

Tehnologii Web

programare Web (I)
protocolul HTTP
script-uri CGI



„Există 2 moduri de a scrie programe fără erori;
doar a treia manieră funcționează.”

Alan Perlis

Ce este Web-ul?

World Wide Web

spațiu informațional compus
din elemente de interes, numite **resurse**,
desemnate de identificatori globali – **URI/IRI**

detalii la www.w3.org/TR/webarch/
recomandare W3C (2004)

resurse Web

Aspecte de interes

identificarea

interacțiunea

reprezentarea prin formate de date

resurse Web

Aspecte de interes

protocol:
HTTP

identificarea

URI/IRI

interacțiunea

reprezentarea prin formate de date

limbaj(e)
de marcarea

Cum are loc interacțiunea
dintre client(i) și server(e) Web?

HTTP

HyperText Transfer Protocol

are ca temelie stiva TCP/IP

HTTP

situat la nivel de aplicație

transfer de hipertext/hipermedia
(HTTP – *HyperText Transfer Protocol*)

transport fiabil via socket-uri
(TCP – *Transmission Control Protocol*)

interconectare rețele + dirijare a datelor
(IP – *Internet Protocol*)

controlul accesului la mediul de transmitere
a datelor (MAC – *Medium Access Control*)

HTTP

HyperText Transfer Protocol

protocol fiabil, de tip cerere/răspuns

port standard de acces: **80**

HTTP

HTTP/1.1

standard Internet: RFC 2616 (1999)

din 2022, specificat de RFC 9110—9112

[httpwg.org/specs/
devdocs.io/http/](http://httpwg.org/specs/devdocs.io/http/)

un tutorial: www.code-maze.com/http-series/

HTTP

HTTP/2

RFC 7540 (2015)

înlocuit de RFC 9113 (2022)

axat asupra performanței

<http2.github.io>

HTTP

HTTP/2

mesaje binare

reutilizarea conexiunii TCP (*a single connection per host*)

multiplexare (*many parallel streams*)

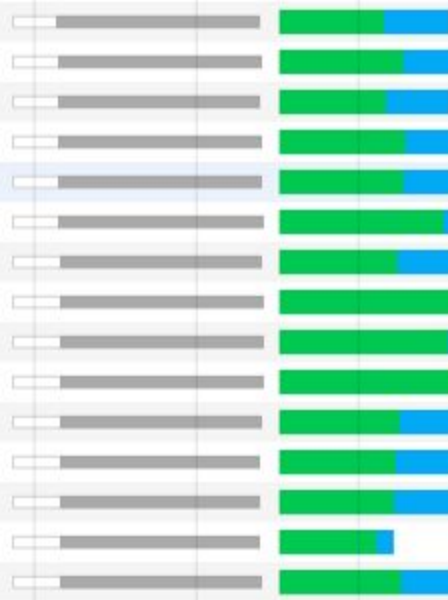
compresia anteturilor – HPACK

trimiterea mesajelor spre client (*server push*)

implementări: github.com/http2/http2-spec/wiki/Implementations

HTTP/2

	h_01.jpg...	...	200	h2	w...	...	1 2...
	h_02.jpg...	...	200	h2	w...	...	1 2...
	h_03.jpg...	...	200	h2	w...	...	1 2...
	h_04.jpg...	...	200	h2	w...	...	1 2...
	h_05.jpg...	...	200	h2	w...	...	1 2...
	h_06.jpg...	...	200	h2	w...	...	1 2...
	h_07.jpg...	...	200	h2	w...	...	1 2...
	h_08.jpg...	...	200	h2	w...	...	1 2...
	h_09.jpg...	...	200	h2	w...	...	1 2...
	h_10.jpg...	...	200	h2	w...	...	1 2...
	h_11.jpg...	...	200	h2	w...	...	1 2...
	h_12.jpg...	...	200	h2	w...	...	1 2...
	h_13.jpg...	...	200	h2	w...	...	1 2...
	h_14.jpg...	...	200	h2	w...	...	1 2...
	h_15.jpg...	...	200	h2	w...	...	1 2...



HTTP/1.1

	h_15.jpg...	...	200	http/1...	w...	...	1 6...
	h_16.jpg...	...	200	http/1...	w...	...	1 6...
	h_17.jpg...	...	200	http/1...	w...	...	9 6...
	h_18.jpg...	...	200	http/1...	w...	...	1 6...
	h_19.jpg...	...	200	http/1...	w...	...	1 6...
	h_20.jpg...	...	200	http/1...	w...	...	1 6...
	h_45.jpg...	...	200	http/1...	w...	...	1 1...
	h_34.jpg...	...	200	http/1...	w...	...	1 1...
	h_95.jpg...	...	200	http/1...	w...	...	1 1...
	h_88.jpg...	...	200	http/1...	w...	...	1 1...
	h_65.jpg...	...	200	http/1...	w...	...	1 1...
	h_67.jpg...	...	200	http/1...	w...	...	1 1...
	h_93.jpg...	...	200	http/1...	w...	...	1 1...
	h_33.jpg...	...	200	http/1...	w...	...	9 1...
	h_69.jpg...	...	200	http/1...	w...	...	1 1...



resurse de interese:

http2-explained.haxx.se

www.tunetheweb.com/blog/http-versus-https-versus-http2/

HTTP

HTTP/3

RFC 9114 (2022)

HTTP over QUIC (Quick UDP Internet Connections)

quicwg.org

HTTP

HTTP/3

UDP, multiplexarea fluxurilor de date, criptarea datelor,
stabilirea rapidă a conexiunilor

explicații:

http3-explained.haxx.se

httpwg.org/specs/rfc9114.html

HTTP

HTTP/3

statistici utilizzare:

w3techs.com/technologies/details/ce-http3
blog.cloudflare.com/http3-usage-one-year-on

HTTP: architectura

Server Web

daemon – „spirit protector”

Client Web

navigator (*browser*), robot (*crawler*), *player*,...

HTTP: arhitectura

Server Web

Apache HTTP Server, Eclipse Jetty, Gunicorn, Lighttpd,
Microsoft Internet Information Services, **NGINX** etc.

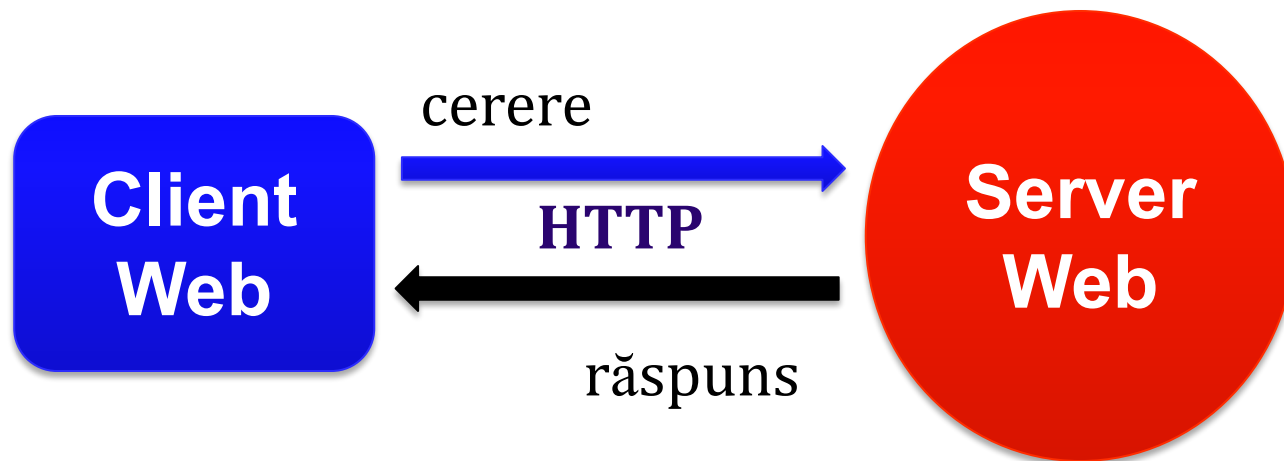
Client Web

Mosaic → Netscape → Mozilla.org → **Firefox**, Microsoft
Internet Explorer, **Chromium**, Edge; **curl**, wget, Echofon, ...
detalii în prezentarea „Arhitectura navigatorului Web”:
profs.info.uaic.ro/~busaco/teach/courses/staw/web-film.html#week5

HTTP

Cererea și răspunsul

accesarea – eventual, modificarea – reprezentării
resursei via URI-ul asociat



HTTP: termeni

Mesaj

unitatea de bază a unei comunicații HTTP
(cerere sau răspuns)

HTTP: termeni

Intermediar

mandatar (*proxy*)

