

Programação Web

Introdução

Prof. Carlos Bazilio

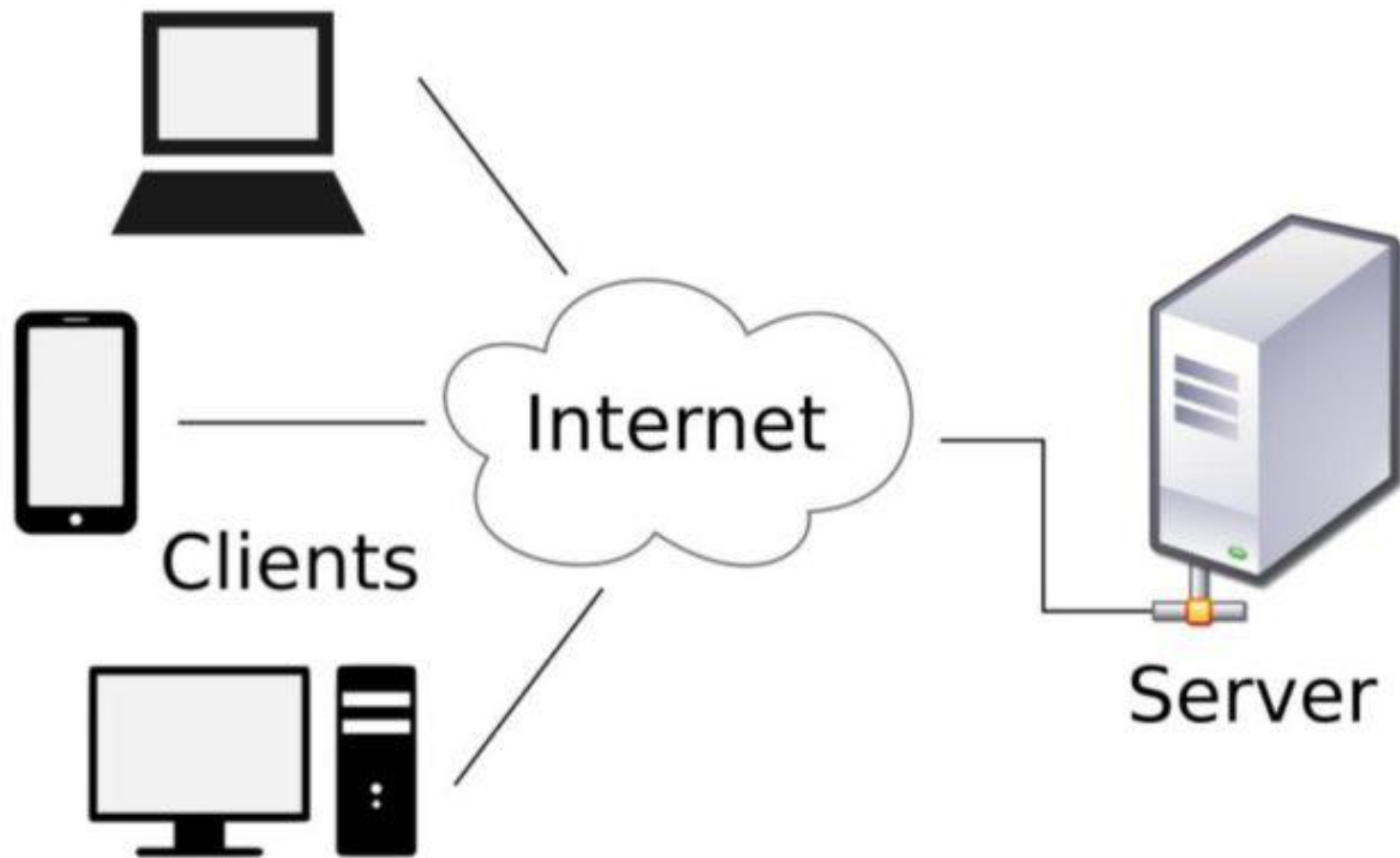
carlosbazilio@id.uff.br

<http://carlosbazilio.github.io>

Lição para Profissional de Computação

“Prostituam-se! Prostituam-se quando o assunto for linguagem de programação, IDE, metodologia, arquitetura, ou qualquer outro tipo de escolha técnica!”

Arquitetura



O Começo

← → ↺ 🏠

🔒 https://en.wikipedia.org/wiki/Tim_Berners-Lee 133% 📄 📁 🌐 🌐 🌐

Current events
Random article
Donate to Wikipedia
Wikipedia store

Interaction

Help
About Wikipedia
Community portal
Recent changes
Contact page

Tools

What links here
Related changes
Upload file
Special pages
Permanent link
Page information
Wikidata item
Cite this page

In other projects

Wikimedia Commons
Wikiquote
Wikisource

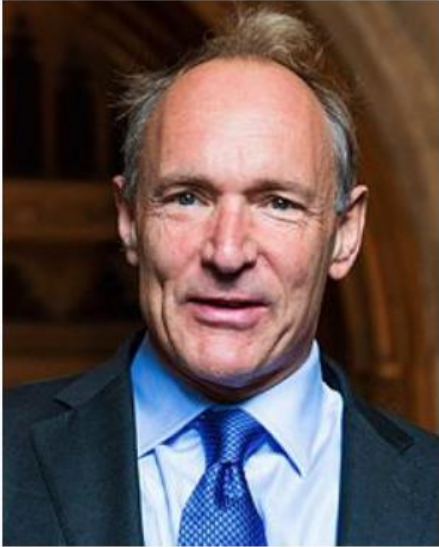
Print/export

Sir Timothy John Berners-Lee OM KBE FRS FREng FRSA FBCS (born 8 June 1955),^[1] also known as **TimBL**, is an English **engineer** and **computer scientist**, best known as the inventor of the **World Wide Web**. He is currently a professor of computer science at the **University of Oxford** and the **Massachusetts Institute of Technology** (MIT).^{[2][3]} He made a proposal for an information management system on 12 March 1989,^{[4][5]} and he implemented the first successful communication between a **Hypertext Transfer Protocol** (HTTP) client and **server** via the **internet** in mid-November the same year.^{[6][7][8][9][10]}

Berners-Lee is the director of the **World Wide Web Consortium** (W3C), which oversees the continued development of the Web. He is also the founder of the **World Wide Web Foundation** and is a senior researcher and holder of the 3Com founders chair at the MIT Computer Science and Artificial Intelligence Laboratory (CSAIL).^[11] He is a director of the **Web Science Research Initiative** (WSRI),^[12] and a member of the advisory board of the **MIT Center for Collective Intelligence**.^{[13][14]} In 2011, he was named as a member of the board of trustees of the **Ford Foundation**.^[15] He is a founder and president of the **Open Data Institute**, and is currently an advisor at **social network MeWe**.^[16]

In 2004, Berners-Lee was **knighted** by Queen **Elizabeth II** for his pioneering work.^{[17][18]} In April 2009, he was elected a **Foreign Associate of the National Academy of Sciences**.^{[19][20]} Named in *Time* magazine's list of the **100 Most**

Sir Tim Berners-Lee
OM KBE FRS FREng FRSA FBCS



Berners-Lee in 2014

Born	Timothy John Berners-Lee 8 June 1955 (age 64) ^[1] London, England, United Kingdom
Other names	TimBL TBL
Education	Emanuel School
Alma mater	University of Oxford (BA)

O Começo



Cronologia

<http://www.evolutionoftheweb.com/#/evolution/night>

<http://www.webfoundation.org/vision/history-of-the-web/>

http://en.wikipedia.org/wiki/World_Wide_Web#History

<http://webdirections.org/history/#0>

<http://www.w3.org/History.html>

Alguns Dados Estatísticos

WORLD INTERNET USAGE AND POPULATION STATISTICS MAY, 2019 - Updated

World Regions	Population (2019 Est.)	Population % of World	Internet Users 31 Mar 2019	Penetration Rate (% Pop.)	Growth 2000-2019	Internet Users %
Africa	1,320,038,716	17.1 %	492,762,185	37.3 %	10,815 %	11.2 %
Asia	4,241,972,790	55.0 %	2,197,444,783	51.8 %	1,822 %	50.1 %
Europe	829,173,007	10.7 %	719,365,521	86.8 %	584 %	16.4 %
Latin America / Caribbean	658,345,826	8.5 %	444,493,379	67.5 %	2,360 %	10.1 %
Middle East	258,356,867	3.3 %	173,542,069	67.2 %	5,183 %	4.0 %
North America	366,496,802	4.7 %	327,568,127	89.4 %	203 %	7.5 %
Oceania / Australia	41,839,201	0.5 %	28,634,278	68.4 %	276 %	0.7 %
WORLD TOTAL	7,716,223,209	100.0 %	4,383,810,342	56.8 %	1,114 %	100.0 %

NOTES: (1) Internet Usage and World Population Statistics estimates for May 9, 2019. (2) CLICK on each world region name for detailed regional usage information. (3) Demographic (Population) numbers are based on data from the [United Nations Population Division](#). (4) Internet usage information comes from data published by [Nielsen Online](#), by the [International Telecommunications Union](#), by [GfK](#), by local ICT Regulators and other reliable sources. (5) For definitions, navigation help and disclaimers, please refer to the [Website Surfing Guide](#). (6) The information from this website may be cited, giving the due credit and placing a link back to www.internetworldstats.com. Copyright © 2019, Miniwatts Marketing Group. All rights reserved worldwide.

