

Banco de Dados

Bancos de Dados de Documentos+Objetos

André Santanchè e Patrícia Cavoto
Instituto de Computação - UNICAMP
Outubro 2015

Document Databases

- XML-based
 - BaseX (<http://basex.org>)
- JSON
 - CouchDB (<http://couchdb.apache.org>)
 - Mongo DB (<http://www.mongodb.org>)

Modelo Lógico

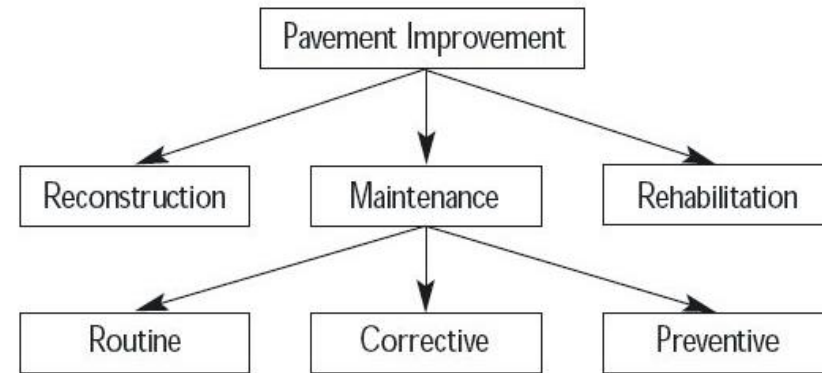
Flat File Model

	Route No.	Miles	Activity
Record 1	I-95	12	Overlay
Record 2	I-495	05	
Record 3	SR-301	33	

Relational Model

Activity Code	Activity Name
23	Patching
24	Overlay
25	Crack Sealing

Hierarchical Model



Object-Oriented Model

Object 1: Maintenance Report Object 1 Instance

Date	01-12-01
Activity Code	24
Route No.	I-95
Daily Production	2.5
Equipment Hours	6.0
Labor Hours	6.0

Object 2: Maintenance Activity

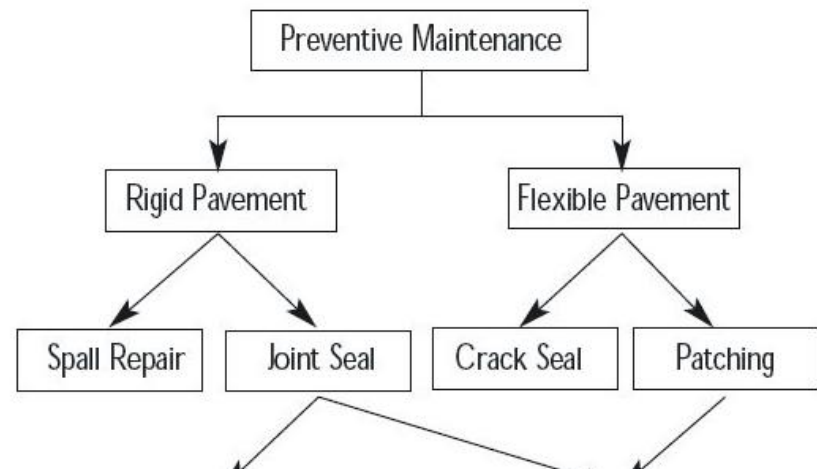
Activity Code	
Activity Name	
Production Unit	
Average Daily Production Rate	

