



SCE 0265

Aula 3 – Arquitetura de software na Web atual:
processamento de lado cliente

Prof: Dra. Renata Pontin de Mattos Fortes

Aluno **PAE**: Willian Massami Watanabe

Sumário da aula

- Aplicações Estáticas x Aplicações Dinâmicas
- Javascript
 - Características da linguagem
 - Núcleo da linguagem
- HTML Events e DHTML
- Por que Javascript?

Sumário da aula

- **Aplicações Estáticas x Aplicações Dinâmicas**
- Javascript
 - Características da linguagem
 - Núcleo da linguagem
- HTML Events e DHTML
- Por que Javascript?

Aplicações Estáticas x Aplicações Dinâmicas

Aplicações Estáticas

Aplicações Estáticas x Aplicações Dinâmicas

- Revisando HTML

Dados Pessoais

Nome: Daniel Marquesim
Nro. USP: 5889619
E-mail: daniel.marquesim@gmail.com

Horário

	SEGUNDA	TERÇA	QUARTA	QUINTA	SEXTA
07:20 - 09:00	-----	-----	SCC0243	-----	-----
07:20 - 10:00	SCC0143	-----	-----	-----	SCC0265
10:10 - 11:50	SCC0207	SME0110	-----	SME0110	SCC0243
13:20 - 16:00	SCC0261	-----	-----	-----	SCC0244
16:20 - 18:50	-----	SCC0206	-----	-----	-----

Aplicações Estáticas x Aplicações Dinâmicas

- Revisando HTML – Formas de Interação
 - Links
 - Âncoras (exemplo: <http://www.w3.org/TR/WCAG10/>)
 - Inputs (exemplo: formulários na aula 2)

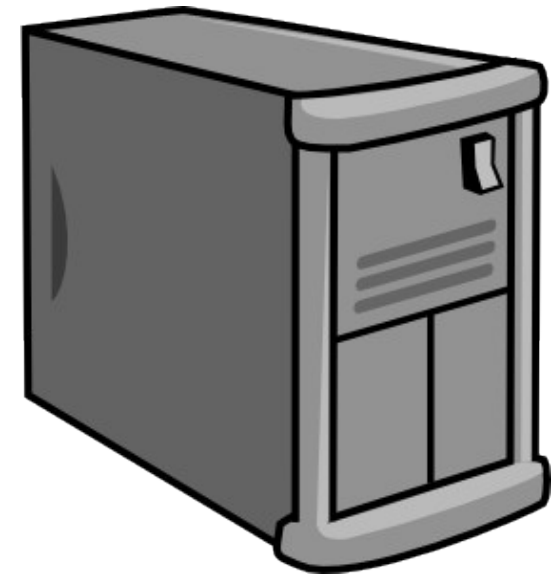
Aplicações Estáticas x Aplicações Dinâmicas

- Revisando HTML



Usuário
interagindo
com o
Navegador

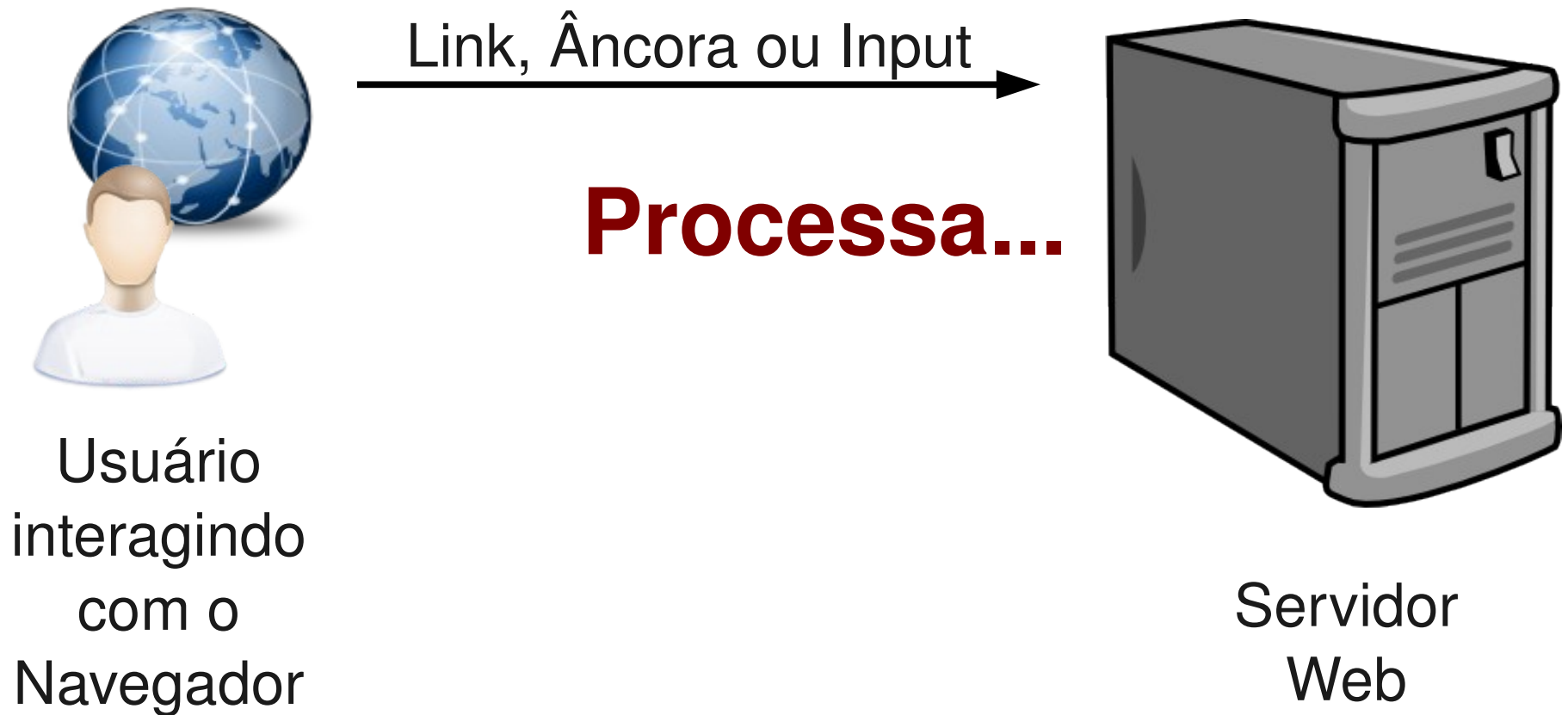
Link, Âncora ou Input

A black arrow pointing from the user icon on the left towards the server icon on the right, indicating the direction of the request.

