



Visualização de Informação com D3.js

Data-Drive Documents

Eric Macedo Cabral
cabral.eric@usp.br

Rosane Minghim
rminghim@icmc.usp.br

26/09/2019



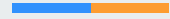
O que é o D3.js?

- Uma biblioteca da linguagem de programação *JavaScript* para visualização de informação
 - Com todas as possibilidades e liberdades que a linguagem *JavaScript* possui
- Autor: Mike Bostock
- Artigo: [D³ Data-Driven Documents](#) (IEEE Vis 2011)



O que o D3.js não é?

- Uma biblioteca gráfica
 - Embora você possa fazer gráficos com ela
- Não é uma biblioteca para desenhar em SVG
 - Embora você com certeza irá desenhar em SVG com ela



Roteiro

1. Páginas Web
2. Abordagem Prática
3. Exemplos

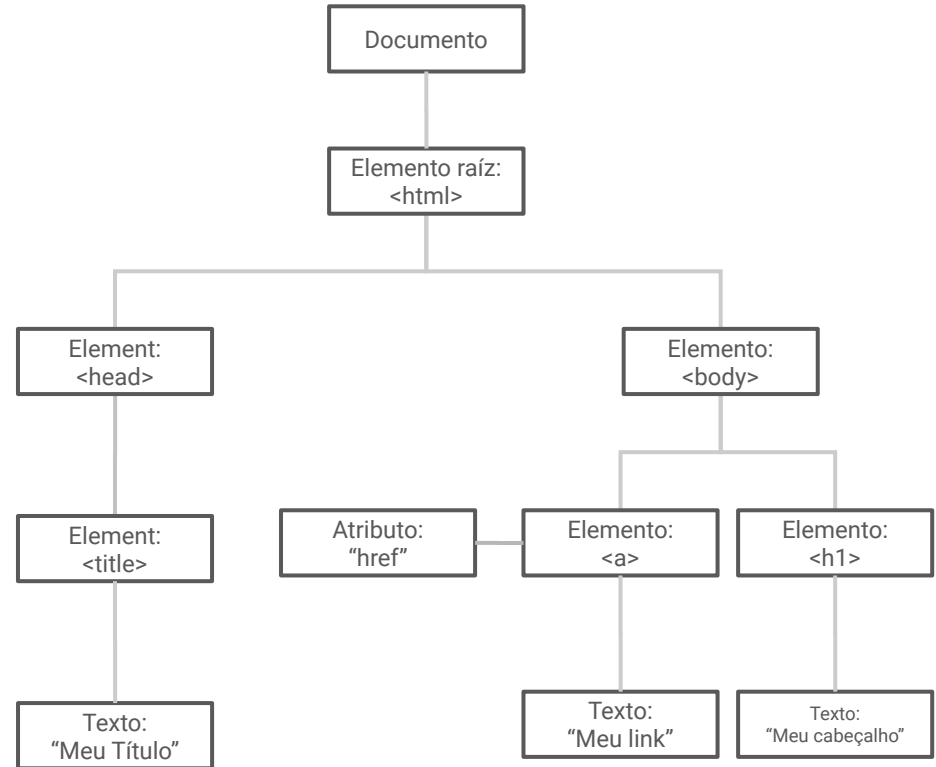


1. Páginas Web

1. DOM
2. Linguagem de Marcação
3. Estilos
4. Gráficos Vetoriais
5. JavaScript
 - a. Eventos
 - b. DOM programming Interface

Páginas Web - DOM

- Document **O**bject **M**odel (**DOM**)
- É um objeto padronizado pela W3C ¹
- Modelo hierárquico



¹ <https://www.w3.org/DOM/>



Páginas Web - Linguagem de Marcação

- HyperText Markup Language (HTML)
- **Não** é uma linguagem de programação
- Tags (Blocos)
- Hierárquico

