



SWCON104
Web & Python Programming

Introduction to Web

Department of Software Convergence

Why web in Python course?

- To understand
 - Common terms used in web technologies
 - Basics of web application development on internet
 - Client-side development techniques (frontend)
 - Server-side development techniques (backend)

- Apply to
 - Web and mobile service development : client and server
 - Data analytics : web and SNS data
 - Servers : big-data processing and backend server programming

Today

- Introduction to Web programming
 - Basics of the web
 - HTML
 - URL
 - HTTP

Practice

- None

What is web programming?

- Web programming refers to the writing, markup, and coding involved in Web development, which includes
 - Web content
 - Web client and server development
 - Network security

- Most common languages used for web programming are
 - Contents: HTML/CSS/Javascript
 - Server development: PHP, Node.js, Python, Go, C++, etc
 - API : REST/RESTful, gRPC

Web development

- Developing a web site for the internet (World Wide Web) or an intranet (a private network).
- From developing the simplest static single page of plain text
- To the most complex web-based internet applications, electronic businesses, and social network services.

Networks and the internet

- Where is a website?

- www.daum.net
- whois.domaintools.com

The image displays two web pages. The top portion is the Daum homepage, featuring the Daum logo with two cartoon characters, a search bar, and a menu bar with links like 카페, 메일, 뉴스, 지도, 증권, 쇼핑, 카카오TV, 웹툰, 블로그, 브런치, 사진, 게임, 벌플레이, 같이가자, and 더보기. Below the menu is a large advertisement for '이사할땐 다방!' (Daum Moving) featuring a smiling man in a blue jacket. The bottom portion is the Whois Lookup page on whois.domaintools.com. It has a dark background with a network graph. At the top, there's a navigation bar with HOME, MONITOR, ACQUIRE, SUPPORT, and WHOIS. The main title 'Whois Lookup' is prominently displayed in white. Below it is a search bar containing the domain 'daum.net' and a green 'Search' button.

Networks and the internet

- Where is a website?
- www.daum.net
 - » 203.133.167.16
(IP address)
 - » Internet Protocol

Domain Profile	
Registrant	Kakao Corp.
Registrant Country	KR
Registrar	Netpla.com, Inc. IANA ID: 130 URL: http://www.ibi.net Whois Server: whois.ibi.net reportabuse@ibi.net (p) 8216445001
Registrar Status	ok
Dates	8,100 days old Created on 1996-03-04 Expires on 2020-03-05 Updated on 2016-08-17
Name Servers	NS1.DAUM.NET (has 190 domains) NS2.DAUM.NET (has 190 domains)
Tech Contact	Kakao Corp. 242, Jeju-si Cheomdan-ro, Jeju-do, KR, Jeju-do, 63309, KR domain@kakaocorp.com (p) 8226710158 (f) 82
IP Address	203.133.167.16 - 7 other sites hosted on this server
IP Location	 - Gyeonggi-do - Yongin - Daou Technology
ASN	 AS9764 DAUM-NET Kakao Corp, KR (registered Dec 10, 1999)
IP History	279 changes on 279 unique IP addresses over 14 years
Registrar History	1 registrar with 3 drops
Hosting History	2 changes on 2 unique name servers over 15 years

Domain Name Service (DNS)

- Using a name (e.g. daum.net), DNS finds an IP address for us
- Runs on...
 - » Hosting service computers
 - » ISP = Internet Service Provider
 - » Smartphone or cellular provider
- ISP: Who do you use to reach the internet?



Domain Name Service (DNS)

- Lets you use names
- No need for complicated numbers
- Protects you from changes to IP address
- For example,
 - 203.133.167.16 belongs to 
 - Daum can change equipments, change IP address as they need to.
 - All you need to remember is, daum.net

IP Information for 203.133.167.16	
— Quick Stats	
IP Location	 Korea, Republic Of Yongin Daou Technology
ASN	 AS9764 DAUM-NET Kakao Corp, KR (registered Dec 10, 1999)
Whois Server	whois.apnic.net
IP Address	203.133.167.16
Reverse IP	8 websites use this address.