Servidor
Web

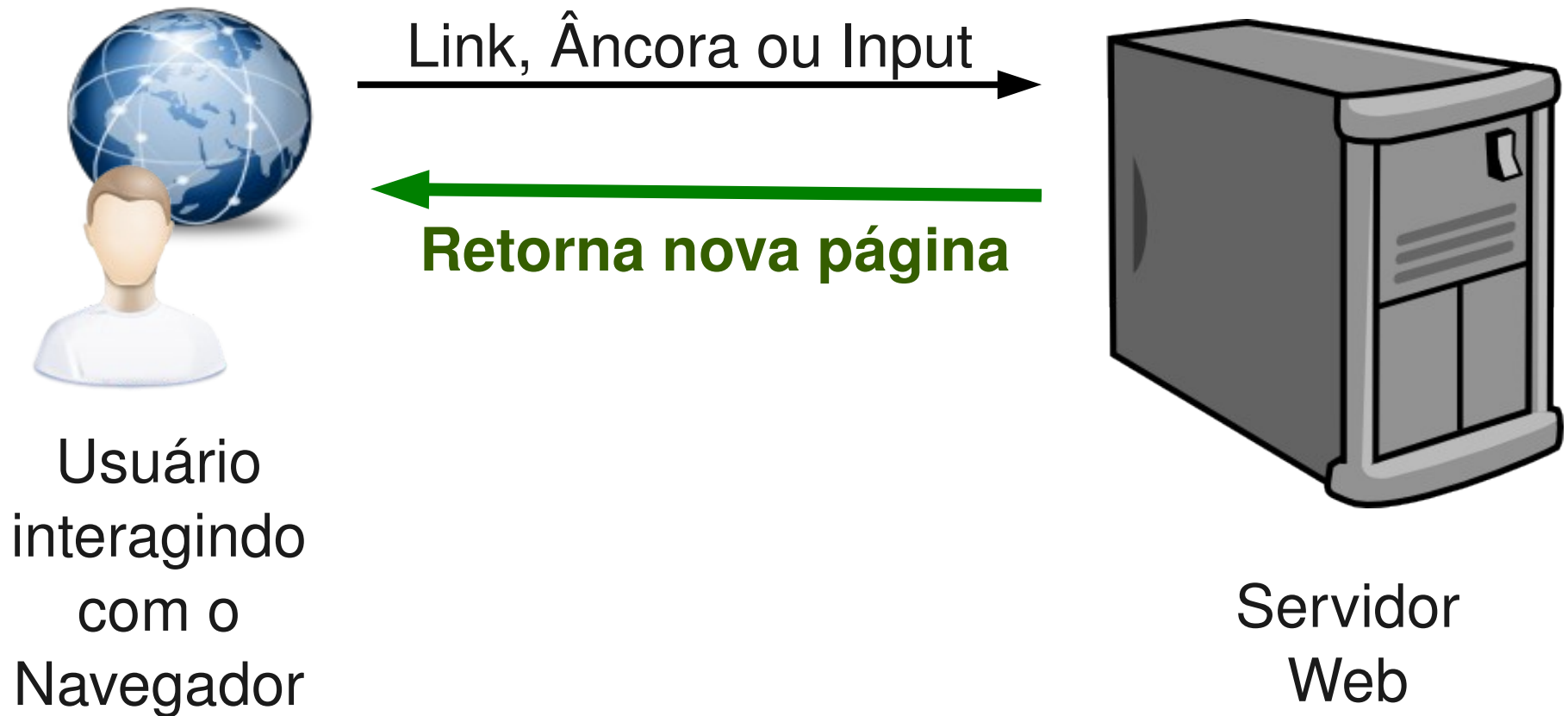
Aplicações Estáticas x Aplicações Dinâmicas

- Revisando HTML



Aplicações Estáticas x Aplicações Dinâmicas

- Revisando HTML



Aplicações Estáticas x Aplicações Dinâmicas

- Revisando CSS – Formas de Interação

- Exemplo menu:

<http://maujor.com/tutorial/ddownmenu/barebones.html>

Home	
About	
Services	Web Design
Contact Us	Internet Marketing
	Hosting
	Domain Names
	Broadband

Aplicações Estáticas x Aplicações Dinâmicas

- Revisando CSS – Formas de Interação

- Exemplo menu:

<http://maujor.com/tutorial/ddownmenu/barebones.html>



Home	
About	
Services	Web Design
Contact Us	Internet Marketing
	Hosting
	Domain Names
	Broadband



**Implementado utilizando apenas CSS.
Dica: Pseudo-classes em CSS**

Aplicações Estáticas x Aplicações Dinâmicas

- Revisando CSS – Formas de Interação
 - Simples e elegante
 - Limitações no desenvolvimento de interações complexas

Aplicações Estáticas x Aplicações Dinâmicas

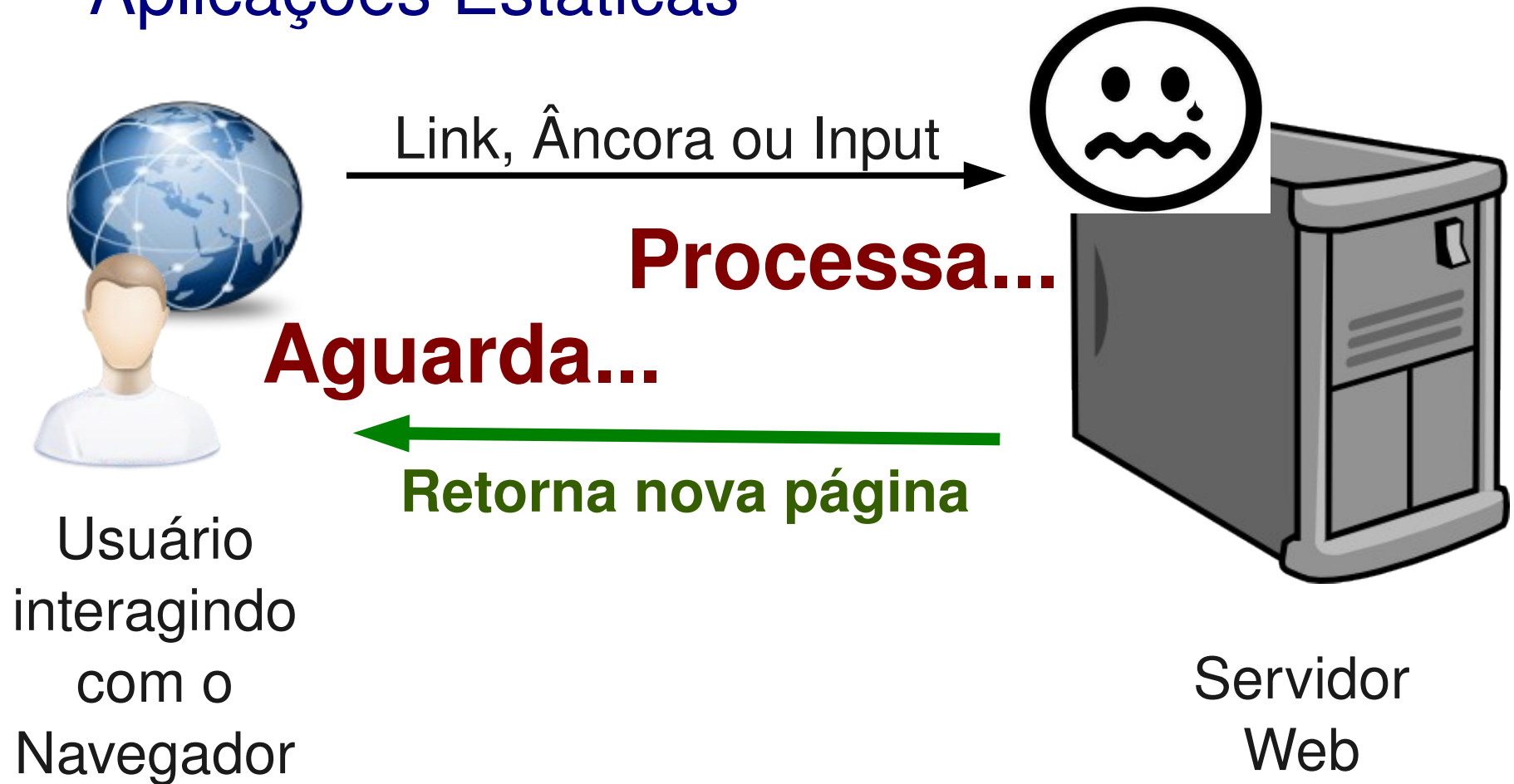
Aplicações Estáticas = HTML + CSS

Aplicações Estáticas x Aplicações Dinâmicas

- Aplicações Estáticas
 - Limitações nas interações com o **usuário**
 - Espera do usuário por atualizações
 - **Sobrecarga no processamento do servidor (Servidor é responsável por todo o processamento)**