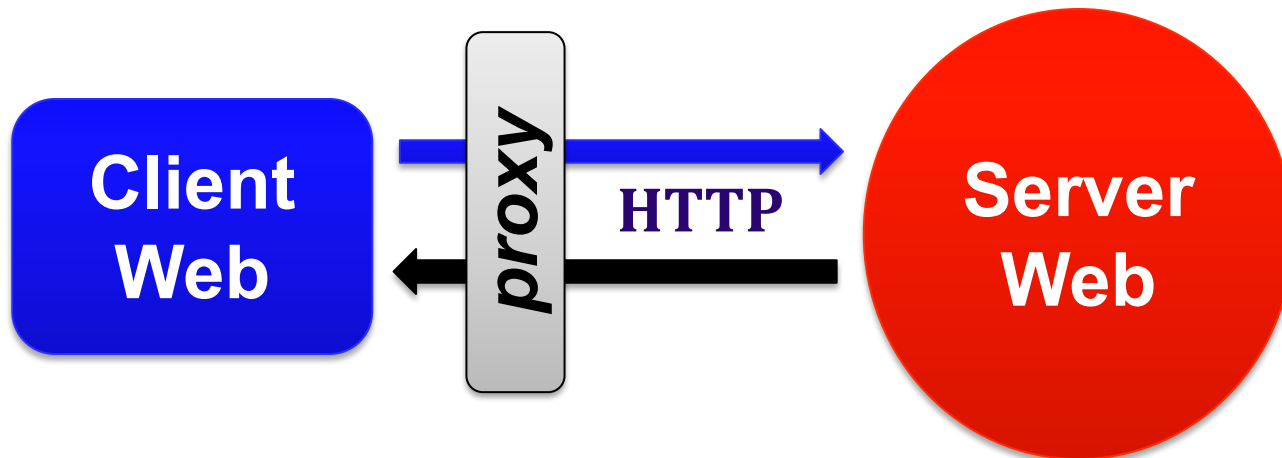
poartă (*gateway*)

tunel (*tunnel*)

HTTP: termeni

Proxy

localizat în proximitatea clientului/serverului
are rol atât de server, cât și de client



HTTP: termeni

Proxy

forward proxy

intermediar pentru un grup de clienți din vecinătate
solicită cereri ca venind din partea clientului

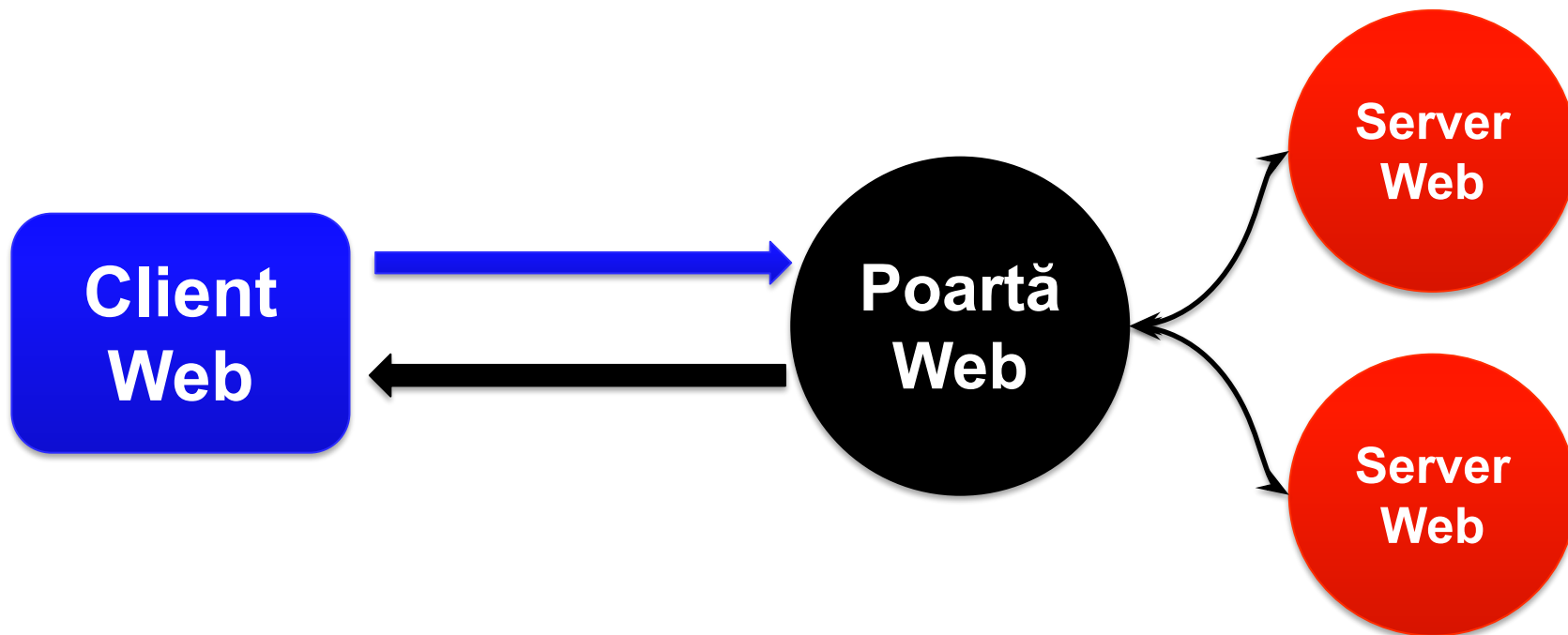
reverse proxy

intermediar pentru un grup de servere din vecinătate

HTTP: termeni

Gateway

intermediar care ascunde serverul țintă,
clientul neștiind aceasta



HTTP: termeni

Gateway

poate asigura:

echilibrarea încărcării – *load balancing*

stocarea temporară a datelor – *caching*

translatarea mesajelor sau cererilor (*e.g.*, HTTPS → HTTP)

alte operații de negociere – rol de mediator/*broker*

HTTP: termeni

Gateway

soluții software în regim deschis (*open source*):

Apache Traffic Server – trafficserver.apache.org

HAProxy – www.haproxy.org

Squid – www.squid-cache.org

Varnish – varnish-cache.org

la nivel de *cloud*: Amazon ELB (*Elastic Load Balancing*)

aws.amazon.com/elasticloadbalancing/

HTTP: termeni

Tunnel

retransmite – uzual, criptat – mesajele HTTP

HTTP: termeni

Tunnel

context:

protocolul **HTTPS** – asigură comunicații „sigure”
HTTP via TLS (*Transport Layer Security*)

autentificare pe baza certificatelor digitale
+ criptare bidirecțională a datelor

HTTP: termeni

Tunnel

tutorial vizual: howhttps.works

a se vedea și inițiativa Let's Encrypt:
letsencrypt.org

detalii despre
o conexiune
HTTPS
oferite de
instrumentele
pentru
dezvoltatori ale
browser-ului Web

▼ Connection:

Protocol version: "TLSv1.3"

Cipher suite: "TLS_AES_128_GCM_SHA256"

Key Exchange Group: "x25519"

Signature Scheme: "ECDSA-P256-SHA256"

▼ Host validator.w3.org:

HTTP Strict Transport Security: "Disabled"

Public Key Pinning: "Disabled"

▼ Certificate:

▼ Issued To

Common Name (CN): "sni.cloudflaressl.com"

Organisation (O): "Cloudflare, Inc."

Organisational Unit (OU): "<Not Available>"

▼ Issued By

Common Name (CN): "Cloudflare Inc ECC CA-3"

Organisation (O): "Cloudflare, Inc."

Organisational Unit (OU): "<Not Available>"

▼ Period of Validity

Begins On: "Mon, 06 Feb 2023 00:00:00 GMT"

Expires On: "Tue, 06 Feb 2024 23:59:59 GMT"

▼ Fingerprints

SHA-256 Fingerprint: "33:0C:6D:2C:73:DF:45:24:BE"

SHA1 Fingerprint: "6C:C9:8A:A3:48:99:C3:50:AD:80"

avansat

sistem de criptare
folosit

date vizând
certificatul digital

HTTP: termeni

Cache

zonă locală de stocare – în memorie, pe disc –
a mesajelor (datelor)

la nivel de server și/sau client

HTTP: termeni

Cache

zonă locală de stocare – în memorie, pe disc –
a mesajelor (datelor)

cererile ulterioare vor fi rezolvate mai rapid

context: asigurarea performanței aplicațiilor Web

HTTP: **mensaje**

Mensaje HTTP = **cabecera + cuerpo**

HTTP: **mesaje**

Antet

include o mulțime de câmpuri

field-name ":" [**field-value**] **CRLF**

CR = *Carriage Return* \r – cod 13

LF = *Line Feed* \n – cod 10

HTTP: **mensaje**

Cerere (*request*) HTTP

Method **Request-URI** **ProtocolVersion** **CRLF**
[**Message-header**] [**CRLF** **MIME-data**]

```
GET /~busaco/teach/courses/web/ HTTP/1.1 CRLF  
Host: profs.info.uaic.ro
```

HTTP: mesaje

Răspuns (*response*) HTTP

HTTP-version Digit Digit Digit Reason
CRLF Content

HTTP/1.1 200 OK CRLF ...

