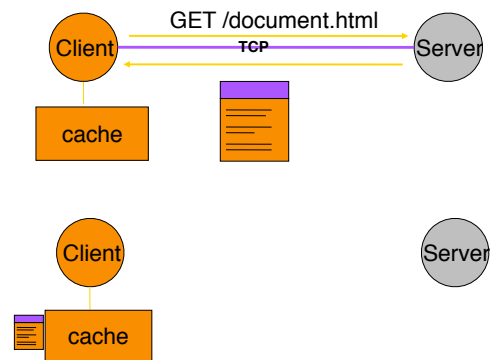


Πρωτόκολλο HTTP

Ανατομία αλληλεπίδρασης HTTP



M. Λικαμάκος, EIT1425

2

Hypertext Transport Protocol (HTTP)

- the set of commands understood by a web server and sent from a browser
- some HTTP commands (your browser sends these internally):
 - GET** filename : download
 - POST** filename : send a web form response
 - PUT** filename : upload
- simulating a browser with a terminal window:

```
$ telnet www.cs.washington.edu 80
Trying 128.208.3.88...
Connected to 128.208.3.88 (128.208.3.88).
Escape character is '^]'.
GET /index.html
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 ...">
<html>
...

```

M. Λικαμάκος, EIT1425

3

Τι είναι το HTTP;

- HyperText Transfer Protocol:
 - Πρωτόκολλο αιτήματος-απάντησης (request-reply).
 - Πρωτόκολλο εφαρμογών (όπως τα ftp και telnet).
 - Επιλήσιμον (stateless) - σε αντίθεση με τα ftp και telnet.
 - Εντολές:
 - GET** filename : download
 - POST** filename : send a web form response
 - PUT** filename : upload
- Πως χρησιμοποιείται το HTTP;
 - Μεταφέρει εντολές από τον πλοηγό στον εξυπηρετητή
 - Μεταφέρει πληροφορίες σε πολλαπλούς μορφότευπους, γλώσσες και συμβολοσειρές.
 - Ο μορφότευπος των οντοτήτων που μεταφέρει το HTTP δεν αφορά το πρωτόκολλο. Η περιγραφή του μορφότευπου βασίζεται στο πρότυπο MIME.

M. Λικαμάκος, EIT1425

4

Internet media ("MIME") types

- sometimes when including resources in a page (style sheet, icon, multimedia object), we specify their type of data

MIME type	file extension
text/html	.html
text/plain	.txt
image/gif	.gif
image/jpeg	.jpg
video/quicktime	.mov
application/octet-stream	.exe

- Lists of MIME types: [by type](#), [by extension](#)

M. Λικαμάκος, EIT1425

5

Βασικές ιδιότητες του HTTP

- Βασίζεται στον μηχανισμό διευθυνσιοδότησης πληροφοριακών πόρων Universal Resource Identifier
- Ανταλλαγή αιτήματος-απάντησης: τα αιτήματα αποστέλλονται από συστήματα πελατών και προκαλούν απαντήσεις από τους εξυπηρετητές. Δοσοληψίες HTTP δεν ενεργοποιούνται από τους εξυπηρετητές.
- Επιλήσιμον (statelessness):
 - Πελάτες και εξυπηρετητές δεν διατηρούν στοιχεία για την αλληλεπίδραση τους μεταξύ διαδοχικών ανταλλαγών αιτημάτων-απαντήσεων (λόγοι επίδοσης).
 - Στην πράξη διατηρούν πληροφορίες κατάστασης, αλλά αυτό δεν αφορά το πρωτόκολλο.
- Μεταδεδομένα πληροφοριακών πόρων:
 - Πληροφορίες σχετικές με τους πόρους του WWW συμπεριλαμβάνονται στα μηνύματα του HTTP.
 - Π.χ. το μέγεθος ενός αρχείου, ο μορφότευπος του περιεχομένου του, η ημερομηνία τελευταίας ενημέρωσής του κλπ.

M. Λικαμάκος, EIT1425

6

HTTP: Ιστορική αναδρομή

3/1990	Το CERN προτείνει το Web
1/1992	Προδιαγραφή HTTP/0.9
12/1992	Πρόταση για πρόσθεση του MIME στο HTTP
3/1993	Πρώτο προσχέδιο HTTP/1.0
6/1993	HTML (προδιαγραφή 1.0)
10/1993	Προδιαγραφή URL
11/1993	Δεύτερο προσχέδιο HTTP/1.0
3/1994	Υιοθέτηση URI στο WWW
5/1996	HTTP/1.0 RFC 1945
1/1997	Εισήγηση HTTP/1.1, RFC 2068
6/1999	Προσχέδιο προδιαγραφής HTTP/1.1, RFC 2616
2001	HTTP/1.1 επίσημη προδιαγραφή

7

HTTP/2

- HTTP/2 is the first new version of HTTP since HTTP 1.1
- Working Group presented HTTP/2 to IETF's Internet Engineering Steering Group for consideration as a Proposed Standard in December 2014.
 - The standardization effort was supported by Chrome, Opera, Firefox, Internet Explorer 11, Safari, Amazon Silk and Edge browsers.
- IESG approved it to publish as Proposed Standard: Feb. 2015
- HTTP/2 specification published as RFC 7540: May 2015.
- Most major browsers added HTTP/2 support by end of 2015.

