HyperText Transfer Protocol HTTP v1.1

hussein suleman
uct cs honours 2009

What is HTTP?

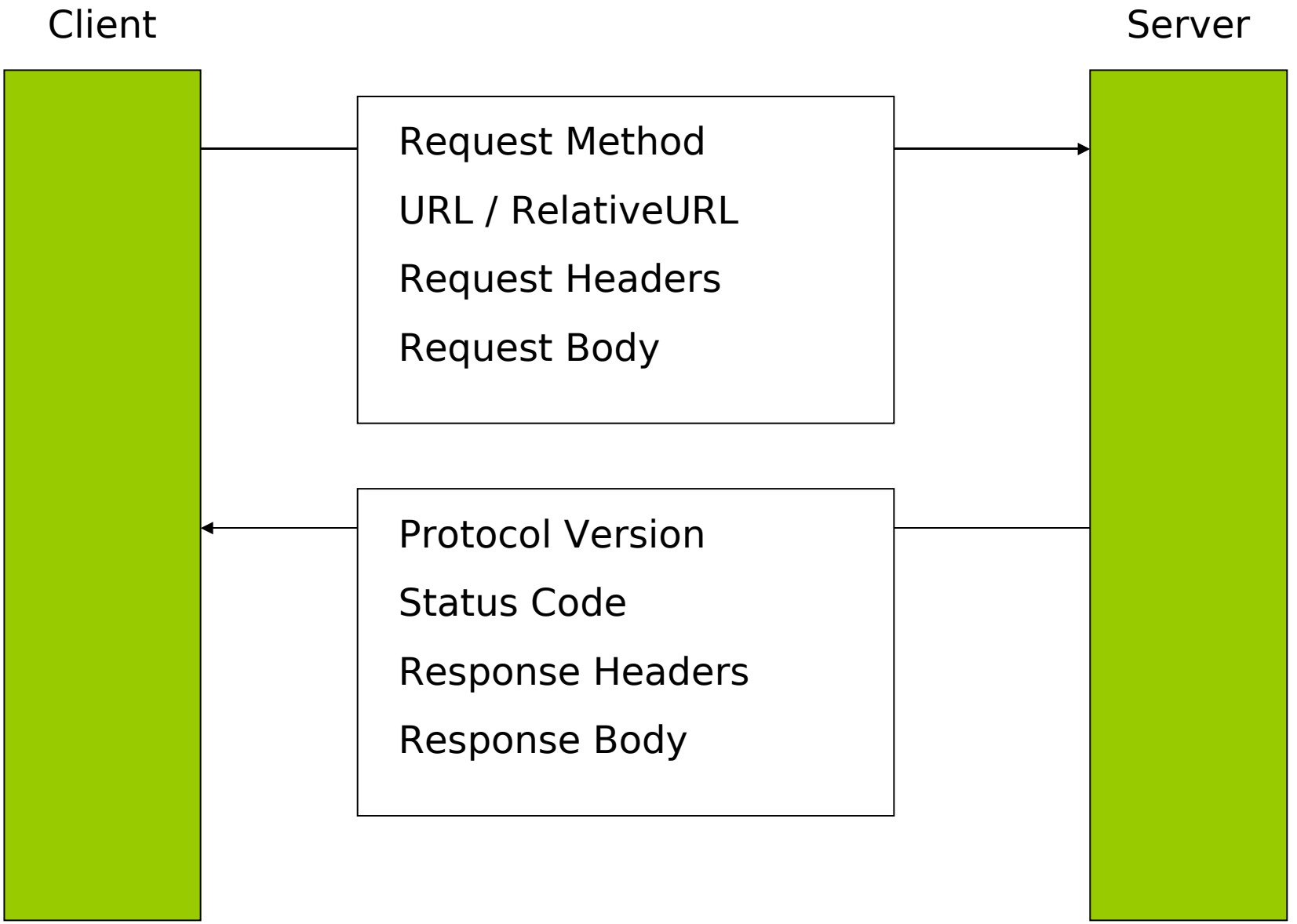
- Protocol for transfer of data between Web servers and Web clients (browsers).
- Popular Web servers:
  - Apache HTTPD
  - JBoss
  - Tomcat
- Popular Web clients:
  - Firefox
  - Opera
  - wget
- Defined formally by IETF as RFC2616.

