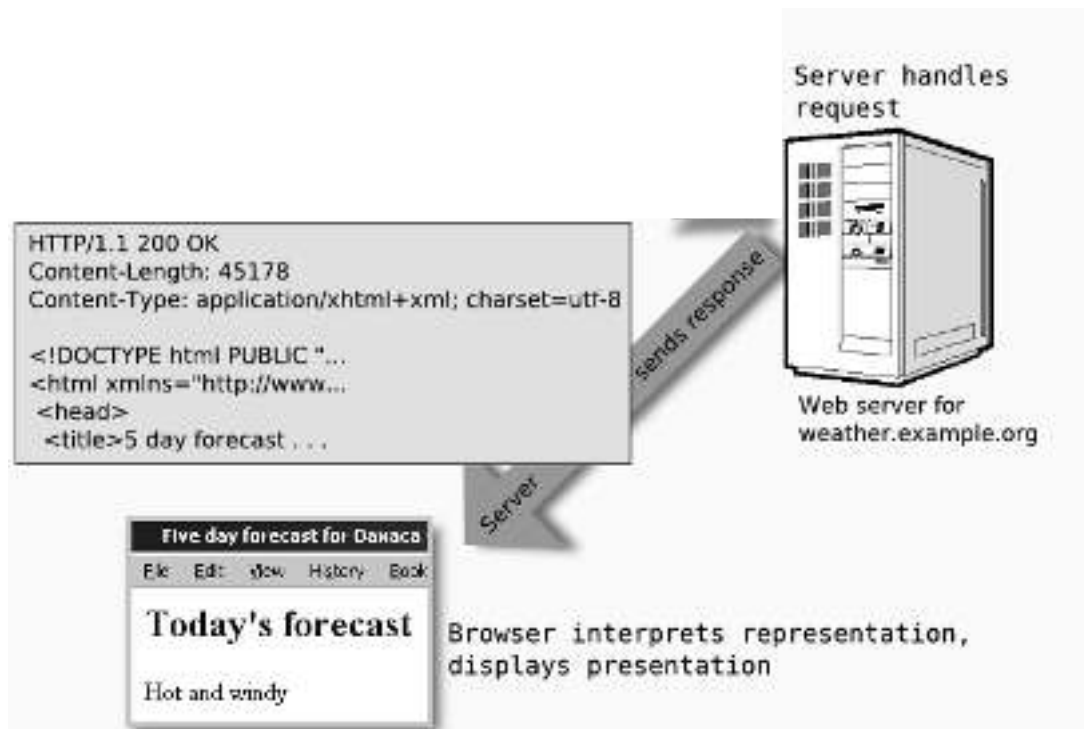
1. Statusna vrstica
2. Glava sporočila (eno ali več polj)
3. Prazna vrstica
4. Telo sporočila - opcijsko

## 1. Statusna vrstica

- Verzija protokola HTTP (HTTP/1.1)
- Statusna koda
- Tekst. Opisna fraza

## Statusne kode (skupine):

- 1xx – Informacijska
- 2xx – Uspešen prenos
- 3xx – Preusmeritev
- 4xx – Napaka odjemalca
- 5xx – Napaka strežnika



<http://www.ltg.ed.ac.uk/~ht/WhatAreURIs/>

## Statusne kode za odgovor

---

- ▶ 100 **Continue** - server je sprejel glavo, odjemalec lahko začne s pošiljanjem telesa v sporočilu.
- ▶ 200 **OK** – zahteva je normalno obdelana.
- ▶ 301 **Moved Permanently** – URI zahtevanega vira je spremenjen in vse naslednje zahteve naj bodo poslane na nov URI, ki je bil podan v odgovoru.
- ▶ 307 **Temporary Redirect** – URI zahtevanega vira je začasno spremenjen.
- ▶ 401 **Unauthorized** – vir je zaščiten z geslom in uporabnik ni posredoval veljavnega gesla.
- ▶ 403 **Forbidden** – vir se nahaja na strežniku, a je zaščiten proti branju.
- ▶ 404 **Not Found** - Noben vir za podan Request–URI se ne nahaja na strežniku
- ▶ 500 **Internal server Error** – program na strežniku je zaznal notranjo napako.