Μ. Λικατίκος, ΕΠΛ425

8

Ορολογία HTTP

- HTTP Μήνυματα (messages)
- Οντότητες (entities)
- Πόροι (resources)
- Διεκπεραιωτής χρήστη (user agent)

Μ. Λικατίκος, ΕΠΛ425

9

Μηνύματα HTTP

- Σειρά οκταδικών ψηφίων που στέλνονται μέσω TCP/IP.
- Διακρίνονται σε αιτήματα (requests) και απαντήσεις (replies).
- Μηνύματα αιτημάτων αρχίζουν με γραμμή αιτήματος.
- Μηνύματα απαντήσεων αρχίζουν με γραμμή κατάστασης.
- Κάθε μήνυμα στο HTTP περιλαμβάνει μηδέν ή περισσότερες επικεφαλίδες, οι οποίες διαχωρίζονται από το προαιρετικό σώμα του μηνύματος με CR ή LF.
- Συντακτική δομή αιτημάτων:
Request-Line
General/Request/Entity Header(s)
CRLF
Optional Message Body

Μ. Λικατίκος, ΕΠΛ425

10

Αίτημα πελάτη HTTP (παράδειγμα)

```
GET /motd HTTP/1.0
Date: Wed, 22 Mar 2000 08:09:01 GMT
Pragma: No-cache
From: mdd@ucy.ac.cy
User-Agent: Mozilla/4.03
CRLF
```

- Γραμμή αιτήματος
- Γενικές επικεφαλίδες: μπορούν να υπάρξουν σε μηνύματα αιτήματος και απάντησης.
- Επικεφαλίδες αιτήματος

Μ. Λικατίκος, ΕΠΛ425

11

Αίτημα πελάτη HTTP (παράδειγμα)

```
POST /motd HTTP/1.0
Date: Wed, 22 Mar 2000 08:09:01 GMT
Pragma: No-cache
From: mdd@ucy.ac.cy
User-Agent: Mozilla/4.03
Content-Length: 23
Allow: GET, HEAD, PUT
CRLF
Welcome to Comer's VAX
```

- Επικεφαλίδες οντότητας
 - Content-Length: καθορίζει το μήκος της οντότητας που μεταφέρει το μήνυμα.
 - Allow: καθορίζει τις εντολές που μπορούν να εκτελεστούν στον πόρο motd

Μ. Λικατίκος, ΕΠΛ425

12

Απαντήσεις HTTP

- Συντακτική δομή απαντήσεων HTTP:
Status-Line
General/Request/Entity Header(s)
CRLF
Optional Message Body
- Π.χ.:
HTTP/1.0 200 OK
Date: Tue, 4 Jul, 1996, 19:57:05 GMT
Server: Netscape-Enterprise/3.5.1
Content-length: 23
CRLF
Welcome to Comer's VAX

Μ. Λαυαράκης, ΕΠΙΛ425

13

Οντότητες HTTP

- Καθορίζονται ως οι πόροι που περικλείονται σε μηνύματα αιτημάτων ή αποκρίσεων HTTP.
- Αποτελούνται από:
 - Επικεφαλίδες οντότητας
 - Προαιρετικό περιεχόμενο οντότητας (entity body)
- Αποτελούν το πιο σημαντικό τμήμα ενός μηνύματος HTTP, αφού μεταφέρουν το περιεχόμενο του μηνύματος.

Μ. Λαυαράκης, ΕΠΙΛ425

14

HTTP Traffic Capture

- LiveHTTPHeaders - plugin του FireFox
- Εγκαταστήστε το και συλλέξτε την διακίνηση HTTP του πλοηγού σας.

```
GET / HTTP/1.1
Host: www.google.com.cy
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.7; rv:10.0) Gecko/20100101 Firefox/10.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip, deflate
DNT: 1
Connection: keep-alive
Cookie: PREF=ID=f3964f3123331b4-U=01e366a1815268b:FF=0:TM=1328257040:LM=1328257040:S=26NwoXRCkgHl-Fh8;NID=56=u01c0ZCM0x8kXAFMfIOV2yIA6ubhEnu7H
HTTP/1.1 200 OK
Date: Sun, 05 Feb 2012 21:53:15 GMT
Expires: -1
Cache-Control: private, max-age=0
Content-Type: text/html; charset=UTF-8
Content-Encoding: gzip
Server: gws
Content-Length: 15584
X-XSS-Protection: 1; mode=block
X-Frame-Options: SAMEORIGIN
```

