PHP

Mohammad Homayoon Fayez

ZIBZAT

Introduction to Dynamic Web Content

- HTTP and HTML
- The Request/Response Procedure
- Using PHP, MySQL, JavaScript, CSS, and HTML5
- The Apache Web Server

- Questions

Introduction to Dynamic Web Content

HTTP and HTML

- CERN (European Laboratory for Particle Physics) was producing incredible amounts of data
- Problem: How to distribute this data to the participating scientists who were spread out across the world ?
- Tim Berners-Lee to the Rescue (Lee was Working at CERN)
  - Early 1990s: Lee created the World Wide Web
- World Wide Web is a constantly evolving network
- Tim Berners-Lee devised Hypertext Transfer Protocol, or HTTP for navigation between resources on computers connected to the Internet

Introduction to Dynamic Web Content

HTTP and HTML

- He also created
  - HTML (Hypertext Markup Language)
  - The first Web server (on Nextcube)
  - The first Web browser (WorldWideWeb)
  - The first http server software (CERN httpd)
  - The first web page
- Thus static pages with text, graphics and hyperlinks were born

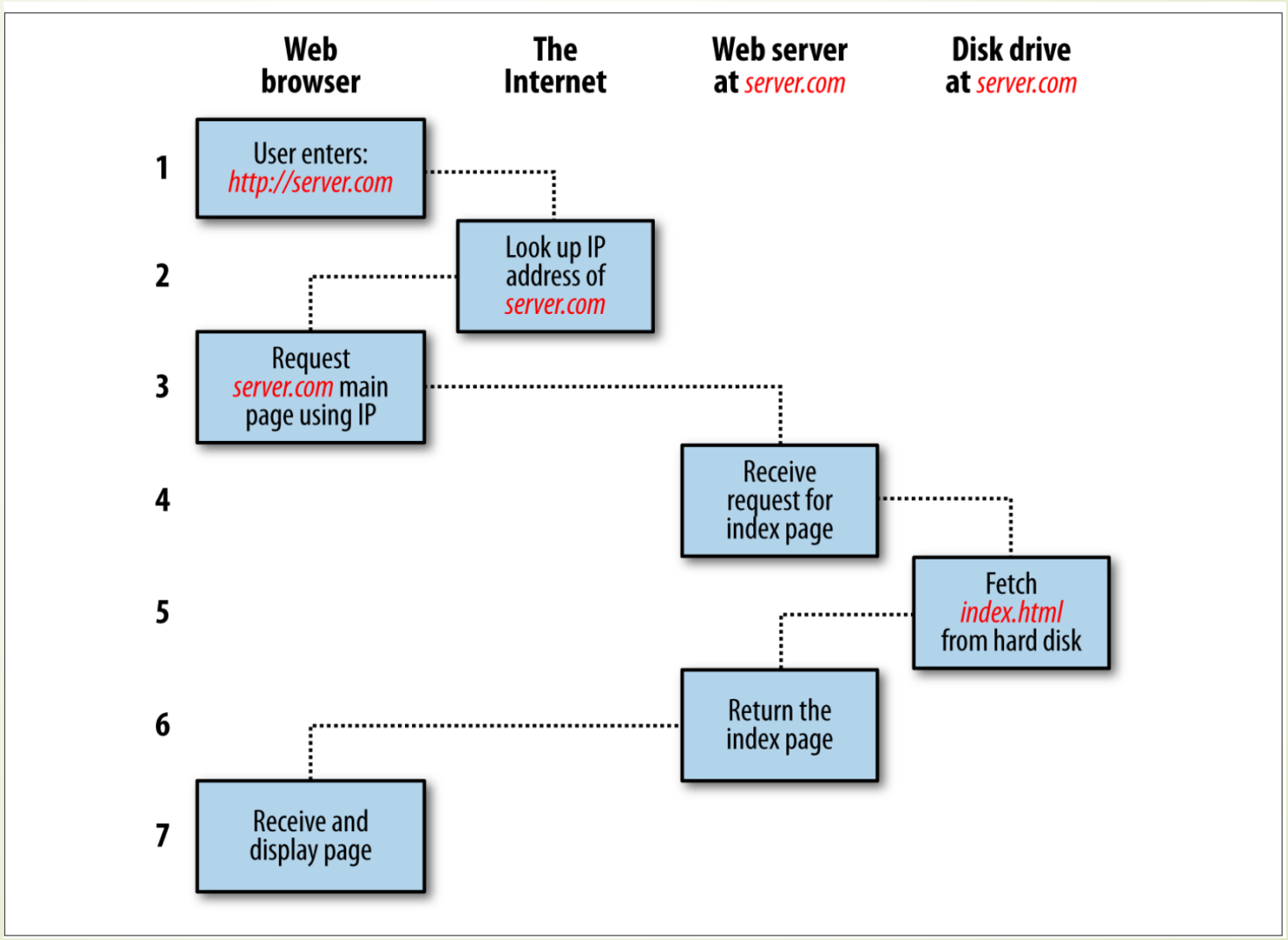
Introduction to Dynamic Web Content

HTTP

- A communication standard primarily used in client server environment
- It transfers request and response messages between the client and server
- The client could be typically a browser on the users machine or another interface e.g. a shell using another software e.g. curl to send http requests
- The server is the process that serves/responds back e.g. a web server
- A web server listens for an incoming connection and thus for a request, handles the request and sends a response back to the client
- A web server can handle multiple simultaneous connections