## 2. Glava sporočila (odgovor HTTP)

---

- ▶ **Date** – čas, ko je bil ustvarjen odgovor.
- ▶ **Server** – informacija o strežniškem programu, ki je ustvaril odgovor.
- ▶ **Last-Modified** – čas, ko je bil vrnjeni vir na to zahtevo nazadnje spremenjen. Preverjanje veljavnosti predpomnjene kopije.
- ▶ **Expires** – čas po katerem bo odjemalec preveril na strežniku predno bo ponovno zahteval vrnjeni vir (predpomnjenje).
- ▶ **ETag** – koda 'hash' vrnjenega vira. Uporabljena za predpomnjenje.
- ▶ **Accept-Ranges** – odjemalec lahko zahteva samo del dokumenta (za dolge datoteke pdf)
- ▶ **Location** – uporabljeno v odgovorih s statusom preusmeritve za določanje novega URI za zahtevan vir.



# Primer:

---

- ▶ Spletna povezava: <http://www.example.com>
- ▶ Zahteva HTTP in odgovor HTTP: <http://web-sniffer.net/> (Preverimo)

- ▶ Sporočilo HTTP z zahtevo:

GET / HTTP/1.1

Host: www.example.com

...

- ▶ Sporočilo HTTP z odgovorom:

Status: HTTP/1.1 200 OK

Date: Fri, 14 Feb 2014 12:24:18 GMT

...

- ▶ Vsebina (HTML koda)



```
Content (11.04 KB)
HTTP/1.1 200 OK
Date: Fri, 14 Feb 2014 12:24:18 GMT
Server: Apache/2.4.6 (Ubuntu)
X-Frame-Options: DENY
X-XSS-Protection: 1; mode=block
X-Content-Type-Options: nosniff
Content-Type: text/html; charset=UTF-8
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Example Domain</title>
</head>
<body>
<div>
Example Domain
</div>
<p>This domain is for use in illustrative examples in documents. You may wish to contact your organization's Internet coordinator to arrange for your domain name to be registered at the Internet Assigned Numbers Authority (IANA) website: http://www.iana.org for more information.</p>
</body>
</html>
```

# Spletni odjemalec - Brskalnik

- ▶ Program za dostop do spletnega strežnika (User agent)
  - Spletni brskalnik (Browser)
  - Brskalnik za tekst
  - Brskalnik, ki govori
- ▶ Zgodovina brskalnikov
  - Mosaic (NCSA), 1993
  - Netscape Navigator (NCC), 1994
  - Internet Explorer (Microsoft), 1995
  - Mozilla Firefox (Mozilla Foundation), 2003
  - Opera (Opera Software), 1996
  - Safari (Apple), 2007
- ▶ Okno brskalnika - območja
  - Področje odjemalca
  - Naslovna vrstica
  - Menijska vrstica
  - Orodna vrstica
  - Statusna vrstica



# Brskalnik – osnovna naloga

---

## ► Izvedba zahtev HTTP

1. Preoblikovanje URL-ja v veljavno sporočilo z zahtevo HTTP
2. Če je strežnik podan z imenom gostitelja (host name), se uporabi DNS za pretvorbo v IP naslov
3. Vpostavitev TCP povezave z uporabo IP naslova podanega spletnega strežnika
4. Pošiljanje zahteve HTTP preko TCP povezave in čakanje na odgovor HTTP, ki ga pošlje strežnik
5. Prikaz dokumenta, ki je vključen v odgovor. Če dokument ni v tekstovnem formatu, ampak v HTML, potem to vključuje:
  - urejanje dokumenta,
  - ustrezno pozicioniranje teksta in slik znotraj okna brskalnika,
  - kreiranje okvirov tabel,
  - uporaba pisav in barv itd.