Μ. Λαυαράκης, ΕΠΙΛ425

15

Διεκπεραιωτής Χρήστη (User Agent)

- Αποτελεί το σύστημα πελάτη (client) το οποίο αρχικοποιεί ένα αίτημα προς κάποιον εξυπηρετητή.
- Μπορεί να είναι:
 - Φυλλομετρητής
 - Αράχνη (crawler)
 - Άλλο σύστημα (προσωπικοί διεκπεραιωτές, απομνημονευτές Ιστού κλπ)
- Παραδείγματα UA:
 - Mozilla/3.01 (X11; SunOS 5.5 sun4m)
 - Mozillas/2.0 (MSIE 2.1; AOL 3.0; Mac)

Μ. Λαυαράκης, ΕΠΙΛ425

16

Μέθοδοι HTTP/1.0

- Οι μέθοδοι αιτημάτων του HTTP προσδιορίζουν τη δράση που πρέπει να εκτελεσθεί στον πληροφοριακό πόρο ο οποίος καθορίζεται από το URI της γραμμής αιτήματος.
- Το HTTP/1.0 καθορίζει μόνο τρεις μεθόδους:
 - **GET**
 - **HEAD**
 - **POST**
- Ορισμένες εκδόσεις πελατών και εξυπηρετητών υλοποιούν και άλλες μεθόδους, όπως **PUT**, **DELETE**, **LINK** και **UNLINK**.
- Ο εξυπηρετητής που δέχεται το αίτημα, εκτελεί την περιγραφόμενη δράση.

Μ. Λαυαράκης, ΕΠΙΛ425

17

Μέθοδοι HTTP/1.0: GET

GET

- Αναζήτησε και επέστρεψε πόρο που καθορίζεται από το URI
- Το URI μπορεί να περιλαμβάνει και ορίσματα (arguments) για την υποβολή δεδομένων εισροής σε πρόγραμμα πύλης (gateway):
[GET /foo/bar.html?x=bar&bam=baz](#)
- Ένα αίτημα GET μπορεί να συνοδεύεται από **τροποποιητική επικεφαλίδα**, η οποία να το μετατρέπει σε conditional GET. Π.χ.:
[GET /foo/bar.html HTTP/1.0](#)
[If-Modified-Since: Sun, 12 Nov 2000 11:12:23 GMT](#)

Μ. Λαυαράκης, ΕΠΙΛ425

18

Μέθοδοι HTTP/1.0: HEAD

- Παρόμοια με την GET, αλλά ζητά την επιστροφή μόνο μεταδεδομένων για κάποιον πόρο. Ζητάει από έναν εξυπηρετητή να στείλει τις επικεφαλίδες απάντησης για κάποιο αρχείο στο διαθέτη, αλλά όχι και τα περιεχόμενα του αρχείου.
- Χρησιμοποιείται από προγράμματα τα οποία εκτελούν αυτόματες αναζητήσεις για κείμενα στον Ιστό.
- Η μέθοδος αυτή είναι ένας γρήγορος τρόπος για να βλέπει κανείς αν ένα αρχείο ή πρόγραμμα gateway είναι πράγματι παρόν στον Ιστό, για την εύρεση γενικών πληροφοριών σχετικά με το αρχείο (τον MIME τύπο του, την ημερομηνία αλλαγής του κλπ).

Μ. Λακτακός, ΕΠΛ425

19

Παράδειγμα αιτήματος HEAD

HEAD /foo.html HTTP/1.0

HTTP/1.0 200 OK

Content-Length: 3219

Last-Modified: Sun, 12 Nov 2000 11:12:23 GMT

Content-Type: text/html

Μ. Λακτακός, ΕΠΛ425

20

Μέθοδοι HTTP/1.0: POST

- Σε αντίθεση με τις GET και HEAD, οι οποίες ζητούν την ανάκτηση πληροφοριών, η POST χρησιμοποιείται για την ενημέρωση ενός πόρου ή την παροχή εισροής σε κάποιο πρόγραμμα επεξεργασίας στον εξυπηρετητή.
- Το «σώμα» μηνυμάτων POST περιλαμβάνει τα δεδομένα που στέλνονται από τον πελάτη στον εξυπηρετητή.
- Δεν είναι ούτε ασφαλής μέθοδος, αφού μπορεί να αλλάξει τα περιεχόμενα ενός πληροφοριακού πόρου, ούτε μηδενοδύναμη (idempotent).

