



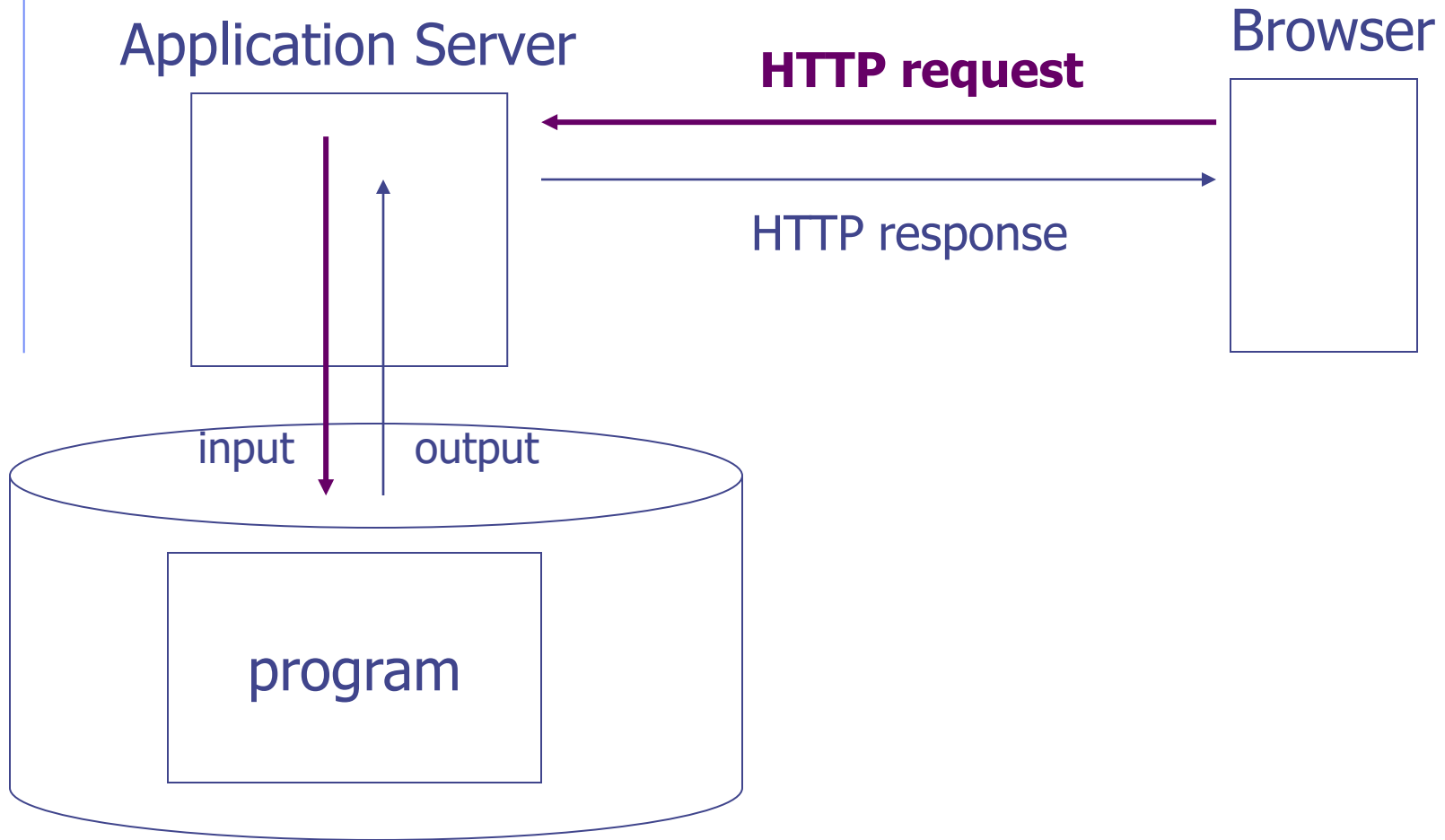
CS3220 Web and Internet Programming

Handling HTTP Requests

Chengyu Sun
California State University, Los Angeles



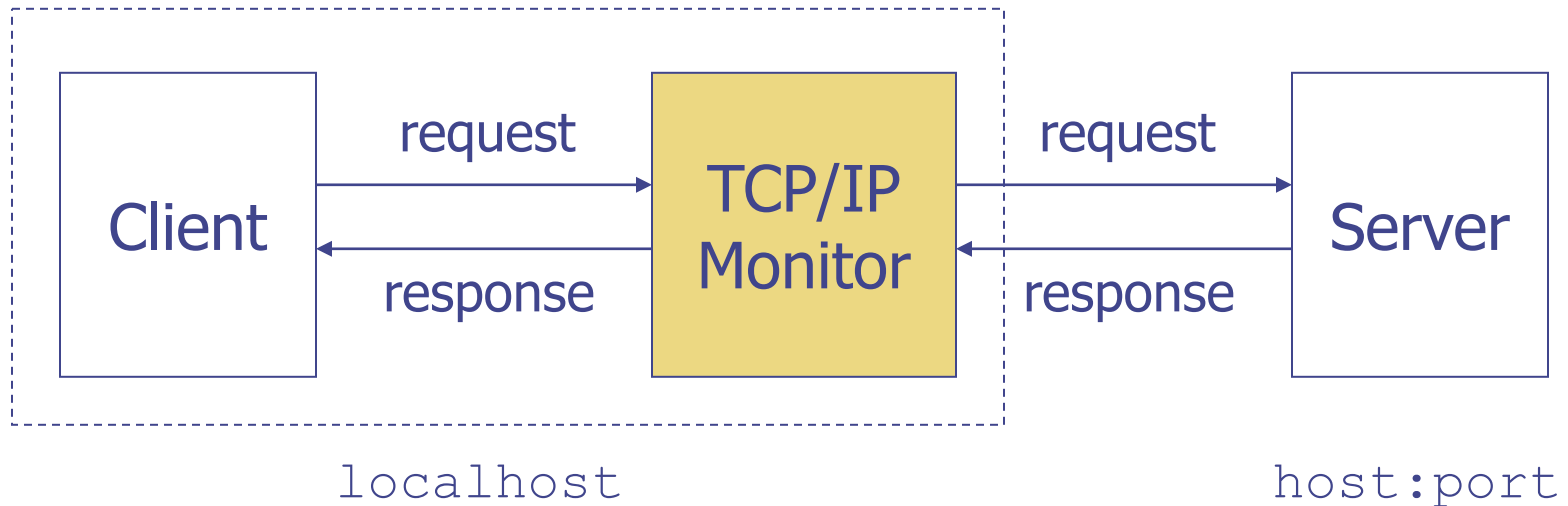
HTTP Request As Input



Examine HTTP Requests and Responses

- ◆ The Network tab in Developer Tools in Chrome/Edge or Firefox, or
- ◆ TCP/IP Monitor in Eclipse

TCP/IP Monitor in Eclipse ...



... TCP/IP Monitor in Eclipse

◆ Window → Preferences →
Run/Debug → TCP/IP Monitor

◆ Example: monitor the access of
<http://cs3.calstatela.edu/~cs3220stu31/test.html>

- *Local Monitoring Port?? Host?? Port??*
- *Browser URL??*

HTTP Request Example

http://cs3.calstatela.edu/foo.html

GET /foo.html HTTP/1.1

Host: cs3.calstatela.edu

User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.0; en-US; rv:1.7.3) ...

Accept: text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,...

Accept-Language: en-us,en;q=0.5

Accept-Encoding: gzip,deflate

Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7

Keep-Alive: 300

Connection: keep-alive

Cookie: nxt/gateway.dll/uid=4B4CF072; SITESERVER=ID=f1675...

HTTP Request

◆ Request line

- Method
- Request URI
- Protocol

◆ Header

◆ [Request body]

Request Methods

◆ Actions to be performed regarding the resource identified by the *Request URI*

◆ Browser

- GET
- POST

◆ Editor

- PUT
- DELETE

◆ Diagnosis

- HEAD
- OPTIONS
- TRACE

HttpServlet Methods

service()

◆ GET	➔	◆ doGet()
◆ POST	➔	◆ doPost()
◆ PUT	➔	◆ doPut()
◆ DELETE	➔	◆ doDelete()
◆ HEAD	➔	◆ doHead()
◆ OPTIONS	➔	◆ doOptions()
◆ TRACE	➔	◆ doTrace()

Override HttpServlet Methods

- ◆ Default implementation returns an HTTP Bad Request error