Basics of the web

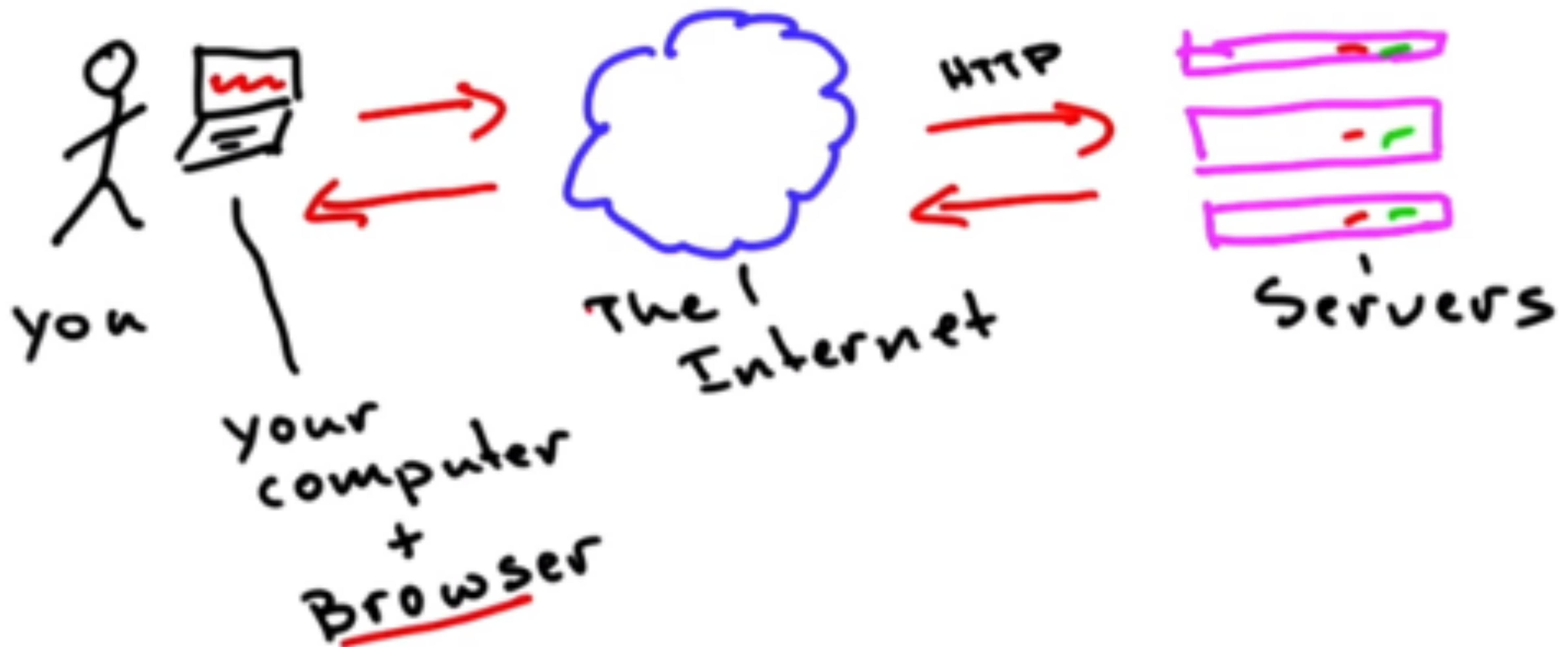
- The WWW (a.k.a, web)
- HTML: document
- URLs: how you refer to documents on the web
- HTTP: protocol (a promise between client and server)
- Web applications or services

World wide web



- Collection of HTML documents
- HTML: HyperText Markup Language
- HTML is the basis for almost every webpage
- There are documents, images, videos, and etc. online – HTML glues them together
- Hyperlinks/links: links between the pages
- Web is invented in 1990s and contains about 30 billion pages.

Major pieces of the web



- Your browser makes requests via the internet to servers.
- These requests use a protocol called HTTP.
- The servers respond with files that your browser displays

Major pieces of the web : Who's Who?

- HTTP: The main protocol of the web
- Servers: Computers that host the files that make up the web
- Internet: The world's largest computer network
- Browser: A program that runs on your computer to display files found on the web



HTML: HyperText Markup Language

- HTMLs are made of:
 - Text content: what you see
 - Markup: what it looks like, how it is arranged
 - References to other documents: images + videos
 - Links to other pages

- HTML practice
 - MS Visual Code or Notepad
 - Save as .html
 - Open the file with a browser (ex. Chrome)
 - » OR, save a Word/HWP file as .html and read via a browser

HTML markup

- Tags
- Opening tag, closing tag
- Element
- Bold tag
- Italic tag

```
<Name>Contents</Name>
```

I am going to learn to **love** HTML.

