



MONASH
University

MONASH
INFORMATION
TECHNOLOGY

HTTP : HyperText Transfer Protocol

HTML : HyperText Markup Language

Introduction

Stateless Protocol

Server does not remember client state between connections

Request-response format

HTTP/1.1 add persistent connections

Multi-homed servers etc

Request

Client makes connection to server
If port not specified, default is port 80
Request must specify a URL
Universal Resource Location
Structure of Request
GET /pathtofile/filename.html HTTP/1.0



MONASH
University

What Happens when making a web request

What happens when we make a request like:

<http://www.monash.edu/index.html>





Client Server Model

Server

Create a socket

Bind socket to address

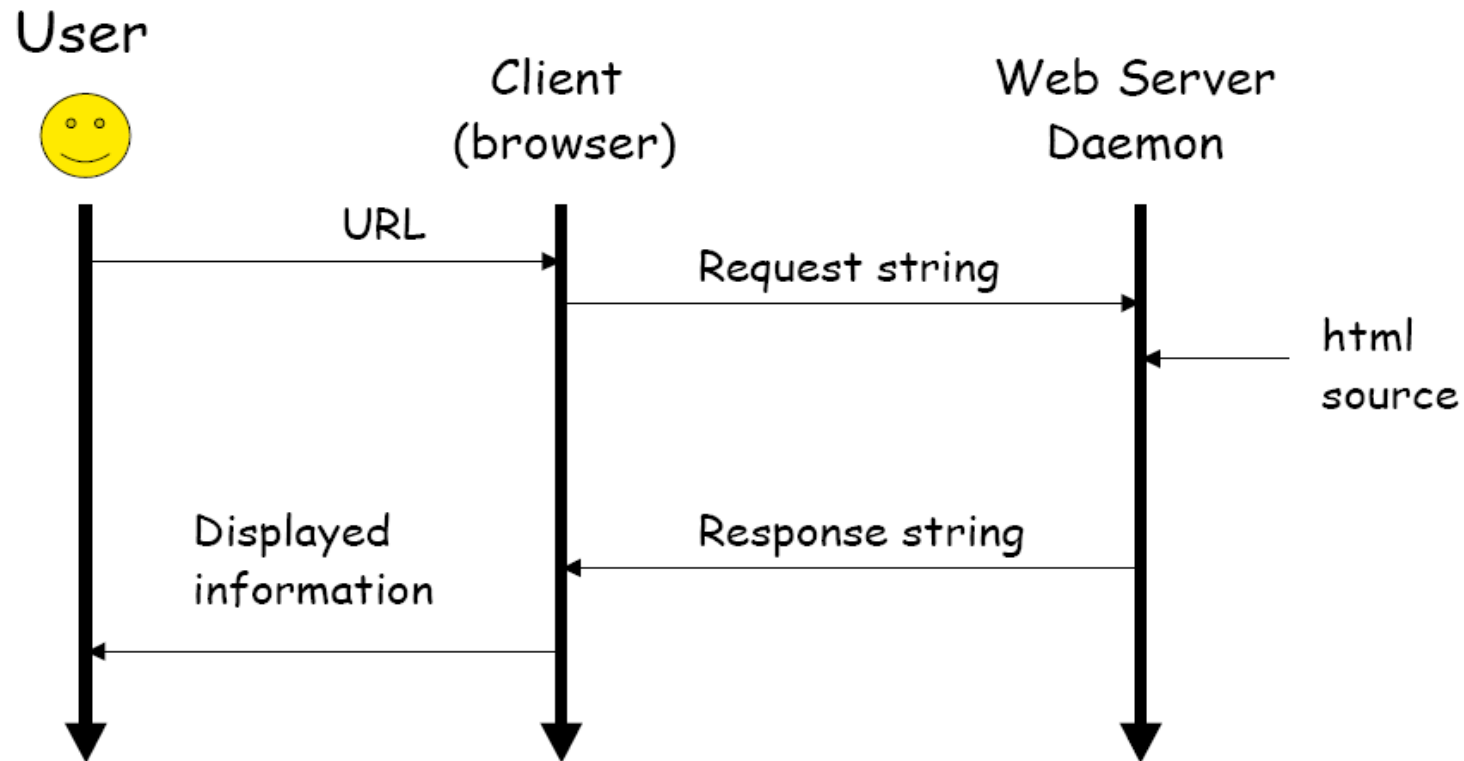
Listen for connections

Accept connection

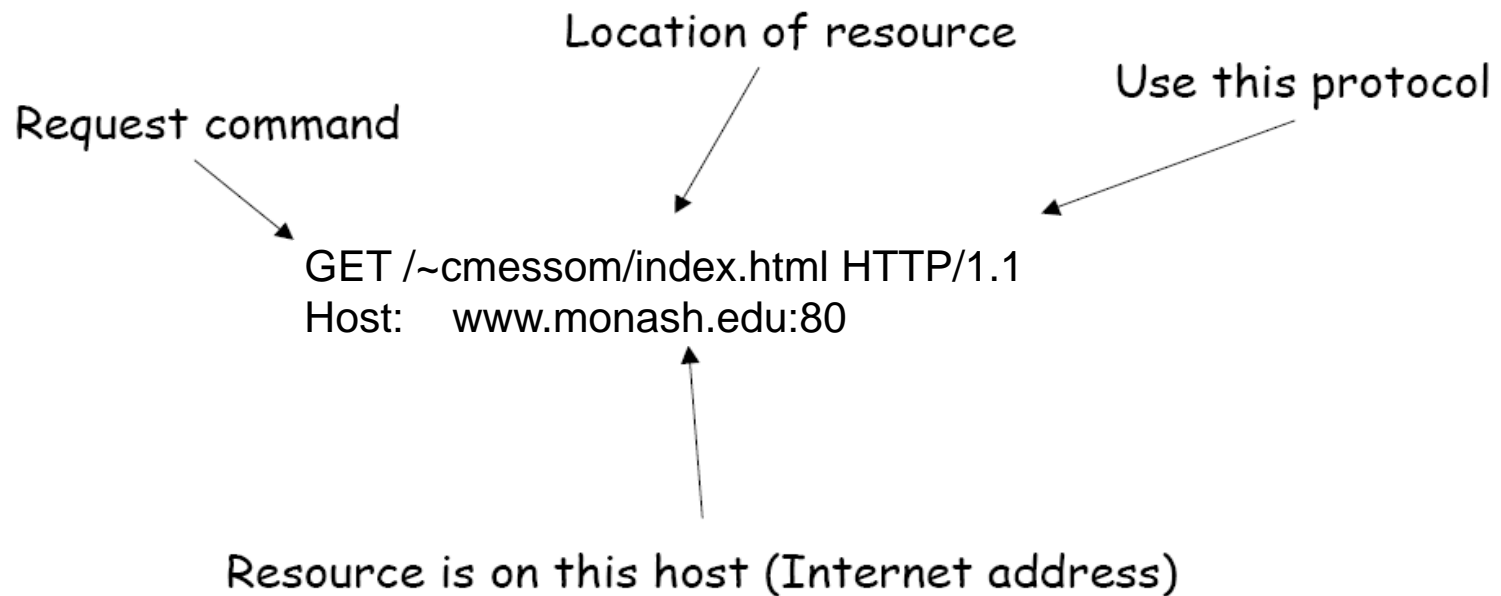
Receive string requesting a resource, e.g. request a web page

Send resource using protocol in URL

Loading a web page



The Request





MONASH
University

Request Commands

OPTIONS - Request for communication options available for the Request-URI.

GET - Retrieve an entity requested in the Request-URI.

HEAD - Identical to GET but no message-body.

POST - Providing a block of data

PUT - Store the supplied Request-URI.

DELETE - Delete the resource identified by the Request-URI.

TRACE

CONNECT



GROUP
OF EIGHT
AUSTRALIA

The Response

HTTP/1.1 200 OK
Date: Mon, 23 May 2005 22:38:34