2021 This Is What Happens In An Internet Minute



Internet

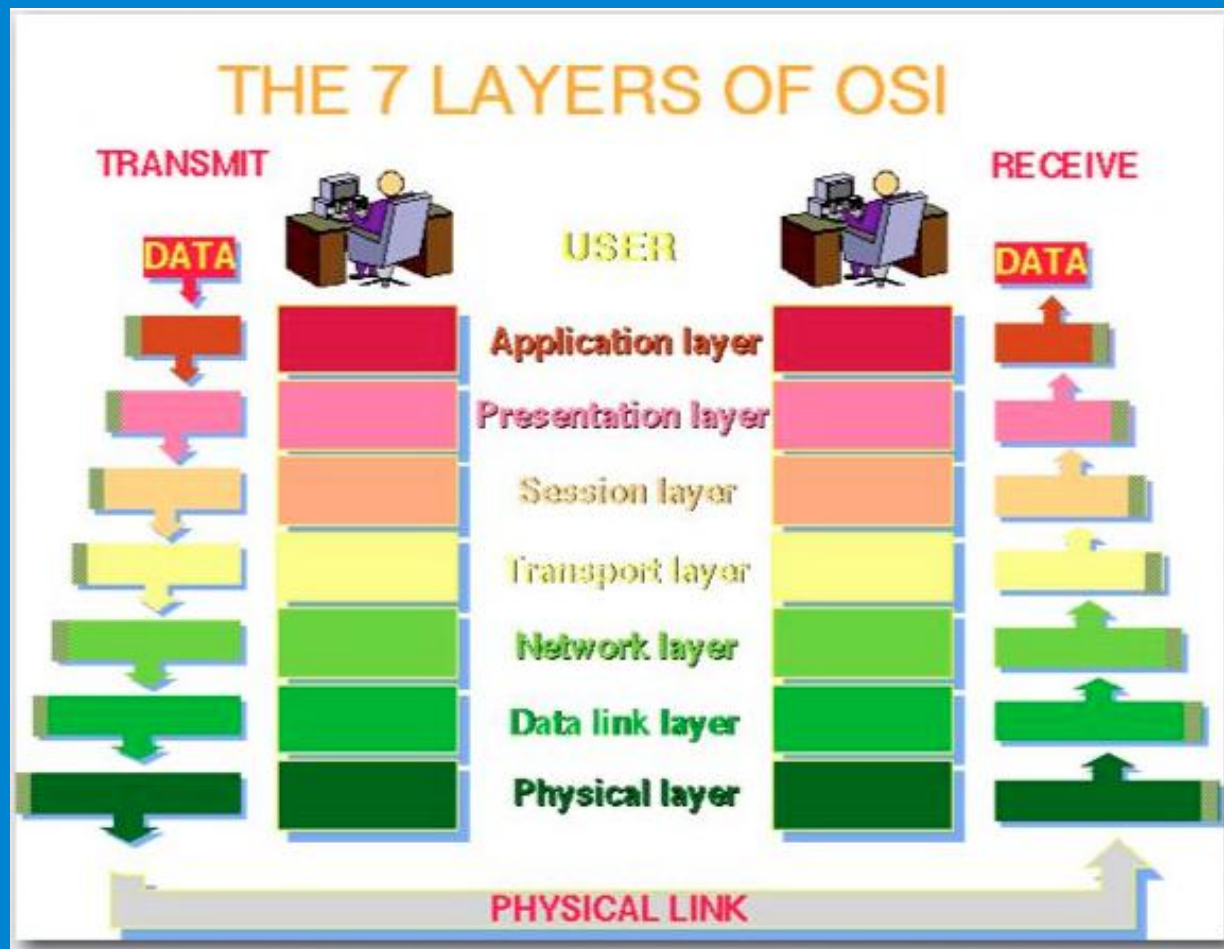
- É um sistema global de redes de computadores interconectados
- www, e-mail, redes sociais, ftp, chat, comércio, jogos online, teleconferência, VoIP, vídeo sob demanda, ...
- O que permite que empresas e clientes diferentes consigam interagir: a existência de **Protocolos**

Protocolos

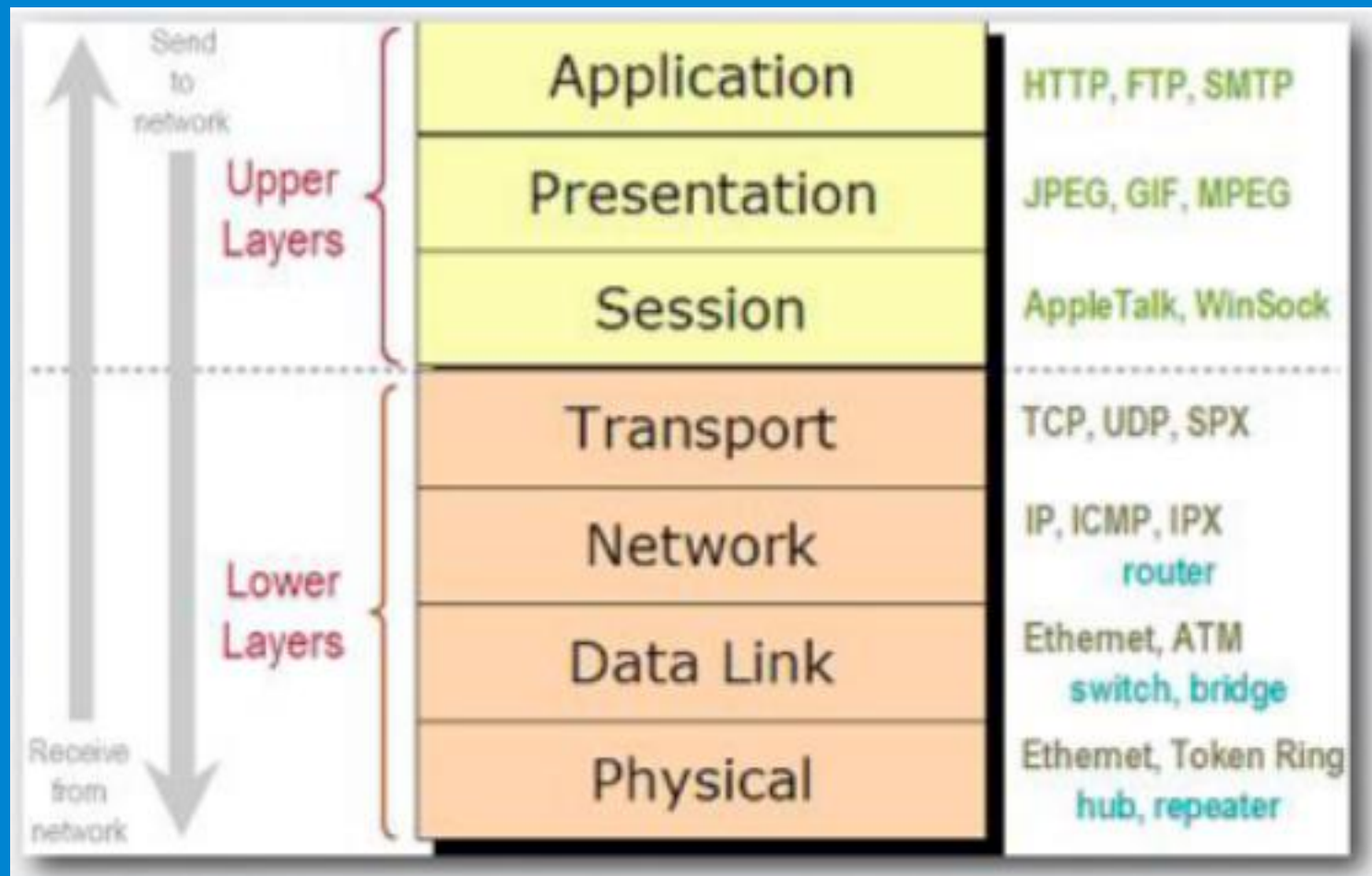
- Protocolo é uma forma de etiqueta
- Protocolos de Internet são padrões que especificam como computadores intergam e trocam mensagens na internet
- Protocolos usualmente definem:
 - O formato das mensagens
 - Como erros são tratados