Aplicações Estáticas x Aplicações Dinâmicas

- Aplicações Estáticas



Aplicações Estáticas x Aplicações Dinâmicas

Aplicações Dinâmicas



Aplicações Estáticas x Aplicações Dinâmicas

- Aplicações Dinâmicas



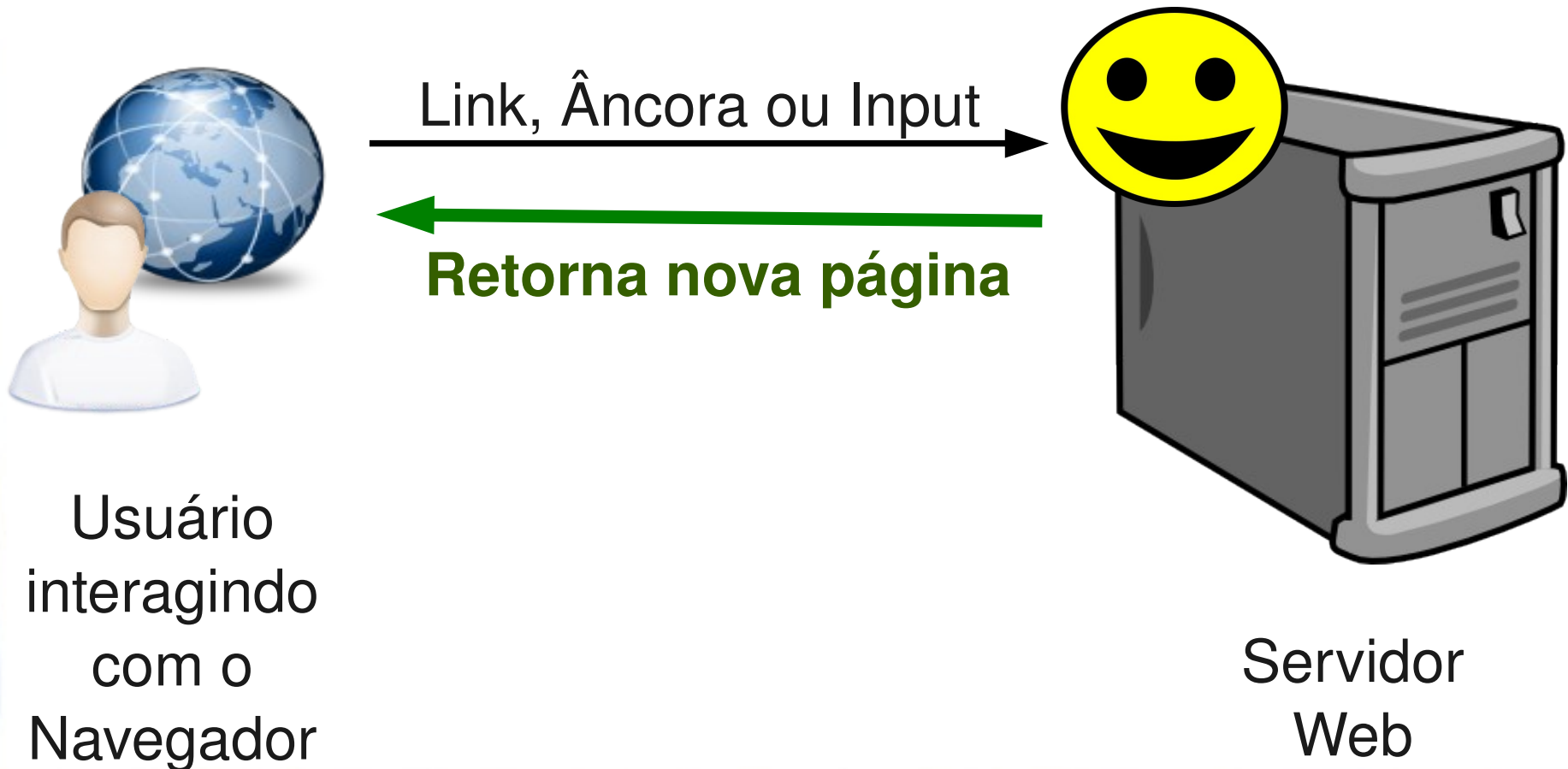
Aplicações Estáticas x Aplicações Dinâmicas

- Aplicações Dinâmicas



Aplicações Estáticas x Aplicações Dinâmicas

- Aplicações Dinâmicas
 - Interações com o servidor menos frequentes



Aplicações Estáticas x Aplicações Dinâmicas

- Aplicações Dinâmicas - Características
 - Interações com o servidor menos frequentes
 - Novas formas de interação com o usuário
 - Menor tempo de espera do usuário

Aplicações Estáticas x Aplicações Dinâmicas

Aplicações dinâmicas = **(X)HTML + CSS + Javascript + DOM HTML + DHTML + XML + HTML Events + XML-Schema + DTD + Webservices + JSON + Ajax + HTML5 + SVG + Canvas + ...**



Aplicações Estáticas x Aplicações Dinâmicas

- Aplicações Dinâmicas
 - **RIA** – *Rich Internet Applications*
 - Resposta aos requisitos da Web 2.0



Usuários participando da autoria de conteúdo na Web

Web 2.0

Aplicações Estáticas x Aplicações Dinâmicas

- Aplicações Dinâmicas - Exemplos
 - Gmail
 - Google Maps
 - Facebook
 - Orkut (depois de adquirido pela Google)
 - Twitter
 - Entre outros.

Sumário da aula

- Aplicações Estáticas x Aplicações Dinâmicas
- **Javascript**
 - Características da linguagem
 - Núcleo da linguagem
- HTML Events e DHTML
- Por que Javascript?

Javascript

- Introdução

- Desenvolvimento de aplicações dinâmicas (Lado Cliente)
- Processamento Local
- Controle sobre eventos
- Customização das interações com os usuários
- Alteração dinâmica do conteúdo ou da apresentação
- Realização de cálculos e computação
- Mais Recursos, não disponíveis no browser

Javascript

- Introdução
 - Não é a mesma coisa que Java
 - Atualmente funciona na maioria dos navegadores
 - Internet Explorer, Firefox, Safari, Chrome, Opera, entre outros.

Javascript

- Histórico
 - Criada pela Netscape Communications Corporation
 - Lançada em 1995 no navegador Netscape
 - Também conhecida como ECMAScript

Sumário da aula

- Aplicações Estáticas x Aplicações Dinâmicas
- **Javascript**
 - **Características da linguagem**
 - Núcleo da linguagem
- HTML Events e DHTML
- Por que Javascript?

Javascript – Características da Linguagem

- Linguagem interpretada
 - O código fonte é interpretado toda vez que o programa é executado
 - Contraste com linguagens compiladas
 - Decremento de desempenho em relação a linguagens compiladas

Javascript – Características da Linguagem

- Tipagem Dinâmica
 - Alteração dinâmica
 - Tipos de variáveis
 - Atributos de **objetos**
 - Métodos de **objetos**

Javascript – Características da Linguagem

- Funções de ordem superior
 - Funções podem ser passadas como argumentos para outras funções ou métodos

Javascript – Características da Linguagem

- Programação Client-side
 - Executada no navegador do cliente
 - Não interfere no processamento de lado servidor

Sumário da aula

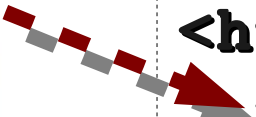
- Aplicações Estáticas x Aplicações Dinâmicas
- **Javascript**
 - Características da linguagem
 - **Núcleo da linguagem**
- HTML Events e DHTML
- Por que Javascript?

Javascript – Núcleo da Linguagem

- Por onde começar?
 - Inserir o código Javascript dentro do elemento `<head>` ou `<body>`
 - Inserir um documento externo

Javascript – Núcleo da Linguagem

- Dentro do elemento <head>

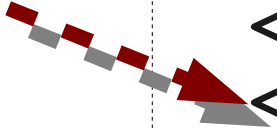


```
<html>  
  <head>  
    <script type="text/javascript">  
      ...  
    </script>  
  </head>  
  ...  
</html>
```