I am going to *learn* to **love** HTML.

HTML attributes

- Anchor - <a> tag

<Tag Attr="value">Contents</Tag>

click here

Attribute
name

Value

Contents

- Image - tag: void tag

Contents

Tags: whitespace

-
: break (void tag)
- <p>Content</p>: paragraph

This is a really long sentence for a line.

This is a really long
sentence for a line.

This is a really long

sentence for a line.

<p>This is a really long</p>
<p>sentence for a line.</p>

Tags: inline and block

-
 - inline
- <p>Contents</p> - block

text
text

text
text

, , <a>,
,

<p>text</p>text

text
text

<p>
<div>, <form>

HTML documents

```
<!DOCTYPE html>
<html>

<head>
<title>Title of the document</title>
</head>

<body>
Content of the <b>document</b>
</body>
</html>
```

html

<head>

Title CSS Javascript

</head>

<body>

Contents

</body>

URLs

- Uniform Resource Locator

http://www.naver.com/

Protocol Host name Path

http://www.udacity.com/cs253x/hipmunk.png

- Protocols: http, https, ftp
- Host name (domain name) translates to IP address, which locates the server that has the documents.
- Path is the document that we want to access.

URL: Query parameters (GET parameters)

`http://example.com/foo?p=1`

Name = value

`http://example.com/foo?p=1&q=neat`

`http://example.com/foo?p=foo&z=p`

URL: fragments

- To reference a particular part of the page you are looking at.
- Not sent to the server, purely exists in the browser

`http://www.example.com/foo#fragment`

`http://www.example.com/foo?p=1#fragment`

URL: port

- Identifier to a web service or a server program
 - Default port to web service : 80
 - `http://ServerDomainName:80/`
- If you want to use a different port, you need to include it between the host and the path separated by a colon.

HTTP

- Hyper Text Transfer Protocol
- Main protocol of the web
- What your browser uses to talk to the servers.

http://www.example.com/

- Request from the browser to the server begins with a request line

GET / HTTP/1.1

- GET : method
- / : path
- HTTP/1.1 : protocol and version

HTTP: request and response

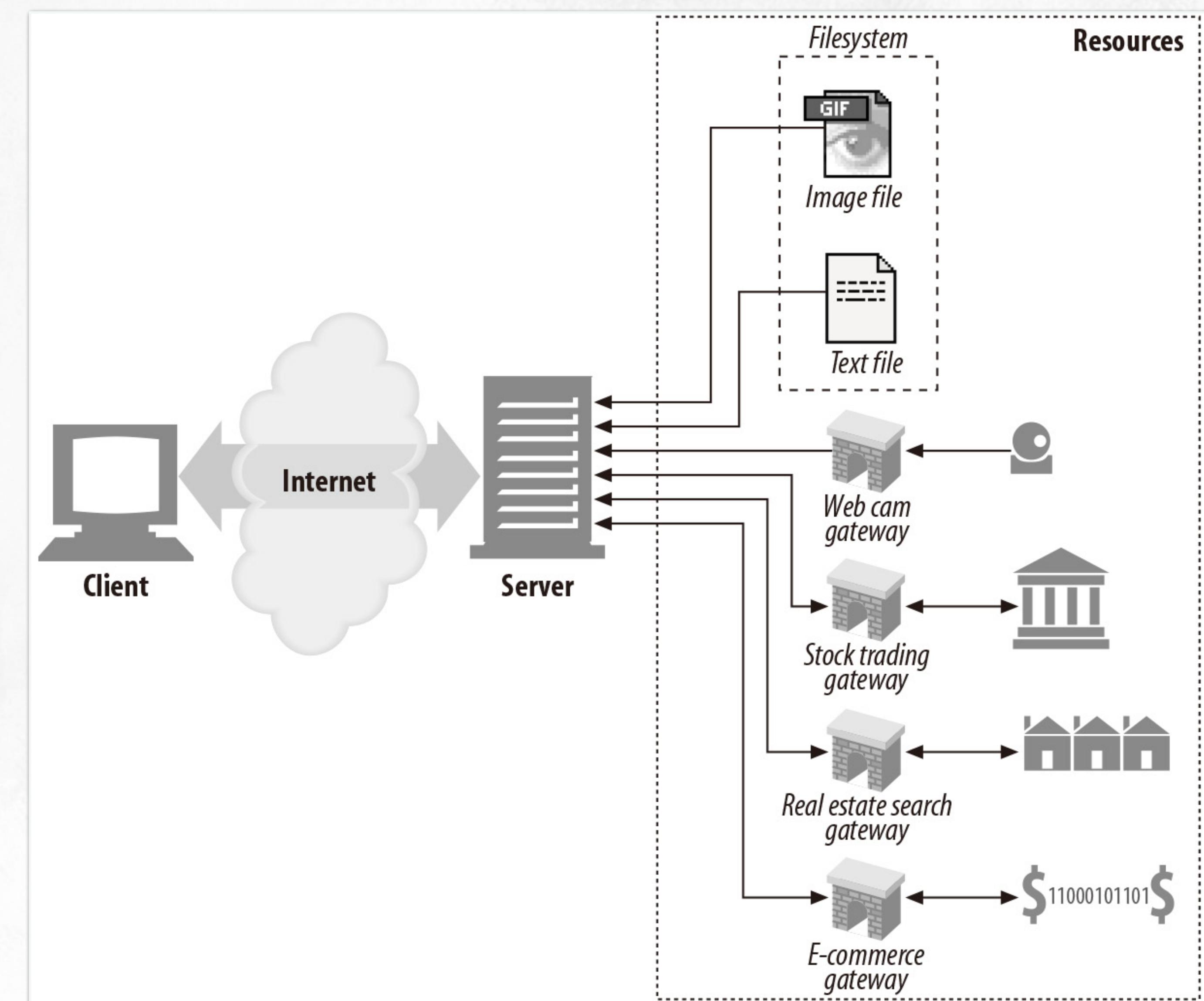
- Simple request and response architecture