POST /foo/bar.cfm HTTP/1.0
Content-Length: 143
<entity body>

-- υποχρεωτική χρήση εδώ

POST /search.cgi HTTP/1.0
Content-Length: 34
CRLF
Query iktinos
Db greek-architecture

Μ. Λακτακός, ΕΠΛ425

21

Μέθοδοι HTTP/1.1

- **OPTIONS:** a request for information about the communication options available on the request/response chain identified by the Request-URI. This method allows the client to determine the options and/or requirements associated with a resource, or the capabilities of a server, without implying a resource action or initiating a resource retrieval
- **GET:** retrieve whatever information (in the form of an entity) is identified by the Request-URI. If the Request-URI refers to a data-producing process, it is the produced data which shall be returned as the entity in the response and not the source text of the process, unless that text happens to be the output of the process.
- **HEAD:** identical to GET except that the server MUST NOT return a message-body in the response
- **POST:** used to request that the origin server accept the entity enclosed in the request as a new subordinate of the resource identified by the Request-URI in the Request-Line.

Μ. Λακτακός, ΕΠΛ425

22

Μέθοδοι HTTP/1.1

- **PUT:** requests that the enclosed entity be stored under the supplied Request-URI. If the Request-URI refers to an already existing resource, the enclosed entity SHOULD be considered as a modified version of the one residing on the origin server
- **DELETE:** requests that the origin server delete the resource identified by the Request-URI. This method MAY be overridden by human intervention (or other means) on the origin server.
- **TRACE:** used to invoke a remote, application-layer loop- back of the request message. The final recipient of the request SHOULD reflect the message received back to the client as the entity-body of a 200 (OK) response.
- **CONNECT:** reserved for use with a proxy that can dynamically switch to being a tunnel

Μ. Λακτακός, ΕΠΛ425

23

Επικεφαλίδες HTTP

- Αποτελούνται από χαρακτήρες ASCII, περιλαμβάνοντας κάποιο όνομα και συνήθως μια τιμή, που ανατίθεται στο όνομα.
- Παίζουν σημαντικό ρόλο στο HTTP, καθώς αποτελούν το βασικό μηχανισμό για τροποποίηση της διαχείρισης των αιτημάτων του. Επιτρέπεται η εισαγωγή νέων επικεφαλίδων, αυθαίρετου μεγέθους.
- Χρήσεις:
 - Παροχή μεταδεδομένων για κάποιο πόρο.
 - Παραμετροποίηση αιτημάτων ή απαντήσεων.
 - Καθορισμός αν μια απάντηση μπορεί να απομνημονευθεί, ή πως μπορεί να αποκωδικοποιηθεί.
 - Επέκταση του πρωτοκόλλου

Μ. Λακτακός, ΕΠΛ425

24

Χρήση επικεφαλίδων στο HTTP

- Ένα μήνυμα HTTP μπορεί να έχει αυθαίρετο αριθμό επικεφαλίδων – καθεμιά από τις οποίες ακολουθείται από CR και LF χαρακτήρες για διαχωρισμό της με τις επόμενες.
- Οι περισσότερες επικεφαλίδες είναι προαιρετικής χρήσης. Δομοστοιχεία του Ιστού μπορούν να αγνοούν τις προαιρετικές επικεφαλίδες.
- Ορισμένες επικεφαλίδες είναι υποχρεωτικές για κάποια μηνύματα αιτήματος-απόκρισης.
- Συντακτικό επικεφαλίδων:
<όνομα>: <τιμή> (<τιμή>)*
Π.χ. Date: Thu, 23 Dec 1999 08:12:31 GMT
Accept-Language: de-CH, en-US

Μ. Λικατίκος, ΕΠΛ425

25

Ιεραρχία επικεφαλίδων HTTP

- Γενικές: χρήση σε αιτήματα και απαντήσεις
- Αιτημάτων:
 - χρησιμοποιούνται σε αιτήματα για εκδήλωση προτίμησης στη μορφή της απάντησης, για παροχή επιπρόσθετης πληροφορίας με το αίτημα ή για καθορισμό κάποιου περιορισμού στον εξυπηρετητή που θα εκτελέσει το αίτημα.
- Απαντήσεων:
 - χρήση σε μηνύματα αποκρίσεων για παροχή περαιτέρω πληροφοριών ή μεταπληροφοριών προς τον πελάτη, ή για έγερση αιτήματος για περισσότερες πληροφορίες από τον πελάτη.
- Οντότητας: παρέχουν πληροφορίες για την οντότητα την οποία αφορά το μήνυμα.
- Η σειρά με την οποία χρησιμοποιούνται οι επικεφαλίδες σε κάποιο μήνυμα δεν είναι σημαντική – συνήθως μπαίνουν πρώτα οι γενικές επικεφαλίδες, ακολουθούν οι επικεφαλίδες αιτημάτων ή απαντήσεων και στο τέλος έπονται οι επικεφαλίδες οντοτήτων.

