

TU

HTML5



# Web Overview

- Packet Switching
- IP addressing
- Routers

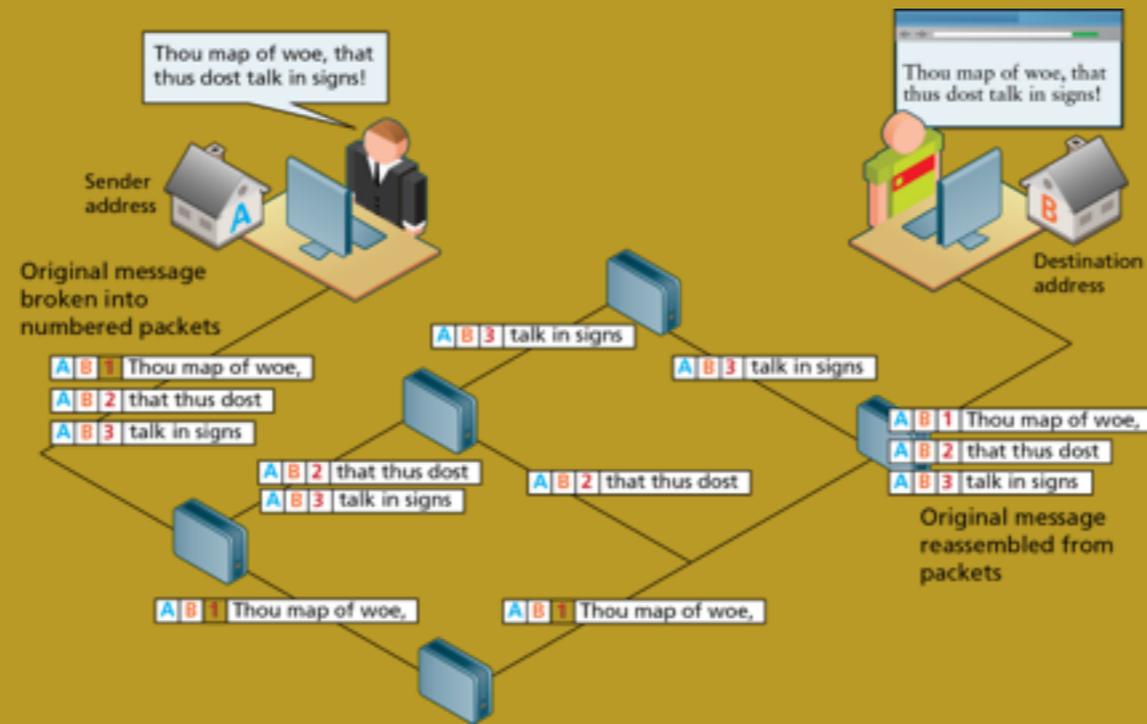
2

Information over the internet is sent/received using packets over tcp/ip. Each packet contains sufficient routing information to deliver the packet to its destination.

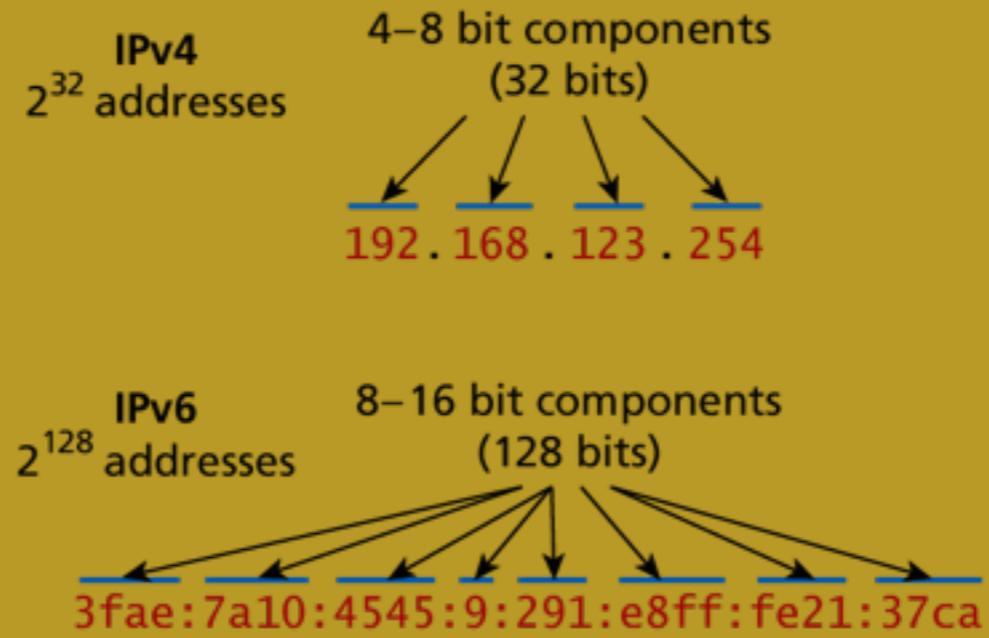
Each resource on the Internet is assigned a unique IP address. Currently ipv4 is the dominant, but ipv6 is up and coming.