GMTServer: Apache/1.3.27 (Unix) (Red-Hat/Linux)
Last-Modified: Wed, 08 Jan 2003 23:11:55
GMTETag: "3f80f-1b6-3e1cb03b"
Accept-Ranges: bytesContent-Length: 438
Connection: close
Content-Type: text/html
Content-Length: 32

--frontier
This is the text body of the document
--frontier

Status line

Headers

Document body



MONASH University Response

Headers + Response Body

Body is requested Document

Headers

- status-line
- general-header
- response-header
- entity-header
 - Content-Length
 - Content-Type
 - Last-Modified



MONASH
University

status-line

200 OK

404 Not Found

301 Moved Permanently

302 Moved Temporarily

303 See Other (*HTTP 1.1 only*)

500 Server Error

501 Not Implemented



MONASH
University

GET Request

Use to request a specific URI

Additional Parameters specified via URL

`http://server/file?variable=val&variable2=val`

Most servers have a limit on the size of a url that can be processed



MONASH
University

POST Request

Submit annotations to requested URI

Form data posted

- URL encoded, spaces changes to +, symbols turned into %XX hexadecimal format

Multipart MIME

CGIs

Common Gateway Interface

Standard for starting programs on the server

Environmental Variables

Specify Query String

Client and Server parameters

Specify output content-type, e.g text/html

Then send body



MONASH University **MIME types**

Multipurpose Internet Mail Extensions

Specifies the file format to follow

E.g.