Μ. Λικατίκος, ΕΠΛ425

26

Γενικές Επικεφαλίδες

- Αφορούν στο μήνυμα HTTP και όχι στο περιεχόμενό του (δηλ. την οντότητα που μεταφέρει).
- Το HTTP/1.0 καθορίζει δύο μόνο γενικές επικεφαλίδες:
 - **Date**: προσδιορίζει την ημερομηνία δημιουργίας του μηνύματος.
 - **Pragma**: επιτρέπει την αποστολή οδηγιών προς τον παραλήπτη του μηνύματος. Οι οδηγίες αυτές αποτελούν ένα τρόπο για να ζητείται από δομοστοιχεία του Ιστού να συμπεριφέρονται με συγκεκριμένο τρόπο, ενώ διαχειρίζονται ένα αίτημα ή μια απόκριση.
- Το πρωτόκολλο δεν επιβάλλει την υπακοή στις οδηγίες αυτές, αλλά απλώς απαιτεί την μετάδοσή τους.
- Το HTTP/1.0 καθορίζει μόνο ένα pragma: no-cache

Μ. Λικατίκος, ΕΠΛ425

27

Επικεφαλίδες Αιτημάτων

- **Authorization**: Χρησιμοποιείται από τον διεκπεραιωτή-πελάτη για την υποβολή των διαπιστευτηρίων (credentials) που χρησιμοποιούνται για την πρόσβαση σε κάποιο πόρο. Π.χ.:
Authorization: Basic YXZpYzRpsS29IDizM1NA==
 - Η λέξη Basic αναφέρεται σε βασικό σχήμα ταυτοποίησης με ταυτότητα χρήστη (userid) και κωδικό (password), κωδικοποιημένα σε Base64 μορφή.
- **From**: επιτρέπει στον χρήστη να παρέχει το ηλ/τ του, για την ταυτοποίησή του (κυρίως για χρήστες-αράχνες).

Μ. Λικατίκος, ΕΠΛ425

28

Επικεφαλίδες Αιτημάτων (συνέχεια)

- **If-Modified-Since**: παράδειγμα επικεφαλίδας *συνθήκης*, η οποία επηρεάζει τον τρόπο με τον οποίο θα τύχει επεξεργασίας ένα αίτημα.
 - Π.χ.:
GET /foo.html HTTP/1.0
If-Modified-Since: Sun, 21 May 2000 07:00:25 GMT
- **Referer**: επιτρέπει στον πελάτη να συμπεριλάβει σε ένα αίτημα το URI του πόρου από τον οποίο ελήφθη το αιτούμενο URI του συγκεκριμένου μηνύματος.
 - Π.χ.:
GET /Disasters/worldglance.html HTTP/1.0
Referer: http://www.cnn.com
 - Συνέπειες: κίνδυνοι για την αποκάλυψη ιδιωτικών δεδομένων.

Μ. Λικατίκος, ΕΠΛ425

29

Η χρήση της τοπικής cache

- Στην περίπτωση που το ζητούμενο αρχείο δεν έχει τροποποιηθεί μετά την ημερομηνία που καθορίζει ο πελάτης, ο εξυπηρετητής επιστρέφει το ακόλουθο:
HTTP/1.0 304 Not Modified
Date: Thu, 03 Aug 1996 16:04:09 GMT
Server: NCSA/1..3
MIME-version: 1.0
- Τότε ο πλοηγός-πελάτης μπορεί να χρησιμοποιήσει την “κρυμμένη” έκδοση του αρχείου που αναζήτησε.

Μ. Λικατίκος, ΕΠΛ425

30

Επικεφαλίδες Αιτημάτων (συνέχεια)

- **User-Agent:** παρέχει πληροφορίες για την έκδοση του πλοηγού που χρησιμοποιεί το σύστημα πελάτη, σε τι λειτουργικό σύστημα τρέχει κοκ.
 - Π.χ.:
User-Agent: Mozilla/4.03 (Macintosh; I; 68K, Nav)

Μ. Λακατάκος, ΕΠΛ425

31

Επικεφαλίδες Απαντήσεων

- **Location:** για παραπομπή ενός αιτήματος σε άλλο URI.
- **Server:** καθορίζει τον τύπο του χρησιμοποιούμενου εξυπηρετητή.
- **WWW-Authenticate:** χρησιμοποιείται για την έγερση από τον εξυπηρετητή απαίτησης προς τον πελάτη για αποστολή διαπιστευτηρίων με σκοπό την παροχή πρόσβασης σε ελεγχόμενο πόρο. Χρησιμοποιείται μαζί με απάντηση 404 Unauthorized. Π.χ.:
 - **WWW-Authenticate: Basic realm="ChaseChem"**