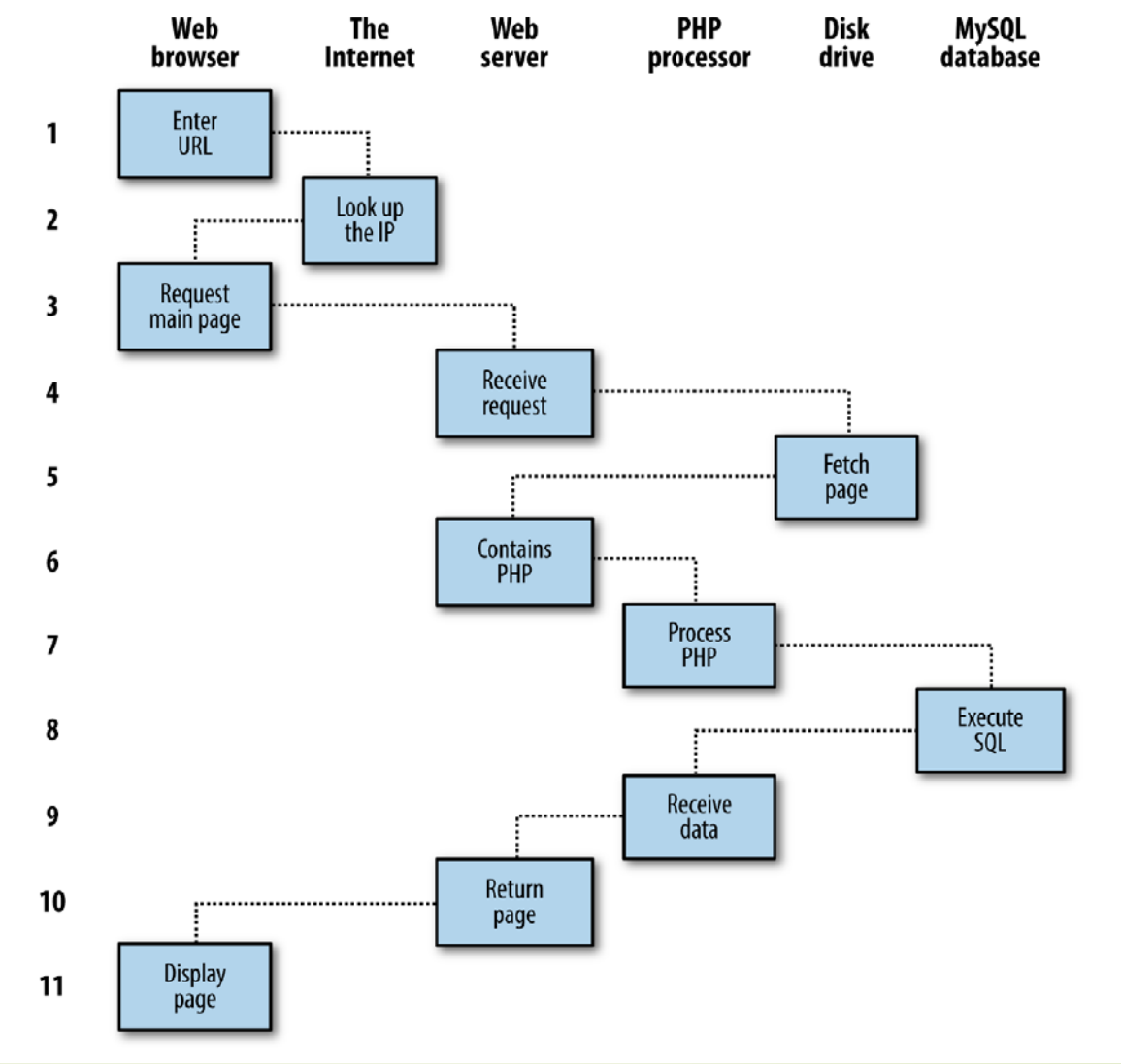
Request Response Process

for static contents



Request Response Process

for dynamic contents



Using PHP, MySQL, JavaScript, CSS, HTML5

- HTML (Heyper Text Markup Language): The markup language that the browsers can read and render into web pages
- CSS: Describes the presentation of a document written in a markup language
  - For the presentation of the web page
- JavaScript: A cross-platform, object-oriented scripting language
  - It runs inside the browser
  - can be connected to the DOM objects
  - Used, among other things, for dynamic user interactions.
- MySQL: Open source relational database management system
  - Manages all the data
- PHP: An open source preprocessor scripting language commonly used on web servers
  - Handles all the main work

They all work together to produce dynamic web content

The Apache Web Server

- Open source
- The web server, serves both the static and dynamic content
- The static content
  - HTML, text files, images, flash files, audio and video are just read from the disc and sent to the client
- The dynamic content
  - Dynamic generation of Text files, images, pdf files etc. need some script e.g. PHP
  - PHP runs as a module in Apache server
  - PHP has its own modules to generate e.g. images etc.

PHP

- A scripting language
- Open source
- Stands for PHP: Hypertext preprocessor
- Commonly used on web servers
- Can be embedded inside HTML
- Can
  - Dynamically generate web content
  - Interact with the database
  - Handle file
  - Send Emails
  - Network communication
  - Much more …

Resources

- www.php.net

PHP
Basics

- PHP commands can be embedded in HTML pages within