Modelo OSI



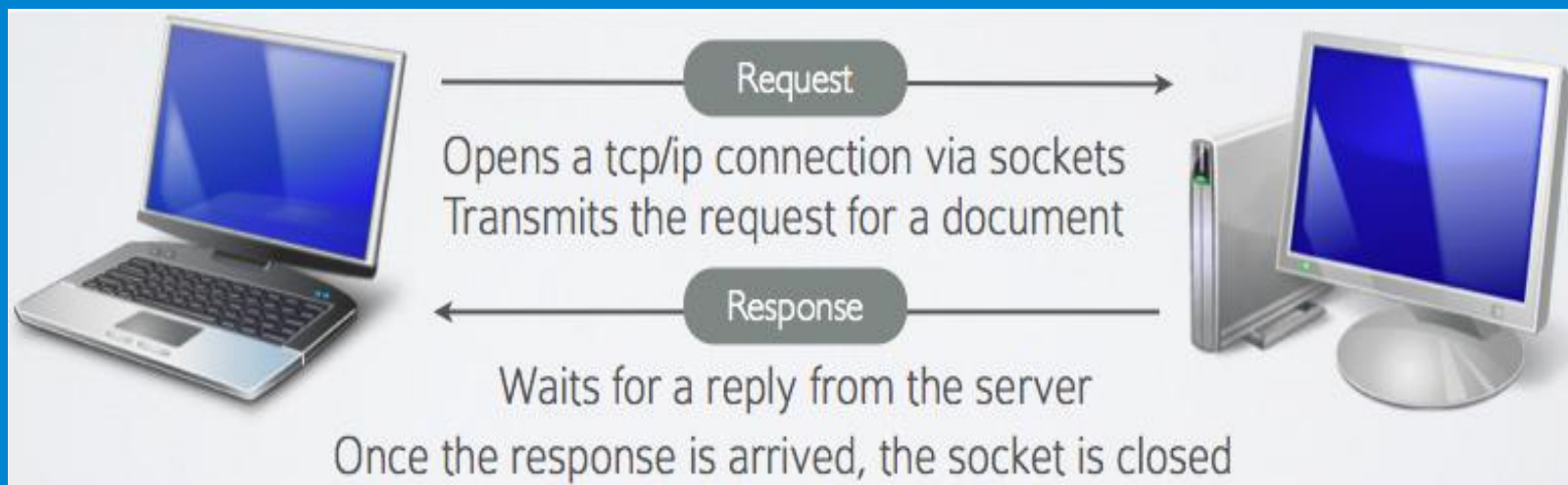
Modelo TCP/IP



<http://tools.ietf.org/html/rfc1180>

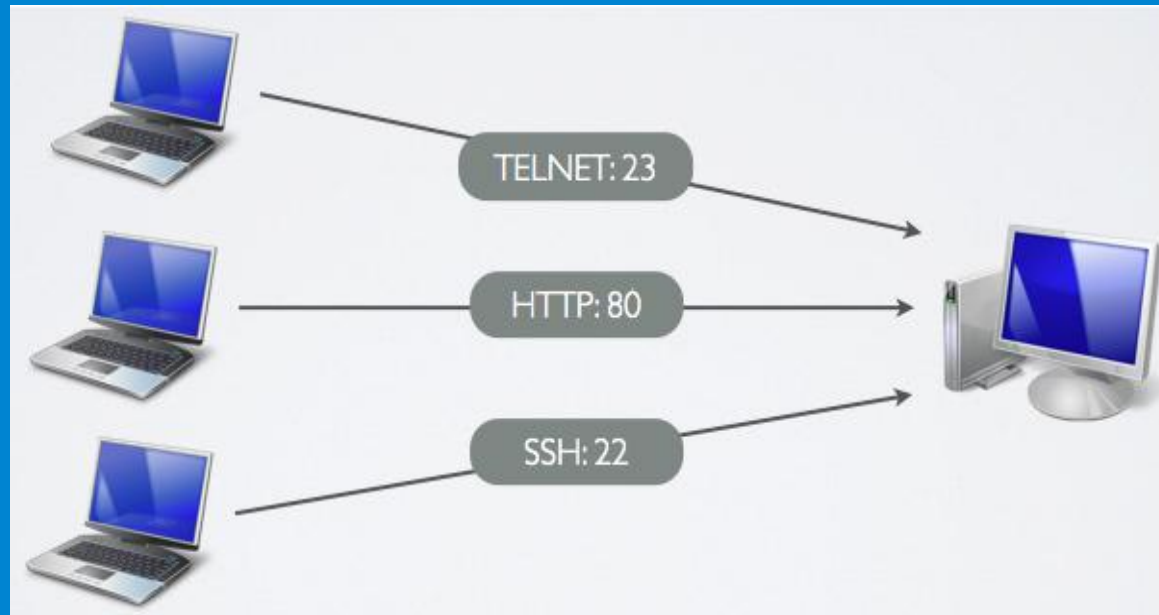
HTTP

- Significa Hypertext Transfer Protocol
- Especifica uma série de padrões que governam como informações são transmitidas na web
- É um protocolo cliente-servidor



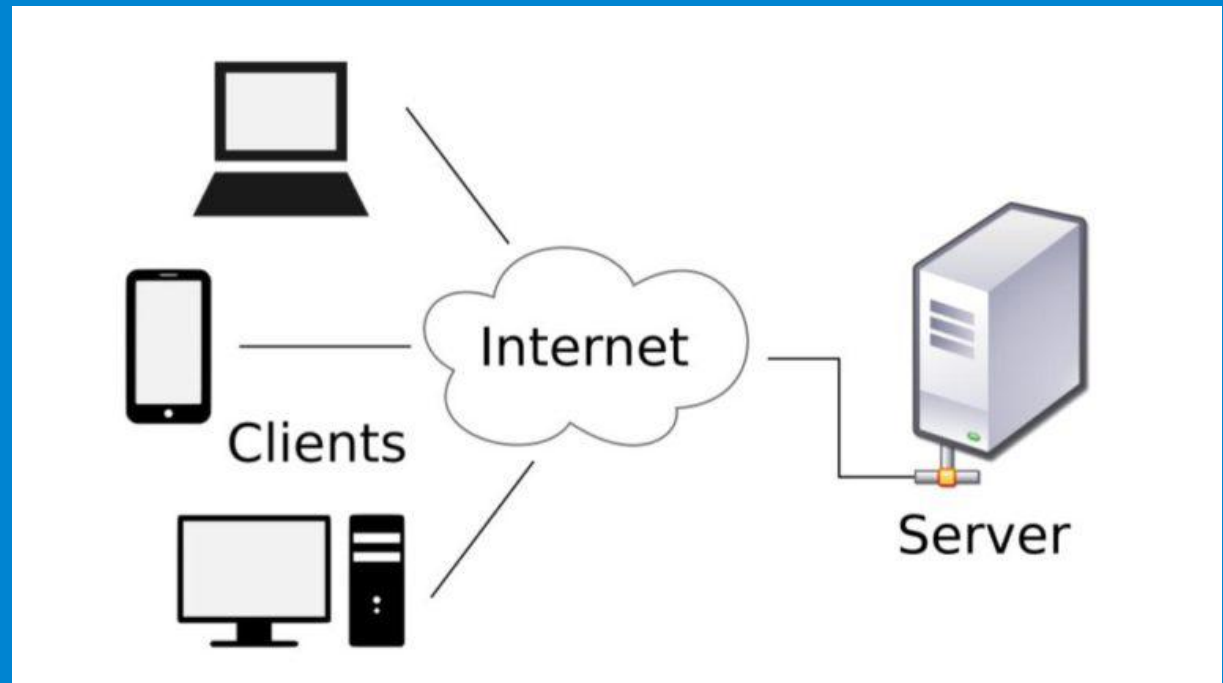
HTTP – Portas

- Uma porta é a maneira uma aplicação cliente especificar uma aplicação particular num servidor na rede



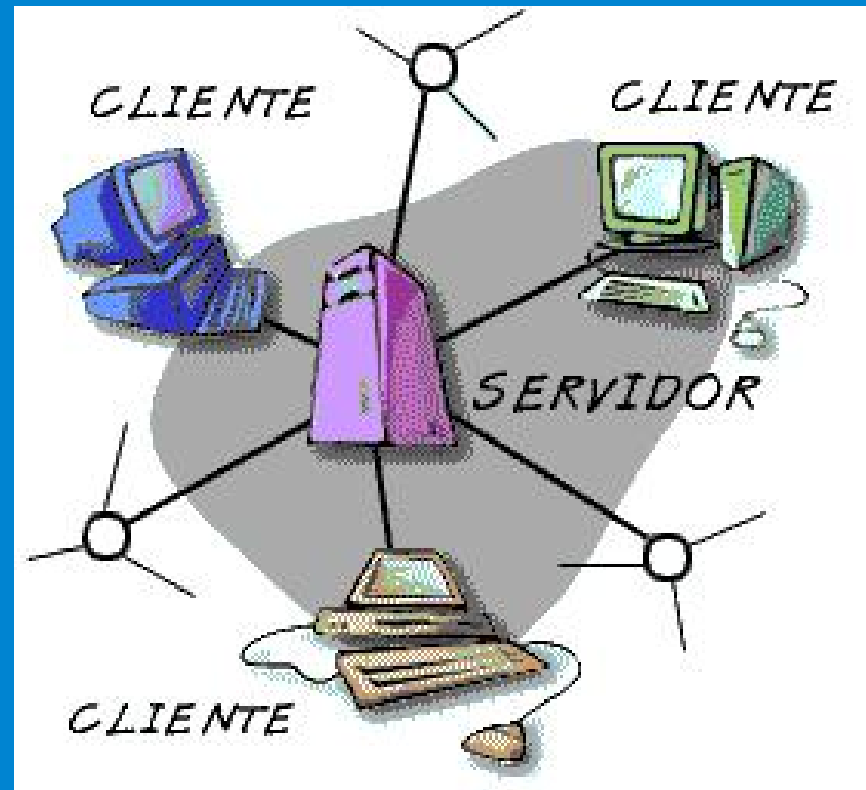
Execução no Cliente (Navegador, Browser)

- HTML
- CSS
- Imagens
- JavaScript
- XML/JSON



Execução no Cliente (Navegador, Browser)