Routers allow for the efficient delivery of packets over the internet. A packet may follow any path from source to destination. Which path is chosen depends on network conditions.

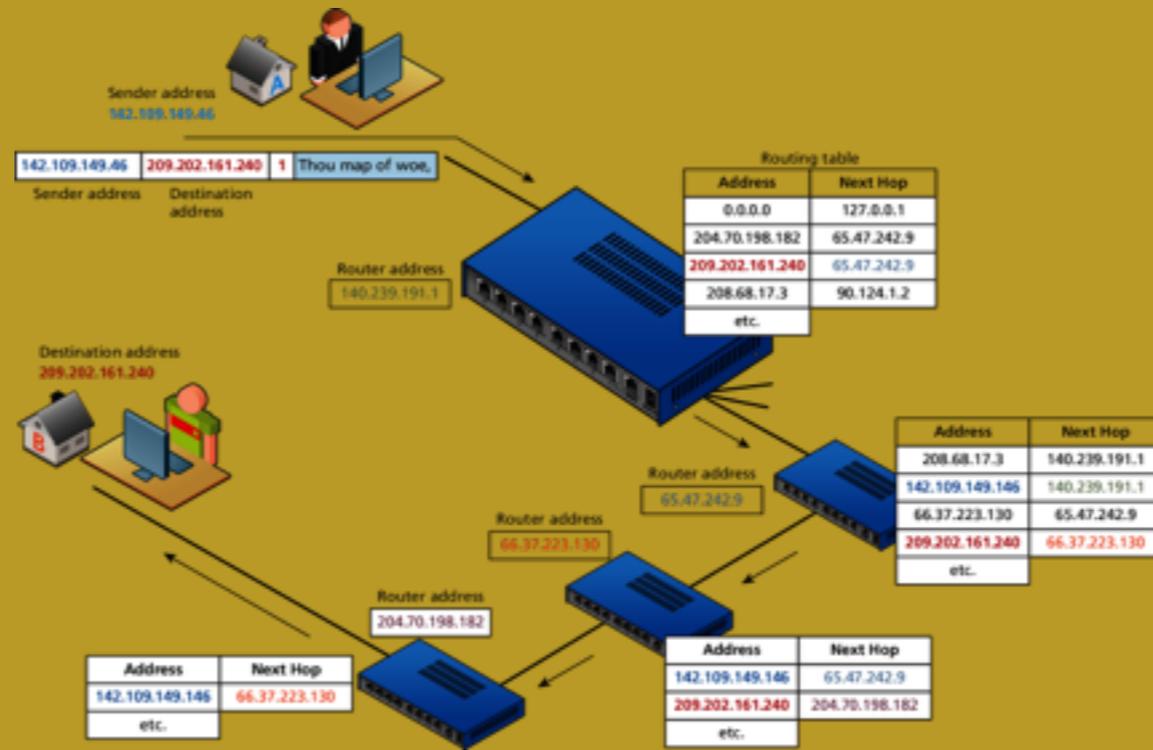
# Packet switching



# IP addressing



# Routers



# Domain Name System

- Domain Levels
- Domain Registration
- Address Resolution

6

Humans prefer text, computers numbers. IP addresses are computer friendly, but urls are human friendly.