Μ. Λακατάκος, ΕΠΛ425

32

Επικεφαλίδες Απαντήσεων (συνέχεια)

- Χρησιμοποιούνται, χωρίς να ορίζονται στο πρωτόκολλο:
 - **Retry-After: (date) or (seconds)**
 - **[Set-]Cookie: Part_Number="Rocket_Launcher_0001"; Version="1"; Path="/acme"**
 - **Title: (title)**

Μ. Λακατάκος, ΕΠΛ425

33

Επικεφαλίδες Οντοτήτων

- **Allow:** χρησιμοποιείται για να προσδιορίσει τις έγκυρες μεθόδους που μπορούν να εφαρμοσθούν σε κάποιον πόρο.
 - Π.χ. αν κάποιος πηγαίος εξυπηρετητής λάβει ένα αίτημα για μη εξουσιοδοτημένη μέθοδο, μπορεί να απαντήσει ενημερώνοντας τον πελάτη για τις μεθόδους που είναι αποδεκτές για το συγκεκριμένο αίτημα.
 - Σε μήνυμα αιτήματος PUT, με το **Allow** ο πελάτης μπορεί να καθορίσει τι μεθόδους επιτρέπει να εκτελούνται στον πόρο που δημιουργεί:
PUT /foo.html HTTP/1.0
Allow: HEAD, GET, PUT
- **Content-Type:** προσδιορίζει τον μορφότυπο περιεχόμενου (media type) της οντότητας. Π.χ.:
POST /chat/chatroom.cgi HTTP/1.0
User-Agent: Mozilla/3.0C
Content-Type: application/x-www-form-urlencoded

Μ. Λακατάκος, ΕΠΛ425

34

Επικεφαλίδες Οντοτήτων (συνέχεια)

- **Content-Encoding:** προσδιορίζει πώς έχει κωδικοποιηθεί το περιεχόμενο της οντότητας και πώς θα πρέπει να αποκωδικοποιηθεί για να πάρουμε την οντότητα στο μορφότυπο που καθορίζεται από το **Content-Type**. Π.χ.:
Content-Encoding: x-gzip
- **Content-Length:**
 - Προσδιορίζει το μέγεθος της μεταφερόμενης οντότητας σε χαρακτήρες.
 - Χρησιμοποιείται για να επικυρώσουμε αν μια οντότητα έχει φθάσει στην ολότητά της, και σαν επικύρωση ότι μια απομνημονευμένη οντότητα δεν διαφέρει από την οντότητα στην τρέχουσα μορφή της.
 - Αποφεύγεται η χρήση της με την μεταφορά δυναμικού περιεχομένου, για λόγους επίδοσης.

Μ. Λακατάκος, ΕΠΛ425

35

Επικεφαλίδες Οντοτήτων (συνέχεια)

- **Expires:** χρησιμοποιείται για τον προσδιορισμό της χρονικής στιγμής μετά από την οποία ένας πόρος μπορεί να θεωρηθεί ως λήξας. Π.χ.:
HTTP/1.0 200 OK
Server: Microsoft-IIS/4.0
Date: Mon, 04 Dec 2000 18:16:45 GMT
Expires: Tue, 05 Dec 2000 16:00:00 GMT
- **Last-Modified:** καθορίζει τη χρονική στιγμή της τελευταίας ενημέρωσης του μεταφερόμενου πόρου.

Μ. Λακατάκος, ΕΠΛ425

36

Κλάσεις Απαντήσεων HTTP/1.0

Οικογένεια κωδικών με πέντε «τύπους»:

- **1xx**: informational
- **2xx**: successful, e.g. 200 OK
- **3xx**: redirection
 - 301 Moved Permanently
 - 304 Not Modified
- **4xx**: Client Error
 - 400 Bad Request
 - 401 Unauthorized
 - 403 Forbidden
 - 404 Not Found
- **5xx**: Server Error
 - 501 Not Implemented
 - 503 Service Unavailable

M. Λικατίκος, EIT1425

37

HTTP error codes

- when something goes wrong, the web server returns a special "error code" number to the browser, possibly followed by an HTML document
- common error codes:

Number	Meaning
200	OK
301-303	page has moved (permanently or
403	you are forbidden to access this page
404	page not found
500	internal server error
complete list	

M. Λικατίκος, EIT1425

38

Η ιστορία ενός κλικ

- Υποθέσατε:
 - Είστε στο: www.yahoo.com/index.html
 - Επιλέγετε το: www.grippy.org/mattmarg/
- Ο πλοηγός χρησιμοποιεί το DNS => IP addr για www.grippy.org
- Ανοίγει σύνδεση TCP με την IP διεύθυνση του www.grippy.org
- Στέλνει το ακόλουθο αίτημα:

get /mattmarg/ HTTP/1.0 Request
User-Agent: Mozilla/2.0 (Macintosh; I; PPC)
Accept: text/html; */* Request Headers
Cookie: name = value
Referer: http://www.yahoo.com/index.html
Host: www.grippy.org
Expires: ...
If-modified-since: ...

