

LECTURE 18: WEB BASED APPLICATIONS: INTRODUCTION

OUTLINE

- Client/server architecture
 - Client/server applications
 - The Web applications
- Web pages, HTML and HTML forms
- Flask framework for web development

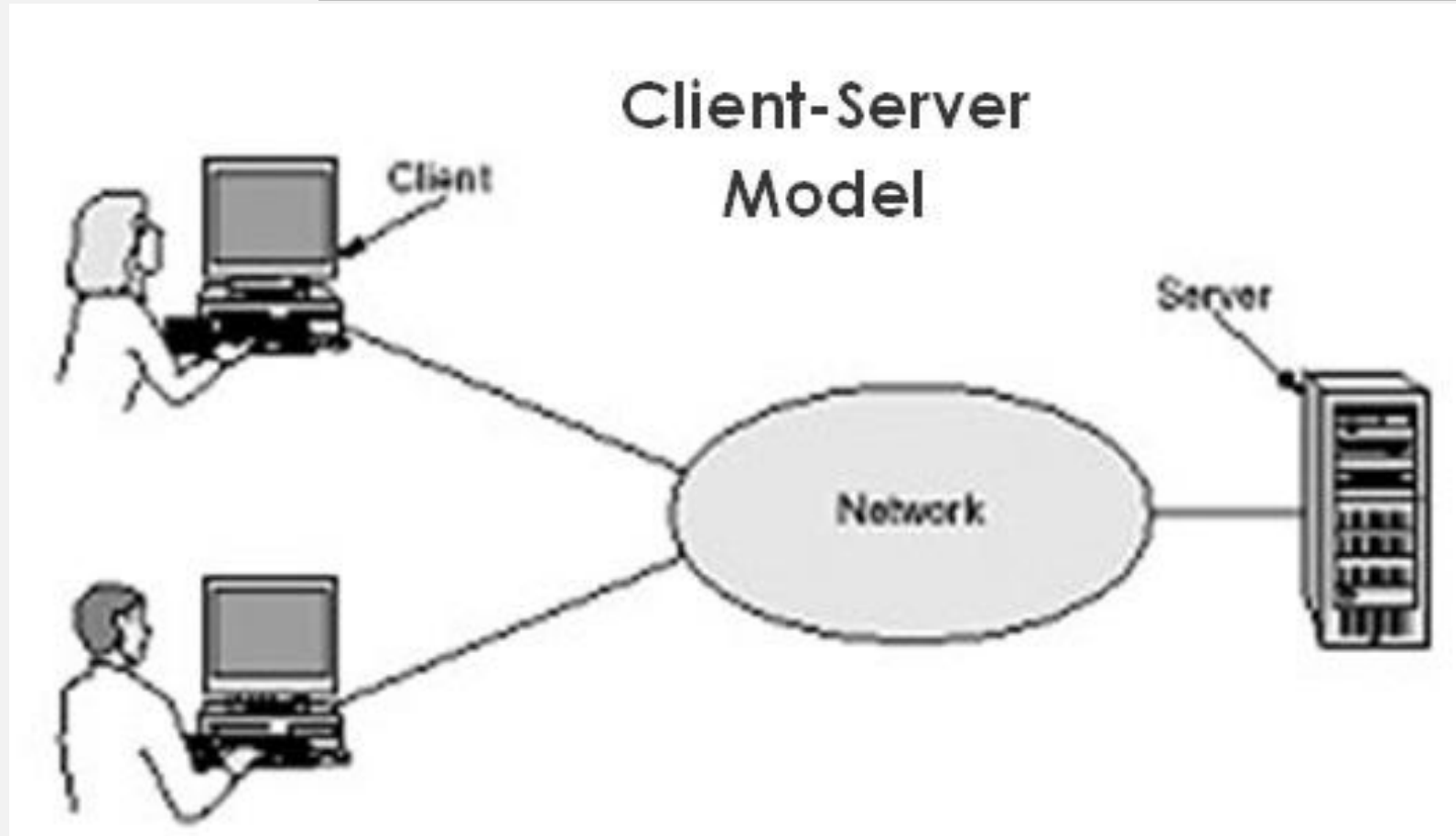
CLIENT/SERVER ARCHITECTURE

- End user runs a client program, which sends request for service, e.g. a URL, to a server
- Server bundles the response, e.g. a web page, and sends it back
- Client program displays the page