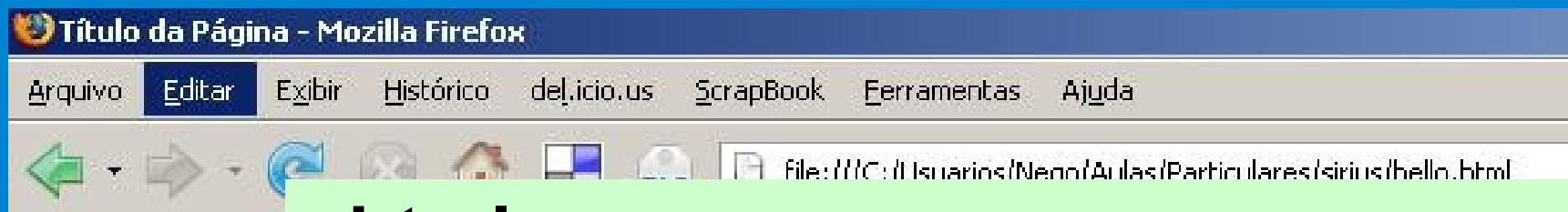
- HTML
- CSS
- Imagens
- JavaScript
- XML/JSON



HTML

- *Hyper Text Markup Language*
- Um arquivo html contém marcadores (tags)
- Estes marcadores indicam para o navegador (browser) como a página deve ser apresentada
- Marcadores usualmente vem em pares:
<tag></tag>

HTML



```
<html>
  <head>
    <title>Título da Página</title>
  </head>
  <body>
    Esta é minha primeira página.
  </body>
</html>
```

HTML



```
<html>
  <head>
    <title>Título da Página</title>
  </head>
  <body bgcolor="black">
    <p align="center">
      <font size="2" face="Verdana" color="white">
        Esta é minha primeira página.
      </font>
    </p>
  </body>
</html>
```

HTML

Formulários

- HTML possibilita a criação de formulários online utilizando tags
- A tag <form> é a mais comum e permite a criação de um formulário de entrada de dados

```
<body> <form>
```

```
Nome: <input type="text" name="firstname"><br />
```

```
Sobrenome: <input type="text" name="lastname"><br />
```

```
Senha: <input type="password" name="senha">
```

```
</form></body>
```



file:///C:/Usuarios/Nego/Aulas/Particulares/sirius/hello.html

Nome:

Sobrenome:

Senha:

HTML

Formulários

- Tipos que podem ser utilizados com a tag `<input>`:
 - button: Insere um botão
 - checkbox: Insere um checkbox; para vários itens, basta inserirmos vários “inputs” deste tipo
 - file: Insere botão seleção de arquivo através de uma caixa de diálogo
 - hidden: Campo pertencente ao formulário, mas que não será exibido na página
 - image: Insere uma imagem como um botão submit
 - password: Insere um campo password (caracteres digitados não aparecem)
 - radio: Insere um radiobox (análogo ao checkbox)
 - reset: Restaura os valores iniciais do formulário
 - submit: Encaminha os dados inseridos para algum arquivo
 - text: Insere um campo de edição de texto

HTML

Formulários

- Quando algum elemento do tipo “submit” é inserido num formulário e acionado, seus dados são enviados para um arquivo
- O arquivo mencionado é indicado pelo atributo “action” do elemento <form>
- Este arquivo deverá estar armazenado num servidor web (Apache, Tomcat, IIS, ...), e estará escrito em alguma linguagem de programação de servidores: jsp, php, asp, ...



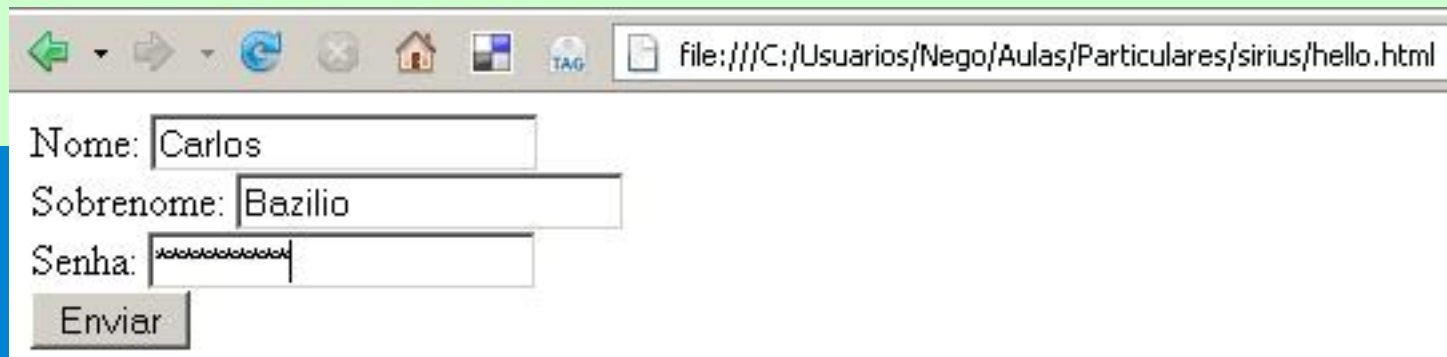
HTML

Formulários

<body> após clicar no
botão: *<form action="/sirius/processaForm.jsp?firstname=Carlos&lastname=Bazilio&senha=abcdefg">*
Nome: <input type="text" name="firstname">

Sobrenome: <input type="text" name="lastname">

Senha: <input type="password" name="senha">
<input type="submit" value="Enviar">
</form>
</body>



The screenshot shows a web browser window with the address bar displaying 'file:///C:/Usuarios/Nego/Aulas/Particulares/sirius/hello.html'. The browser's toolbar includes navigation buttons (back, forward, home, stop, reload) and a search icon. The main content area displays a form with three input fields: 'Nome:' containing 'Carlos', 'Sobrenome:' containing 'Bazilio', and 'Senha:' containing a masked password 'xoxoxoxoxoxoxoxox'. Below the password field is a button labeled 'Enviar'.

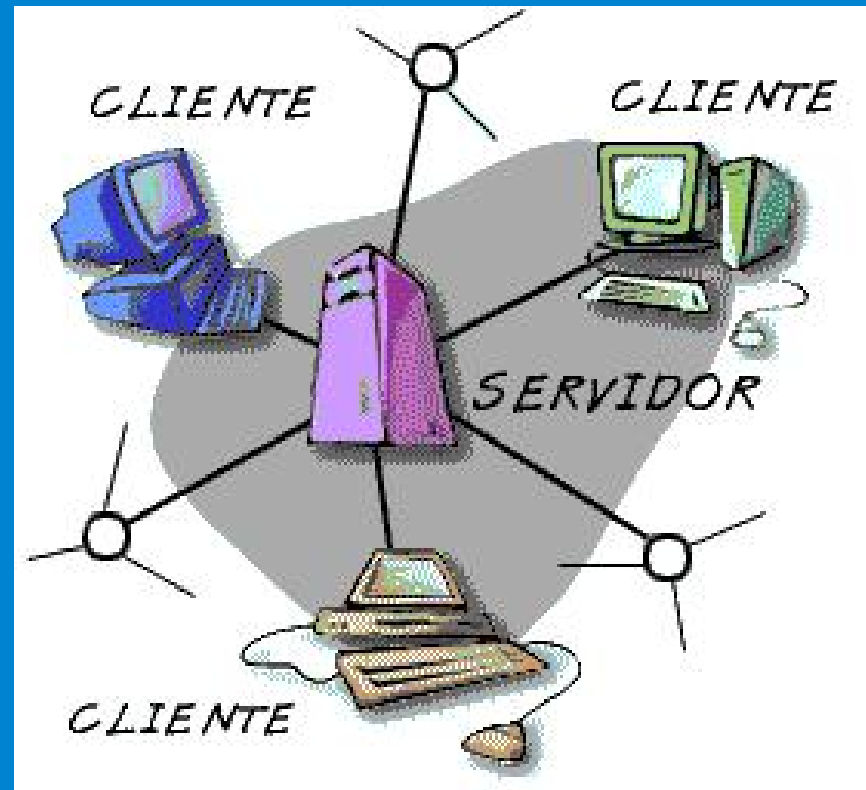
HTML

Referências

- <http://www.w3schools.com/html/default.asp>
p
– Lista de tags possíveis e exemplos diversos
- <http://www.w3.org/TR/html401/>
– Link do consórcio WWW que define padrões de linguagens web (em especial, HTML)

Execução no Cliente (Navegador, Browser)

- HTML
- CSS
- Imagens
- JavaScript
- XML/JSON



CSS

- *Cascading Style Sheets*
- Estilos definem como elementos html (textos, tabelas, links, ...) devem ser apresentados
- Permite a separação entre definição de conteúdo (informação, texto, ...) e formatação (cores, fontes, ...) em HTML
- Atualmente é o padrão para inserção de estilo na construção de páginas html

CSS

Como definir (1/3)

- Existem 3 maneiras de definir CSS para um documento HTML
 - Criando um arquivo CSS

```
/* Conteúdo do arquivo meuestilo.css */  
body { background-color: black }  
p { font-family: "Verdana"; text-align: center; color: white; font-size: 200% }
```

```
<html>
```

```
<head><link rel="stylesheet" type="text/css"  
href="meuestilo.css" />
```

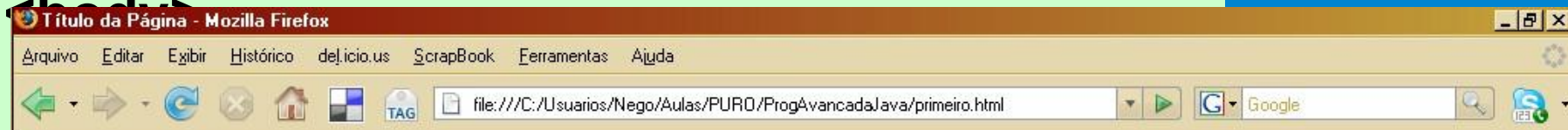
```
<title>Título da Página</title>
```

```
</head>
```

```
<body>
```