HTTP: metode

GET

cerere – efectuată de un client – pentru accesul la reprezentarea unei resurse

HTTP: metode

GET

cerere – efectuată de un client – pentru accesul la reprezentarea unei resurse

document HTML, foaie de stiluri CSS, imagine în format JPEG/PNG, ilustrație vectorială SVG, program JavaScript, date în format JSON (*JavaScript Object Notation*), model 3D, flux de știri RSS (XML), prezentare PDF, arhivă ZIP, video,...

HTTP: metode

HEAD

similară cu GET
uzual, furnizează doar meta-date

HTTP: metode

HEAD

similară cu GET

uzual, furnizează doar meta-date

e.g., tipul MIME (*Media Type*) al resursei,
ultima actualizare,...

HTTP: metode

PUT

actualizează o reprezentare de resursă sau
eventual creează o resursă la nivel de server Web

A blue, cloud-like bubble with a soft shadow, containing white text. It has three small circles at the top right, suggesting a trail or movement.

amănunte în cursul
privind serviciile Web

HTTP: metode

POST

creează o resursă, trimittând uzual entităţi
(date, acţiuni) spre server

HTTP: metode

POST

creează o resursă, trimittând uzual entităţi
(date, acţiuni) spre server

e.g., datele introduse de utilizator
în câmpurile unui formular Web

HTTP: metode

DELETE

șterge o resursă – reprezentarea ei – de pe server

HTTP: metode

Remarcă

tradițional, *browser*-ul Web permite doar folosirea metodelor GET și POST

HTTP: metode

O metodă e considerată **sigură** (*safe*) dacă nu conduce la modificarea stării serverului *i.e.* pe server nu au loc acțiuni având efecte colaterale

GET și HEAD sunt *safe*
POST, PUT și DELETE nu sunt *safe*

HTTP: metode

O metoda e considerată **idempotentă** în cazul în care cereri identice vor conduce la returnarea aceluiasi răspuns (aceeași reprezentare)

GET, HEAD, PUT și DELETE sunt idempotente
POST nu este idempotentă

HTTP: reprezentări ale resursei

Codificări ale setului de caractere (*encodings*)

ISO-8859-1

ISO-8859-2

KOI8-R

ISO-2022-JP

UTF-8

UTF-16 Little Endian

...

HTTP: reprezentări ale resursei

Codificarea mesajelor (conținutului)

comprimare, asigurarea identității
și/sau integrității

abordare tradițională: gzip – gzip.org

abordări moderne:

Brotli – brotli.org

Zstandard – zstd.net

HTTP: reprezentări ale resursei

Formatul reprezentării

text

HTML, CSS, text obișnuit, cod JavaScript, document XML

sau

binar

image, document PDF, resursă multimedia, arhivă .gzip

HTTP: reprezentări ale resursei

Tipul conținutului resursei

media types

HTTP: câmpuri (attribute)

Content-Type

permite transferul datelor de orice tip

Content-Type: tip/subtip

HTTP: câmpuri (attribute)

Content-Type

specificat prin Media Types – MIME
(*Multipurpose Internet Mail Extensions*)

desemnează un set de **tipuri primare de conținut**
+ **sub-tipuri** adiționale

inițial, utilizat în contextul poștei electronice

HTTP: câmpuri (attribute)

Tipuri MIME:

text desemnează formate textuale

text/plain – text neformatat

text/html – document HTML (*HyperText Markup Language*)

text/css – foaie de stiluri CSS (*Cascading Style Sheet*)

HTTP: câmpuri (attribute)

Tipuri MIME

image specifică formate grafice

image/gif – imagini GIF (*Graphics Interchange Format*)

image/jpeg – fotografii JPEG (*Joint Picture Experts Group*)

image/png – imagini PNG (*Portable Network Graphics*)

image/webp – imagini WebP (*Web Picture Format*)

image/avif – imagini (animate) optimizate pentru Web

image/svg+xml – ilustrații SVG (*Scalable Vector Graphics*)

HTML `` – developer.mozilla.org/docs/Web/HTML/Element/img

HTTP: câmpuri (attribute)

Tipuri MIME:

audio desemnează conținuturi sonore

audio/mpeg – resursă codificată în format MP3
specificația privitoare la date audio a standardului MPEG
(*Motion Picture Experts Group*) – tools.ietf.org/html/rfc3003

audio/ac3 – resursă audio compresată
conform standardului AC-3 – www.atsc.org/standards/

HTML `<audio>` – developer.mozilla.org/docs/Web/HTML/Element/audio

HTTP: câmpuri (attribute)

Tipuri MIME:

video definește conținuturi video: animații, filme

video/av1 – format deschis: *Alliance for Open Media*

aomedia.org/av1-features/

video/h265 – resursă în format *High Efficiency Video Coding*

www.itu.int/rec/T-REC-H.265

standard W3C pentru comunicații multimedia în timp-real:

WebRTC (2023): www.w3.org/TR/webrtc/

HTML **<video>** – developer.mozilla.org/docs/Web/HTML/Element/video

HTTP: câmpuri (attribute)

Tipuri MIME:

application desemnează formate care vor putea fi procesate de aplicații disponibile la nivel de client

application/epub+zip – *e-book* EPUB în format ZIP

application/javascript – program JavaScript

application/json – date JSON (*JavaScript Object Notation*)

application/octet-stream – „șuvoi” arbitrar de octeți

application/senml+xml – date XML de la senzori: RFC 8428

HTTP: câmpuri (attribute)

Tipuri MIME:

multipart utilizat la transferul datelor compuse

multipart/mixed – conținut mixt

multipart/alternative – conținuturi alternative



e.g., calități diferite ale
stream-uri multimedia

N. Freed *et al.*, *Media Types* (29 februarie 2024)
www.iana.org/assignments/media-types/media-types.xhtml

calendar+json	application/calendar+json	Calendar în format JSON
csv	text/csv	Date în format CSV
heic	image/heic	High Efficiency Image File
markdown	text/markdown	Text în format Markdown
msword	application/msword	Document Word (MS Office)
vnd.rar	application/vnd.rar	Format proprietar – <i>vendor</i>
VP8	video/VP8	Format video VP8: RFC 7741
woff2	font/woff2	Format de fonturi tipografice
x3d	model/x3d+xml	Model 3D în format X3D XML
zip	application/zip	Arhivă ZIP

(în loc de) pauză

Today

Are you https? Because I'm ://
without you

Nah, I'm just plain old HTTP
because I'm insecure

Delivered

HTTP: câmpuri (attribute)

Location

Location ":" "http(s)://" **authority** [":" **port**] [**abs_path**]

redirecționează clientul spre o altă reprezentare a resursei
(*HTTP redirect*)

Location: http://undeva.info:8080/s-a_mutat.html

HTTP: câmpuri (attribute)

Referer

desemnează URI-ul resursei Web
care a referit resursa curentă

folosit pentru a determina de unde provin
cererile privind un document dat (*back-links*)
pentru a efectua statistici, jurnalizări, *caching*,...

