

Web Application Development

Dr. Vivek Nallur
vivek.nallur@ucd.ie

University College Dublin

Class Logistics

Lecture: Tuesdays from 1330 to 1505 in Room 218
(TB4)

Labs: Thursdays from 0955 to 1130 in Room 414
(starts in week 3)

Class Logistics - II

Self-register on Moodle:

<https://csmoodle.ucd.ie/moodle/login/index.php>

Search for module code: 3019J

Look for “COMP 3019J - Web Application Development – 2019/20”

Use enrolment key: 2019-3019JVN

Grading

Exam → 40%

In-Class Graded Assignment → 30%

Personal Project → 30%

Attendance at labs is mandatory

Submit your own work before the deadline

Any submission after will not be considered.

What we are going to learn

- HyperText Markup Language (HTML) [5%]
- Cascading Style Sheets (CSS) [5%]
- JavaScript — client-side [10%]
- Server-side programming Using Python [10%]
- Database [5%]
- JQuery Framework [5%]

What we want to achieve

Build a simple web site that can present information efficiently and nicely.

- Select suitable web page style
- The web pages can respond to users' actions
- Information can be loaded from database(s)



Plagiarism & UCD Computer Science

- **Plagiarism is a serious academic offence**
 - [Student Code, sections 6.2 & 6.3] or [UCD Registry Plagiarism Policy] or [CS Plagiarism policy and procedures]
- Our staff and demonstrators are **proactive** in looking for possible plagiarism in all submitted work
- Suspected plagiarism is reported to the CS Plagiarism subcommittee for investigation
 - Usually includes an interview with student(s) involved
 - 1st offence: **usually** 0 or NG in the affected components
 - 2nd offence: may be referred to the **University disciplinary committee**
- Student who enables plagiarism is equally responsible

http://www.ucd.ie/registry/academicsecretariat/docs/plagiarism_po.pdf

http://www.ucd.ie/registry/academicsecretariat/docs/student_code.pdf

<http://libguides.ucd.ie/academicintegrity>

What is World Wide Web (WWW)

Internet = Web?

The answer is no

The World-Wide Web (WWW or simply the Web) is certainly what most people think of when they see the word “internet.”

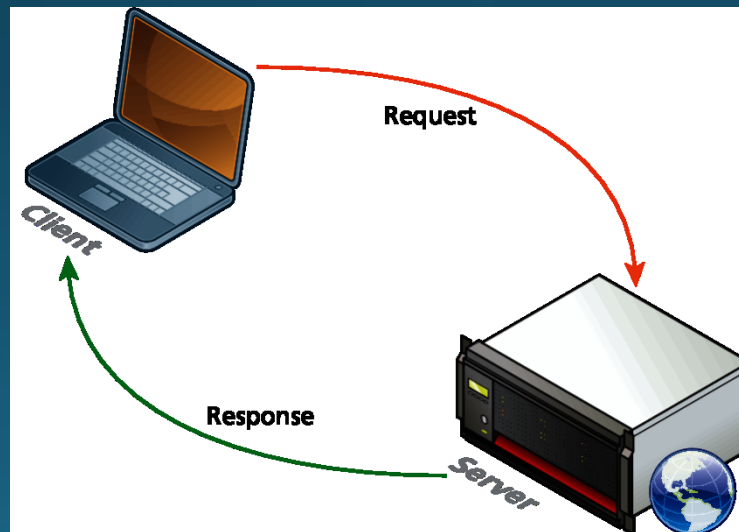
But the WWW is only a subset of the Internet.



Request-Response Loop

Within the client-server model, the **request-response loop** is the most basic mechanism on the server for receiving requests and transmitting data in response.

The client initiates a **request** to a server and gets a **response** that could include some resource like an HTML file, an image or some other data.