To locate the ip address of a resource given its url, it must be translated by a DNS server. Each domain must be registered and assigned a unique ip address.

DNS server are spread throughout the internet and serve as domain to ip mappers. Each DNS server will have a table of currently know-about ip addresses.

# Domain levels

server1	4LD
.www	3LD
.funwebdev	2LD
.com	TLD

7

server1.www.funwebdev.com

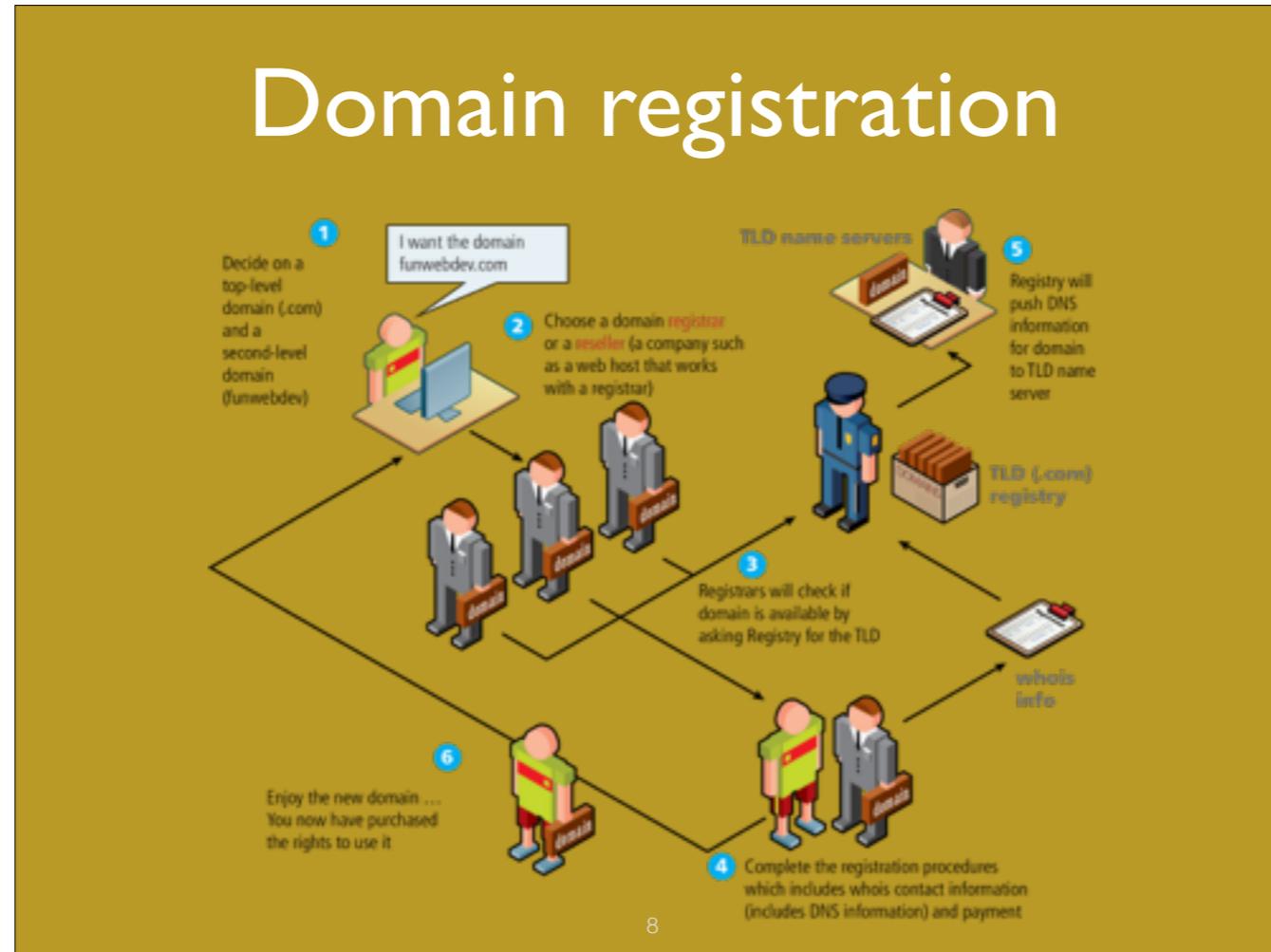
A url includes a domain that identifies the server hosting the requested resource. The components are as follows:

server1.www is the subdomain

funwebdev is the actual domain that is registered

.com is the top level domain

# Domain registration

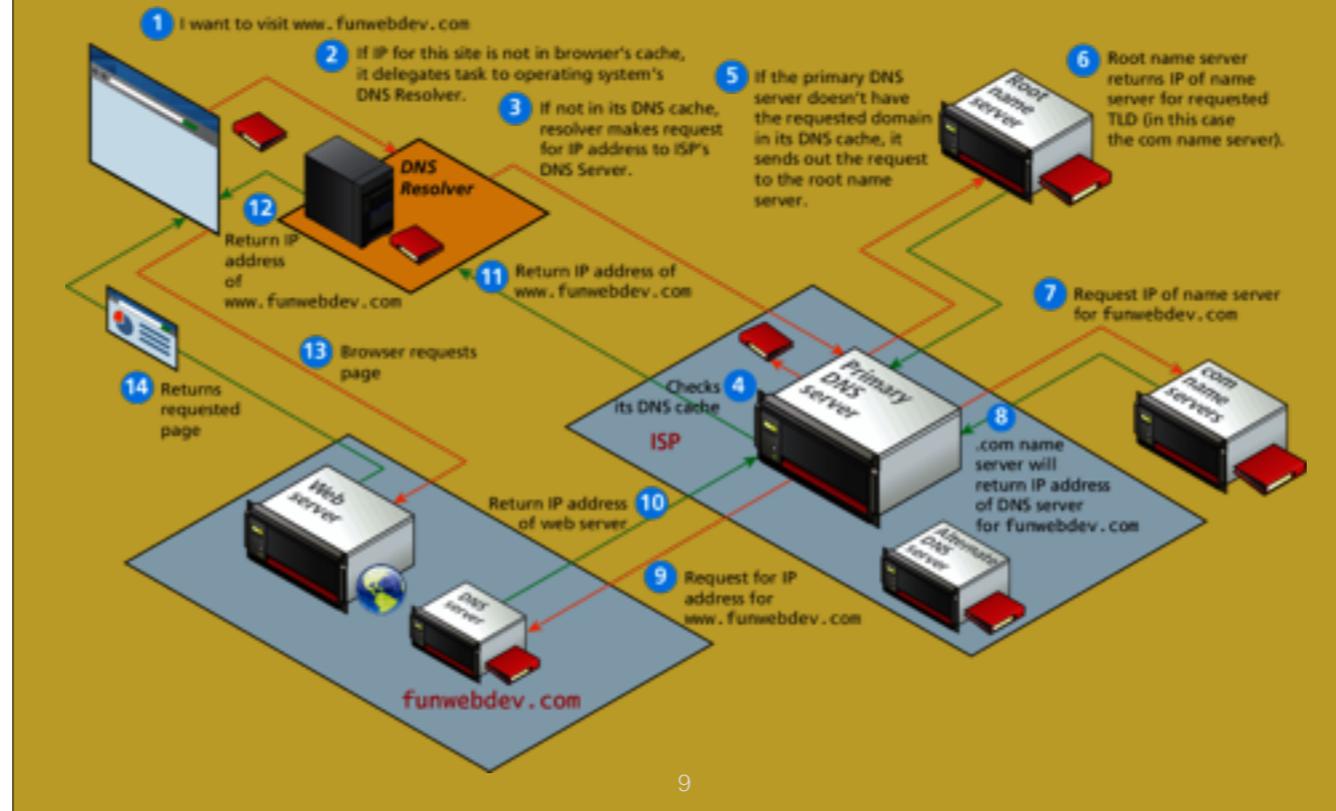


A domain is registered with an approved registrar (e.g. GoDaddy)

You opt the TLD that identifies the nature of your domain (.edu, .com, .net etc)

Your registered domain will now be available at the TLD name server

# Address resolution



A domain has to eventually map to its ip address, and this is where the DNS system comes in to play.