HTTP: câmpuri (attribute)

Host

specifică adresa – IP sau simbolică – a mașinii
de pe care se solicită accesul la o resursă

HTTP: câmpuri (attribute)

Sunt definite și altele, referitoare la:

conținut acceptat (*content negotiation*) – *e.g.*, Accept
autentificare + autorizare – WWW-Authenticate Authorization
acces condiționat la resurse – If-Match, If-Modified-Since, ...
cache – Age, Cache-Control, Expires, ETag etc.
proxy – Proxy-Authenticate, Proxy-Authorization, Via
livrare de mesaje (*HTTP push*) – Topic, TTL, Urgency
...și multe multe altele

developer.mozilla.org/docs/Web/HTTP/Headers

HTTP: starea

Coduri de informare (1xx)

100 Continue, 101 Switching Protocols, 102 Processing




Status	Method	Domain	File	Initiator	Type	Transferred
101	GET	 public-api.wordpress.com	newest-note-data	websocket	plain	244 B

comutarea protocolului
aici, de la HTTP la WebSocket (RFC 6455)

HTTP: starea

Coduri de succes (2xx)

200 OK, 201 Created, 202 Accepted,
204 No Content, 206 Partial Content,...

200	OPTIONS	 people-pa.clients6.g...	people?person_id= xhr	html	681 B
200	POST	 clients6.google.com	v2internal?\$ct=mu m=b:353 (xhr)	mixed	1.88 kB
200	GET	 people-pa.clients6.g...	people?person_id= m=b:353 (xhr)	json	7.63 kB

OPTIONS – determină facilități ale serverului sau cerințe vizând o resursă
transferurile sunt asincrone (**xhr** – se utilizează **XMLHttpRequest**)

204	POST	 docs.google.com	naLogImpressions?id=1i0F1p 1526657789-waffle_js_...	plain	6.43 kB
-----	------	---	---	-------	---------

HTTP: starea

Coduri de redirectionare (3xx)

300 Multiple Choices, 301 Moved Permanently, 302 Found, 303 See Other, 304 Not Modified, 305 Use Proxy etc.

Status	Method	Domain	File	Initiator	Type	Transferred
301	GET	tinyurl.com	fii-web-zoom	document	html	3.96 kB
200	GET	us02web.zoom.us	83547687804?pwd=bk1YT2EvaTh	document	html	6.75 kB
304	GET	static.ada.support	embed.js	script	js	cached
200	304 Not Modified	us02web.zoom.us	lres	script	js	cached
302	GET	www.slideshare.net	/busaco/	document	html	10.35 kB 37.79 kB
200	GET	www.slideshare.net	/busaco/	document	html	11.08 kB 37.79 kB
	GET	www.linkedin.com	noauthtracker?type=le	Blocked By uBlock ...		
200	GET	public.slidesharec...	ss-core.css?bc41874			

Slow server response time (1.06 s). The recommended limit is 500 ms.

HTTP: starea

Coduri de eroare la nivel de client (4xx)


400 Bad Request, 401 Unauthorized, 403 Forbidden, 405 Method Not Allowed, 408 Request Timeout, 410 Gone, 414 Request-URI Too Long, 415 Unsupported Media Type, 423 Locked, 429 Too Many Requests,...

400	GET	 i0.wp.com	Logo_design_jam_iasi.png	img	html
400	400 Bad Request	i0.wp.com	Logo_design_jam_iasi.png	img	html
401	GET	 api.nytimes.com	articlesearch.json?q=w&api-key=1	document	

HTTP: starea

Coduri de eroare la nivel de server (5xx)

500 Internal Server Error, 501 Not Implemented,
502 Bad Gateway, 503 Service Unavailable,
505 HTTP Version Not Supported, 508 Loop Detected,...

501	PATCH	 httpbin.org	501	swagger-ui-bundle.js:72 (fetch)	html
5	501 Not Implemented	tpbin.org	501	swagger-ui-bundle.js:72 (fetch)	html

transferurile sunt asincrone (**fetch** – se recurge la **Fetch API** oferit de HTML5)

Error 502

Ray ID: 6ebb1499bb939229 • 2022-03-14 06:51:22 UTC

avansat

Bad gateway



You

Browser
Working



Frankfurt

Cloudflare
Working



emojipedia.org

Host
Error

Cloudflare oferă servicii de distribuție de conținut, asigurând performanța și securitatea aplicațiilor Web și are rol de *reverse proxy*, fiind situat între *browser*-ul Web al utilizatorului și situl găzduit pe serverul Web țintă

HTTP: jurnalizare

Cererile adresate serverului Web sunt jurnalizate

Common Log Format

format standardizat de fișier text

Apache HTTP Server: modulul **mod_log_config**

<httpd.apache.org/docs/current/logs.html>

NGINX: configurare via directiva **access_log**

docs.nginx.com/nginx/admin-guide/monitoring/logging/

w10.uaic.ro - msi2019 [13/Feb/2020:14:53:14 +0200]
"GET /~vidrascu/MasterSI2/note/Restanta.pdf HTTP/1.1" 206 25227
"https://profs.info.uaic.ro/~vidrascu/MasterSI2/index.html" "...Chrome/79.0.3945.130"
82-137-8-231.rdsnet.ro - - [13/Feb/2020:15:38:23 +0200]
"POST /~computernetworks/login.php HTTP/1.1" 302 1115
"https://profs.info.uaic.ro/~computernetworks/login.php"
"...X11; Ubuntu; Linux x86_64 ... Firefox/73.0"
ec2-23-21-0-202.compute-1.amazonaws.com - - [13/Feb/2020:15:48:29 +0200]
"GET /~busaco/teach/courses/web/presentations/web01ArhitecturaWeb.pdf HTTP/1.1"
200 2081804 "-" "HTTP_Request2/2.3.0 (http://pear.php.net/package/http_request2)..."
199.16.156.126 - - [13/Feb/2020:15:58:58 +0200]
"GET /robots.txt HTTP/1.1" 404 182 "-" "Twitterbot/1.0"
psihologie-c-113.psih.uaic.ro - - [13/Feb/2020:16:03:04 +0200]
"GET /~busaco/ HTTP/1.1" 200 1942 "-" "... Firefox/72.0..."
psihologie-c-113.psih.uaic.ro - - [13/Feb/2020:16:03:04 +0200]
"GET /~busaco/csb.css HTTP/1.1" 200 852 "https://profs.info.uaic.ro/~busaco/"
"... Firefox/72.0..."
proxy-220-255-2-224.singnet.com.sg - - [13/Feb/2020:16:23:23 +0200]
"GET /favicon.ico HTTP/1.1" 200 1406 "-" "...UCBrowser/11.3.8.976..."
c2.uaic.ro - - [13/Feb/2020:16:33:43 +0200]
"GET /~busaco/teach/courses/web/ HTTP/1.1" 304 - "-" "...Chrome/80.0.3987.106"
220.181.51.219 - - [13/Feb/2020:19:20:20 +0200]
"HEAD /%7Ebusaco/music/09.Sabin%20Buraga%20-...mp3 HTTP/1.0" 200 - "-"
"NSPlayer/10.0.0.4072 WMFSDK/10.0"

host-static-92-114-239-107.moldtelecom.md - - [15/Feb/2021:10:49:18 +0200]
"POST / HTTP/1.1" 200 8923 "https://webmail-studs.info.uaic.ro/?_action=login&_url=" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) ... Chrome/88.0.4324.150 Safari/537.36"
host-static-92-114-239-107.moldtelecom.md - - [15/Feb/2021:10:49:18 +0200]
"GET /skins/default/common.css?s=1322121180 HTTP/1.1" 200 3668
"https://webmail-studs.info.uaic.ro/" " ... Chrome/88.0.4324.150 Safari/537.36"
host-static-92-114-239-107.moldtelecom.md - - [15/Feb/2021:10:49:18 +0200]
"GET /program/js/jquery.min.js?s=1315898178 HTTP/1.1" 200 32833
"https://webmail-studs.info.uaic.ro/" " ... Chrome/88.0.4324.150 Safari/537.36"
82.77.237.116 - - [15/Feb/2021:10:51:02 +0200]
"POST /autodiscover/autodiscover.xml HTTP/1.1" 404 6872 "-" " ... Thunderbird/78.7.1"
89.137.143.178 - - [15/Feb/2021:10:53:13 +0200]
"GET /passwd/ HTTP/1.1" 503 564 "https://webmail-studs.info.uaic.ro/" "Mozilla/5.0..."
adsl92-86-48-213.romtelecom.net - - [15/Feb/2021:11:06:09 +0200]
"GET /skins/default/images/abook_toolbar.png HTTP/1.1" 200 16933
"https://webmail-studs.info.uaic.ro/?_action=login&_url=" " ... Android 9; ANE-LX1..."
89.137.143.178 - - [15/Feb/2021:11:11:20 +0200] "-" 408 145 "-" "-" *request timeout*
ec2-54-234-157-199.compute-1.amazonaws.com - - [15/Feb/2021:11:17:20 +0200]
"GET /~adrian.schipor/lab9/1.exe HTTP/1.1" 404 7340 "-" " ... Safari/537.36"
79-112-26-111.iasi.fiberlink.ro - - [15/Feb/2021:11:18:49 +0200]
"GET /skins/default/images/mail_footer.png HTTP/1.1" 200 1338
"https://webmail-studs.info.uaic.ro/?_task=mail&_mbox=INBOX&_refresh=1"
" ... CPU iPhone OS 14_3 ... Mobile/15E148 Safari/604.1"

GET /~busaco/teach/courses/web/ HTTP/1.1
Host: profs.info.uaic.ro
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:110.0)
Gecko/20100101 Firefox/110.0
Accept: [text/html](#),[application/xhtml+xml](#),[application/xml](#);
q=0.9,[image/avif](#),[image/webp](#),[*/*](#);q=0.8
Accept-Language: en-GB,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Referer: <https://profs.info.uaic.ro/~busaco/>

HTTP: exemplu de cerere

HTTP/1.1 200 OK

Date: Fri, 23 Feb 2024 08:33:24 GMT

Server: Apache/2.4.35

Last-Modified: Tue, 20 Feb 2024 10:53:12 GMT

Content-Encoding: gzip

Content-Length: 1060

Keep-Alive: timeout=5, max=100

Connection: Keep-Alive

Content-Type: text/html; charset=UTF-8

câmpuri-antet
(meta-date)

<!DOCTYPE html>

<html lang="ro">

...