Server Types

A server is rarely just a single computer

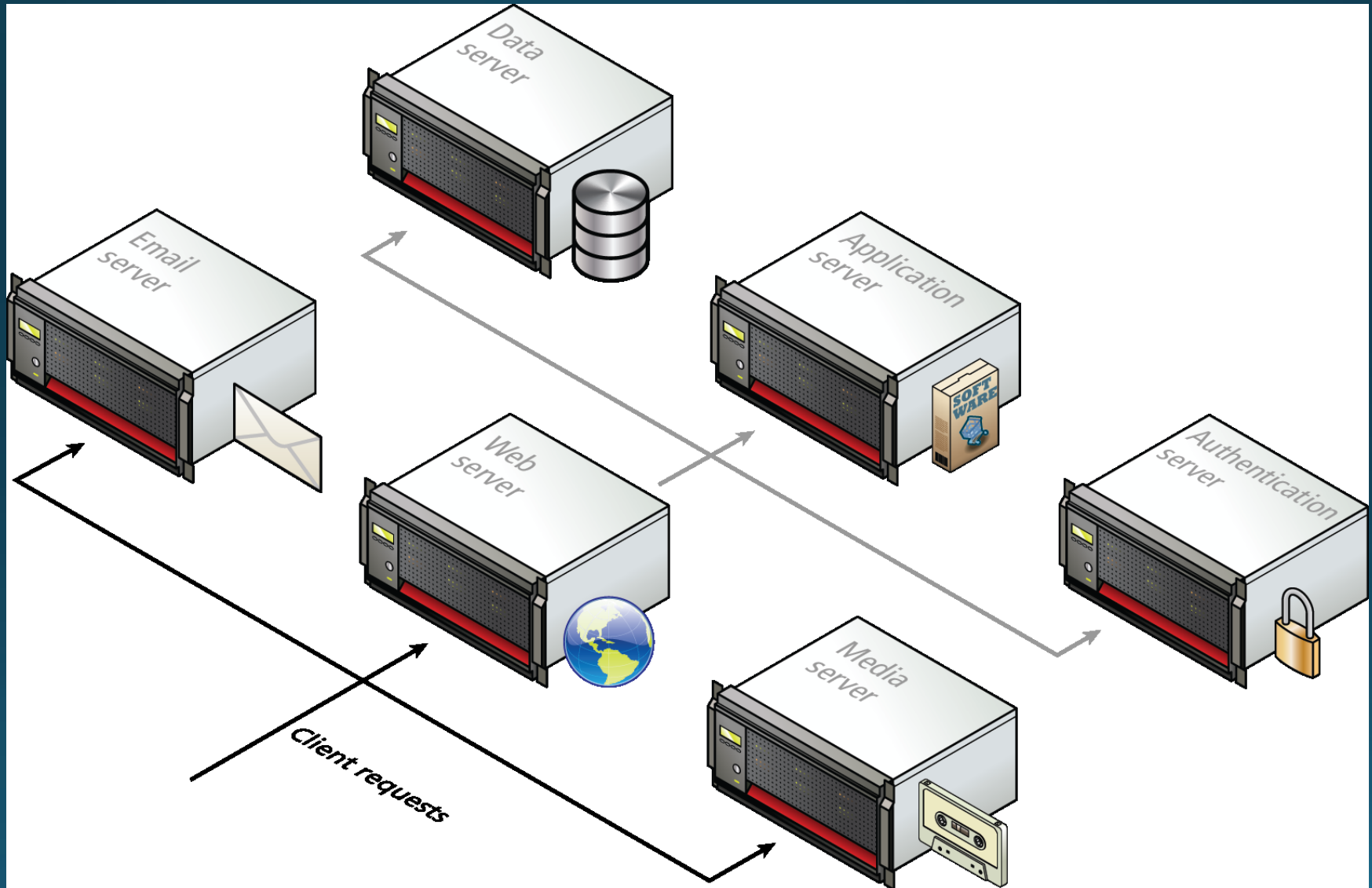
Earlier, the server was shown as a single machine, which is fine from a conceptual standpoint.

Clients make requests for resources from a URL; to the client, the server *is* a single machine.

However, most real-world web sites are typically not served from a single server machine, but by many servers.

It is common to split the functionality of a web site between several different types of server.