A CLIENT/SERVER APPLICATION

- Consists of several parts
 - Client program: the code that runs on the client machine; interacts with the user and the server
 - Server program: the code that runs on the server machine; accepts and serves client requests
 - Application protocol: types/order/format of the messages transferred between client and server programs
- Today's client/server applications commonly run on the internet

A CLIENT/SERVER APPLICATION



Bp2010.hprastiawan (https://commons.wikimedia.org/wiki/File:Client-Server_Model.jpg), „Client-Server Model“, <https://creativecommons.org/licenses/by-sa/3.0/legalcode>

THE WEB APPLICATION

- Web is a client/server Internet application
- A web browser serves as the client
 - Firefox, Safari, Chrome...
- A web server runs behind every active Web active
 - Most popular: Apache, Nginx, IIS
- The HTTP protocol is used to send messages between Web clients and the Web server
 - A web browser implements an HTTP client
 - A web server implements an HTTP server

WEB PAGES

- Simple text files stored on a Web server
- Are written using HTML commands that are read by the Web browser for formatting
 - HTML (HyperText Markup Language)
- See [simple.html](#)

HTML FORMATTING COMMANDS

- The commands are called tags
- Many in matched pairs: <html>, <head>, <title>, <body>, <h1>
- Many applications allow users to design pages without knowing these commands

FORM-BASED WEB PAGES

- A simple way to interact with webpages is HTML forms
- A form allows to put graphical components(e.g., text boxes, buttons) on a page for user input
- A form allows to specify a program that is executed on the Web server, to handle the values that user entered into the form
- See [form4eliza.html](#)

COMMON GATEWAY INTERFACE (CGI)

- A server-side script for dynamically generated webpages
- Defines a standard method for web servers to delegate the generation of webpages to executable files
 - Such executable files are known as CGI scripts. They are programs, usually written in a scripting language
- A CGI script is invoked by an HTTP server, usually to process user input submitted through an HTML form

INVOKING CGI SCRIPTS

- CGI scripts, rather than pre-written pages, are stored in a separate directory on the Web server, e.g., the cgi-bin directory
- If a Web browser requests a URL that points to a CGI script, then instead of sending that script file to the browser, the Web server runs the script and passes the output of the script to the Web browser
- CGI allows a browser to send information to the script via the URL or an HTTP POST request

HOW TO RUN FORM4ELIZA.HTML

- Copy form4eliza.html and cgi-bin directory to your desktop
- Proceed to Desktop(Using command line)
- Make eliza.py executable in Mac/Linux (`chmod u+x eliza.py`)
- run “`python -m CGIHTTPServer 18987`”
- Open a browser, and go to
<http://localhost:18987/form4eliza.html>

CGI PROGRAMMING

- Many web applications can be built using CGI
 - Blogs
 - Wiki
 - Shopping carts
- All have similar architecture as eliza, except the CGI script part is app specific
 - There is commonly a database behind the scenes

FLASK

- Flask is a micro web framework
- Supports cookies
- Extensions available
- Lots of 3rd party libraries
- Install: pip(or pip3) install flask

FLASK HELLO WORLD

```
from flask import Flask
app=Flask(__name__)
@app.route("/")
def hello():
    return"Hello World!"
if __name__=="__main__":
    app.run()
```

```
export FLASK_APP=hello.py flask run
```

Source: <http://flask.pocoo.org/>

URL BUILDING

See

`flask_url_building.py`

Source: <http://flask.pocoo.org/docs/1.0/quickstart/>

FIN!