M. Λικατίκος, EIT1425

39

Απόκριση HTTP

- One click => several responses
- HTTP1.0: new TCP connection for each page element
- HTTP1.1: KeepAlive - several requests/connection

HTTP/1.0 200 Found Status
Date: Mon, 10 Feb 1997 23:48:22 GMT Response
Server: Apache/1.1.1 HotWired/1.0 1st header
Content-type: text/html *Image/jpeg, ...*
Last-Modified: Tues, 11 Feb 1999 22:45:55 GMT

M. Λικατίκος, EIT1425

40

Σύνδεση HTTP 1.0

- Μία σύνδεση HTTP αποτελείται από τα ακόλουθα τέσσερα στάδια:
 1. **Έναρξη της σύνδεσης**: ο πελάτης επικοινωνεί με τον εξυπηρετητή σε ένα URL και σε αριθμό πύλης (port number), τα οποία καθορίζονται από το URL (κατά συνθήκη τιμή του αριθμού πύλης το 80).
 2. **Πραγματοποίηση της σύνδεσης**: ο πελάτης στέλνει ένα μήνυμα στον εξυπηρετητή, ζητώντας εξυπηρέτηση και καθορίζοντας την HTTP-μέθοδο της δοσοληψίας (συνήθως GET ή POST).
 3. **Αποστολή απάντησης**: επικεφαλίδα (περιγραφή κατάστασης της δοσοληψίας, περιγραφή δεδομένων) και δεδομένα.
 4. **Κλείσιμο της σύνδεσης**.

M. Λικατίκος, EIT1425

41

Ενσωμάτωση Εικόνων

- Ένα υπερκείμενο στον Ιστό αποθηκεύεται συνήθως σε περισσότερα του ενός αρχεία (ίσως και κόμβους): το κείμενο σε ένα αρχείο και κάθε εικόνα σε διαφορετικό αρχείο.
- Ωστόσο, όταν επιλέξουμε να ανοίξουμε ένα αρχείο, στο παράθυρο του πλοηγού μας εμφανίζεται ολόκληρη η πληροφορία μαζί, μορφοποιημένη με τον κατάλληλο τρόπο.
- Η αναζήτηση και εμφάνιση ολόκληρης της πληροφορίας γίνεται υπ' ευθύνη του πελάτη-πλοηγού (πώς;). Ο εξυπηρετητής «βλέπει» απλά μία σειρά από αιτήματα, πρώτα για το αρχείο HTML και στη συνέχεια για κάθε ενσωματωμένη εικόνα.
- Η αυτόματη αναζήτηση των ενσωματωμένων εικόνων, ωστόσο, έχει συνέπειες για τον εξυπηρετητή (υπερφόρτωση).

M. Λικατίκος, EIT1425

42

Ταυτόχρονη Εξυπηρέτηση Πολλών Αιτημάτων

- Οι εξυπηρετητές Ιστού είναι σχεδιασμένοι ώστε να μπορούν να εξυπηρετήσουν ταυτόχρονα, πολλές ταυτόχρονες αιτήσεις HTTP.
- Αυτό επιτυγχάνεται με διάφορους τρόπους:
 - Με κλωνοποίηση ενός νέου αντιγράφου του httpd προγράμματος για κάθε νέα αίτηση. Κάθε κλώνος του httpd διαχειρίζεται ακριβώς μία αίτηση.
 - Με χρήση πολυνηματικής εκτέλεσης (multithreading): το πρόγραμμα httpd έχει πολλά νήματα εκτέλεσης. Κάθε νήμα διαχειρίζεται μια αίτηση. Το πρόγραμμα δίνει τον έλεγχο σε άλλο νήμα όταν το εκτελούμενο νήμα πρέπει να περιμένει κάποια εξυπηρέτηση από το Λειτουργικό Σύστημα.
 - Με κατανομή της εργασίας σε πολλά υποβοηθητικά προγράμματα: το httpd απλά δέχεται αιτήσεις και τις κατανέμει σε άλλα προγράμματα για περαιτέρω εξυπηρέτηση.