Abstract

“The Hypertext Transfer Protocol (HTTP) is an application-level protocol for distributed, collaborative, hypermedia information systems. It is a generic, stateless, protocol which can be used for many tasks beyond its use for hypertext, such as name servers and distributed object management systems, through extension of its request methods, error codes and headers [47]. A feature of HTTP is the typing and negotiation of data representation, allowing systems to be built independently of the data being transferred.

HTTP has been in use by the World-Wide Web global information initiative since 1990. This specification defines the protocol referred to as "HTTP/1.1", and is an update to RFC 2068 [33].”

Basic Operation



Example HTTP Communication

- Client→Server:

```
GET / HTTP/1.1
Host: www.cs.uct.ac.za
```

- Server→Client:

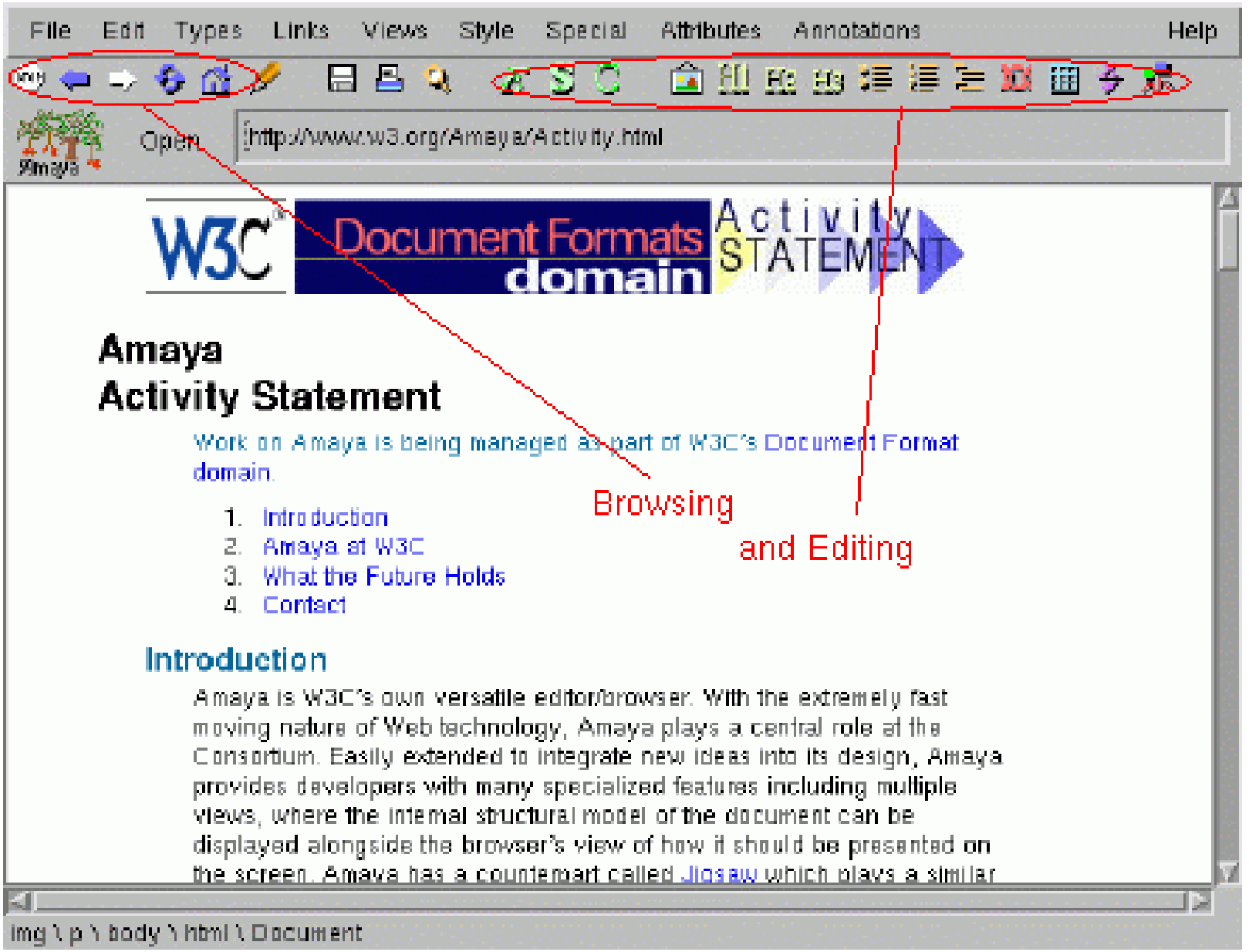
```
HTTP/1.1 200 OK
Content-type: text/html
Content-length: 1024

<html>Really old webpage!</html>
```