Server Types



Real-World Server Installations

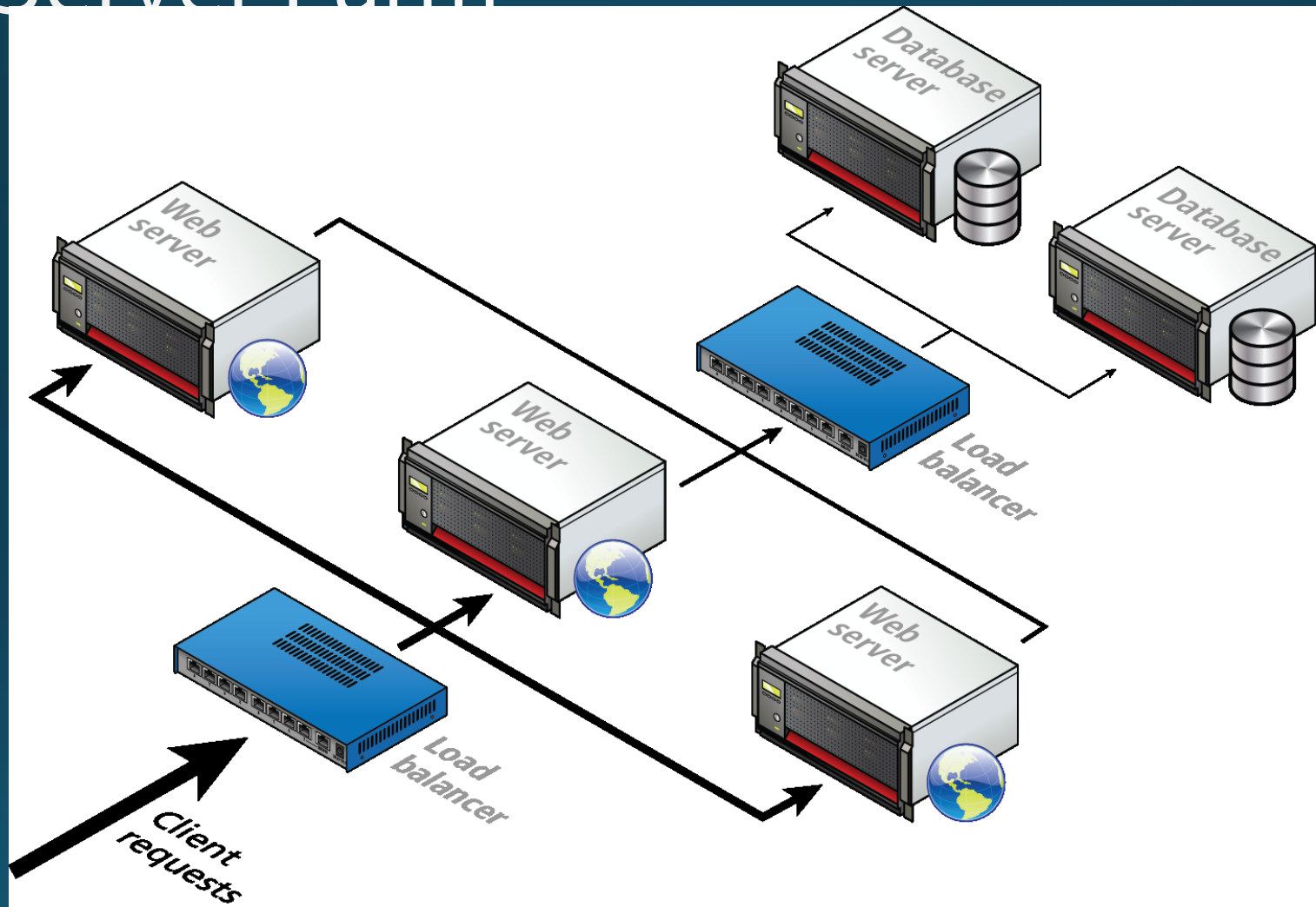
Not only are there different types of servers, there is often replication of each of the different server types.

A busy site can receive thousands or even tens of thousands of requests a second; popular sites such as Amazon receive millions of requests a second.

Server Farms

A single web server that is also acting as an application or database server will be hard-pressed to handle more than a few hundred requests a second, so the usual strategy for busier sites is to use a **server farm**.