Javascript – Núcleo da Linguagem

- Dentro do elemento `<body>`

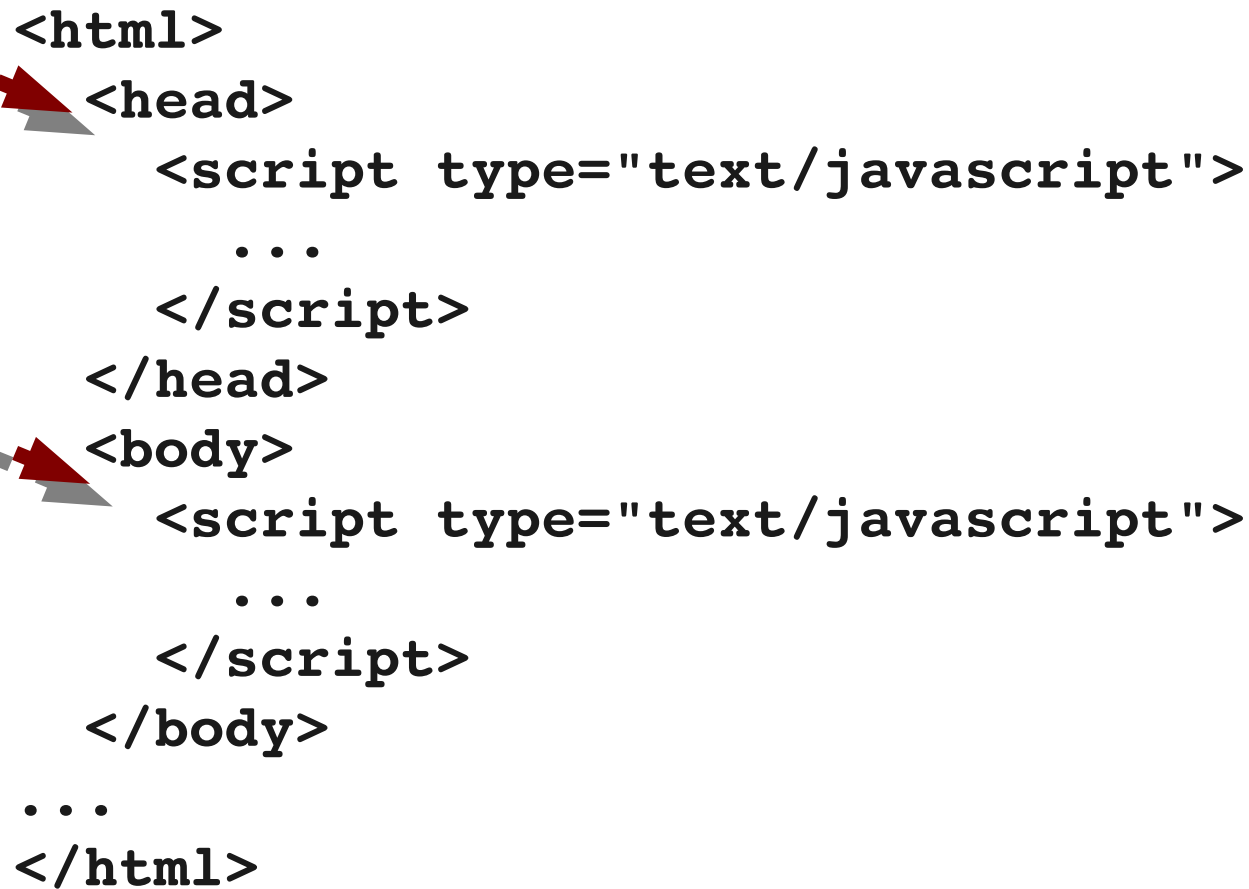
```
<html>  
  <head>  
  </head>  
  <body>  
    <script type="text/javascript">  
      ...  
    </script>  
  </body>  
  ...  
</html>
```



Javascript – Núcleo da Linguagem

- Dentro do elemento <head> e <body>

```
<html>  
  <head>  
    <script type="text/javascript">  
      ...  
    </script>  
  </head>  
  <body>  
    <script type="text/javascript">  
      ...  
    </script>  
  </body>  
  ...  
</html>
```

A diagram illustrating the placement of JavaScript code within an HTML document. The code is shown in a monospaced font within a dashed-line box. Two red dashed arrows point from the left towards the opening tags of the <script> elements inside the <head> and <body> sections. The code structure is as follows: <html>, <head>, <script type="text/javascript">, ..., </script>, </head>, <body>, <script type="text/javascript">, ..., </script>, </body>, ..., </html>.

Javascript – Núcleo da Linguagem

- Declaração externa

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

Javascript – Núcleo da Linguagem

- Exemplo 1

```
<html>
  <head>
    <script type="text/javascript">
      document.write("<h1>Exemplo 1</h1>");
    </script>
  </head>
  <body>
    <h2>Introdução JavaScript</h2>
  </body>
</html>
```

Javascript – Núcleo da Linguagem

- Refletindo sobre o Exemplo 1
 - `<h1>` e `<h2>`. Faz diferença qual utilizar?
 - O que significa **ç** e **ã**?
(*Relembrando HTML*)
 - Por que o `<h1>` apareceu antes do `<h2>`?



Javascript – Núcleo da Linguagem

- Exemplo

```
<html>
  <head>
    <title>Exemplo 2 Javascript</title>
  </head>
  <body>
    <h1>Operadores Aritmético JavaScript</h1>
    <script type="text/javascript">
      document.write("<code>1 + 2 + 3 = " + (1+2+3) +
" </code>");
    </script>
  </body>
</html>
```

Javascript – Núcleo da Linguagem

- Refletindo sobre o Exemplo 2
 - Por que utilizar o elemento `<code>` ?
(*Relembrando mais HTML*)
 - Por que o `<h1>` apareceu antes do `<code>`?
 - Faz diferença o lugar onde o elemento `<script>` é inserido ?



Javascript – Núcleo da Linguagem

- Operadores aritméticos

Operador	Operação	Exemplo
+	Adição	$x+y$
-	Subtração	$x-y$
*	Multiplicação	$x*y$
/	Divisão	x/y
%	Módulo (resto da divisão inteira)	$x\%y$
-	Inversão de sinal	$-x$
++	Incremento	$x++$ ou $++x$
--	Decremento	$x--$ ou $--x$

Javascript – Núcleo da Linguagem

- Operadores de comparação

Operador	Função	Exemplo
==	Igual a	(x == y)
!=	Diferente de	(x != y)
===	Idêntico a (igual e do mesmo tipo)	(x === y)
!==	Não Idêntico a	(x !== y)
>	Maior que	(x > y)
>=	Maior ou igual a	(x >= y)
<	Menor que	(x < y)
<=	Menor ou igual a	(x <= y)

Javascript – Núcleo da Linguagem

- Operadores bit a bit

Operador	Operação	Exemplo
&	E (AND)	(x & y)
	OU (OR)	(x y)
^	Ou Exclusivo (XOR)	(x ^ y)
~	Negação (NOT)	~x
>>	Deslocamento à direita (com propagação de sinal)	(x >> 2)
<<	Deslocamento à esquerda (preenchimento com zero)	(x << 1)
>>>	Deslocamento à direita (preenchimento com zero)	(x >>> 3)

Javascript – Núcleo da Linguagem

- Operadores de atribuição

Operador	Exemplo	Equivalente
=	x = 2	Não possui
+=	x += y	x = x + y
-=	x -= y	x = x - y
*=	x *= y	x = x * y
/=	x /= y	x = x / y
%=	x %= y	x = x % y
&=	x &= y	x = x & y
=	x = y	x = x y
^=	x ^= y	x = x ^ y
>>=	x >>= y	x = x >>= y
<<=	x <<= y	x = x <<= y
>>>=	x >>>= y	x = x >>>= y

Javascript – Núcleo da Linguagem

- Operadores Lógicos

Operador	Função	Exemplo
&&	E Lógico	(x && y)
 	OU Lógico	(x y)
!	Negação Lógica	! x

Javascript – Núcleo da Linguagem

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

```
var d = new Date();
```

```
var time = d.getHours();
```

```
if (time<10){
```

```
    document.write("<b>Good morning</b>");
```

```
} //pode terminar aqui, ou ter um else ou else
```

```
if
```

```
else if (time>10 && time<16){
```

```
    document.write("<b>Good day</b>");
```

```
} // e ainda pode ter um else
```

```
else{
```

```
    document.write("<b>Hello World!</b>");
```

```
}
```

```
</script>
```

If e else

Javascript – Núcleo da Linguagem

```
<script type="text/javascript">  
// Note that Sunday=0,Monday=1, Tuesday=2, etc.
```

Switch

```
var d=new Date();  
theDay=d.getDay();  
switch (theDay) {  
  case 5:  
    document.write("que bom sexta!!");  
    break;  
  case 6:  
    document.write("Super Sabado");  
    break;  
  default:  
    document.write("esperando o fim-de-semana!");  
}  
</script>
```

Javascript – Núcleo da Linguagem

```
<html>
  <body>
    <script type="text/javascript">
      var i=0;
      for (i=0;i<=8;i++) {
        document.write("The number is " +
i);
        document.write("<br />");
      }
    </script>
  </body>
</html>
```

For

Resultado:

The number is 0
The number is 1
The number is 2
The number is 3
The number is 4
The number is 5
The number is 6
The number is 7
The number is 8

Javascript – Núcleo da Linguagem

While

```
<html>
  <body>
    <script type="text/javascript">
      var i=0;
      while (i<=10) {
        document.write("The number is " + i);
        document.write("<br />");
        i=i+1;
      }
    </script>
  </body>
</html>
```

Resultado:

The number is 0
The number is 1
The number is 2
The number is 3
The number is 4
The number is 5
The number is 6
The number is 7
The number is 8
The number is 9
The number is 10

Javascript – Núcleo da Linguagem

function

```
<html>
  <head>
    <script type="text/javascript">
      function displaymessage(message) {
        window.alert(message);
      }
      displaymessage("utilizando
funcoes...");
    </script>
  </head>

  <body>
  </body>
</html>
```

Javascript – Núcleo da Linguagem

- Outras estruturas de controle:
 - Break Loops
 - For...In
 - Try...Catch
 - Throw
 - ...