...

</html>

HTTP: exemplu de răspuns

A simple HTTP Request & Response Service.

Run locally: `$ docker run -p 80:80 kennethreitz/httpbin`

Schemes

HTTPS

inspectarea online a mesajelor HTTP
via aplicația Web httpbin.org

HTTP Methods Testing different HTTP verbs

DELETE

`/delete` The request's DELETE parameters.

GET

`/get` The request's query parameters.

PATCH

`/patch` The request's PATCH parameters.

POST

`/post` The request's POST parameters.

PUT

`/put` The request's PUT parameters.

Auth Auth methods

GET

`/basic-auth/{user}/{passwd}` Prompts the user for authorization using HTTP Basic Auth.

eventual, pot fi furnizate
date vizând
autentificarea clientului

GET /services/feeds/photos_public.gne?tags=FII,lasi

avansat

Host: www.flickr.com

...

HTTP/2 200 OK

Content-Type: **application/atom+xml**; charset=utf-8

date în format Atom
(procesate de client)

Date: Fri, 23 Feb 2024 08:49:57 GMT

Server: Apache/2.4.55 (Ubuntu)

expiră în trecut
(nu va fi păstrat în *cache*)

Expires: Mon, 26 Jul 1997 05:00:00 GMT

Last-modified: Sun, 02 Nov 2014 06:58:25 GMT

Cache-control: private, no-store, no-cache, must-revalidate

Pragma: no-cache

X-Frame-Options: SAMEORIGIN

X-Cache: Miss from cloudfront

Via: 1.1 46d5c1a4d1e3a5c8a14bdb9b6676ba11.cloudfront.net (CloudFront)

X-Firefox-Spdy: h2

câmpurile X- nu
sunt standardizate

obținerea informațiilor despre fotografii publice oferite de Flickr

HTTP: jurnalizare – formatul HAR

Interacțiunea dintre *browser* și serverul Web (cereri + răspunsuri) poate fi stocată în fișiere HAR (*HTTP ARchive*)

format bazat pe JSON

www.softwareishard.com/blog/har-12-spec/

exemplificare: gist.github.com/igrigorik/3495174

HTTP: jurnalizare – formatul HAR

Interacțiunea dintre *browser* și serverul Web (cereri + răspunsuri) poate fi stocată în fișiere HAR (*HTTP ARchive*)

scop principal: analizarea traficului Web

aspect de interes: performanța

a se vedea și inițiativa *HTTP Archive*:

httparchive.org

HTTP: API-uri (biblioteci)

cURL + libcurl

(C, Java, Haskell, .NET, PHP, Ruby,...) – curl.se

Apache HttpComponents (Java) – hc.apache.org

HTTP (modul Node.js) – nodejs.org/api/http.html

http.client (Python 3)

LibHTTP (bibliotecă C): www.libhttp.org



detalii în cursul
despre servicii Web

HTTP: instrumente la nivel de client

Google Chrome Developer Tools
developer.chrome.com/docs/devtools/

Firefox Developer Tools
firefox-source-docs.mozilla.org/devtools-user/

Fiddler – *free Web debugging proxy
for MacOS, Windows, and Linux*
www.telerik.com/fiddler

Inspector Console Debugger Performance Network Storage DOM Application

Filter URLs

All HTML CSS JS XHR Fonts Images Media WS Other

Status	Method	Domain	File	Initiator	Type	Transferred	Size
304	GET	occ-0-3018-3466....	AAAABb89JNT487jNJ_mRF6AaGV3q9UWNzNxNKbsfF0suBLNc	img	jpeg	cached	41.04 KB
304	GET	occ-0-3018-3466....	AAAABeFa98bPMrT-JmX04T3-naWQ7ZH9kcmQRWHxaSSJN4C	img	jpeg	cached	25.01 KB
200	GET	codex.nflxext.com	none	script	js	3.96 KB	9.01 KB
200	GET	codex.nflxext.com	none	script	js	777.11 KB	2.62 MB
	GET	assets.nflxext.com	nficon2016.ico	FaviconLoader.jsm:1...	x-icon	16.56 KB (...)	16.56 KB
101	GET	push.prod.netflix.c...	ws	websocket	plain	175 B	0 B
	GET	ae.nflximg.net	adtech_iframe_target_05.html?data={"membership_status":"Cl	none:11 (subdocume...		Blocked B...	
200	POST	www.netflix.com	pathEvaluator?webp=false&drmSystem=widevine&isVolatileBillk	none:9 (xhr)	json	7.28 KB	28.72 KB
200	GET	www.netflix.com	probe?monotonic=false&device=web&iter=0	cadmium-playercore...	json	2.42 KB	1.12 KB
200	POST	www.netflix.com	pathEvaluator?webp=false&drmSystem=widevine&isVolatileBillk	none:9 (xhr)	json	2.48 KB	27 B
200	GET	nfrt-nrdp-nlb.prod....	probe?data=5120&device=web&ord=4&recipe=probnikNoFront	cadmium-playercore...	octet-stream	5.78 KB	5 KB
200	GET	nfrt-mobile-direct....	probe?data=5120&device=web&ord=0&recipe=probnikNoFront	cadmium-playercore...	octet-stream	5.79 KB	5 KB
200	GET	nfrt-mobile-nlb.pro...	probe?data=5120&device=web&ord=2&recipe=probnikNoFront	cadmium-playercore...	octet-stream	5.79 KB	5 KB
200	GET	nfrt-mobile-direct...	probe?data=5120&device=web&ord=1&recipe=probnikNoFront	cadmium-pl	am	5.79 KB	5 KB
200	GET	nfrt-nrdp-direct.pr...	probe?data=5120&device=web&ord=3&recipe=probnikNoFront	cadmium-p	stream	5.79 KB	5 KB
304	GET	occ-0-3018-3466....	AAAABZSqcXJDH3nWFZk9HJx5FMAabIXDeFHovYYVlcDQg1d...	none:6 (img)	jpeg	cached	35.59 KB
304	GET	occ-0-3018-3466....	AAAABYIET22cXmL7amS5zxBdaguxTnVQspnUjsapLjyKXeOQC...	none:6 (img)	jpeg	cached	18.54 KB
304	GET	occ-0-3018-3466....	AAAABWdYKiWuRs8MbYQ_6TiWe8exVZZECeva_6tP7OiNCLkrk...	none:6 (img)	jpeg	cached	16.78 KB

100 requests 6.49 MB / 1.14 MB transferred Finish: 2.09 min DOMContentLoaded: 1.57 s load: 5.04 s

inspectarea cererilor HTTP efectuate de *browser-ul* Web

Care e arhitectura serverului Web?

HTTP: server Web

Deservește cereri multiple provenite de la clienți
pe baza protocolului HTTP

HTTP: server Web

Deservește cereri multiple provenite de la clienți
pe baza protocolului HTTP

fiecare cerere e considerată independentă de alta,
chiar dacă provine de la același client Web
→ nu e păstrată starea conexiunii – *stateless*

HTTP: server Web

Tradițional, implementarea serverului Web este una *pre-forked* sau *pre-threaded*

se creează un număr de procese copil ori fire de execuție (*threads*) la inițializare, fiecare proces/fir interacționând cu un anumit client



vezi și suplimentul privitor
la Apache HTTP Server

Cum dezvoltăm aplicații Web
pe partea de server?

necesitate

Generarea **dinamică** – la nivel de **server** –
de **reprezentări** ale unor resurse
solicitate de clienți

soluții

CGI – *Common Gateway Interface*

Servere de aplicații Web

Cadre de lucru (*framework-uri*) Web

soluție: cgi

Interfață de programare, independentă de limbaj,
facilitând interacțiunea dintre clienți
și programe invocate la nivel de server Web

standard de facto

RFC 3875 – datatracker.ietf.org/doc/html/rfc3875
www.w3.org/CGI/

cgi

Un program (*script*) CGI se invocă pe server

explicit

i.e., preluarea datelor dintr-un formular Web
după apăsarea butonului de tip *submit*

cgi

Un program (*script*) CGI se invocă pe server

implicit

exemplu: la fiecare vizită se generează o nouă reclamă (*e.g.*,
banner publicitar)

cgi: caracterizare

Script-urile CGI pot fi concepute
în **orice** limbaj disponibil pe server

limbaje interpretate

bash, Perl – *e.g.*, modulul Perl::CGI –, Python, Ruby,...