HTTP: resources

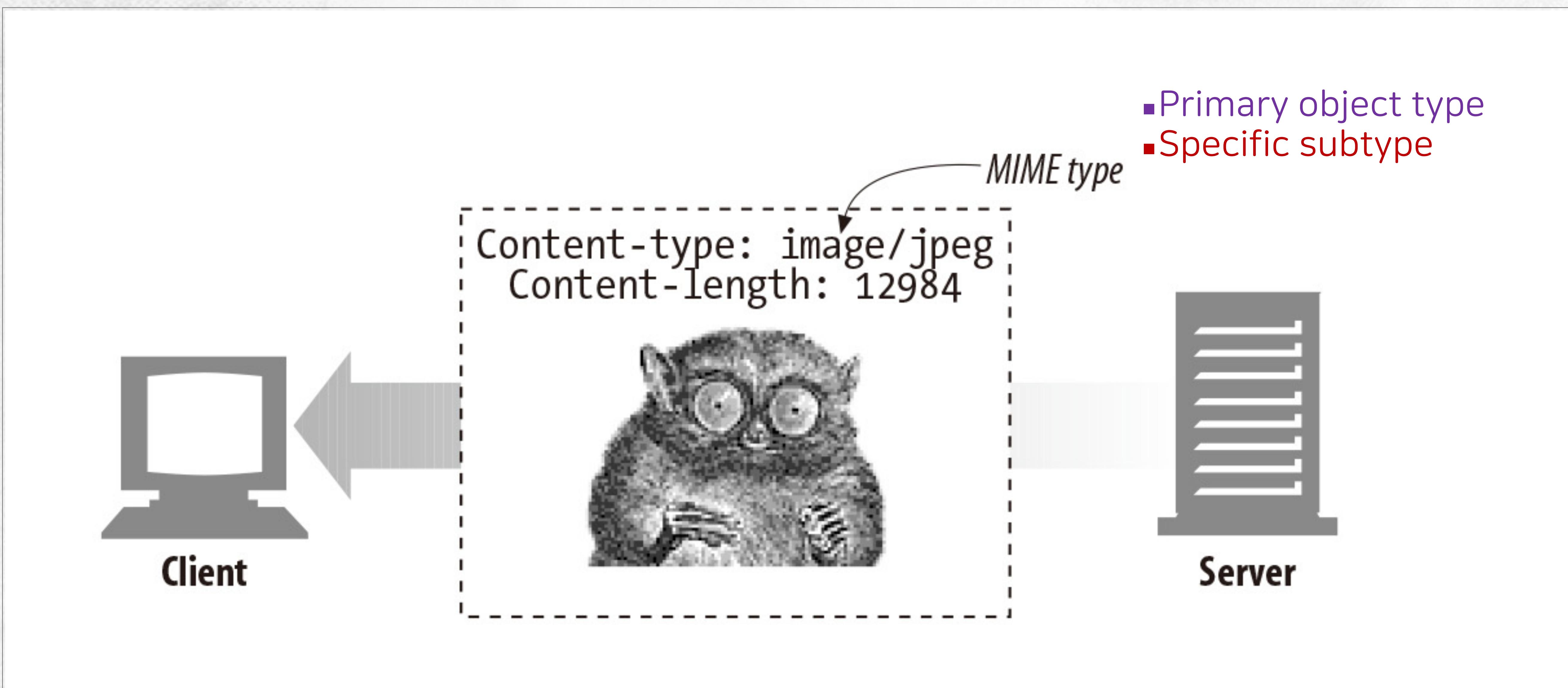
- Static content
 - File system
 - MS word file, Image file, **Video Clip**, etc.