HTTP Request

- Format:
  - Method URI HttpVersion

Method	Description
OPTIONS	capabilities of resource/server
GET	retrieve resource
HEAD	retrieve headers for resource
POST	submit data to server
PUT	replace/insert resource on server
DELETE	remove resource from server
TRACE	trace request route through Web

Amaya


File Edit Types Links Views Style Special Attributes Annotations Help
Open
http://www.w3.org/Amaya/Activity.html
W3C
Document Formats domain
Activity STATEMENT
Amaya
Activity Statement
Work on Amaya is being managed as part of W3C's Document Format domain.
1. Introduction
2. Amaya at W3C
3. What the Future Holds
4. Contact
Introduction
Amaya is W3C's own versatile editor/browser. With the extremely fast moving nature of Web technology, Amaya plays a central role at the Consortium. Easily extended to integrate new ideas into its design, Amaya provides developers with many specialized features including multiple views, where the internal structural model of the document can be displayed alongside the browser's view of how it should be presented on the screen. Amaya has a counterpart called Jigsaw which plays a similar
Browsing
and Editing
img \ p \ body \ html \ Document

Tim Berners-Lee's WWW Vision

- The WWW is meant to be a place for accessing and authoring content, not just the former.
- Amaya is W3C's experimental 2-way browser that works with their 2-way server Jigsaw.
- Is access more important than content creation? Why (not)?

URLs, URNs and URIs

- Every resource accessible through HTTP is identified by a Uniform Resource Location (URL), which is a location-specific identifier.
  - For example,
    - http://www.cs.uct.ac.za:80/
    - ftp://ftp.cs.uct.ac.za/
- A Uniform Resource Identifier (URI) is a standard format (<scheme>:<identifier>) generic identifier.
  - For example,
    - mailto:hussein@cs.uct.ac.za
    - oai:www.ndltd.org:123456-789
- A Uniform Resource Name (URN) is one example of a location-independent URI.
  - For example,
    - urn:isbn:123-456-789
- Note: Every URL and URN is also a URI!

HTTP Response

- Format:
  - HTTPVersion StatusCode Reason

Status	Reason	Description
200	OK	Successful request
206	Partial Content	Successful request for partial content
301	Moved Permanently	Resource has been relocated
304	Not Modified	Conditional GET but resource has not changed
400	Bad Request	Request not understood
403	Forbidden	Access to resource not allowed
404	Not Found	URI/resource not found on server
500	Internal Server Error	Unexpected error