Server Farm



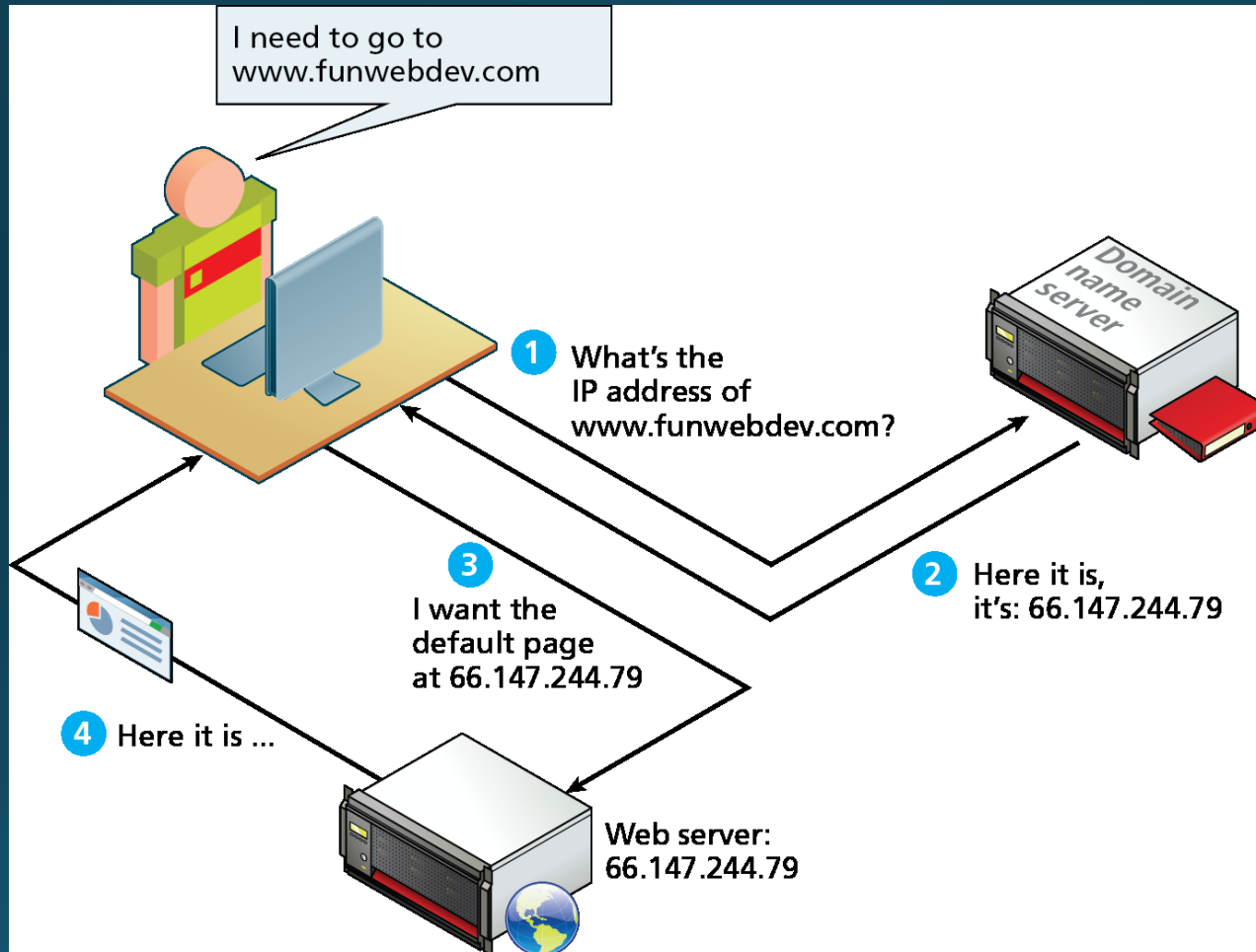
DOMAIN NAME SYSTEM (DNS)

Domain Name System

Why do we need it?

As elegant as IP addresses may be, human beings do not enjoy having to recall long strings of numbers. Instead of IP addresses, we use the **Domain Name System (DNS)**

DNS Overview



DNS Address Resolution

While domain names are certainly an easier way for users to reference a web site, eventually, your browser needs to know the IP address of the web site in order to request any resources from it.

The Domain Name System provides a mechanism for software to discover this numeric IP address.