Esta é minha primeira página.

```
</body>
```



CSS

Como definir (2/3)

- Internamente (dentro da própria página HTML a ser exibida)

```
<html>
  <head>
    <style type="text/css">
      body { background-color: black }
      p { font-family: "Verdana";
        text-align: center; color: white;
        font-size: 200% }
    </style>
  </head>
  <body>
    Esta é minha primeira página.
  </body>
</html>
```

CSS

Como definir (3/3)

- Definindo de forma inline

```
<html>  
  <head</head>  
  <body style="background-  
color: black; text-align: center">  
    Esta é minha primeira página.  
  </body>  
</html>
```

- Caso existam várias definições para um mesmo elemento, a prioridade é: inline, interna e externa

CSS

- Formato geral:
`seletor { propriedade: valor }`
- Uma definição de estilo em CSS será composta por uma sequência de definições como esta acima
- Exemplos:
 - `body { color: black }`
 - `p { font-family: "Verdana"; text-align: center; color: white }`
 - `p {margin-left: 20px}`

CSS - Classes

- Classes em CSS permitem que um mesmo elemento seja exibido de diferentes formas
- Formato usando classes:

seletor.classe { propriedade: valor }

- Exemplos:
 - p.direita {text-align: right} // Alinha à direita
 - p.centro {text-align: center} // Centraliza
 - .esquerda {text-align: left} // Aplicado a qualquer elemento html que contenha classe *esquerda*

```
<p class="direita">Este parágrafo será alinhado à direita</p>
```

```
<p class="centro">Este parágrafo será centralizado</p>
```

```
<p class="esquerda">Este será à esquerda</p>
```

CSS – Pseudo-classes

- Pseudo-classes são usadas para adição de efeitos especiais a alguns seletores
- Formato:
seletor:pseudo-classe {propriedade: valor}
- Exemplos:
 - `a:link {color: #FF0000} /* link não visitado*/`
 - `a:visited {color: #00FF00} /* link visitado */`
 - `a:hover {color: #FF00FF} /* mouse sobre link */`
 - `a:active {color: #0000FF} /* link selecionado */`

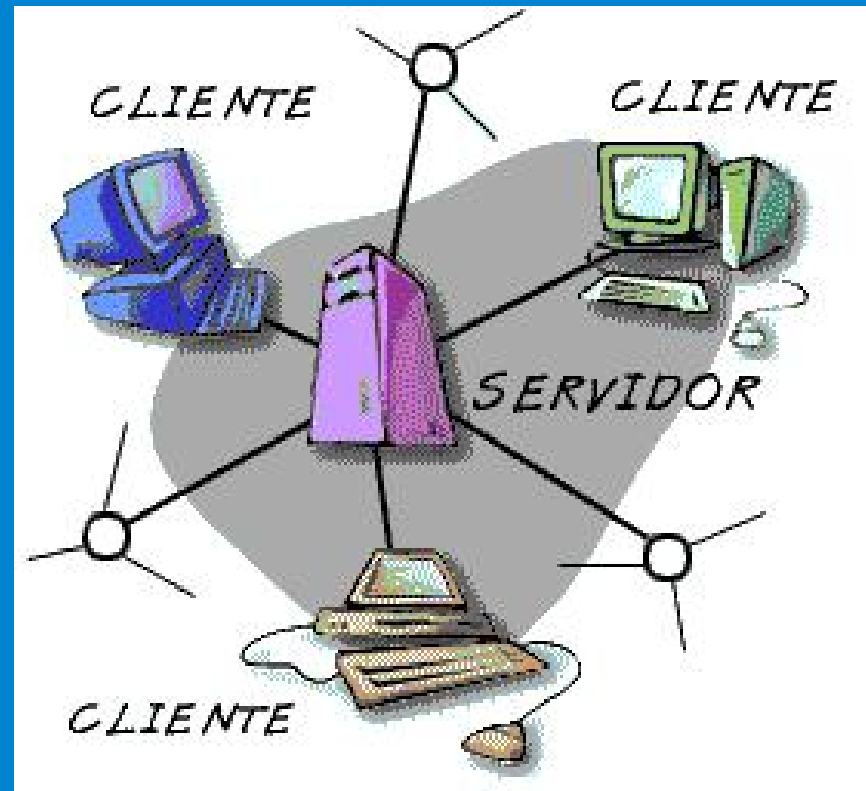
CSS

Referências

- Com CSS podemos formatar:
 - Background, Textos, Fontes, Margens, Bordas, Listas, Tabelas, ...
- <http://www.w3schools.com/css/default.asp>
 - Site com descrição de diversas construções disponíveis em CSS
- <http://www.csszengarden.com/>
 - Site com demonstração do poder expressivo de CSS

Execução no Cliente (Navegador, Browser)

- HTML
- CSS
- Imagens
- JavaScript
- XML/JSON



JavaScript

- É a linguagem de script utilizada por milhões de páginas web
- Foi projetada para aumentar interatividade das páginas web:
 - Validação de formulários, interação com o usuário (p.ex., tratamento de cliques de botões), detecção de navegadores, etc
- É reconhecida pela maioria dos navegadores
- Seu processamento é feito na máquina cliente (browser)
- Não há relação com Java

JavaScript

Janelas Popup

- Funções para criação de janelas popup:

- Alerta (`alert("Texto a ser exibido");`)

- Confirm

- em janelas*

- Entradas

- entradas*

- Janelas quando

```
<body>
```

```
  <script type="text/javascript">
```

```
    var nome = prompt("Seu nome");
```

```
    if (nome != null && nome != "")
```

```
      document.write("Oi " + nome + " !!");
```

```
    else
```

```
      document.write("Oi anônimo!");
```

```
  </script>
```

```
</body>
```

[exemplo-javascript.html](#)

JavaScript - Sintaxe

- Possui construções existentes na maioria das linguagens de programação (sintaxe similar a C):
 - Declaração de variáveis (*var x;*)
 - Comandos condicionais, repetições, definição de funções de usuário
 - Operadores de atribuição, comparação, ...
 - Tipos pré-definidos, classes de objetos, ...

JavaScript – Eventos

- Eventos são ações que podem ser detectadas por um script
- Exemplos de eventos:
 - Clique do mouse, abertura de uma página web ou imagem, envio de um formulário html, uma tecla pressionada, etc
- O tratamento destes eventos pode ser a chamada de funções do script

JavaScript – Eventos

```
<html>
  <head><script src="event.js"></script>
</head>
<body>
  
</body>
</html>
```

```
/* Conteúdo do arquivo event.js */
function mouseOver()
{
  document.imagem.src = "quadrado2.gif";
}
function mouseOut()
{
  document.imagem.src = "quadrado1.gif";
}
```

[exemplo2-javascript.html](#)

JavaScript – Objetos

- JavaScript é uma linguagem OO
- Com isto, algumas classes utilitárias padrões estão disponíveis, as quais possuem métodos e propriedades:
 - String: manipulação de strings no script

```
/* Exemplo de função de script que calcula  
o tamanho de uma string */  
function tamanho(msg)  
{  
    return msg.length;  
}
```

- http://www.w3schools.com/jsref/jsref_obj_string.asp

JavaScript – Objetos

- Date: manipulação de datas no script

```
/* Exemplo de função de script que manipula  
datas */  
function bug(data)  
{  
    var x = new Date();  
    x.setFullYear(2000,0,0);  
    if (data > x)  
        alert("Bug do milênio!");  
}
```