- Dynamic content
 - Generated by SW program
 - Live image, Trade stocks, etc.



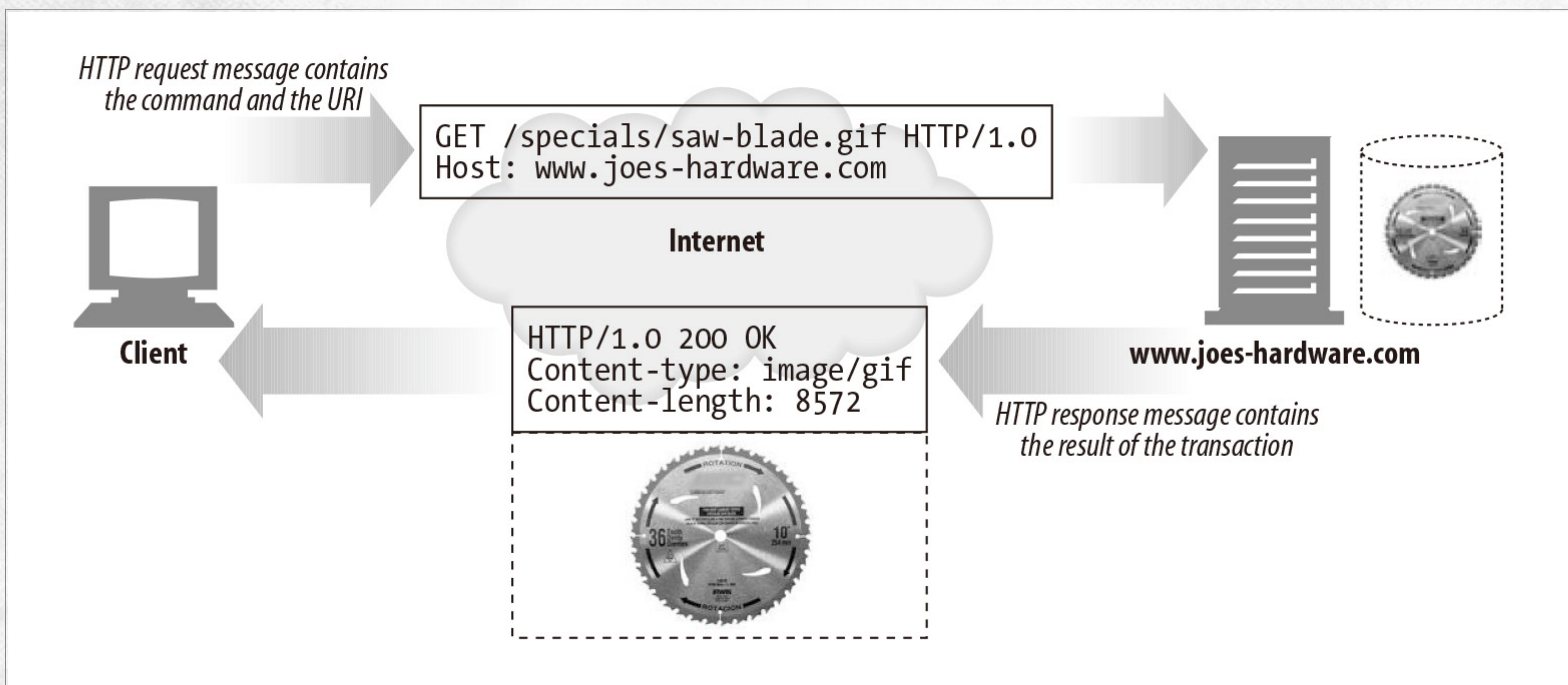
HTTP: media types

- Media Types
- MIME (Multipurpose Internet Mail Extensions)
- Textual label



HTTP: transactions

- HTTP Messages
 - Request
 - Response



HTTP: transactions

- HTTP Methods

- Supports several request commands
- Server what action to perform

HTTP method	Description
GET	Send named resource from the server to the client.
PUT	Store data from client into a named server resource.
DELETE	Delete the named resource from a server.
POST	Send client data into a server gateway application.
HEAD	Send just the HTTP headers from the response for the named resource.