Key = 24

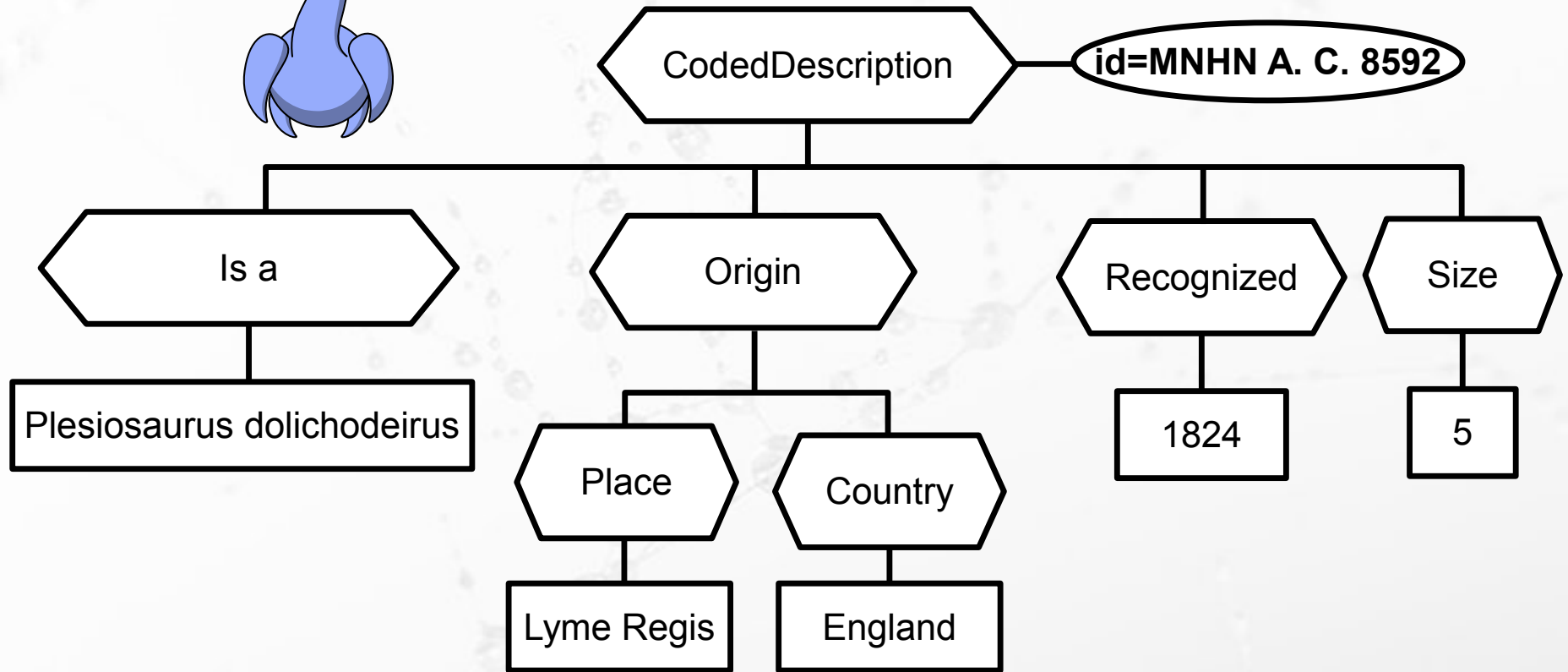
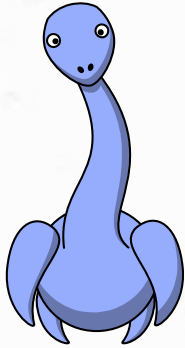
Activity Code	Date	Route No.
24	01/12/01	I-95
24	02/08/01	I-66

