Introduction to HTML & PHP

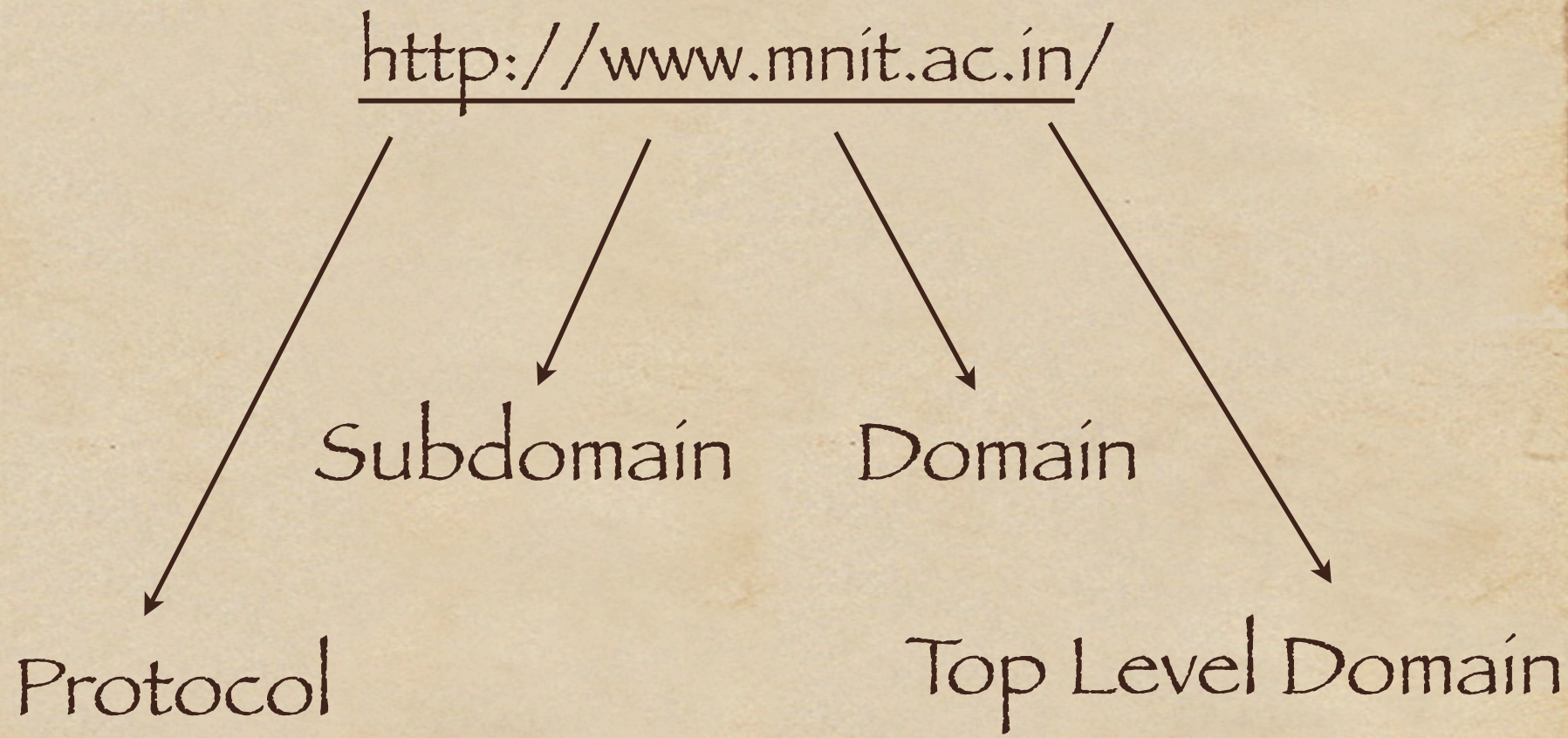
Anant Narayanan

17 August 2007

Overview of the WWW

- Client Server Architecture
- Any network always works in layers
- We work on the topmost "Application" layer
- The protocol used is HTTP. Another lower layer is at work: TCP/IP

Dissecting the URL



IP Addresses

- Every computer on a network is required to have an "address"
- Version 4 is the most popular and is of the form: a.b.c.d
- Combination of 4 numbers, each in the range 0-255

Distributing Addresses

- The scheme leaves us with a combination of 4,29,49,67,296 addresses
- There are certainly more computers than that!
- Not to mention, certain addresses are considered invalid

The DNS

- Domain Name System
- Converts domain names to IP Addresses
- Works hierarchically

The path of a web page

- User requests http://www.mnit.ac.in/
- Browser asks OS to "resolve"
- Browser connects to IP address
- Sends an HTTP "Request"
- Web Server responds with an HTTP "Response"

HTML

- Hypertext Markup Language
- Used to deliver content over HTTP
- Based on the superset XML
- The language used to design web pages

Example

```
<html>
  <head>
    <title>My First Web Page</title>
  </head>
  <body>
    <h1>Hello World!</h1>
  </body>
</html>
```

Tags & Attributes

- Everything enclosed in <> is known as a tag
- Every tag can have a set of associated attributes
- Learn the tags *yourself*!

`<a>, <p>, <br>, <form>, <input>, <table>, <div>, <textarea>, <span>, <h1>, <h2> etc...`

What's wrong with HTML

- It's static!
- Once you create an HTML page, it stays like that forever
- Although you can create forms in HTML, how do you actually process them?

Server side scripting

- Browsers understand only HTML
- Solution: Do the processing on the web server, but return HTML pages
- Different HTML pages are generated for different users

Hypertext Preprocessor

- PHP: Hypertext Preprocessor
- As the name suggests you do some processing and output HTML
- Sometimes also called a scripting language
- You can "submit" HTML forms to PHP scripts

A Sample Form

```
<html>
<head><title>Form</title></head>
<body>
 <form name="test" action="submit.php"
method="post">
  Please Enter your name:
  <input type="text" name="uname"/>
 </form>
</body>
</html>
```

submit.php

```
<?php
if (isset $_POST[‘uname’]) {
  $message = “Hello ”.
             $_POST[‘uname’].“!”;
} else {
  $message = “Hello Unnamed!”;
}
?>
<html>
 <head><title>Hello!</title></head>
 <body>
  <h1><?php echo $message;?></h1>
 </body>
</html>
```

PHP for Programming

- boolean, integer, float, string
- array, object
- resource, NULL

Booleans

- TRUE
- FALSE
- (Case-Insensitive)
- 0, 0.0, "", "0", array(), NULL are all FALSE
- Everything else is TRUE (including -1)

Integers

- Very Similar to C
- $a = 1234; $b = -123; $c = 0123; $d = 0x1A
- Any overflow will automatically become a float
- $x = 2147483647; // (int)
  $y = 2147483648; // (float)

Strings

- Single Quote, Double Quote, Heredoc
- You can include variables and escape sequences in double quoted strings
- $x = “Hello!”; echo “$x”; echo ‘$x’

Arrays

- array([key =>] value, ...)
- key may be an integer or string, value can be anything (including an array)
- $arr1 = (“Hello”, “World”); echo $arr1[0];
- $arr2 = array(“name” => “PHP”);
  echo $arr2[‘name’];

Objects

- Classes, Properties, Methods

Objects

- Classes, Properties, Methods

Oh wait, you're getting bored!
Let's leave this for another day

Questions?

Thank You!

http://foss.mnit.ac.in/