- http://www.w3schools.com/jsref/jsref_obj_date.asp

JavaScript – Objetos

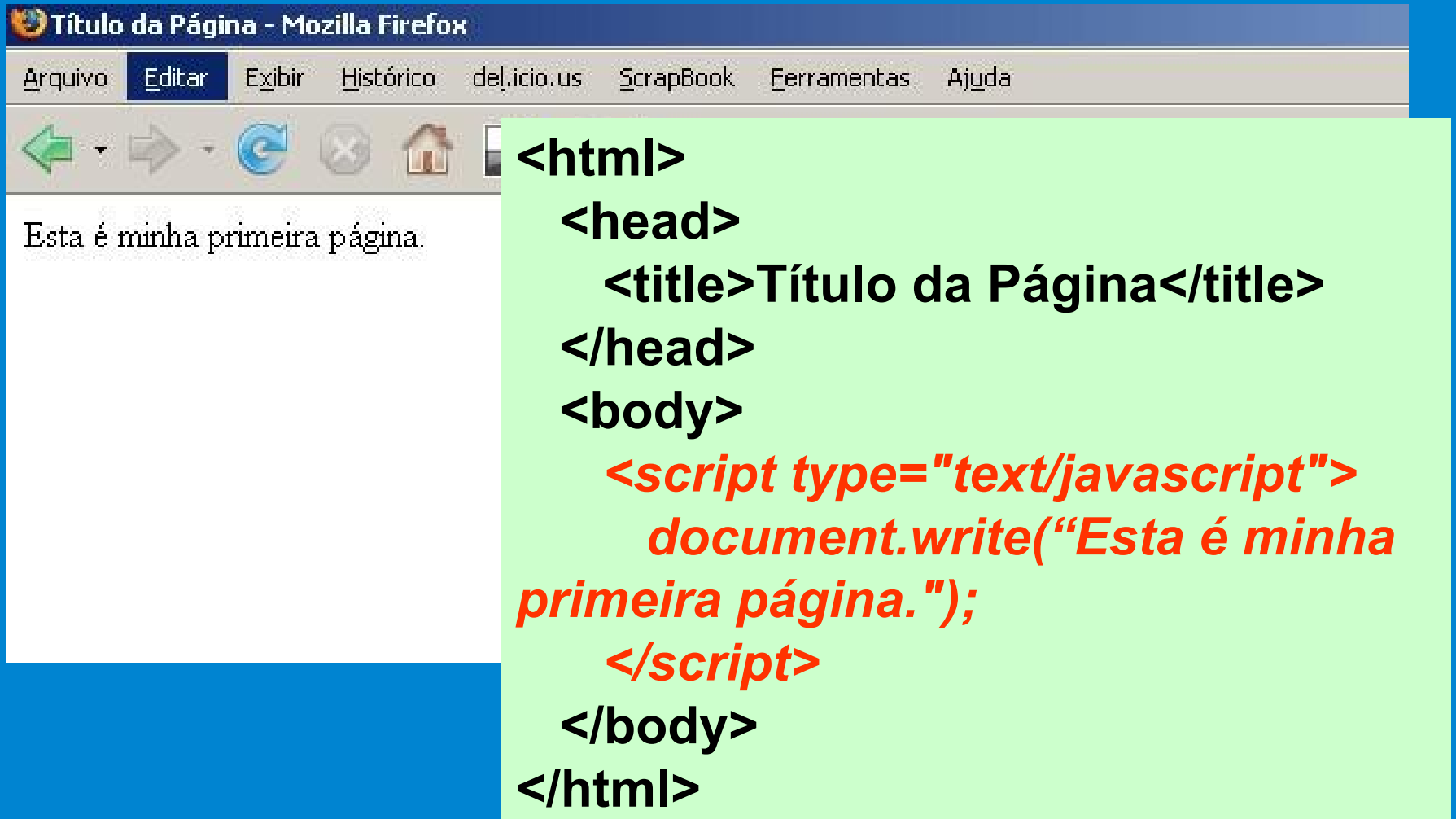
- Outras classes disponíveis na linguagem JavaScript:
 - Array: armazenamento de conjuntos de valores
 - http://www.w3schools.com/js/js_obj_array.asp
 - Boolean: manipulação de valores booleanos
 - http://www.w3schools.com/js/js_obj_boolean.asp
 - Math: manipulação de valores com operações matemáticas
 - http://www.w3schools.com/jsref/jsref_obj_math.asp
- Além destas, todos os objetos HTML DOM podem ser acessados através de scripts
 - http://www.w3schools.com/js/js_obj_htmlDOM.asp

JavaScript – Objetos

Exemplo

```
/* Exemplo de função de script que manipula objetos  
de JavaScript */  
function exemplo_objetos()  
{  
    var d=new Date();  
    var dias=new Array(7);  
    dias[0]="Domingo"; dias[1]="Segunda";  
    dias[2]="Terça"; dias[3]="Quarta"; dias[4]="Quinta";  
    dias[5]="Sexta"; dias[6]="Sábado";  
    document.write("Hoje é" + dias[d.getDay()]);  
    document.write("Dia qualquer: " +  
    dias[Math.round(Math.random()*6)]);  
    document.write(navigator.appCodeName);  
}
```

JavaScript - Exemplo



The image shows a screenshot of a Mozilla Firefox browser window. The title bar reads "Título da Página - Mozilla Firefox". The menu bar includes "Arquivo", "Editar", "Exibir", "Histórico", "del.icio.us", "ScrapBook", "Ferramentas", and "Ajuda". The address bar shows a home icon and a search bar. The main content area displays the text "Esta é minha primeira página." on a white background. To the right of the browser window, the HTML code for the page is displayed on a light green background.

```
<html>  
  <head>  
    <title>Título da Página</title>  
  </head>  
  <body>  
    <script type="text/javascript">  
      document.write("Esta é minha  
primeira página.");  
    </script>  
  </body>  
</html>
```

JavaScript – Onde ocorrem

- Uma tag `<script/>` pode ser definida numa seção head, numa seção body e também pode ser definida externamente:
 - Na seção head, os scripts são executados quando são chamados ou quando algum evento ocorre;
 - Na seção body, os scripts são executados na carga da página web
 - Para definição externa, um arquivo “.js” precisa ser fornecido com as funções necessárias

JavaScript – Exemplo

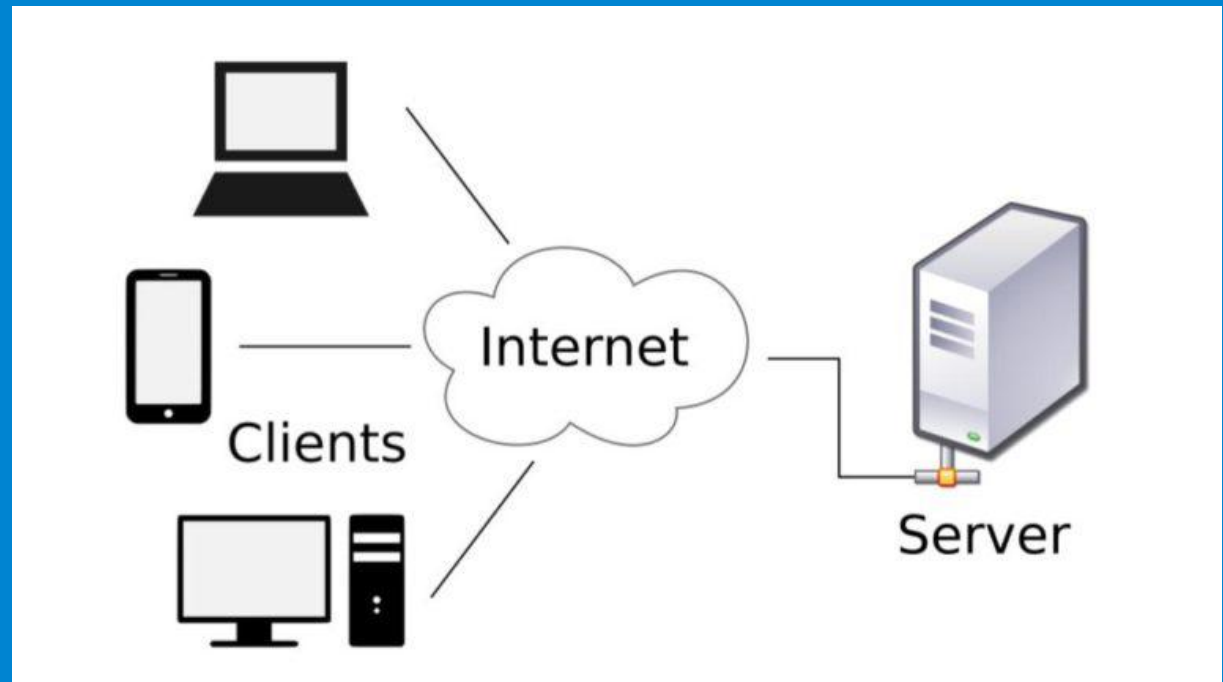
```
<html>
  <head>
    <title>Título da Página</title>
    <script src="hello.js"></script>
  </head>
  <body onload="message()">
    <script type="text/javascript">
      document.write("<h1>Esta é minha primeira
página.</h1>");
    </script>
  </body>
</html>
```

JavaScript

- Mais Usos
 - Criação de cookies
 - Validação de entrada
 - Disparo de funções com retardo (tempo)
 - Criação de objetos próprios
- Referências
 - <http://www.w3schools.com/js/default.asp>
 - <http://oopweb.com/JavaScript/Documents/jsintro/Volume/tutorial.htm>

Execução no Cliente (Navegador, Browser)

- HTML
- CSS
- Imagens
- JavaScript
- XML/JSON



XML

```
<?xml version="1.0" encoding="UTF-8"?>
<livros>
  <livro isbn="0001">
    <titulo>JavaServer Pages</titulo>
    <autor>Nick Todd</autor>
    <editora>Campus</editora>
    <assunto>JSP</assunto>
  </livro>
  <livro isbn="0002">
    <titulo>Meu pé de laranja lima</titulo>
    <editora>Vozes</editora>
    <autor>Brilhante</autor>
  </livro>
</livros>
```

XML

- Um arquivo XML é definido por:
 - Um arquivo em formato ASCII
 - Tags aninhadas hierarquicamente (número indeterminado de sub-elementos)
 - Um único elemento (<tag></tag>) raiz
 - Indeterminado número de atributos (<tag atr1="val1" ...></tag>) numa tag

Justificativa para Uso de XML

- Integração de aplicações
- Formato flexível (linguagem de propósito geral)
- Tem sido amplamente utilizada
 - [Meus dados na declaração do IRPF](#)
 - [OpenOffice.org XML Essentials](#)
 - Canais de notícia (RSS, Feeds):
 - [Artigos de Desenvolvimento Web na IBM](#)
 - [Informações sobre tecnologia do site do Globo](#)



Exemplo de Integração usando XML

IRPF 2008 - CARLOS BAZILIO MARTINS - Declaração com Desconto Simplificado

Declaração Preenchimento Fichas Ferramentas Ajuda

Declaração

- Identificação do Contribuinte
- Rend. Trib. Receb. de PJ
- Rend. Trib. Receb. de PF/Ext
- Rend. Isentos e Não-Tributá
- Rend. Sujeitos à Trib. Excl
- Imposto Pago
- Dependentes
- Alimentandos
- Pagamentos e Doações Efet
- Bens e Direitos
- Dívidas e Ônus Reais
- Informações do Cônjuge
- Espólio
- Doações a Part. Políticos, Co
- Atividade Rural
- Ganhos de Capital
- Moeda Estrangeira
- Renda Variável
- Resumo da declaração