By Marcel Douwe Dekker
https://en.wikipedia.org/wiki/File:Database_models.jpg

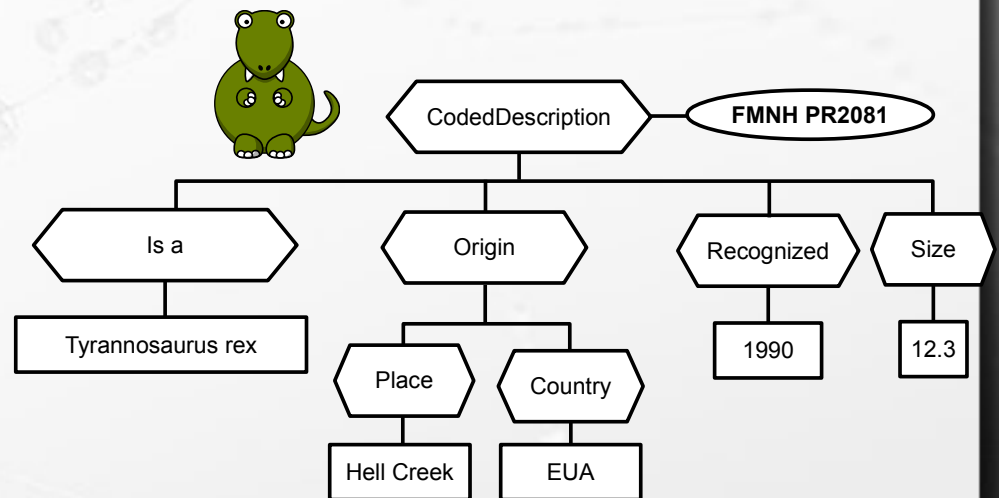
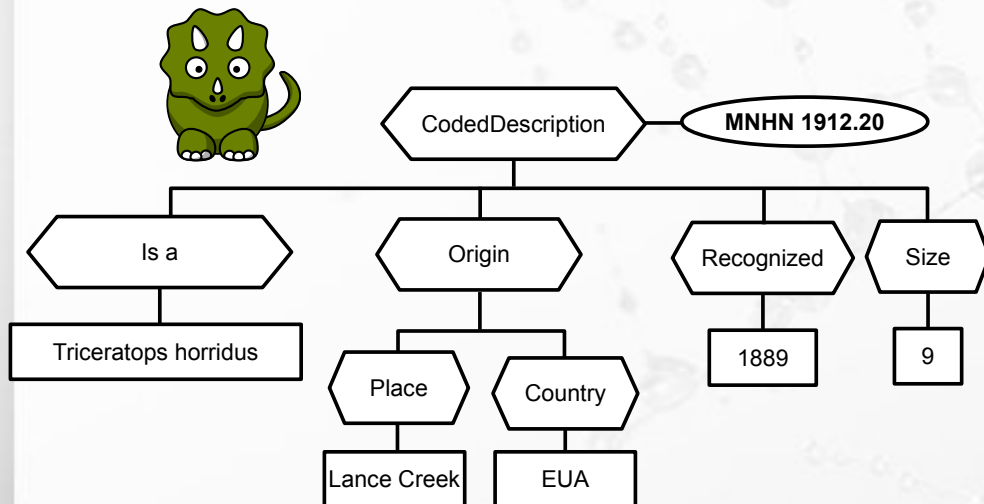
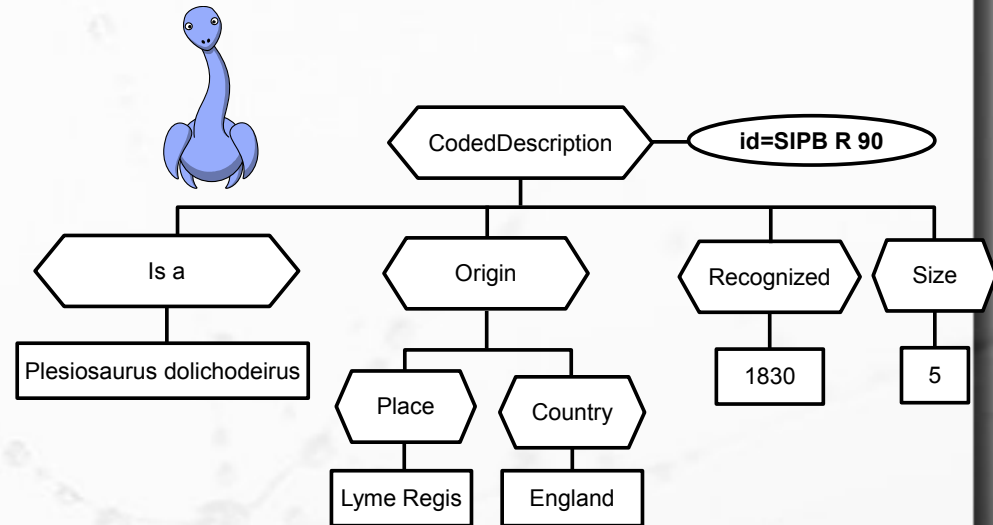
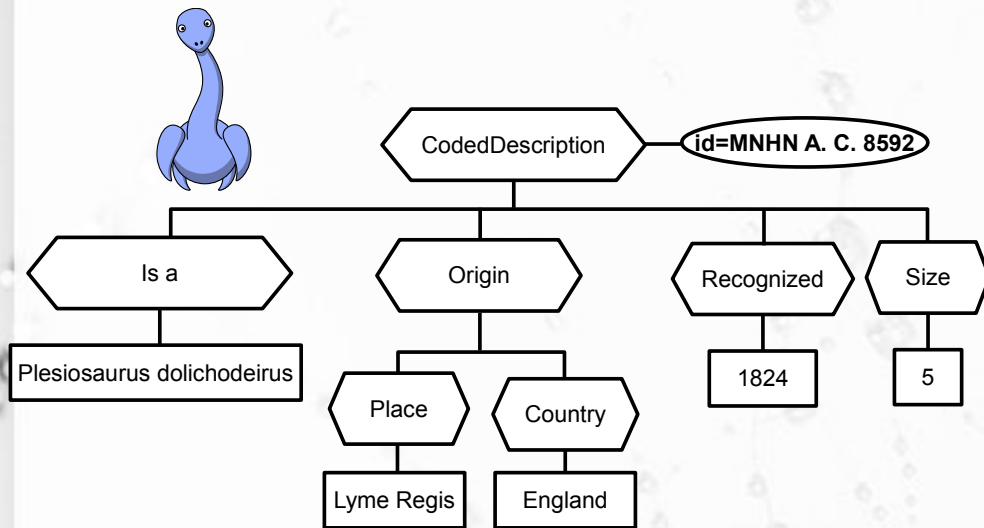
Network Model