HTTP: transactions

- Status Codes
 - HTTP response contain status code
 - Three-digit numeric code
 - Explanatory textual “reason phrase”

HTTP status code	Description
200	OK. Document returned correctly.
302	Redirect. Go someplace else to get the resource.
404	Not Found. Can't find this resource.

HTTP: messages

- Request and Response
- No other kinds of HTTP Messages
- Line-oriented sequences of characters
- Easy for humans to read and write

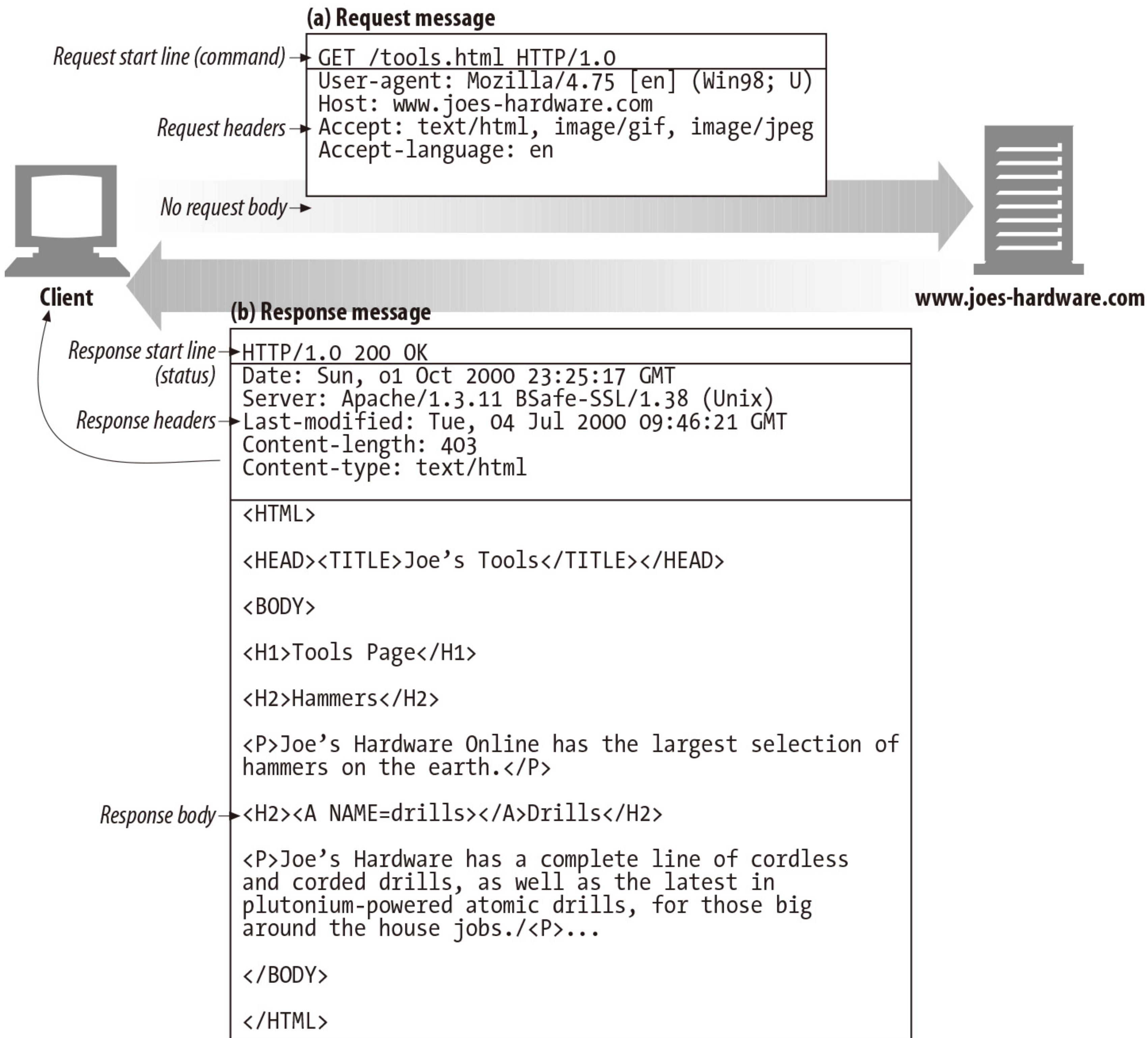
(a) Request message

```
GET /test/hi-there.txt HTTP/1.0
Accept: text/*
Accept-Language: en,fr
```