- ◆ doGet(), doPost()
 - If they do the same thing, simply call one method from the other

Request Header Fields

- ◆ User-Agent
- ◆ Accept
- ◆ Accept-Charset
- ◆ Accept-Encoding
- ◆ Accept-Language
- ◆ Host
- ◆ Connection/Keep-alive
- ◆ Content-Length
- ◆ Cookie
- ◆ Referer

RFC2616, Section 14

<http://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html>

HttpServletRequest

- ◆ getXxx() methods
- ◆ <http://download.oracle.com/javaee/7/api/javax/servlet/http/HttpServletRequest.html>

Example: RequestInfo

- ◆ Display some request information
 - Method, request URI, request URL ...
- ◆ Display the IP address of the client
- ◆ Determine if gzip compression supported

Request Parameters

- ◆ Query string

- *?param1=value1¶m2=value2&...*

- ◆ Form data

Example: Add

- ◆ Get two integers from request parameter and display the sum
 - Using query string
 - Using form: GET vs POST

Parameter Methods

◆ Inherited from *ServletRequest*

- `String getParameter(String p)`
- `Enumeration getParameterNames()`
- `String[] getParameterValues(String p)` –
useful for handling multiple parameters
with the same name but different values
(e.g. checkboxes)
- `Map getParameterMap()`

<form>

- ◆ **action** – URL of the server-side program that will process the request
- ◆ **method** – the method of request used to submit the form
 - **GET**: request parameters sent in query string
 - **POST**: request parameters sent in request body

Summary

◆ HTTP Request

- Request Line, Headers, Request Body
- Request Methods

◆ Request parameters

- Query string and form
- Handle request parameters in servlet