Identificação do Contribuinte

Nome: CARLOS BAZILIO MARTINS

Data de Nascimento: 07/07/1975 Título Eleitoral:

Endereço: ☒ No Brasil ☐ No Exterior

Houve mudança de endereço? ☒ Sim ☐ Não

Tipo: Estrada Logradouro: RIO PETROPOLIS, KM 83

Número: 38-A Complemento: CX. POSTAL 17E Bairro/Distrito: DUGUES

UF: RJ - Rio de Jan Município: Petrópolis

CEP: 25650-001 Consulta CEP: 0000xx 24 Telefone: 2235-9433

Ocupação Principal

Natureza da ocupação: 22 Servidor público de autarquia ou fundação federal

Ocupação Principal: 294 Professor do ensino superior

Esta declaração é retificadora? ☐ Sim ☒ Não

Nº do recibo da última declaração entregue do exercício de 2007:

XML
(CEP)



XML (Dados
Postais)

WebService = XML sobre HTTP

Esquema XML

- Documentos XSD (esquemas), assim como DTDs, definem uma gramática para documentos XML
- Ou seja, definem o que vale e o que não vale num documento XML
- Para o exemplo dos Correios, a definição pode ser feita em comum acordo (cliente e servidor) ou determinadas pelo servidor

XML

```
<?xml version="1.0" encoding="UTF-8"?>
<livros>
  <livro isbn="0001">
    <titulo>JavaServer Pages</titulo>
    <autor>Nick Todd</autor>
    <editora>Campus</editora>
    <assunto>JSP</assunto>
  </livro>
  <livro isbn="0002">
    <titulo>Meu pé de laranja lima</titulo>
    <editora>Vozes</editora>
    <autor>Brilhante</autor>
  </livro>
</livros>
```

DTD

```
<?xml version="1.0" encoding="UTF-8"?>
<!--DTD generated by XMLSpy v2008
(http://www.altova.com)-->
<!ELEMENT titulo (#PCDATA)>
<!ELEMENT livros ((livro+))>
<!ATTLIST livros
    xmlns:xsi CDATA #FIXED
    "http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation CDATA #IMPLIED>
<!ELEMENT livro ((titulo, ((editora, autor) | (autor, editora,
assunto))))>
<!ELEMENT editora (#PCDATA)>
<!ELEMENT autor (#PCDATA)>
<!ELEMENT assunto (#PCDATA)>
```

Esquema XML (XSD)

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="livros">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="livro" maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="livro">
    <xs:complexType>
      <xs:all>
        <xs:element name="titulo" type="xs:string"/>
        <xs:element name="autor" type="xs:string"/>
        <xs:element name="editora" type="xs:string" minOccurs="0"/>
        <xs:element name="assunto" type="xs:string" minOccurs="0"/>
      </xs:all>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

XML

Vantagens e Desvantagens

- Vantagens:
 - Documento texto
 - Formato flexível
 - Reuso de ferramentas de manipulação
- Desvantagens:
 - Documento verboso (em torno de 70% são tags)
 - Manipulação manual ou através de bibliotecas

XML

Referências

- <http://www.w3.org/XML/>
 - Site do consórcio W3C sobre a linguagem
- <http://www.w3schools.com/xml/default.asp>
 - Site com informações básicas e muitos exemplos
- <http://www.xml.com/>
 - Site notícias para desenvolvimento avançado utilizando XML
- <http://xmlsucks.org/>
 - Site que fala sobre desvantagens do formato XML

JSON

- *JavaScript Object Notation*
- Formato aberto para **representação de dados**
- Usualmente utilizado para armazenamento e troca de informações, da mesma maneira que XML
- Herdado do formato de representação de estruturas de dados e vetores associativos de JavaScript
- É utilizado com uma **alternativa ao XML** por ser menos verboso

JSON - Exemplo

```
{
  "firstName": "John",
  "lastName": "Smith",
  "age": 25,
  "address": {
    "streetAddress": "21 2nd Street",
    "city": "New York",
    "state": "NY",
    "postalCode": "10021"
  },
  "phoneNumber": [
    {
      "type": "home",
      "number": "212 555-1234"
    },
    {
      "type": "fax",
      "number": "646 555-4567"
    }
  ]
}
```

Tableless

- “Antigamente” (antes da definição de CSS), os sites eram estruturados utilizando tabelas
- Conteúdos web eram disponibilizados de forma rígida e em posições fixas
- Esta prática vai contra a tendência atual de computação ubíqua (disponibilização de informações em diversos meios e momentos)
- Tableless (*less tables*) é o movimento que tenta desestimular esta prática (uso excessivo de tabelas)
- Tabelas devem ser utilizadas **apenas** para definição de tabelas (de dados, números, etc). Cada tag HTML tem uma semântica própria e deve ser usada para tal.

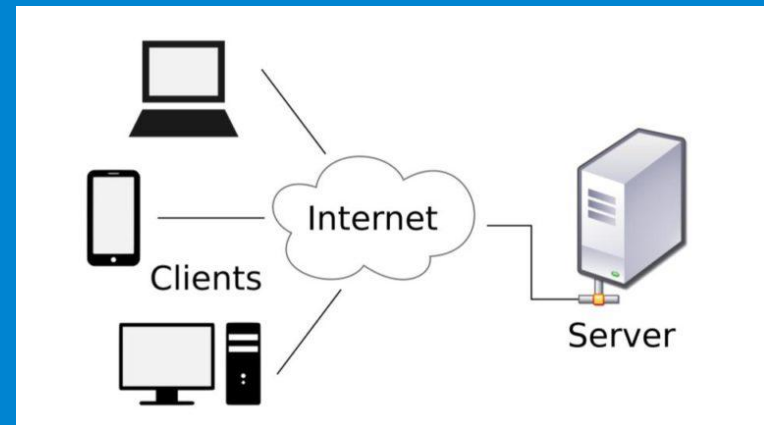
Padrões Web

Web Standards

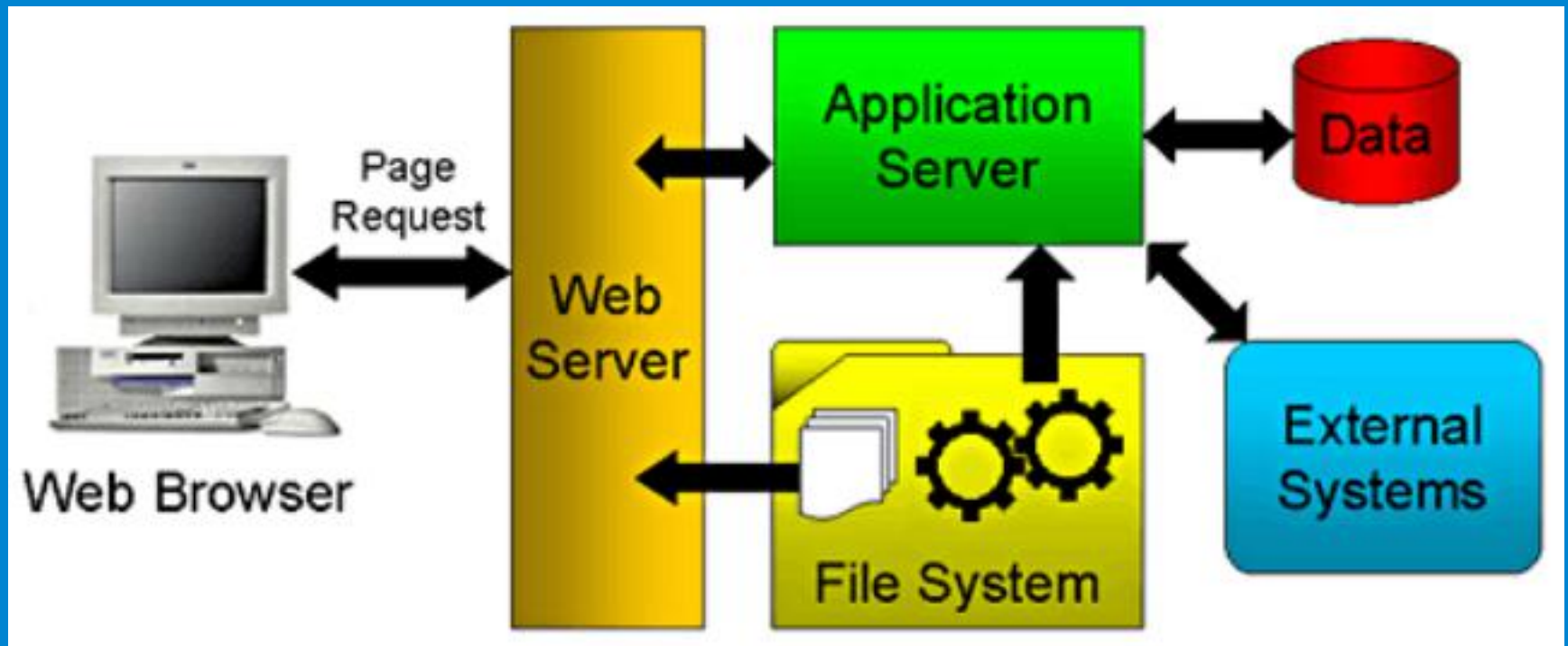
- Podem ser vistos como boas práticas na construção de páginas web
- Estes padrões procuram garantir interoperabilidade na web
- Como exemplo, podem ser abordadas questões como usabilidade e acessibilidade (permitir acesso a portadores de deficiência)
- Ou seja, é um termo mais geral e abrangente que Tableless

Execução no Servidor (Servidor Web)

- Servidores Web
 - Hospedam recursos a serem solicitados por navegadores
 - Exemplos: Apache, IIS, “*Tomcat*”, ...
- Linguagens de Script
 - Combinam comandos linguagem de programação com tags HTML para produzir tags HTML mais dinâmicas
 - Exemplos: ASP, JSP, PHP, ...