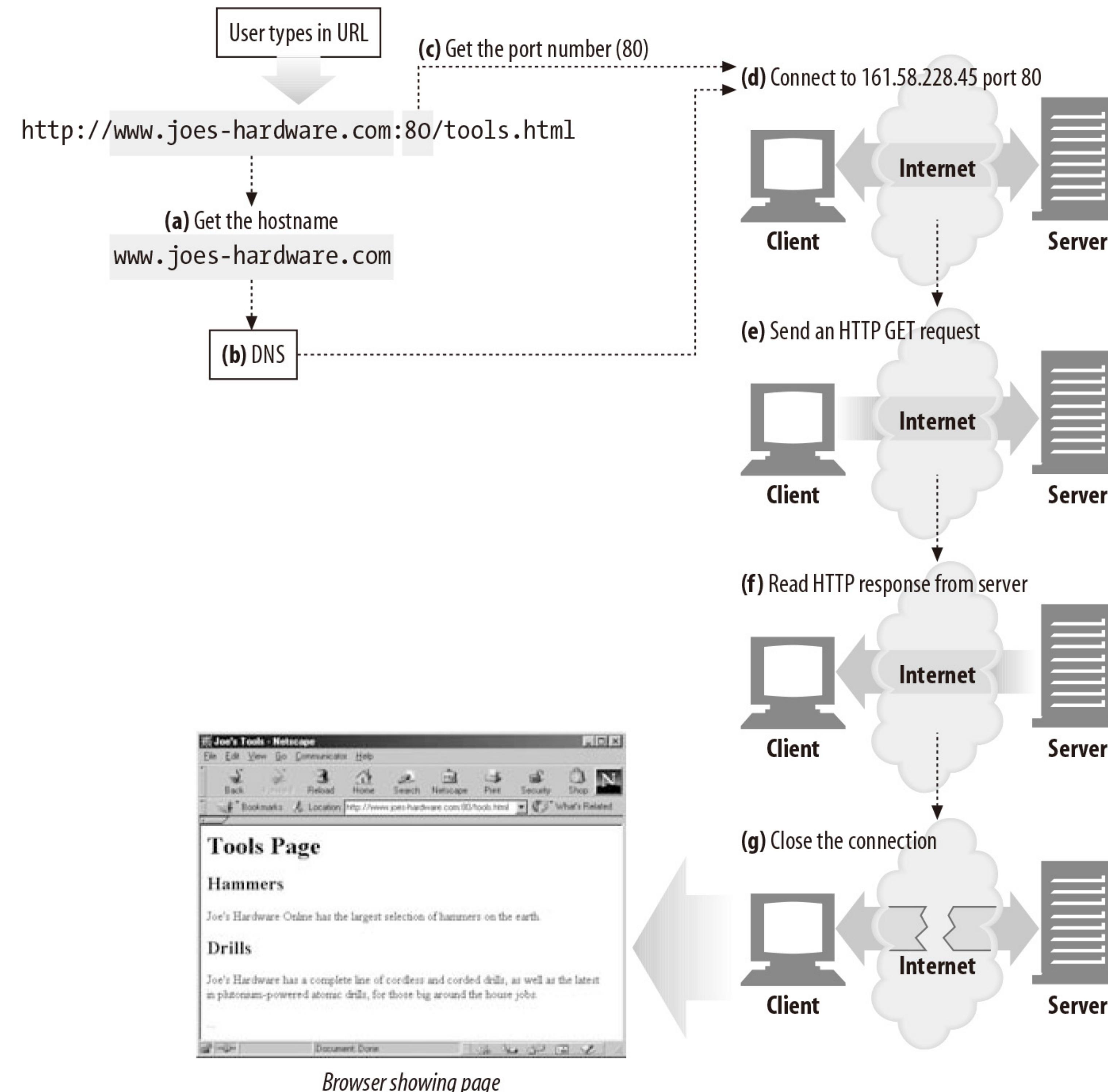
(b) Response message

Start line	HTTP/1.0 200 OK
Headers	Content-type: text/plain Content-length: 19
Body	Hi! I'm a message!