Modelo Hierárquico (Documentos/XML)

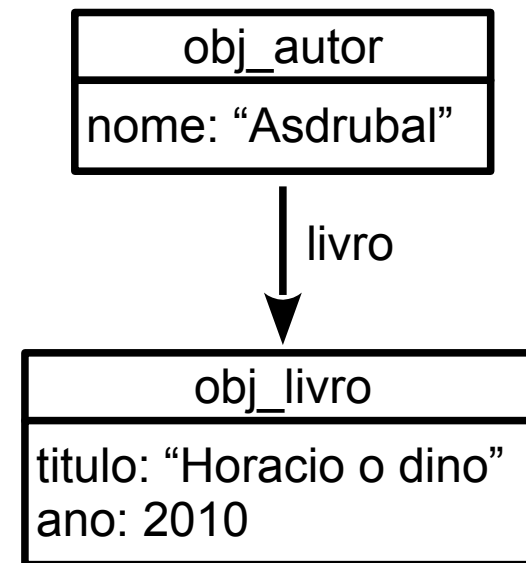


Modelo Hierárquico (Documentos/XML)



Modelo Hierárquico (JSON - Documentos + OO)

```
{  
  "nome": "Asdrubal",  
  "livro": {  
    "titulo": "Horacio o dino",  
    "ano": 2010  
  }  
}
```

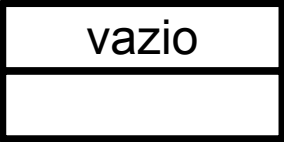
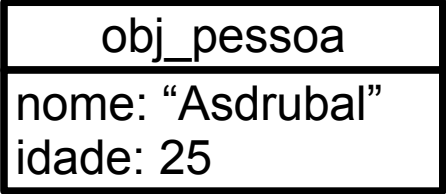
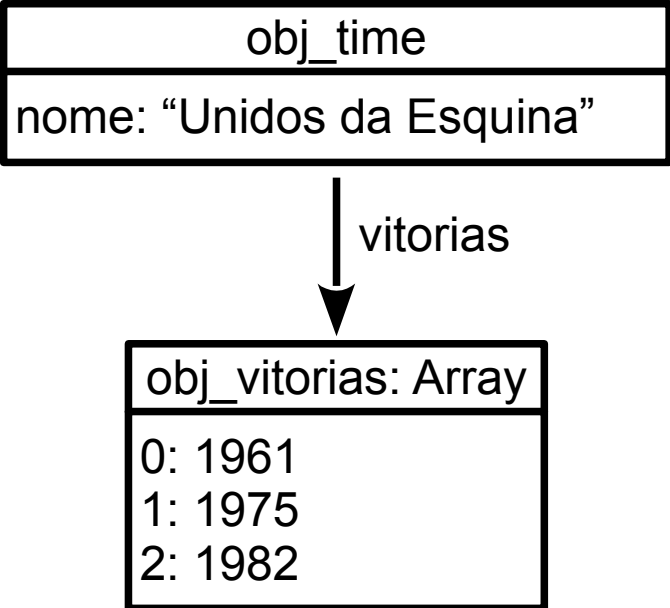


JavaScript

- Inventada por Brendan Eich na Netscape
- Incorporada no Internet Explore como Jscript
- ECMAScript → padronização (Ecma, 2011)
- Originalmente para pequenos scripts no navegador
- Limitações de acesso ao hardware local
 - segurança
 - independência de plataforma