[Exercício prático \(para casa\)](#)

```
<nome-da-tag  
  id="identificador-unico"  
  atributo="valor"  
  class="nome-da-classe">  
  <tag-interna></tag-interna>  
  <tag-interna></tag-interna>  
</nome-da-tag>
```



Páginas Web - Estilos

- Cascading **S**tyle **S**heets (CSS)
- Descreve como os elementos do DOM devem ser visualizados
- Seletores
- Classes

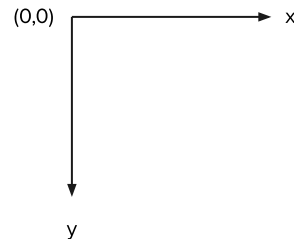
[Exercício prático \(para casa\)](#)

```
tag {  
    atributo1: valor;  
    atributo2: valor;  
}  
#id {  
    atributo: valor;  
}  
.classe {  
    atributo: valor;  
}
```


Páginas Web - Gráficos Vetoriais

- Scalable Vector Graphics (SVG)
- Define gráficos vetoriais em formato XML
- Hierárquico
- É um elemento DOM
- Primitivas básicas

[Documentação](#)



Desenha um círculo amarelo com borda verde:

```
<svg width="100" height="100">  
  <circle  
    cx="50"  
    cy="50"  
    r="40"  
    stroke="green"  
    stroke-width="4"  
    fill="yellow"/>  
</svg>
```



Páginas Web - JavaScript

- Linguagem de programação (não é de marcação)
- Objetos
- Funções

[Exercício práticos \(para casa\)](#)

```
var x = {  
    item1: 1,  
    "item2": function(param) {return 2},  
    item3: {} // Objeto vazio  
};
```

```
x.item1
```

```
<- 1
```



Páginas Web - JavaScript - Eventos

- Mudanças no DOM
- Listeners
- Callbacks
 - Função anônima

[Lista de Eventos \(Mozilla Docs\)](#)

[Exercício prático \(para casa\)](#)

Eventos em elementos:

```
<elemento event="handler()"></elemento>
```

Listeners:

```
elemento.addEventListener(  
    tipo_de_evento,  
    function(event) {} // Callback  
)
```



Páginas Web - JavaScript - DOM Programming Interface

- Acessar, manipular e remover elementos DOM
- Seletores
- Elementos DOM são definidos como objetos JavaScript

Referência

```
<div id="paragrafo"></div>
```

```
<script>
```

```
var paragrafo = document.getElementById("paragrafo");  
paragrafo.append("p").innerHTML = "Hello World";
```

```
</script>
```



2. Abordagem Prática

1. D3.js
 - a. Recapitulando
 - b. D3 Gallery
 - c. Importando
2. Dados
 - a. Seleções
 - b. Associação
 - c. Modificadores
 - d. *Data Join*
 - e. Escalas
 - f. Eixos
3. Como ler arquivos JSON?

Resumindo:



Dados

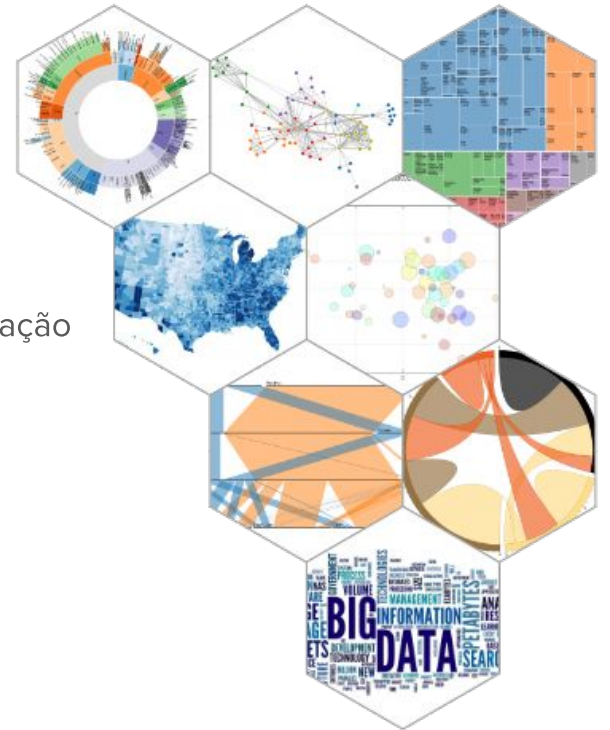


Representação
Visual

D3.js - Gallery

- Exemplos de visualizações prontas
- Plugins para problemas específicos
- Sempre veja primeiro essa galeria antes de fazer sua visualização
 - Alguém já pode tê-lo feito
 - Pode existir um plugin que facilite o desenvolvimento

[D3 Gallery](#)





D3.js - Importando

- Para adicionar qualquer script de JavaScript numa página web, usa-se a o elemento `<script>`.

Logo, para adicionar o D3.js:

```
<head>  
  <script src="https://d3js.org/d3.v5.min.js">  
  </script>  
</head>
```


Etapa 1:

1. Faça download do Projeto Inicial
2. Importe D3 no HTML
3. Adicione o script “main.js” ao final do elemento `<body>` (Por quê?)



2 minutos



Dados - Seleções

Imperativo

```
var paragraphs = document.getElementsByTagName("p");
for (var i = 0; i < paragraphs.length; i++) {
    var paragraph = paragraphs.item(i);
    paragraph.style.setProperty("color", "blue", null);
}
```

Vs.

Declarativo

- Mais intuitivo e compacto

```
d3.selectAll("p").style("color", "blue");
```

[How Selections Work](#)



Dados - Associação

- Os dados são atribuídos ao elemento DOM
- Propriedade `__data__`

Em JavaScript puro, você pode:

```
document.body.__data__ = 42;
```

Mas esse o jeito “D3” de fazer isso:

```
d3.select("body").datum(42);
```



Dados - Modificadores

- Modificadores de elementos
 - .text()
 - .property()
 - .style()
 - .attr()
- “Data-driven”

```
svg.selectAll("rect")  
  .data(dataset)  
  .enter()  
  .append("rect")  
    .attr("x", function(d, i) { return i * 5; })  
    .attr("y", function(d) { return d * 5; })  
    .attr("width", 5)  
    .attr("height", function(d) { return d * 5; });
```

Etapa 2:

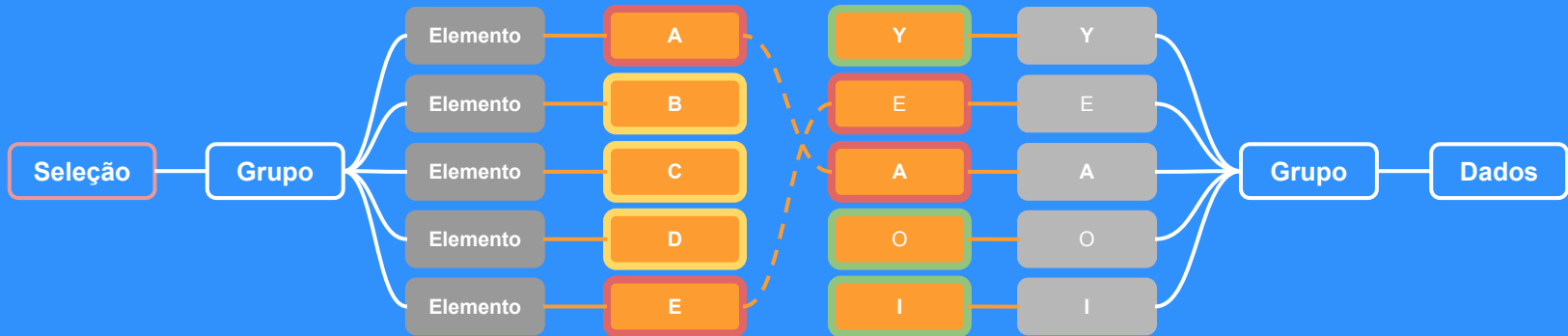
1. Use a função de eventos `.on` do D3 para “escutar” pelo evento de “mouseover” para mudar a cor da barra para vermelho
2. Faça o mesmo para o evento “mouseout” para mudar a barra de volta para azul
3. Procure a documentação sobre eventos no D3