A series of DNS servers (alternate, primary, root etc) is checked until the one containing your domain is located. At that point its ip address is returned to the browser which in turn requests the page from the web server

# URL components

http://	protocol
<u>www.funwebdev.com</u>	domain
/index.php	resource path
?page=17	query string
#article	fragment

10

A url address consists of:

protocol - http, ftp, file, etc

path to resource

an optional query string to be used by server side script (e.g. php). This is a list of key=value pairs.

a location inside the document requested (fragment)

# HTTP

- GET v POST
- Query String

11

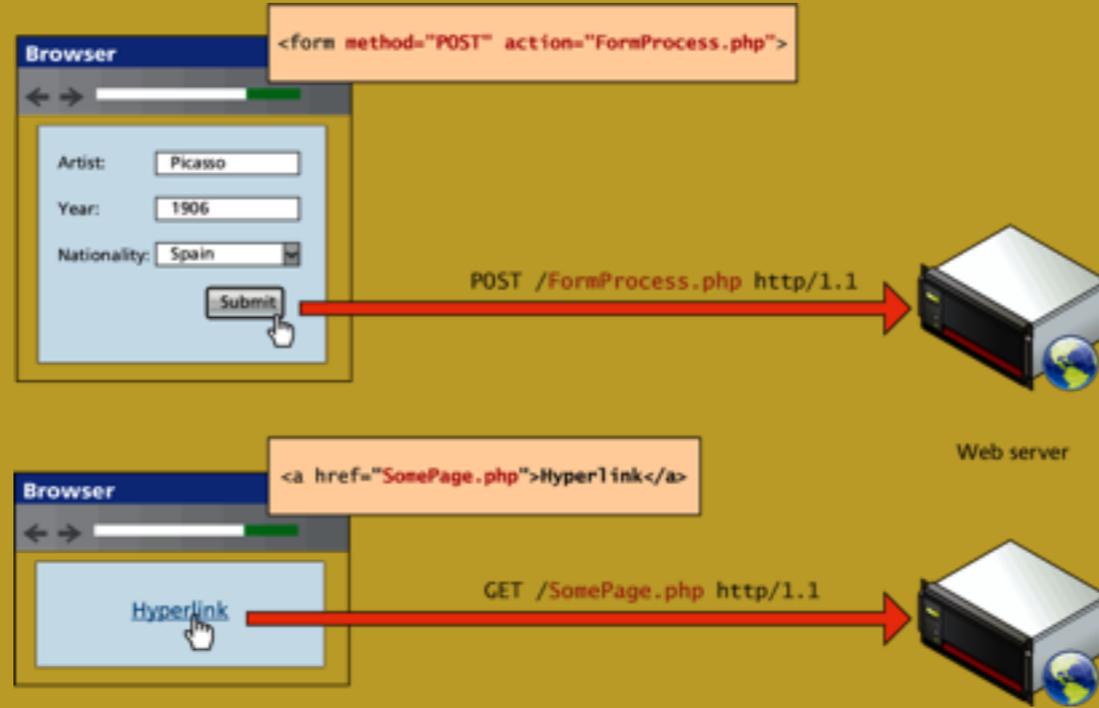
When requesting a page from the web server the browser has two options.

GET - this option requests a page and optionally provides data to the server through the url itself. Not as secure as the POST method and is limited by the length of the string that can be sent up.

POST - the preferred method of uploading data to the server. Used as a more secure method, since the data is bundled in the HTTP headers.

The query string that results is available for processing by the server-side script (php in our case) through PHP global arrays.