Javascript – Núcleo da Linguagem

- Objetos Javascript ?

- String
- Date
- Array
- Boolean
- Math
- RegExp



<http://www.w3schools.com/>

Sumário da aula

- Aplicações Estáticas x Aplicações Dinâmicas
- Javascript
 - Características da linguagem
 - Núcleo da linguagem
- **HTML Events e DHTML**
- **Por que Javascript?**

HTML Events e DHTML

- Considerações a respeito de Javascript
 - Processamento em lado cliente da aplicação web
 - Disponibiliza o ferramental necessário para desenvolvimento de rotinas complexas no cliente

HTML Events e DHTML

- Considerações a respeito de Javascript
 - Processamento em lado cliente da aplicação web
 - Disponibiliza o ferramental necessário para desenvolvimento de rotinas complexas no cliente

Como utilizar isso para melhorar a interação com o usuário ?



HTML Events e DHTML

- Considerações a respeito de Javascript
 - *Javascript sozinho é insuficiente para incrementar a interação no cliente*

Necessidade de determinar formas de controle sobre os elementos HTML e interações no navegador



HTML Events e DHTML

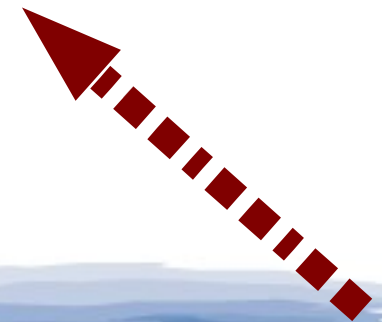
- DOM HTML
 - **Modelo de acesso e manipulação dos elementos HTML**
- HTML Events
 - **Modelo de manipulação e controle sobre eventos no HTML**

HTML Events e DHTML

- Exemplo de DOM HTML

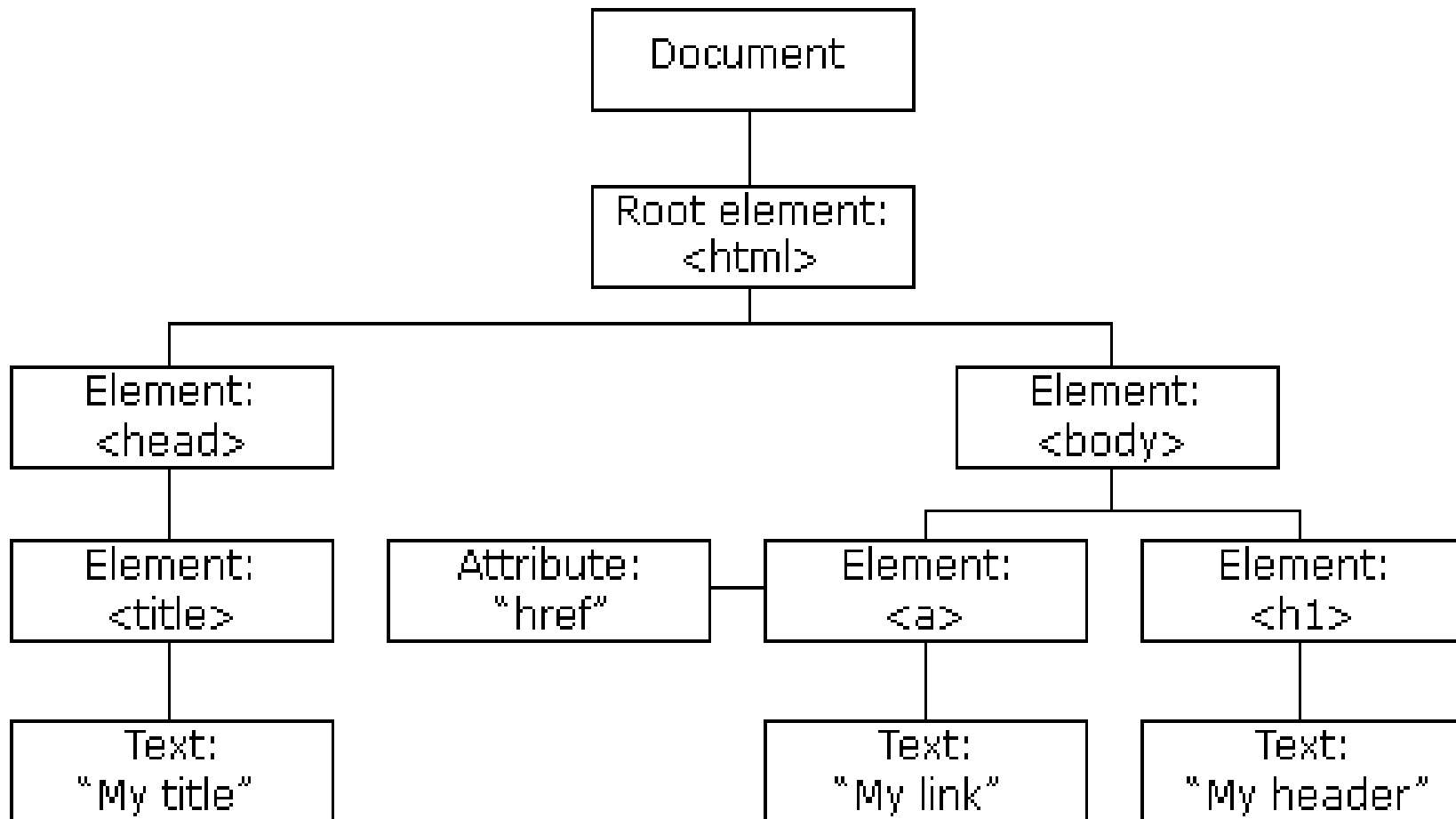


```
<html>
  <head>
    <title>Exemplo 3 Javascript</title>
  </head>
  <body>
    <h1>Javascript</h1>
    <p>Introdu&ccedil;&atilde;o ao Javascript em uma aula super
legal.<p>
    <script type="text/javascript">
      alert(document.body.childNodes[3].innerHTML);
    </script>
  </body>
</html>
```