10 minutos

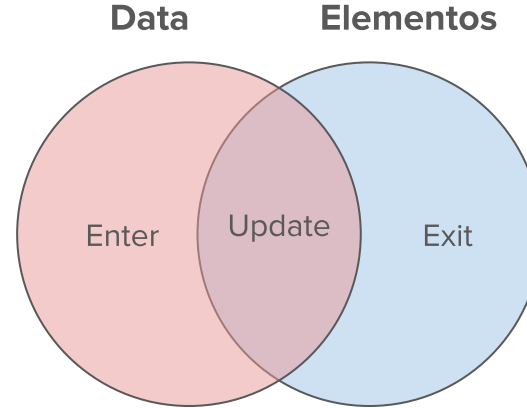
Data Join:

- Update
- Enter
- Exit



Dados - *Data Join*

- Operação de conjuntos entre dados e elementos



`selection.data.enter()`

Retorna os dados que não possuem elementos no grupo selecionado

`selection.data.update()`

Contém os elementos que já existem na seleção e devem ser atualizados

`selection.data.exit()`

Contém elementos que existem no grupo mas que não estão no novo conjunto de dados

Etapa 3:

1. Mude o tipo do elemento “#barChat” para **SVG**
2. Modifique o script de forma que as barras sejam representadas por elementos “rect” verticais
3. A altura da barra é dada por “value”



10 minutos



Dados - Escalas

- Funções
- Mapeiam um intervalo em outro

[Documentação de Escalas \(Github\)](#)

```
var yScale = d3.scaleLinear()  
  .range([svgHeight, 0]) // int. alvo  
  .domain([ // int. original  
    0, d3.max(dataset)  
  ]);
```



Dados - Eixos

- Mostra os valores para os quais as escalas mapeiam
- Auxilia a visualização

```
var yAxis = d3.axisLeft(yScale);
```

```
svg.append("g")  
  .attr("transform", "translate(30, 0)")  
  .call(yAxis);
```

[Documentação para Eixos \(Github\)](#)

Etapa 4:

1. Faça um mapeamento dos valores de “value” de *svgHeight* até 0 (Por quê?)
2. Crie um eixo vertical com os valores escalares criados em Y

Visite a documentação do D3



10 minutos



4. Lendo arquivos JSON

1. O padrão JSON
2. Como ler arquivos JSON locais com D3



O Padrão JSON

- JavaScript Object Notation (JSON)
- Muito utilizado para troca de mensagens entre serviços na web
- Sintaxe do objetos em JavaScript
- Não é utilizado apenas em JavaScript

[Documentação](#)

```
var myObj = {
    name: "John",
    age: 31,
    city: "New York"
};
var myJSON = JSON.stringify(myObj); //Objeto-String
// --- //
var myJSON = '{"name":"John", "age":31, "city":"New York"}';
var myObj = JSON.parse(myJSON); // String-Objeto
```



Como ler arquivos JSON locais com D3?

Etapa 5:

1. Leia o arquivo “data.json” e atribua à variável “data”
2. Faça as alterações necessárias para adaptar os dados aos exemplos anteriores

Visite a documentação do D3



15 minutos



Referências

- [Curso do VICG - Visualização de Informação na Web com D3.js](#) (Fábio Sikansi e Samuel Fadel)
- Tutoriais e exercícios: [W3C Schools](#)
- [How Selections Work](#) (Mike Bostock)
- [a fun, difficult introduction to d3](#) (Tom MacWright)
- [D3.js Github's Wiki Page](#)

Obrigado!

Dúvidas?

Contato: cabral.eric@usp.br