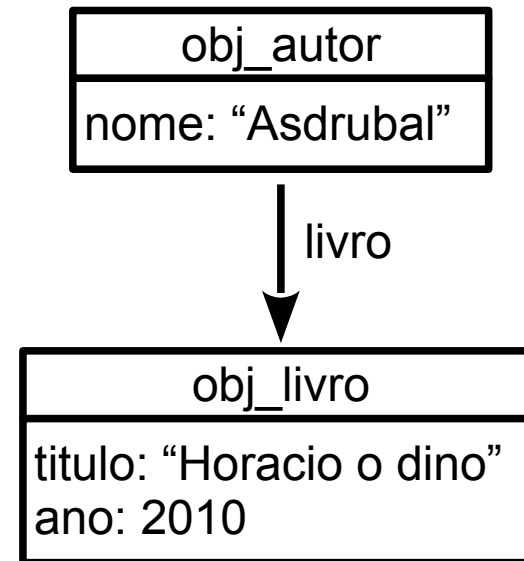
Objetos em JavaScript

Objetos JS

<pre>{ }</pre>	
<pre>{ "nome": "Asdrubal", "idade": 25 }</pre>	
<pre>{ "nome": "Unidos da Esquina", "vitorias": [1961, 1975, 1982] }</pre>	

Objetos JS

```
{  
  "nome": "Asdrubal",  
  "livro": {  
    "titulo": "Horacio o dino",  
    "ano": 2010  
  }  
}
```



JavaScript

Objeto com Atributos (1)

```
function exemploObjetoAtributos1() {  
  var circulo = {  
    centroX : 5,  
    centroY : 3,  
    raio : 10  
  };  
  
  alert("Circulo: centro(" + circulo.centroX + ", " +  
    circulo.centroY + "), raio " + circulo.raio);  
}
```

JavaScript

Objeto com Atributos (2)

```
function exemploObjetoAtributos2() {  
    var circulo = {};  
  
    circulo.centroX = 5;  
    circulo.centroY = 3;  
    circulo.raio = 10;  
  
    alert("Circulo: centro(" + circulo.centroX + ", " +  
circulo.centroY + "), raio " + circulo.raio);  
}
```

JavaScript

Objeto com Método (1)

```
function exemploObjetoMetodo1() {  
  var circulo = {  
    centroX : 5,  
    centroY : 3,  
    raio : 10,  
    area : function() {  
      return 3.1416 * this.raio * this.raio;  
    }  
  };  
  
  alert("Circulo: centro(" + circulo.centroX + ", " +  
    circulo.centroY + "), raio " + circulo.raio + ",  
    area " + circulo.area());  
}
```

JavaScript

Objeto com Método (2)

```
function exemploObjetoMetodo3() {  
    var circulo = {};  
  
    circulo.centroX = 5;  
    circulo.centroY = 3;  
    circulo.raio = 10;  
    circulo.area = function() {  
        return 3.1416 * this.raio * this.raio;  
    }  
  
    alert("Circulo: centro(" + circulo.centroX + ", " +  
        circulo.centroY + "), raio " + circulo.raio +  
        ", area " + circulo.area());  
}
```

JSON

JavaScript Object Notation

- Padrão aberto de intercâmbio de objetos
- Baseado na notação JavaScript
- Incorporado ao ECMAScript (Ecma, 2011)
- Adotado por diversas linguagens (<http://json.org/>)

Stringify

- Serializando

```
var pessoa = {  
  "nome": "Asdrubal",  
  "idade": 25  
};  
var pessoaStr = JSON.stringify(pessoa);
```

- Desserializando

```
var pessoa2 = JSON.parse(pessoaStr);
```

MongoDB
(<http://www.mongodb.org>)

Iniciando Servidor

- `mongod --dbpath "c:\Andre\db\mongodb\mc536-2015-2-a"`

MongoDB Shell

- `mongo.exe`

Insert simples

```
db.autores.insert(  
  {  
    "nome": "Asdrubal",  
    "idade": 25  
  }  
);
```

563c873745400d49a72753c5

nome: "Asdrubal"
idade: 25

Insert Hierárquico (embedded)

```
db.autores.insert(  
  {  
    "nome": "Doriana",  
    "livro": {  
      "titulo": "Horacio o dino",  
      "ano": 2010  
    }  
  }  
);
```