- text/html
- image/jpg
- application/pdf

Your server side application must specify the correct MIME type it is delivering



MONASH
University

HTML:

HyperText Markup Language

Introduction

Mark up to describe structure of document

Basically the markup is a suggestion to the browser how to present the content.

Basic HTML is very simple, it shouldn't be used for sophisticated layout of a document.

HTML tags + attributes define how content is displayed



Basic Document Structure

Document type declaration

Each standard conforming page should start with the document type declaration

- `<!DOCTYPE HTML PUBLIC "-//W3C/DTD HTML 4.0//EN">`

Document: `<html> </html>`

Contains Head and body

Head Section: `<head> </head>`

This contains the HTML description of the page

Body Section: `<body> </body>`

This is what will be rendered on the browser and contains the content of the page

<title> </title>

Required element, used by browsers and displayed on browser windows and titles of book marked pages

<base href=“...”>

Optional element, specifies where relative URLs are referenced from

<meta name=“...” content=“...”>

Optional element to specify information about the document, used by search engines for indexing

<meta http-equiv=“...” content=“.....”>

Optional element used by the server to generate matching HTTP headers

- e.g. Expires: Fri, 16 Dec 2004 23:59:59 GMT
- Or Refresh 1800;
URL=<http://www.monash.edu/~chmessom/FIT5032/index.html>

Document Body

`<h1> </h1>`, `<h2> </h2>`, etc

Large heading sizes

`
`

Line Breaks

`<p> </p>`

Paragraphs

`<pre> </pre>`

Paragraphs with whitespace preserved,
useful for formatted code

`<xmp> </xmp>`

Html interpretation switched off, useful
for displaying HTML

Special Characters

`<`, `>`, `&`, `"`, ` `



More Markup

`
`

Unordered list

`
`

Ordered lists

`` bold, `<I>` italic, `<TT>` fixed
width

Inserting Images

``

Image is referenced by the url

height

Displayed height of image

Width

Displayed width of image

Alt

Alternate text for text only browsers of
browsers with images switched off



Links

`SomeText`

Provides a link to the given url

Linking images

` `

Layout in Tables

`<table> </table>`

Defines table

`<tr> </tr>`

Defines table row

- `align="center/right/justify"`
- `bgcolor="color"`
- `valign="top/middle/bottom/baseline"`

`<td> </td>`

Defines table division


```
<form method="post" action="ACT.php"> </form>
```

Used for submitting user data

```
<input type="TYPE" > </input>
```

Input elements

Type=submit: button to submit form

Type=text: text box

Type=password: password input (displays asterix)

Type=reset: resets the form

Type=hidden: provides a hidden field (not displayed in browser)

Type=file: provides a control for users to specify a file to submit

Addition <form> Elements

```
<select name="....." multiple>
```

```
<option value="..." selected>
```

```
OPTION1</option>
```

```
</select>
```

Specifies a list box of selections

```
<textarea name="..." cols=".."
```

```
rows=".."> </textarea>
```

Specifies a large input text area

List Boxes

Radio Buttons

```
<input type="radio" name="g1" value=1 checked>
```

```
<input type="radio" name="g1" value=2>
```

- Select one of the options

Check Boxes, select many

```
<input type="checkbox" name="c1" value=1 checked>
```

```
<input type="checkbox" name="c2" value=2>
```

- Select many of the options

The Form tag specifies a method attribute

GET submits form data using the get method

- The form data is encoded into the URL and visible in most browsers
- There are practical limits to the size of encoded URIs received by servers
- Some characters are not allowed in the URI (only ASCII)

POST submits the form data using the post method

- Form Data is encoded using the Enctype specified, default encoding is url encoding

HTML Editors

Text Editor ☺

- Manual editing, good for learning and experts

XML editor

- Source editing with tools for autocompletion using DTD

Netscape Composer

- Quick editing of HTML in both WYSIWYG and source format

Macromedia DreamWeaver

- Fully integrated web development environment



<body> attributes

background="filename"

Background image

bgcolor="color"

Background colour

text="color"

Text colour

link="color"

Text link colour

alink="color"

vlink="color"

Visited link colour

See Reference for full lists of tags and
attributes

<table> attributes

align="center/right"

background="image"

bgcolor="color"

border="value"

bordercolor="color"

cellpadding="value"

cellspacing="value"

summary="text"

width="value"

[BACK](#)

<td> attributes

align="center/right/justify"
bgcolor="color"
colspan="value"
height="value"
rowspan="value"
valign="top/middle/bottom/baseline"
width="value"



Main points to remember:

- HTTP Protocol
- CGI Programming
- HTML

Thanks and See you in the Studio!