HTML Events e DHTML

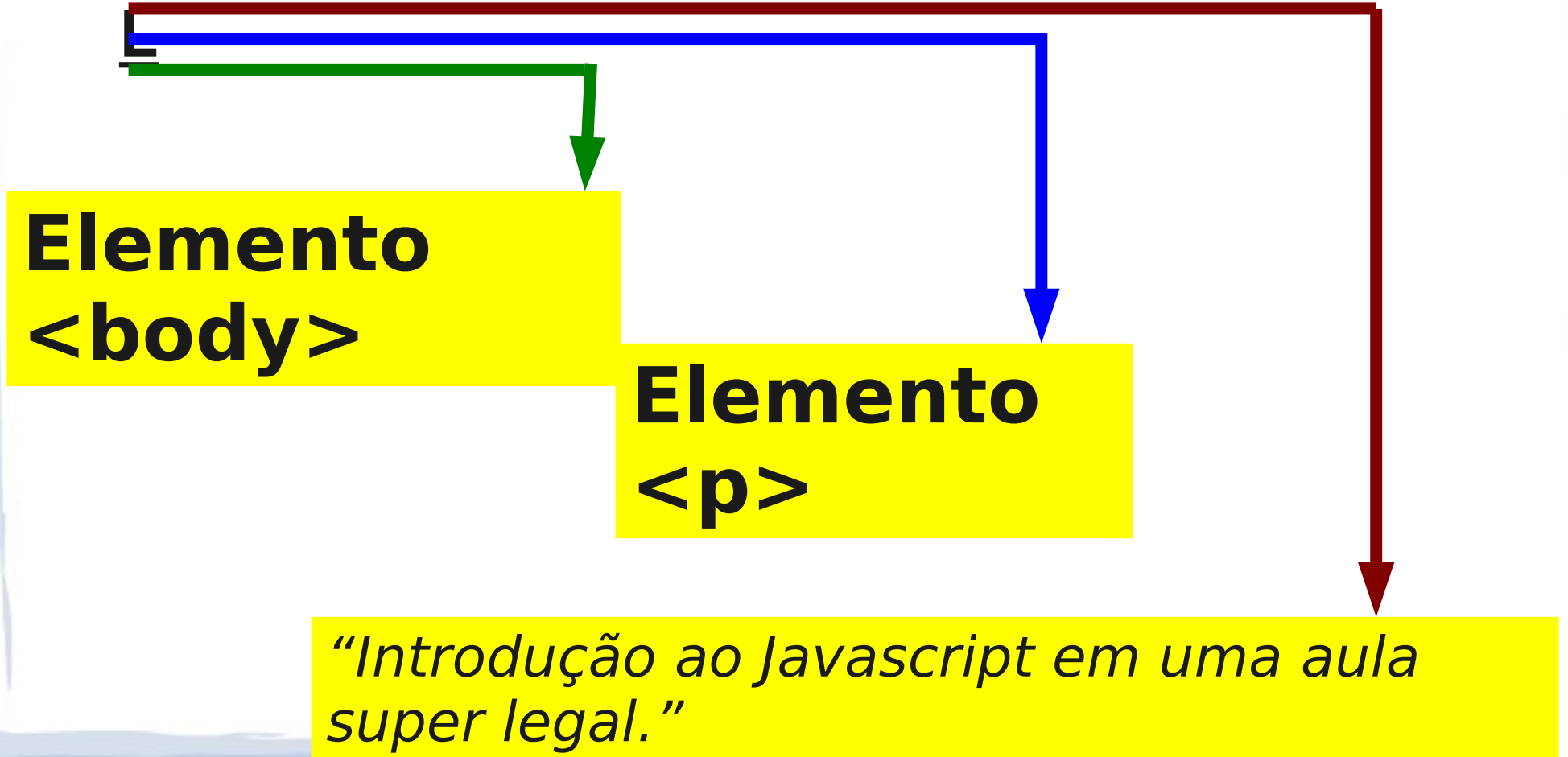
- Exemplo DOM HTML



HTML Events e DHTML

- Exemplo DOM HTML

document.body.childNodes[3].innerHTML



HTML Events e DHTML

- Exemplo DOM HTML

```
<html>
  <head>
    <title>Exemplo 4 Javascript</title>
  </head>
  <body>
    <h1>Javascript</h1>
    <p>Introdu&ccedil;&atilde;o ao Javascript em uma aula super
legal.<p>
    <script type="text/javascript">
      document.body.childNodes[3].innerHTML =
“AAAAAAAAAAAA”;
    </script>
  </body>
</html>
```

HTML Events e DHTML

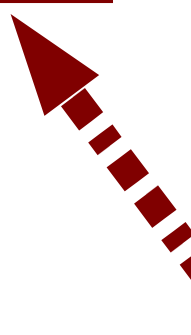
DOM HTML == DOM XML ?



HTML Events e DHTML

- Exemplo HTML Events

```
<html>
  <head>
    <title>Exemplo 5 Javascript</title>
  </head>
  <body>
    <h1>Javascript</h1>
    <p>Introdu&ccedil;&atilde;o ao <a href="#" title="teste
click">Javascript</a>.<p>
    <script type="text/javascript">
      document.getElementsByTagName("a")[0].onclick =
function(){
        alert("Evento click no link");
      };
    </script>
  </body>
```



HTML Events e DHTML

- Outra forma de declarar um *handler* de eventos

```
<a href="#" onclick="alert('click no elemento link')">link</a>
```

HTML Events e DHTML

- **Outro** exemplo HTML Events

```
<html>
```

```
<head>
```

```
<title>Exemplo 5 Javascript</title>
```

```
</head>
```

```
<body>
```

```
<h1>Javascript</h1>
```

```
<p>Introdu&ccedil;&atilde;o ao <a href="#" title="teste  
click">Javascript</a>.<p>
```

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

```
document.getElementsByTagName("p")[0].onclick =  
function(){  
    alert("Evento click");  
};
```

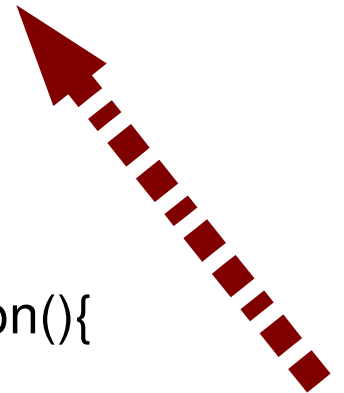
```
document.getElementsByTagName("a")[0].onclick = function(){  
    alert("Evento click no link");
```

```
};
```

```
</script>
```

```
</body>
```

```
</html>
```