563c871c45400d49a72753c4

nome: "Doriana"

livro

titulo: "Horacio o dino"
ano: 2010

Find

```
db.autores.find();
```

```
db.autores.find({"nome": "Doriana"});
```

```
db.autores.find({"livro.titulo": "Horacio o  
dino"});
```

Update

```
db.autores.update(  
  { "nome" : "Doriana" },  
  { $set: { "livro.ano": 2015 } }  
);
```


Update

```
db.autores.update(  
  { "nome": "Asdrubal" },  
  { "nome": "Asdrubal",  
    "livro": {  
      "titulo": "Vida sentimental dos dinossauros",  
      "ano": 2013  
    }  
  }  
);
```

Remove

```
db.autores.remove({"nome": "Doriana"});
```

Modelos de Relações



Embedded

Livro Embedded em Autor bloco autores

```
db.autores.insert(  
  {  
    "nome": "Doriana",  
    "livro": {  
      "titulo": "Horacio o dino",  
      "ano": 2015  
    }  
  }  
);
```

563c871c45400d49a72753c4
nome: "Doriana"



titulo: "Horacio o dino" ano: 2015

Livro Embedded em Autor bloco autores

```
db.autores.insert(  
  { "nome": "Asdrubal",  
    "livro": {  
      "titulo": "Vida sentimental dos dinossauros",  
      "ano": 2013  
    }  
  }  
);
```

563c873745400d49a72753c5

nome: "Asdrubal"

livro

titulo: "Vida sentimental dos dinossauros"
ano: 2013

Referências Cruzadas

Insert com “_id” manual bloco livros

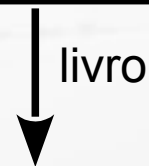
```
db.livros.insert(  
  {  
    "_id": "livro-como-entender",  
    "titulo": "Como entender seu dino",  
    "ano": 2015  
  }  
)
```

livro-como-entender
titulo: "Como entender seu dino" ano: 2015

Insert referência a livro bloco autores

```
db.autores.insert(  
  {  
    "_id": "autor-alcebiades",  
    "nome": "Alcebiades",  
    "idade": 32,  
    "livro": "livro-como-entender"  
  }  
)
```

autor-alcebiades
nome: "Alcebiades" idade: 32



livro-como-entender
titulo: "Como entender seu dino" ano: 2015

Insert com “_id” automático

bloco livros

```
livro_id = ObjectId()
```

```
db.livros.insert(  
  {  
    "_id": livro_id,  
    "titulo": "Dinos para leigos",  
    "ano": 2014  
  }  
)
```

563c878645400d49a72753c6

titulo: "Horacio o dino"
ano: 2015

Insert referência a livro bloco autores

```
db.autores.insert(  
  {  
    "nome": "Melissa",  
    "idade": 29,  
    "livro": livro_id  
  }  
)
```

563c878745400d49a72753c7
nome: "Melissa" idade: 29



563c878645400d49a72753c6
titulo: "Horacio o dino" ano: 2015



Iteração

Guardando find em variável bloco autores

```
var todosAutores = db.autores.find();
```

```
todosAutores;
```

Iterando variável

```
var todosAutores = db.autores.find();  
while (todosAutores.hasNext()) {  
    print(tojson(todosAutores.next()));  
}
```

```
var umLivro = db.livros.find();  
while (umLivro.hasNext()) {  
    print(tojson(umLivro.next()));  
}
```

Simulando um Join

Simulando Join

```
var todosLivros = db.livros.find();
while (todosLivros.hasNext()) {
    var umLivro = todosLivros.next();
    print(tojson(umLivro));
    var todosAutores =
        db.autores.find({"livro": umLivro._id});
    while (todosAutores.hasNext()) {
        print(tojson(todosAutores.next()));
    }
}
```


Filtrando um Livro

```
var todosLivros = db.livros.find({"titulo": "Dinos
para leigos"});

while (todosLivros.hasNext()) {
    var umLivro = todosLivros.next();
    print(tojson(umLivro));
    var todosAutores = db.autores.find({"livro":
umLivro._id});
    while (todosAutores.hasNext()) {
        print(tojson(todosAutores.next()));
    }
}
```



André Santanchè

<http://www.ic.unicamp.br/~santanche>

License

- These slides are shared under a Creative Commons License. Under the following conditions: Attribution, Noncommercial and Share Alike.
- See further details about this Creative Commons license at: <http://creativecommons.org/licenses/by-nc-sa/3.0/>