HTTP: example

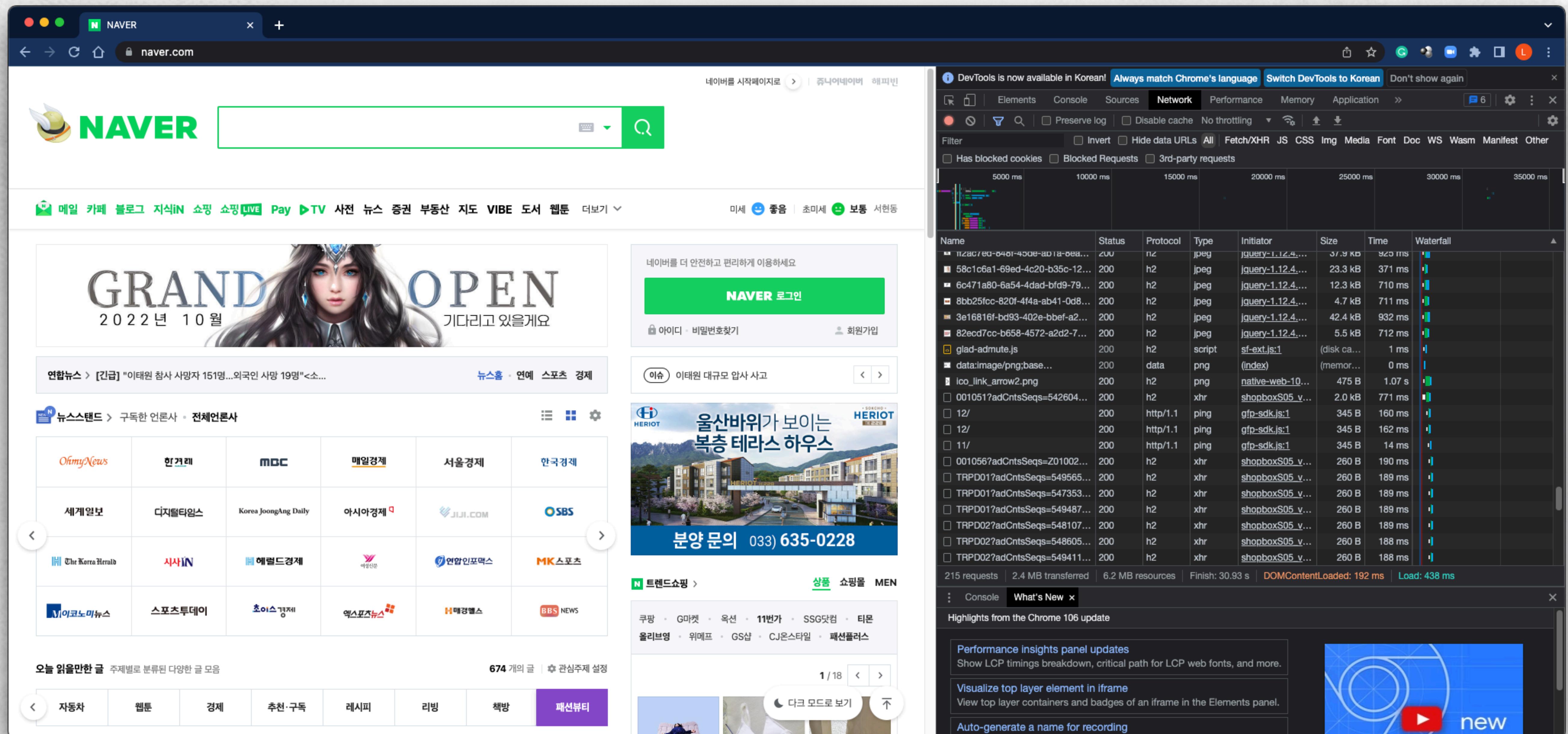


HTTP: example (put it all together)



Tools: Chrome browser

- 메뉴 >> 도구 더 보기 >> 개발자 도구 >> Elements [HTML]
- 메뉴 >> 도구 더 보기 >> 개발자 도구 >> Network [HTTP]



Reference





Thank you