# Get v Post



# Query string

?	delimiter
username=john	key=value
&	delimiter
password=abcdef	key=value

# HTML5

- Structure
- Example

# Structure

```
<!DOCTYPE html>  
<html>
```

```
<head>
```

```
<title></title>
```

```
<meta>
```

```
<link>
```

```
</head>
```

```
<body>
```

```
<script></script>
```

```
</body>
```

```
</html>
```

```
<!-- File: add-content.html  
Author: Socratis Tornaritis -->
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Constructive & Co.</title>
```

```
<meta charset="utf-8">
```

```
<link rel="stylesheet" href="css/c01.css" />
```

```
</head>
```

```
<body>
```

```
<h1>Constructive & Co.</h1>
```

```
<p>For all orders and inquiries please  
call<em>555-3344</em></p>
```

```
<script src="js/add-content.js"></script>
```

```
</body>
```

```
</html>
```

# Block elements



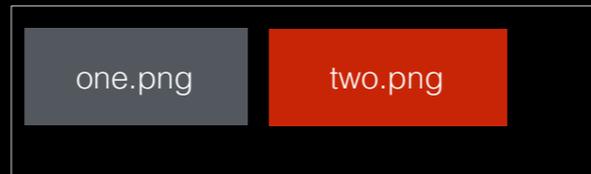
```
<div class="gray">  
  ...  
</div>  
  
<div class="red">  
  ...  
</div>
```

17

Block elements such as `<div>`, `<section>`, `<table>`, `<form>` etc begin with a newline. In other words, on a line by themselves. Using CSS you can make them appear next to each other on the same line, and we shall do that a bit later on.

They tend to fill (width wise) their containers, so usually a width can be specified to control how wide they are, especially if they are floated using CSS.

# Inline elements



```

```

```

```



elements are free to float based  
on container's width

Inline elements such as `<span>`, `<img>` etc, are floated automatically by the browser's layout algorithm. Adjusting the width of the window will force the elements to be repositioned if need be.

# Common elements

- `<article>`
  - `<section>`
  - `<aside>`
  - `<details>`
  - `<figure>` and `<figcaption>`
  - `<header>` and `<footer>`
  - `<nav>`
- `<form>`
  - `<table>`
  - `<div>`
  - `<img>`
  - `<a>`
  - `<ul>` and `<ol>`
  - `<span>`

These are just some of the common elements we may encounter along the way. Make sure you refer to online resources as well as our text book for a more in depth coverage of these elements.

```
<!DOCTYPE html><!-- html5-ulist.html -->
<html>
<head lang="en">
  <meta charset="UTF-8">
  <title>&lt;ul&gt; example</title>
</head>
<body>
  <ul>Unordered List
    <li>Item 1</li>
    <li>Item 2</li></ul>
  <p>Some text between the two lists</p>
  <ul>Unordered List
    <li>Item A</li>
    <li>Item B</li>
    <li>Item C</li></ul>
</body>
</html>
```

These are just some of the common elements we may encounter along the way. Make sure you refer to online resources as well as our text book for a more in depth coverage of these elements.

### Unordered List

- Item 1
- Item 2

Some text between the two lists

### Unordered List

- Item A
- Item B
- Item C

```
<!DOCTYPE html><!-- File: html5-table.html -->
<html>
<head lang="en">
  <meta charset="UTF-8">
</head>
<body>
  <table><caption>caption</caption>
  <thead><tr><th>th</th>
    <th>th</th></tr></thead>
  <tbody><tr><td>td</td>
    <td>td</td></tr>
    <tr><td>td</td>
    <td>td</td></tr></tbody>
  <tfoot><tr><td>tf</td>
    <td>tf</td></tr></tfoot></table>

</body>
</html>
```

These are just some of the common elements we may encounter along the way. Make sure you refer to online resources as well as our text book for a more in depth coverage of these elements.

caption

**th th**

td td

td td

tf tf

```
<!DOCTYPE html><!-- html5-form.html -->
<html>
<head lang="en"><meta charset="UTF-8"></head>
<body>
<form method="get" action="#">
  <fieldset><legend>legend</legend>
    <p><label for="first">First:</label>
      <input type="text" name="first"
        placeholder="Enter your first name">
    <br><label for="last">Last:</label>
    <input type="text" name="last"
      placeholder="Enter your last name"></p>
    <p><input type="checkbox" name="isStudent"
      checked="checked"> Student</p>
    <input type="submit" value="Sign Up">
  </fieldset></form>
</body></html>
```

24

These are just some of the common elements we may encounter along the way. Make sure you refer to online resources as well as our text book for a more in depth coverage of these elements.

html5-form.html?first=Soc&last=Tor&isStudent=on#

legend

First:

Last:

Student