# Brskalnik - uporabniške možnosti

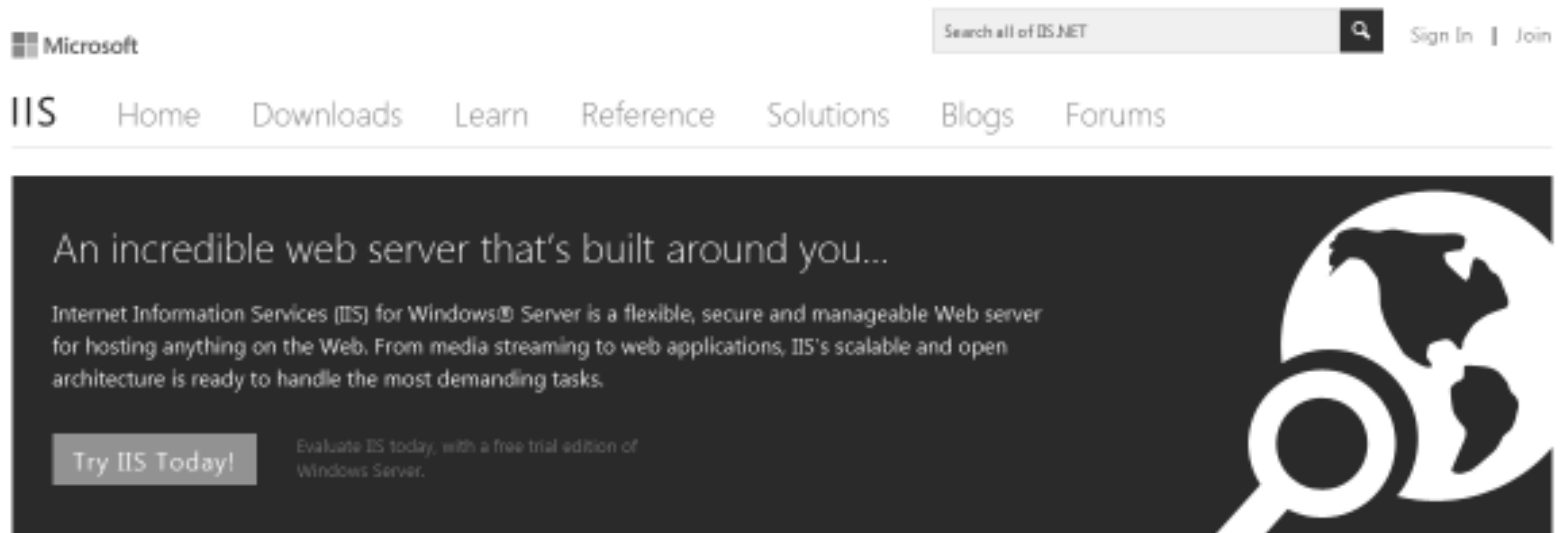
---

- ▶ Shranjevanje dokumentov v odjemalčev datotečni sistem
- ▶ Iskanje na strani (podobno kot v urejevalnikih)
- ▶ Avtomatsko polnjenje obrazcev
- ▶ Preference – željene nastavitve
  - Jezik
  - Niz znakov
  - Predpomnilniške nastavitve
  - HTTP nastavitve
- ▶ Definicija stila: velikost črk, barve ozadja, ...
- ▶ Meta podatki dokumenta: MIME tipi, koda HTML, ...
- ▶ Teme
- ▶ Zgodovina
- ▶ Zaznamki

# Spletni strežniki - IIS

---

- ▶ IIS - Internet Information Server (Microsoft),
- ▶ Microsoft, 1993
- ▶ ASP.NET Web applications
- ▶ <http://www.iis.net/>



# Spletni strežniki – Apache

---

- ▶ Apache Software Foundation
- ▶ 1995
- ▶ <http://www.apache.org/>



Home

About ▼

Projects

People ▼



## OPEN.

### THE APACHE SOFTWARE FOUNDATION

provides support for the Apache Community of open-source software projects, which provide software products for the public good.

## INNOVATION.

### THE APACHE PROJECTS ARE DEFINED

by collaborative consensus based processes, an open, pragmatic software license and a desire to create high quality software that leads the way in its field.

# Spletni strežniki - nginx

---

- ▶ nginx
- ▶ 2004
- ▶ <http://nginx.org/en/>, <http://wiki.nginx.org/Main>
- ▶ nginx plus