HTML Events e DHTML

- Outro exemplo HTML Events

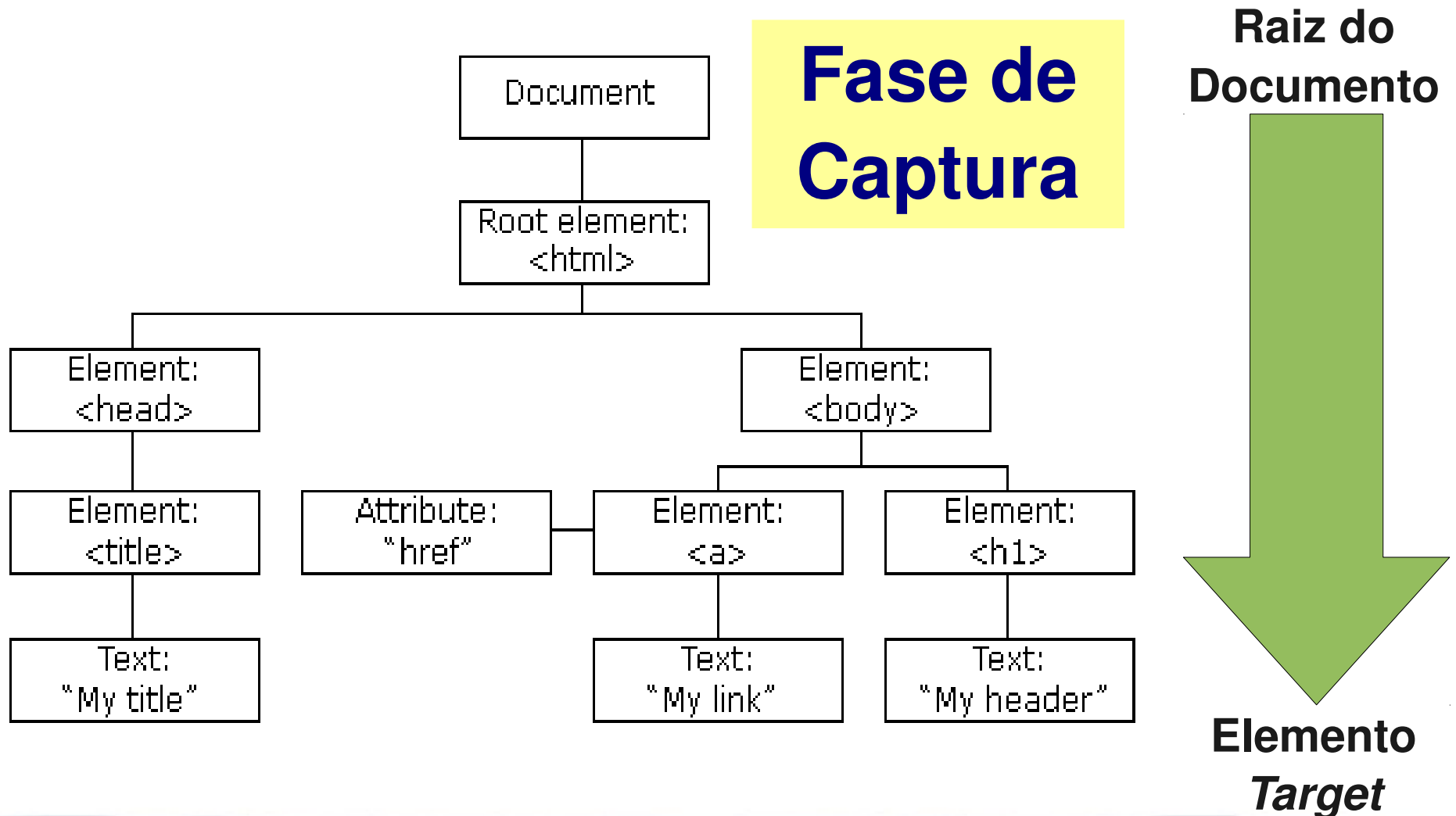


**Clique no link:
2 Alerts ?**

```
<html>
  <head>
    <title>Exemplo 5 Javascript</title>
  </head>
  <body>
    <h1>Javascript</h1>
    <p>Introdu&ccedil;&atilde;o ao <a href="#" title="teste
click">Javascript</a>.<p>
    <script type="text/javascript">
      document.getElementsByTagName("p")[0].onclick =
function(){
        alert("Evento click");
};
      document.getElementsByTagName("a")[0].onclick = function(){
        alert("Evento click no link");
      };
    </script>
```

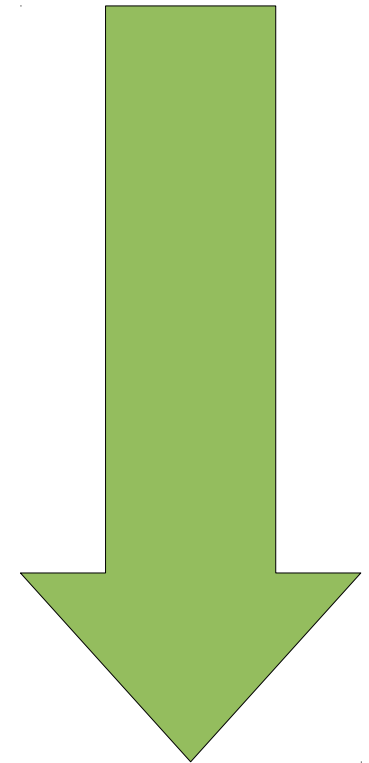
HTML Events e DHTML

- Fluxo de eventos no HTML



Fase de Captura

Raiz do Documento



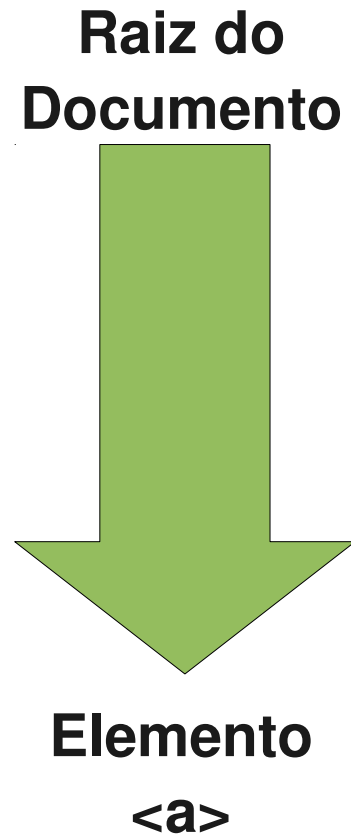
Elemento Target

HTML Events e DHTML

- Fluxo do evento no HTML no exemplo

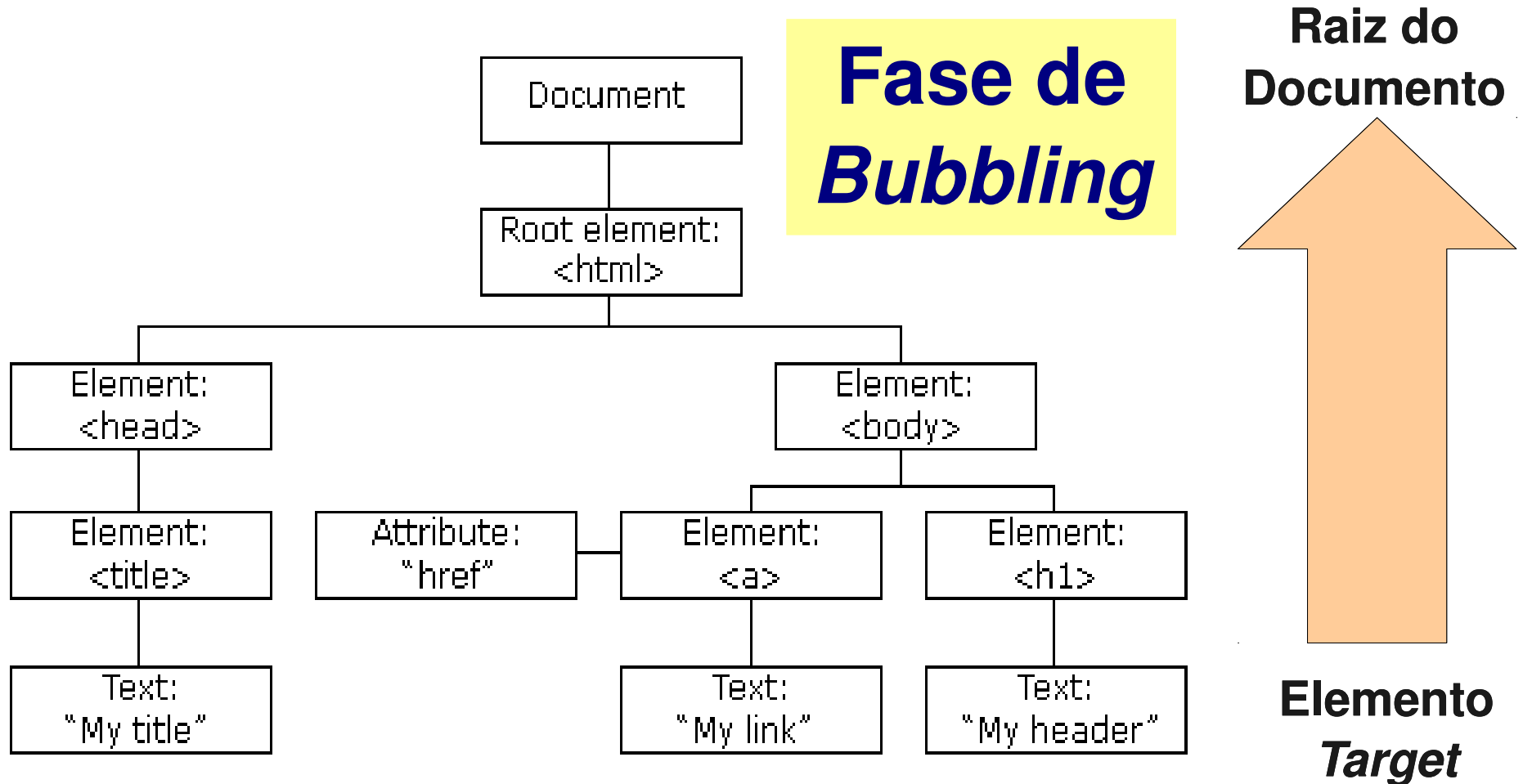
**Fase de
Captura**

*Fluxo da raiz do documento
diretamente até o elemento que
foi responsável pelo
lançamento do evento*



HTML Events e DHTML

- Fluxo de eventos no HTML

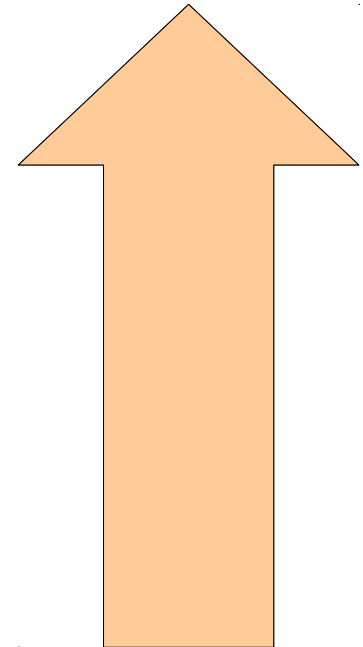


HTML Events e DHTML

- Fluxo do evento no exemplo

**Fase de
*Bubbling***

**Raiz do
Documento**



**Elemento
*Target***

Fluxo do elemento Target até a raiz do documento ativando todos os respectivos handlers para o mesmo evento.