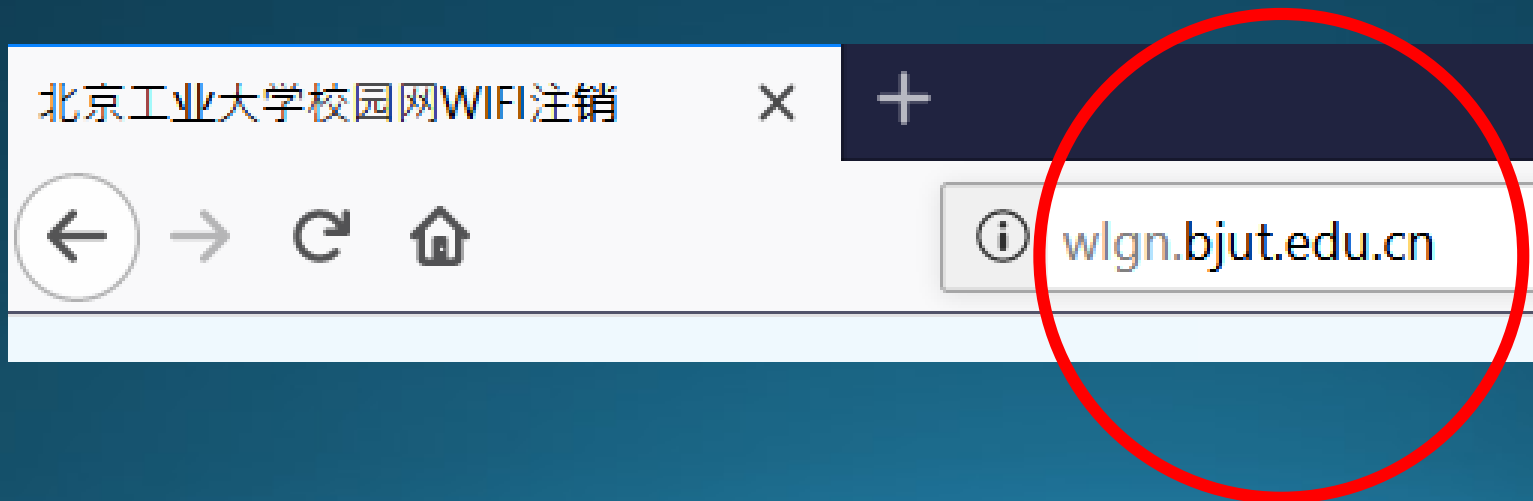
This process is referred to here as **address resolution**.

Uniform Resource Locators (URL)

URL Components

In order to allow clients to request particular resources from the server, a naming mechanism is required so that the client knows how to ask the server for the file.

For the web that naming mechanism is the **Uniform Resource Locator (URL)**.



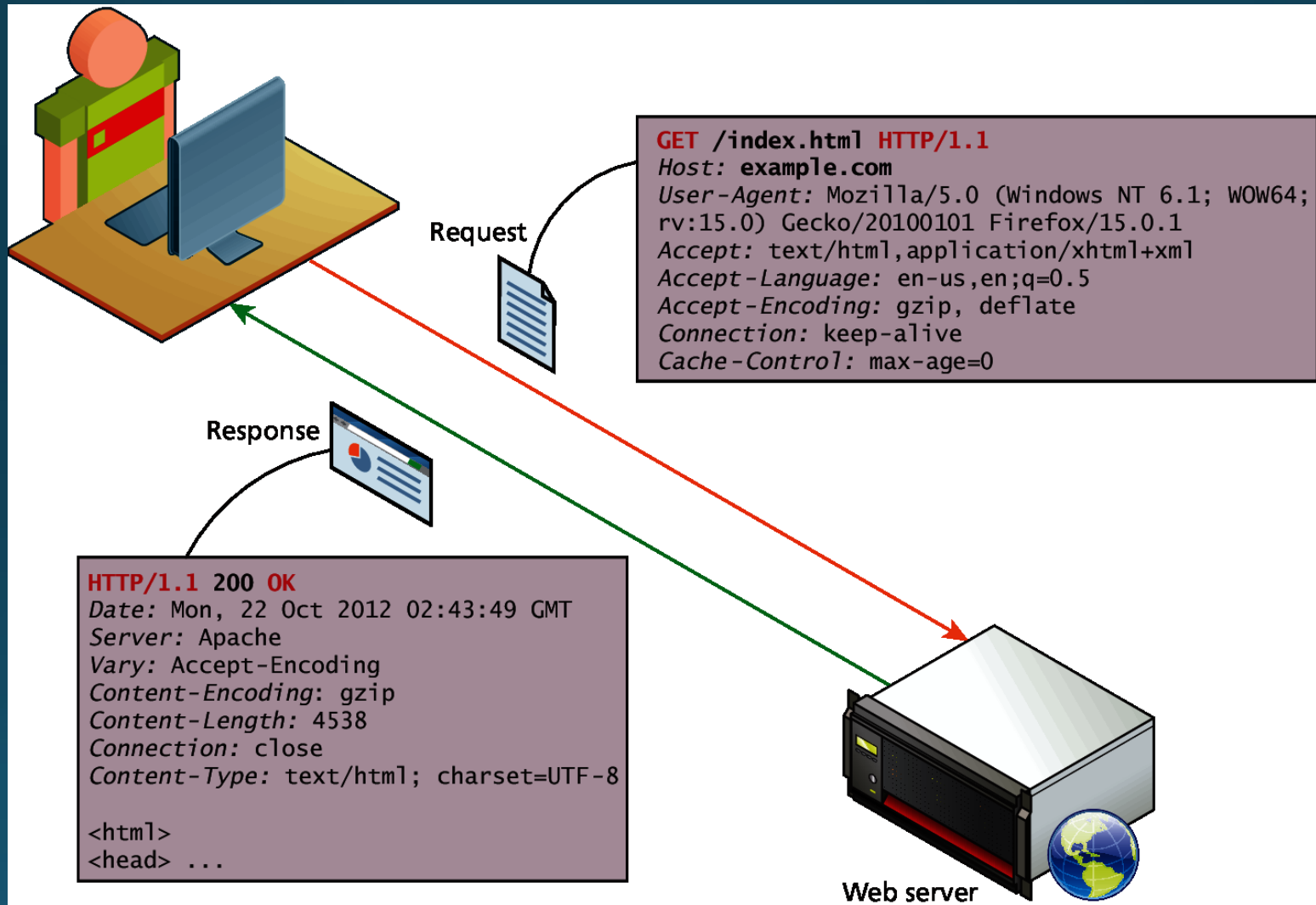
Hypertext Transfer Protocol (HTTP)