Μ. Αικαίδος, ΕΠΛ425

43

Πολλαπλοί Εξυπηρετητές σε έναν κόμβο

- Σε έναν κόμβο του Διαδικτύου μπορούν να τρέχουν περισσότεροι του ενός εξυπηρετητές Ιστού, εφόσον χρησιμοποιούν διαφορετική πύλη επικοινωνίας (port).
- Η κατά συνθήκη πύλη επικοινωνίας μέσω HTTP είναι η 80, ενώ πύλες διαθέσιμες στους χρήστες αριθμούν από 1025 και πάνω. Π.χ.:
`http://www.server.org/inter.html`
`http://www.server.org:8080/inter.html`
`http://www.server.org:8081/inter.html`
- Ορισμένες φορές είναι επιθυμητό να τρέχουν περισσότεροι του ενός εξυπηρετητές σε έναν κόμβο, αλλά κάθε διαθέτης να έχει διαφορετικό όνομα κόμβου (host name).
- Αυτό γίνεται με την υποστήριξη του Λειτουργικού Συστήματος στην παροχή *Virtual Host Support* (υποστήριξη εικονικού κόμβου) και την αντιστοίχιση περισσότερων της μίας διευθύνσεων IP στο ίδιο μηχάνημα.

Μ. Αικαίδος, ΕΠΛ425

44

Τα μειονεκτήματα του HTTP 1.0

- Απλό μοντέλο αιτημάτων/απαντήσεων, το οποίο οδηγεί σε προβλήματα:
 - Επίδοσης και Απόδοσης (performance & efficiency)
 - Υλοποίησης εφαρμογών που απαιτούν διαδοχικές αλληλεπιδράσεις πελάτη-εξυπηρετητή.
 - Πολύ φτωχά μοντέλα συνέπειας κρυφής μνήμης (cache consistency)
 - Δυσκολία συνύπαρξης εξυπηρετητών με διαφορετικά ονόματα πεδίου, πάνω στην ίδια διεύθυνση IP. Αυτό που είναι δυνατόν, είναι η χρήση virtual hosts – δηλαδή η ύπαρξη περισσότερων εξυπηρετητών WWW στην ίδια μηχανή, όπου ο κάθε εξυπηρετητής έχει διαφορετικό όνομα πεδίου και IP address.
 - Δυσκολία υπολογισμού μεγέθους περιεχομένου (content-lengths)
 - Διακοπή συνόδου ισοδυναμεί με απώλεια δεδομένων – δεν υπάρχει η δυνατότητα τεμαχισμού της ανακτούμενης πληροφορίας (chunking)

Μ. Αικαίδος, ΕΠΛ425

45

Επίμονο HTTP (Persistent HTTP)

- Ορισμένα από τα προβλήματα του HTTP/1.0 αντιμετωπίστηκαν με την υιοθέτηση του Επίμονου HTTP – Persistent HTTP (P-HTTP), σύμφωνα με το οποίο, μετά από μια δοσοληψία HTTP, η αντίστοιχη TCP σύνδεση δεν διακόπτεται, ούτως ώστε τυχόντα αιτήματα που έπονται από τον πελάτη προς τον διαθέτη να εξυπηρετηθούν μέσα από αυτή.
- Στο HTTP/1.0 με χρήση επικεφαλίδας: Connection: Keep-Alive
- Τι πετυχαίνει το P-HTTP προσφέρει βελτιώσεις:
 - Κυρίως στην επίδοση του εξυπηρετητή παρά στο δίκτυο.
 - Επιτρέπει την διασωλήνωση (pipelining) πολλών αιτημάτων στην ίδια σύνοδο TCP
- Δυσκολίες:
 - how does a client know when document is returned?
 - when does the connection get dropped?
 - idle timeouts on server side
 - client drops connections
 - server needs to reclaim resources

Μ. Αικαίδος, ΕΠΛ425

46

HTTP/2



- According to W3Techs, as of December 2015 **5.6% of the top 10 million websites supported HTTP/2**
- Negotiation mechanism allows clients and servers to elect to use HTTP 1.1, 2.0, or potentially other non-HTTP protocols.
- Maintain high-level compatibility with HTTP 1.1 (for example with methods, status codes, and URIs, and most header fields)
- Decrease latency to improve page load speed in web browsers by considering:
 - Data compression of HTTP headers
 - Server push technologies
 - Pipelining of requests
 - Fixing the head-of-line blocking problem in HTTP 1
 - Multiplexing multiple requests over a single TCP connection
- Support common existing use cases of HTTP, such as desktop web browsers, mobile web browsers, web APIs, web servers at various scales, proxy servers, reverse proxy servers, firewalls, and content delivery networks

Μ. Αικαίδος, ΕΠΛ425

47