  <?php //php commands; ?>

  OR

  <? //php commands; ?>

- Each line of PHP code is terminated with a semi colon (;)
- Example

```
<?php
        echo "Hello world";
?>
```

OR

Within Html

```
<html>
<head></head>
  <body>
        <?php  echo "Hello world";  ?>
  </body>
</html>
```

PHP
Comments

- Single line comment

```
// This is a comment
$x += 10; // Increment $x by 10
```

- Multiline comment

```
/* This is a section
of multiline comments
which will not be
interpreted */
```

- Do not nest multiline comments

PHP

The $ symbol

- In PHP, however, you must place a $ in front of *all* variables.

```
<?php
    $mycounter = 1;
    $mystring = "Hello";
    $myarray = array("One", "Two", "Three");
?>
```

PHP

Variables

- String
  - strings can be enclosed inside single or double quotes

    “Hello” OR ‘Hello’

    $username = "Fred Smith";
- You can use the variable like this

  echo $username; //Prints Fred Smith
- Or assign it to another variable

  $current_user = $username;
- Variables are case sensitive
  - $variable differs from $Variable
  - Variable name must start with a letter or underscore

PHP
Variables

```
<?php // test1.php
  $username = "Fred Smith";
  echo $username;
  echo "<br>";
  $current_user = $username;
  echo $current_user;
?>
```

PHP
Constants

- Constants are defined without $ sign
- Constants are UPPERCASE by convention

```
<?php
    define (‘NAME’ , ‘Rose’);
    define (‘AGE’ , 23);
    echo NAME;
    echo ‘is’;
    echo AGE;
    // Rose is 23
?>
```

Single or double quotes

- Try this

```
<?php
        $name = ‘Rose’;
        $age = 23;
        echo ‘$name is $age’;
        echo “$name is $age”;
        //what it prints?
?>
```