HTTP

The HTTP protocol establishes a TCP connection on port 80 (by default).

The server waits for the request, and then responds with a response code, headers and an optional message (which can include files).

HTTP



Web Requests

While we as web users might be tempted to think of an entire page being returned in a single HTTP response, this is not in fact what happens.

In reality the experience of seeing a single web page is facilitated by the client's browser which requests the initial HTML page, then parses the returned HTML to find all the resources referenced from within it, like images, style sheets and scripts.

Only when all the files have been retrieved is the page fully loaded for the user

What we will use

Browser – Firefox

Editor – Notepad++

Server-side Programming – Flask (Using Python)

Bring your laptop to class

This is important. This class will have a mixture of lecturing and coding!

HyperText Markup Language

HTML

HTML Syntax

What is a markup language?

HTML is defined as a **markup language**.

- A markup language is simply a way of annotating a document in such a way to make the annotations distinct from the text being annotated.
- The term comes from the days of print, when editors would write instructions on manuscript pages that might be revision instructions to the author or copy editor.

Markup

What is it again?

At its simplest, **markup** is a way to indicate *information about the content*

- This “information about content” in HTML is implemented via **tags** (aka elements).
- Markup examples...
- HTML does the same thing but uses textual tags.

What is the W3C?

Standards

The W3C is the main standards organization for the World Wide Web.

To promote compatibility the W3C produces **recommendations** (also called **specifications**).

In 2008, W3C announced the first draft of HTML5



HTML5

Three main aims

There are three main aims to HTML5:

- Specify unambiguously how browsers should deal with invalid markup.
- Provide an open, non-proprietary programming framework (via Javascript) for creating rich web applications.
- Be backwards compatible with the existing web.

HTML SYNTAX

Elements and Attributes

More syntax

HTML documents are composed of textual content and HTML elements.

An **HTML element** can contain text, other elements, or be empty. It is identified in the HTML document by tags.

HTML elements can also contain attributes. An **HTML attribute** is a name=value pair that provides more information about the HTML element.

In XHTML, attribute values had to be enclosed in quotes; in HTML5, the quotes are optional.

What HTML lets you do

- Insert images using the `` tag
- Create links with the `<a>` tag
- Create lists with the ``, `` and `` tags
- Create headings with `<H1>`, `<H2>`, ..., `<H6>`
- Define metadata with `<meta>` tag
- And much more...

Elements

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>This is a Heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

DOCTYPE



(short for **Document Type Definition**)

Tells the browser (or any other client software that is reading this HTML document) what type of document it is about to process.

Notice that it does not indicate what version of HTML is contained within the document: it only specifies that it contains HTML.

HTML, Head, and Body

HTML5 does not require the use of the `<html>`, `<head>`, and `<body>`.

However, we continue to use them for adding style and script information.

The `<html>` element is sometimes called the **root element** as it contains all the other HTML elements in the document.

Head and Body

HTML pages are divided into two sections: the **head** and the **body**, which correspond to the `<head>` and `<body>` elements.

The head contains descriptive elements *about* the document

The body contains content that will be displayed by the browser.

Inside the head

There are no brains

The `<head>` element can contain other elements inside it.

Some of these are *meta* tags (or tags that describe the document)

As the course progresses, you will add:

- `<style>` → to add CSS to your documents
- `<script>` → to add javascript to your documents

We'll stop for now

Questions?

Remember to:

- Register on Moodle
- Download Firefox and Notepad++
- Bring laptop to class!