limbaje compilate

C, C++, Rust etc.

cgi: programare

Orice program CGI va scrie datele
– reprezentarea resursei Web –
la ieșirea standard (*stdout*)

cgi: programare

Pentru a desemna tipul reprezentării generate, se folosesc anteturi HTTP – MIME (*Media Types*)

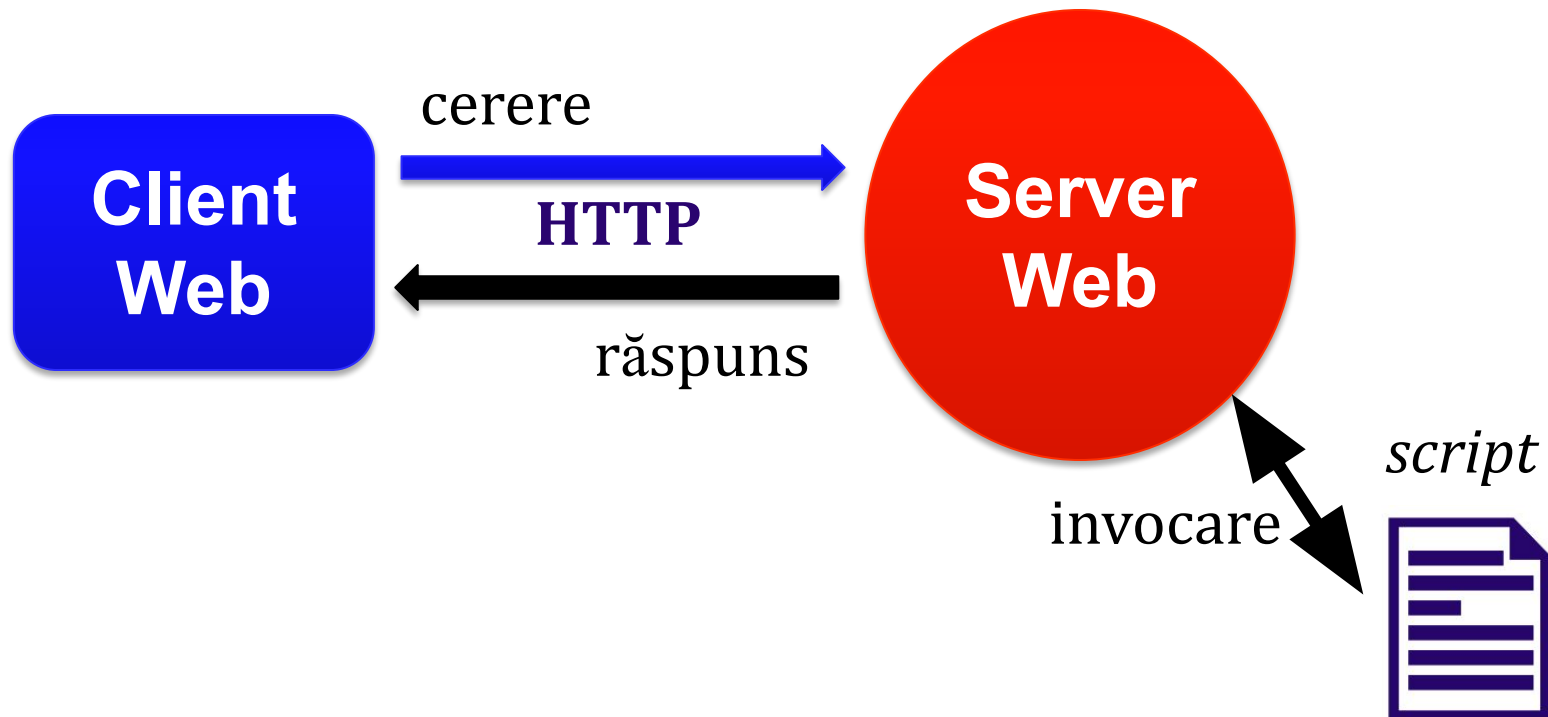
example:

Content-type: text/html

Content-type: image/webp

cgi: programare

Interacțiunea dintre clientul și serverul Web



cgi: variabile

Un *script* CGI are acces la variabile de mediu
specifice unei cereri transmise programului CGI:

- REQUEST_METHOD** – metoda HTTP (GET, POST,...)
- QUERY_STRING** – șir de interogare: date trimise de client
- REMOTE_HOST, REMOTE_ADDR** – adresa clientului
- CONTENT_TYPE** – tipul conținutului conform MIME
- CONTENT_LENGTH** – lungimea în octeți a conținutului

cgi: variabile

Variabile suplimentare
generate, uzual, de serverul Web:

HTTP_ACCEPT – tipurile MIME acceptate de *browser*

HTTP_COOKIE – date despre *cookie*-uri

HTTP_HOST – informații despre gazdă (client)

HTTP_USER_AGENT – informații privind clientul

...și altele

cgi: variabile

obținerea valorilor variabilelor de mediu ale sistemului
în urma invocării prin **GET** a *script*-ului *variabile.cgi*
stocat la nivel de server Web
(programul trebuie să aibă drepturi de citire și execuție)

```
#!/bin/bash
```

```
# Stabilim tipul conținutului
```

```
echo "Content-type: text/plain";
```

```
echo
```

```
# Executăm comanda 'set' din Linux
```

```
# pentru a afișa variabilele de mediu
```

```
set
```

← → ↺ 🏠 https://profs.info.uaic.ro/~busaco/cgi/bash/variabile.cgi?date_de_intrare

```
GATEWAY_INTERFACE=CGI/1.1
GROUPS=()
HOSTNAME=profs.info.uaic.ro
HOSTTYPE=x86_64
HTTPS=on
HTTP_ACCEPT='text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8'
HTTP_ACCEPT_ENCODING='gzip, deflate, br'
HTTP_ACCEPT_LANGUAGE='en-GB,en;q=0.5'
HTTP_CONNECTION=keep-alive
HTTP_DNT=1
HTTP_HOST=profs.info.uaic.ro
HTTP_SEC_GPC=1
HTTP_UPGRADE_INSECURE_REQUESTS=1
HTTP_USER_AGENT='Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:85.0) Gecko/20100101 Firefox
IFS=$' \t\n'
MACHTYPE=x86_64-redhat-linux-gnu
OPTERR=1
OPTIND=1
OSTYPE=linux-gnu
PATH=/usr/local/bin:/usr/bin:/bin
PIPESTATUS=( [0]="0")
PPID=25485
PS4='+ '
PWD=/home/thor/profs/busaco/html/cgi/bash
QUERY_STRING=date_de_intrare
REMOTE_ADDR=109.102.38.36
REMOTE_PORT=50606
REQUEST_METHOD=GET
REQUEST_SCHEME=https
REQUEST_URI='/~busaco/cgi/bash/variabile.cgi?date_de_intrare'
SCRIPT_FILENAME=/thor/profs/busaco/public_html/cgi/bash/variabile.cgi
SCRIPT_NAME=/~busaco/cgi/bash/variabile.cgi
SERVER_ADDR=85.122.23.20
SERVER_ADMIN=pimi@info.uaic.ro
SERVER_NAME=profs.info.uaic.ro
SERVER_PORT=443
SERVER_PROTOCOL=HTTP/1.1
SERVER_SIGNATURE=
SERVER_SOFTWARE='Apache/2.4.35 (IUS) OpenSSL/1.0.2k-fips mod_fcgid/2.3.9 PHP/7.2.34'
```