PHP

- Numeric variables
- Numbers should not be enclosed in quotes

  $age = 30;
- OR a floating-point number (containing a decimal point)

  $count = 17.5;
- Arrays

  $team = array('Bill', 'Mary', 'Mike', 'Chris', 'Anne');
  - The elements of the array are numbered called index, the first element is assigned number 0
  - If we then wanted to know who player 4 is, we could use this command:

    echo $team[3]; // Displays the name Chris

Why HTML

- **HTML:** For describing the structure of hypertext and (partly) the layout.

```
<html>
    <head>
        <title>Hello</title>
    </head>
    <body>
        Hello World
    </body>
</html>
```

CSS
Cascading Style Sheet

- To apply style to your web pages
- CSS is connected to the DOM
- Due to its integration to DOM you can easily restyle any element e.g.
  - You can restyle/override the default font, size and color of heading tags
    - h1 { color:red; font-size:3em; font-family:Arial; }
- You can add style to a page By:
  - Adding it in the <head></head> tag
  - Importing a style sheet using @import directive
  - Importing a style sheet using html <link> tag
  - Using it directly in a html tag

CSS Examples

- Adding in the <head> tag

```
<!DOCTYPE html>
<html>
    <head>
        <title>Hello World</title>
        <style>
            h1 { color:red; font-size:3em; font-family:Arial; }
        </style>
    </head>
    <body>
        <h1>Hello there</h1>
    </body>
</html>
```

CSS Examples

- Importing a Style Sheet using @import directive

```
<style>
    @import url('styles.css');
</style>
```

- This statement tells the browser to fetch a style sheet with the name *styles.css*.
- With the @import you can create style sheets that themselves pull in other style sheets, and so on.

CSS Examples

- Importing CSS from Within HTML using <link> tag

```
<link rel='stylesheet' type='text/css' href='styles.css'>
```

- This has the exact same effect as the @import directive, except that <link> is an HTML-only tag and is not a valid style directive, so it cannot be used from within one style sheet to pull in another, and also cannot be placed within a pair of <style>...</style> tags.

CSS Examples

- inserting style declarations directly within HTML

  ```
  <div style='font-style:italic; color:blue;'>Hello there</div>
  ```

- But this should be reserved only for the most exceptional circumstances, as it breaks the separation of content and presentation.
- A better solution for setting the style of an element is to assign an ID to it in the HTML, like this:

  ```
  <div id='welcome'>Hello there</div>
  ```

- And then in your style sheet access the element with welcome id like this

  ```
  #welcome { font-style:italic; color:blue; }
  ```

CSS Examples

- Using Classes
  - If you would like to apply the same style to many elements, you do not have to give
  - each one a different ID because you can specify a class to manage them all, like this:

<div **class='welcome'** >Hello

  - This states that the contents of this element (and any others that use the class) should have applied to them the style defined in the welcome class. A class is defined by its name prefixed by a Dot (**.**)

.welcome { font-style:italic; color:blue; }

CSS Frameworks

- Makes your job easier
- Provides responsiveness and a ton of other features to your web page
- Among others the following tow are worth using
  - Foundation (http://foundation.zurb.com/)
  - Bootstrap (http://getbootstrap.com)

General HTML tags

- You should know about the following tags in HTML:
- Headings: <h1>. . .</h1>, . . . , <h4>. . .</h4>
- Rules: <hr />
- Paragraphs and line breaks: . . . ,
- Quotes: <blockquote>. . .</blockquote>
- Centering: . . .
- Bold text: . . .
- Italic text: . . .
- Underlined text: <u>. . .</u>
- Ordered lists: <ol>. . .
- Unordered lists: <ul>. . .
- List items: <li>. . .

General HTML tags

- Hyper links: name
- Local *named* hyper links: . . .
- References to a name: The Name
- Mail-to links: nh@itu.dk
- Tables: <table>. . .</table>, <tr>. . .</tr>, <th>. . .</th> and <td>. . .</td>
- Images:
- Colors: . . . and <body bgcolor="yellow">. . .</body>

Other HTML tags

- Visit http://www.w3schools.com/html/ and try
  - Form
  - Textarea
  - Select
  - Radio buttons
  - Input tags
  - And their relevant attributes