HTML Events e DHTML

- Mais detalhes a respeito de HTML Events:

<http://www.w3.org/TR/DOM-Level-2-Events/events.html>

HTML Events e DHTML

- Mais um exemplo de HTML Events

```
<html>
  <head>
    <script type="text/javascript">
      alert(document.getElementsByTagName("h2")[0].innerHTML);
    </script>
  </head>
  <body>
    <h2>Introdu&ccedil;&atilde;o JavaScript</h2>
  </body>
</html>
```

HTML Events e DHTML

- Mais um exemplo de HTML Events

```
<html>
  <head>
    <script type="text/javascript">
      alert(document.getElementsByTagName("h2")[0].innerHTML);
    </script>
  </head>
  <body>
    <h2>Introdu&ccedil;&atilde;o JavaScript</h2>
  </body>
</html>
```



Por que não funciona ?

HTML Events e DHTML

- Mais um exemplo de HTML Events

```
<html>
  <head>
    <script type="text/javascript">
      alert(document.getElementsByTagName("h2")[0].innerHTML);
    </script>
  </head>
  <body>
    <h2>Introdu&ccedil;&atilde;o JavaScript</h2>
  </body>
</html>
```

Evento Onload ?



HTML Events e DHTML

- Importante:
 - DOM Level 2 Events não especifica todos os eventos (ex: keypress)

**Novas especificações
sob desenvolvimento**



HTML Events e DHTML

DHTML

HTML Events e DHTML

**DHTML = (X)HTML + CSS + JAVASCRIPT +
DOM HTML + HTML Events**

HTML Events e DHTML

**DHTML = (X)HTML + CSS + JAVASCRIPT +
DOM HTML + HTML Events**



Aplicações Dinâmicas

HTML Events e DHTML

- O que falta?
 - AJAX e Ajax
 - HTML 5
 - CSS 3.0
 - Canvas
 - SVG
 - Entre outras.

Sumário da aula

- Aplicações Estáticas x Aplicações Dinâmicas
- Javascript
 - Características da linguagem
 - Núcleo da linguagem
- HTML Events e DHTML
- **Por que Javascript?**

Sumário da aula

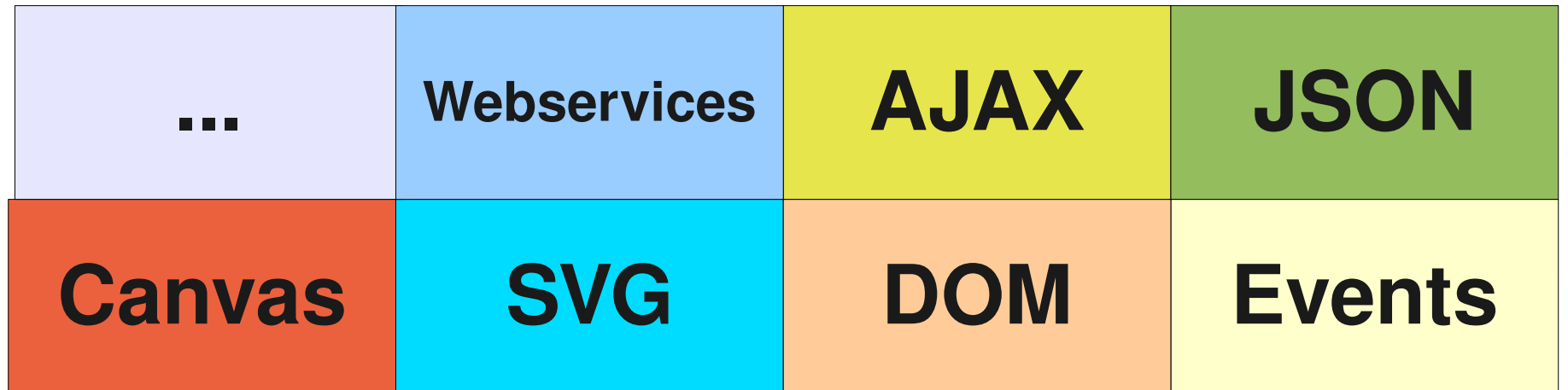
- Aplicações Estáticas x Aplicações Dinâmicas
- Javascript
 - Características da linguagem
 - Núcleo da linguagem
- HTML Events e DHTML
- **Por que Javascript?**

Por que Javascript ?

- Limitações do Javascript
 - Acesso limitado dos recursos do computador (sem banco de dados, câmera, entre outros)
 - *Same-origin policy* (limitação AJAX)
 - Limites de segurança definido pelas especificações do navegador
 - Diferenças de implementação entre os navegadores

Por que Javascript ?

- Framework Javascript



Javascript

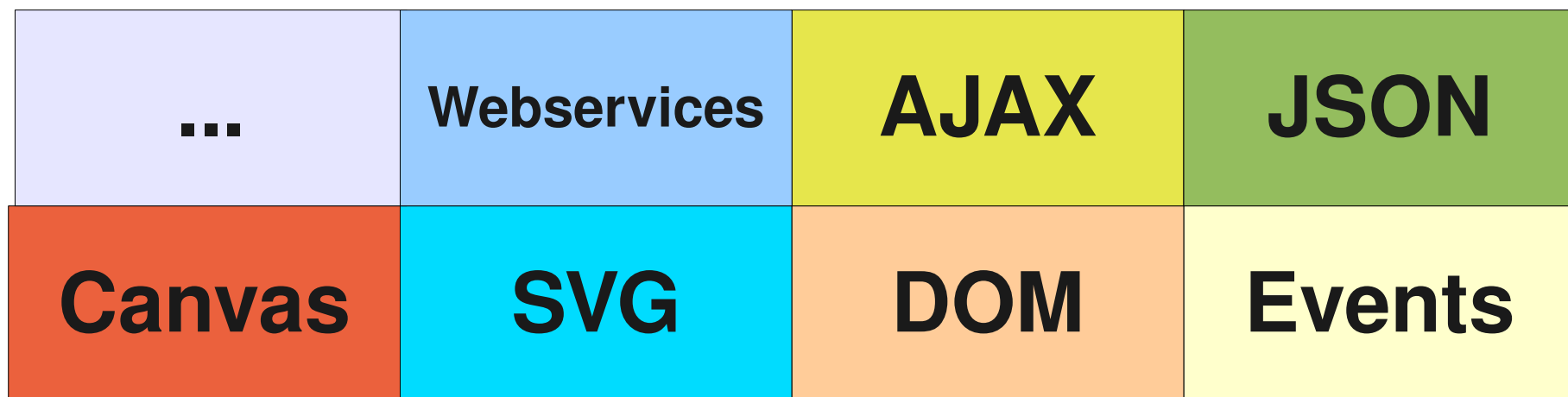
CSS

HTML

Navegador

Por que Javascript ?

- Framework Javascript – **padrões Web**



Javascript

CSS

HTML

Navegador

Tarefa para próxima aula:

- Submeter como Atividades do agora.tidia-ae.usp.br
- A partir do **html da sua grade horária**, voce deve apresentar as ementas resumidas das disciplinas ao clicar no nome OU no código de cada uma delas.
- As ementas devem ser coletadas das informações do Jupiterweb previamente e devem estar no conteúdo Html da página , sem que seja apresentado. Use CSS para fazer a apresentação do html, sem apresentar as ementas previamente coletadas. Use **Javascript** para que reaja a interação de clique sobre a disciplina mostrando a ementa devidamente apresentada com um CSS apropriado.
- As ementas devem ser “adicionadas” dinamicamente com o uso de **Javascript, DOM HTML, Events e CSS (DHTML)**.
- Tentem evitar o uso de alerts.