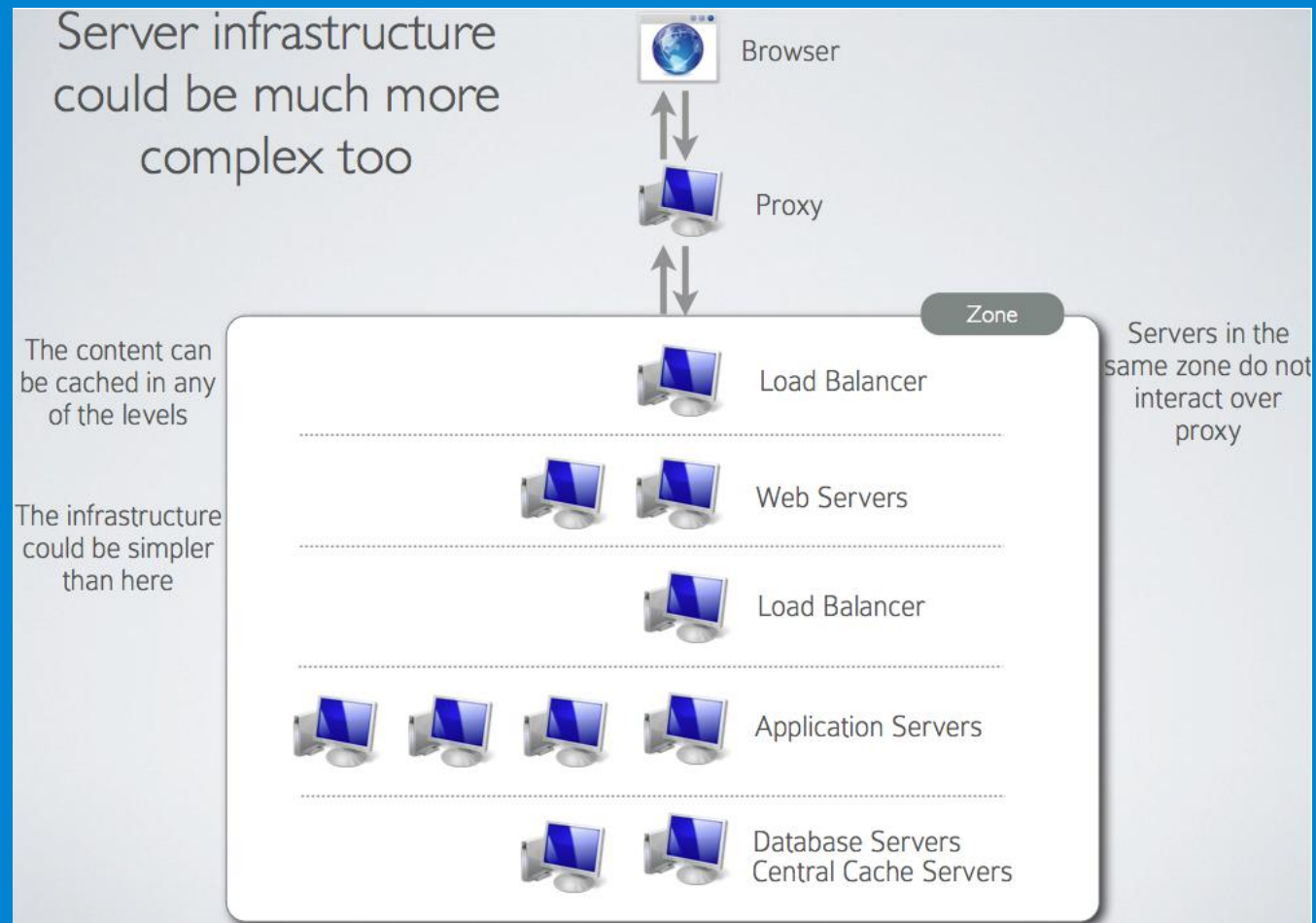
Arquitetura Web Canônica



Infraestrutura



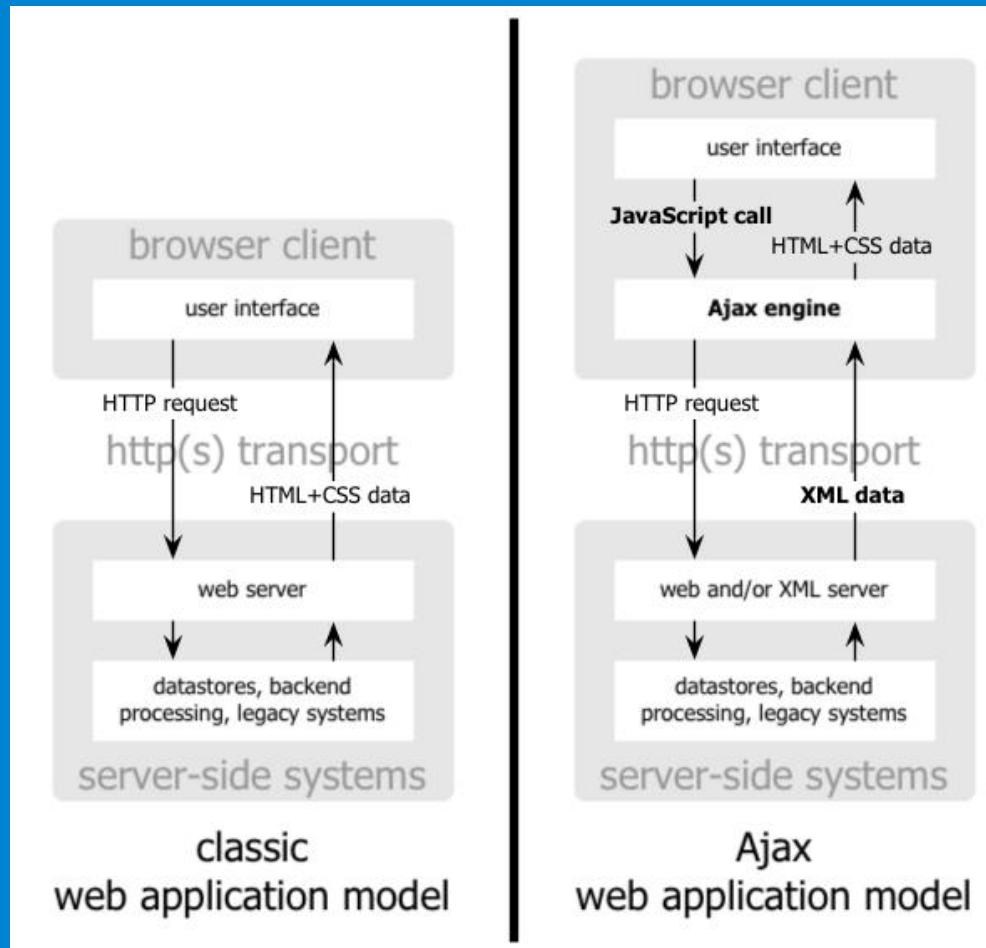
Infraestructura



AJAX

- *Asynchronous Javascript And XML*
- Construção de páginas mais dinâmicas através de chamadas assíncronas ao servidor
- Não é uma nova tecnologia ou linguagem, mas sim uma combinação de tecnologias padrões já existentes:
 - XHTML + CSS + XML + JavaScript + XSLT

AJAX



Web 2.0



- Termo que define uma tendência de projeto de websites



- Estes valorizam a criatividade, o compartilhamento de informações (APIs, RSS, ...), a colaboração entre usuários, a inferência de novos conteúdos, folksonomia (tags), etc



- **Não** está associado a nenhuma nova especificação da Web, apesar de se beneficiar



Blogger



BuscaPé



Web – Tendências

- Algumas tecnologias, arquiteturas e linguagens “novas”, das quais a suposta Web 2.0 tira proveito:
 - AJAX: XHTML + CSS + XML + JavaScript + XSLT
 - Web Services
- E na Web 3.0:
 - Web Semântica
 - Big Data
 - ...

Tableless, AJAX

Referências

- <http://www.hideout.com.br/usp/semacomp2005/slides/minicurso.html>
 - Minicurso apresentado na USP em 2005 sobre web 2.0
- <http://en.wikipedia.org/wiki/AJAX> e [http://www.w3schools.com/Ajax/Default.As](http://www.w3schools.com/Ajax/Default.asp)
[p](http://www.w3schools.com/Ajax/Default.asp)
 - Tutoriais sobre AJAX

Considerações Finais

- Em se tratando de web, muito ainda está por vir:
 - Melhor integração com dispositivos móveis, veículos, eletrodomésticos, ...
 - Mudança nas relações trabalho/local de trabalho, estudo/local de estudo, ...
 - Autêntico BBB
 - Extinção do dinheiro em papel
 - E muito trabalho para nós:
- Novas linguagens, novos ambientes de execução, o papel crucial da engenharia de software, novos clientes, questões filosóficas, morais, ...

That's all Folks!