HTTP Headers

- Accept: Indicates which data formats are acceptable.
  - Accept: text/html, text/plain
- Content-Language: Language of the content
  - Content-Language: en
- Content-Length: Size of message body
  - Content-Length: 1234
- Content-Type: MIME type of content body
  - Content-Type: text/html
- Date: Date of request/response
  - Date: Tue, 15 Nov 1994 08:12:31 GMT
- Expires: When content is no longer valid
  - Expires: Tue, 15 Nov 1994 08:12:31 GMT
- Host: Machine that request is directed to
  - Host: www.cs.uct.ac.za
- Location: Redirection to a different resource
  - Location: http://myserver.org/
- Retry-After: Indicates that client must try again in future
  - Retry-After: 120

Other HTTP Features

- Authentication
- Persistent connections
- GET-if-modified
- Byte ranges
- Content type negotiation
- Cache control
- Proxy support

Non-static content

- HTTP can support content that is not static.
- For a GET request, data is appended to the request – for a POST request, data is contained in the request body.
- Responses are generated by a piece of software and are similar to the non-static version.

Common Gateway Interface

- Common Gateway Interface (CGI) defines how parameters are passed to Web applications.
- For a GET request, the URL contains
  - http://host:port/path/file?var1=value1&var2=value2&var3=valu
    ...
  - These are called URL-encoded parameters.
- The part beyond '?' is passed in the environment of the Web application as a QUERY_STRING.
- The application interprets the QUERY_STRING, generates an HTTP response and writes it to stdout, with at least a Content-type header.
- HTML forms generate GET requests that can easily be converted to support CGI.

Notes on URL-Encoding

- URLs assign special semantics for some characters so if they are needed, they must be inserted as character codes.
  - e.g., http://host:port/test?math=1+%2B+2+%3D+3

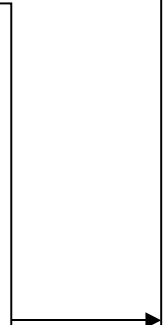
Character	Regular Use	Code
:	Separates port from host	%3A
?	Separates parameters from file	%3F
=	Separates var from value	%3D
&	Separates parameters	%26
+	Indicates a space	%2B
/	Separates elements of path	%2F

CGI POST

- GET cannot handle file uploads.
- File uploads are handled as Multipart-MIME messages sent from the client to the server.

recursive example

if you fill in the form embedded here, this is the data that gets sent to the server



```
----------------------------41184676334
Content-Disposition: form-data;
name="var1" something
----------------------------41184676334
Content-Disposition: form-data; name="var2"; filename="testpost.html“
Content-Type: text/html

<html>
<body>
<form action="http://banzai.cs.uct.ac.za/~hussein/cgi-bin/testpost/testpost.pl"
method="POST" enctype="multipart/form-data">
<input type="text" name="var1" size="40"/>
<br/>
<input type="file" name="var2" size="40"/>
<br/>
<input type="submit"/>
</form>
</body>
</html>
----------------------------41184676334--
```

Not-So-Common Gateway Interfaces

- Instead of QUERY_STRING and stdin and stdout for data,

- Java servlets use methods to acquire parameters and output data.
- PHP defines global variables for GET/POST query parameters.

References

- Achour, Mehdi, Friedhelm Betz, Antony Dovgal, Nuno Lopes, Philip Olson, Georg Richter, Damien Seguv and Jakub Vrana (2006) PHP Manual. Available http://www.php.net/manual/en/
- Fielding, R., J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach and T. Berners-Lee (1999) Hypertext Transfer Protocol – HTTP/1.1, RFC 2616, Network Working Group, IETF. Available ftp://ftp.rfc-editor.org/in-notes/rfc2616.txt
- NCSA (1996) The Common Gateway Interface. Available http://hoohoo.ncsa.uiuc.edu/cgi/
- URI Planning Interest Group (2001) URIs, URLs, and URNs: Clarifications and Recommendations 1.0, W3C. Available http://www.w3.org/TR/uri-clarification/
- Wilson, Brian (2003) URL Encoding. Available http://www.blooberry.com/indexdot/html/topics/urlencoding.htm