rezultatul obținut
de clientul Web

← → ↺ 🏠 https://profs.info.uaic.ro/~busaco/cgi/bash/variabile.cgi?date_de_intrare

```
GATEWAY_INTERFACE=CGI/1.1
GROUPS=()
HOSTNAME=profs.info.uaic.ro
HOSTTYPE=x86_64
HTTPS=on
HTTP_ACCEPT='text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8'
HTTP_ACCEPT_ENCODING='gzip, deflate, br'
HTTP_ACCEPT_LANGUAGE='en-GB,en;q=0.5'
HTTP_CONNECTION=keep-alive
HTTP_DNT=1
HTTP_HOST=profs.info.uaic.ro
HTTP_SEC_GPC=1
HTTP_UPGRADE_INSECURE_REQUESTS=1
HTTP_USER_AGENT='Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:85.0) Gecko/20100101 Firefox
IFS=$' \t\n'
MACHTYPE=x86_64-redhat-linux-gnu
OPTERR=1
OPTIND=1
OSTYPE=linux-gnu
PATH=/usr/local/bin:/usr/bin:/bin
PIPESTATUS=( [0]="0")
PPID=25485
PS4='+ '
PWD=/home/thor/profs/busaco/html/cgi/bash
QUERY_STRING=date_de_intrare
REMOTE_ADDR=109.102.38.36
REMOTE_PORT=50606
REQUEST_METHOD=GET
REQUEST_SCHEME=https
REQUEST_URI='/~busaco/cgi/bash/variabile.cgi?date_de_intrare'
SCRIPT_FILENAME=/home/thor/profs/busaco/public_html/cgi/bash/variabile.cgi
SCRIPT_NAME=/~busaco/cgi/bash/variabile.cgi
SERVER_ADDR=85.122.23.20
SERVER_ADMIN=pimi@info.uaic.ro
SERVER_NAME=profs.info.uaic.ro
SERVER_PORT=443
SERVER_PROTOCOL=HTTP/1.1
SERVER_SIGNATURE=
SERVER_SOFTWARE='Apache/2.4.35 (IUS) OpenSSL/1.0.2k-fips mod_fcgid/2.3.9 PHP/7.2.34'
```

rezultatul obținut
de clientul Web

```

/* hello.c

(compilare cu gcc hello.c -o hello.cgi) */
#include <stdio.h>

int main() {
    int mesaje;

    printf ("Content-type: text/html\n\n");

    for (mesaje = 0; mesaje < 10; mesaje++) {
        printf("<p>Hello, world!</p>");
    }

    return 0;
}

```

programe CGI scrise în C,
bash, Python generând același
conținut marcat în HTML

```

#!/bin/bash
# hello.sh.cgi

echo "Content-type: text/html"
echo

MESAJE=0
while [ $MESAJE -lt 10 ]
do
    echo "<p>Hello, world!</p>"
    let MESAJE=MESAJE+1
done

```

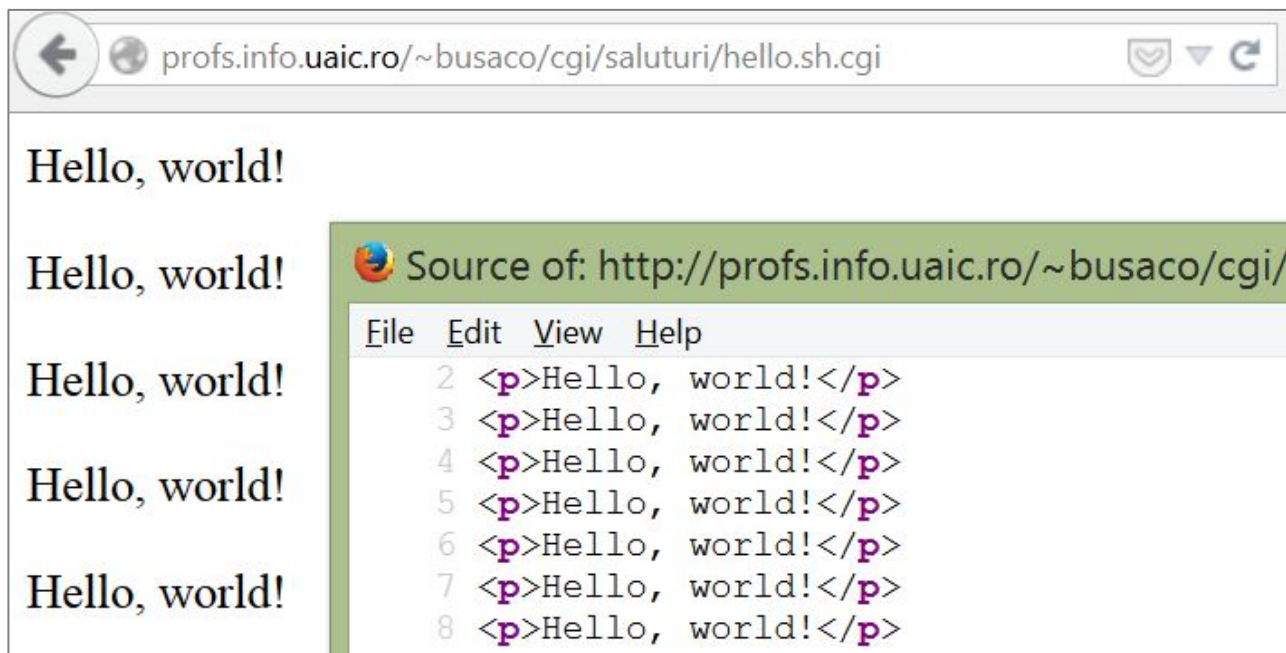
```

#!/usr/bin/python
# hello.py.cgi

print "Content-type: text/html\n"

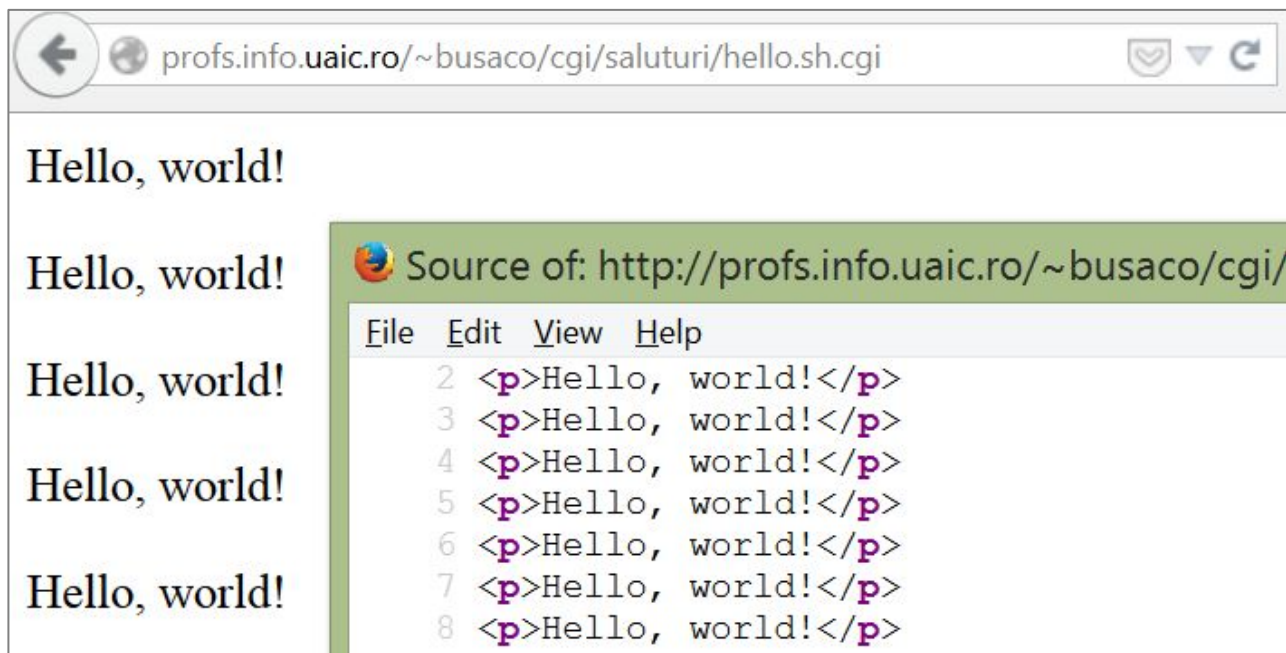
for mesaje in range (0, 10):
    print "<p>Hello, world!</p>"

```

clientul – *i.e.* navigatorul Web – primește ca răspuns reprezentarea – aici, pagina folosind marcaje HTML – generată de programul CGI invocat de serverul Web

această reprezentare a datelor este procesată și, eventual, afișată într-o (zonă dintr-o) fereastră a *browser*-ului



experimentând alte tipuri MIME, *browser*-ul prelucrează datele primite și redă următoarele:

```
<p>Hello, world!</p>
<p>Hello, world!</p>
<p>Hello, world!</p>
<p>Hello, world!</p>
<p>Hello, world!</p>
```

Content-type: **text/plain**

XML Parsing Error: junk after document element
Location: http://profs.info.uaic.ro/~busaco/cgi/hello
Line Number 2, Column 1:

```
<p>Hello, world!</p>
^
```

Content-type: **text/xml**

cgi: invocare

```
<form action="https://profs.info.uaic.ro/~.../get-max.cgi"
      method="GET">
  <p>Enter two numbers:
  <input type="text" name="no1" />
  <input type="text" name="no2" />
  </p>
  <input type="submit" value="Compute maximum" />
</form>
```

invocare dintr-un formular Web interactiv
în acest caz, folosind metoda **GET**

https://profs.info.uaic.ro/~busaco/cgi/max.html

Enter two numbers:

Compute maximum

URL
special
cazul GET

profs.info.uaic.ro/~busaco/cgi/get-max.cgi?no1=7&no2=4

Maximum of two numbers

$\max(7, 4) = 7$

cgi: invocare

Pentru fiecare câmp al formularului, se generează o pereche **nume_câmp=valoare** delimitată de **&** și adăugată URL-ului programului CGI de invocat pe server

<https://profs.info.uaic.ro/~busaco/cgi/get-max.cgi?no1=7&no2=4>

cgi: invocare

Exemple concrete:

<https://usabilitygeek.com/?s=design+web>

https://www.youtube.com/watch?v=_0GWG0yCu-w#t=110

<https://twitter.com/search?q=web%20development&src=typd>

<https://developer.mozilla.org/search?q=ajax&topic=api>

revezi primul curs

acest URL este codificat – *URL encoding*

prin convenție, caracterul spațiu inclus în valoarea unui câmp de intrare al unui formular Web este substituit de „+”

cgi: invocare

Serverul va invoca *script*-ul CGI
pasându-i datele preluate de la client
via *variabile de mediu* (*environment*) – uzual, cazul **GET**
sau
la *intrarea standard* (*stdin*) – dacă se recurge la **POST**

cgi: invocare

Procesarea datelor când s-a recurs la metoda **GET**

date disponibile în variabila de mediu **QUERY_STRING**

cgi: invocare

Procesarea datelor când s-a folosit metoda **POST**

datele vor fi preluate de la *stdin*, lungimea în octeți a acestora fiind specificată de variabila **CONTENT_LENGTH**

cgi: invocare

Procesarea datelor – **GET** și/sau **POST**

folosind servere de aplicații ori *framework*-uri, informațiile sunt încapsulate în structuri/tipuri de date specifice

ASP.NET (C# *et al.*) – clasa **HttpRequest**

Node.js (JavaScript) – **http.ClientRequest**

PHP – tablouri asociative: **\$_GET[] \$_POST[] \$_REQUEST[]**

Play (Java, Scala) – **play.api.mvc.Request**

Python – clasa **cgi.FieldStorage**

GET vs. POST

Metoda **GET** se folosește pentru obținerea de reprezentări ale resurselor cerute serverului Web

e.g., documente HTML, imagini JPEG, PNG sau WebP, ilustrații SVG (XML), resurse PDF, arhive în format ZIP etc.

starea serverului nu trebuie să se modifice

GET vs. POST

Metoda **GET** se folosește pentru obținerea de reprezentări ale resurselor cerute serverului Web

preluând datele via GET, utilizatorul poate stabili un *bookmark* pentru acces ulterior la o resursă Web (utilizând URL-ul reprezentării generate)

e.g., <https://duckduckgo.com/?q=web+dev&iar=videos>

GET vs. POST

Metoda **POST** se utilizează atunci când datele transmise serverului au dimensiuni mari (*e.g.*, conținut de fișiere ce a fost transferat prin *upload*) sau sunt „delicate” – exemplu tipic: parole

GET vs. POST

Metoda **POST** se utilizează atunci când datele transmise serverului au dimensiuni mari (*e.g.*, conținut de fișiere ce a fost transferat prin *upload*) sau sunt „delicate” – exemplu tipic: parole

de asemenea, când invocarea programului poate conduce la modificări ale stării pe server: adăugarea unei înregistrări, alterarea unui fișier,...

cgi: suport

Serverul Web trebuie să ofere suport pentru invocarea de *script*-uri CGI

exemplificare:

Apache HTTP Server – se utilizează modulul **mod_cgi**

cgi: ssi

Script-urile CGI pot fi invocate direct dintr-un document HTML via **SSI** (*Server Side Includes*)

www.ssi-developer.net/ssi/

Apache: httpd.apache.org/docs/trunk/howto/ssi.html

NGINX: nginx.org/en/docs/http/nginx_http_ssi_module.html

cgi: fastcgi

FastCGI

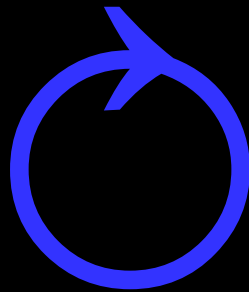
alternativă la CGI axată asupra performanței

implementări:

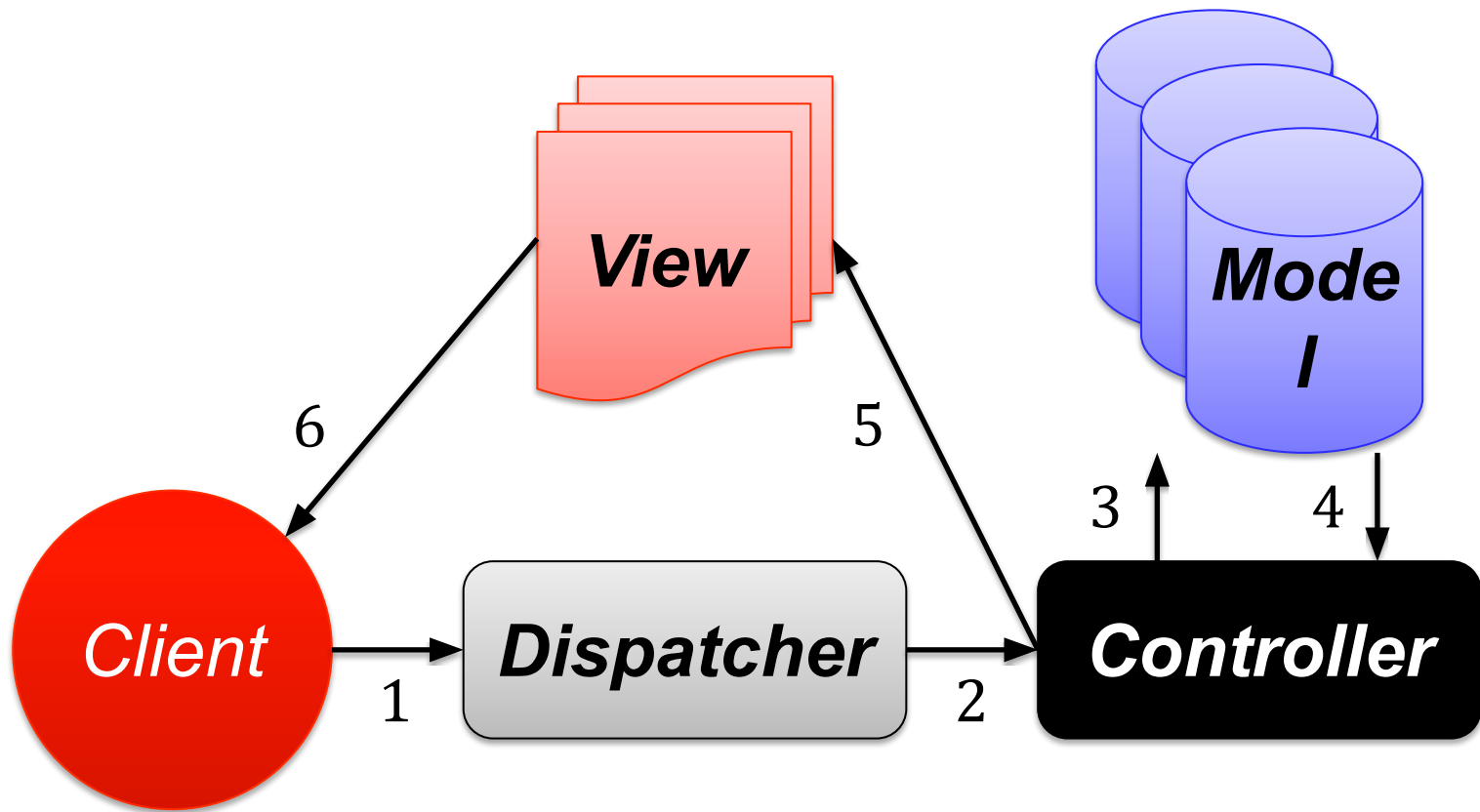
Apache HTTP Server – httpd.apache.org/mod_fcgid/

NGINX – nginx.org/en/docs/http/nginx_http_fastcgi_module.html

rezumat



HTTP (*HyperText Transfer Protocol*)
script-uri CGI (*Common Gateway Interface*)



episodul viitor: **programare Web**
cookie-uri, sesiuni Web,
servere de aplicații Web, arhitectura aplicațiilor Web