

The background is a dark blue gradient with a subtle pattern of white dots. Overlaid on this are several concentric circles and arcs in a lighter blue color. Some of these arcs have degree markings, such as 140, 150, 160, 170, 180, 190, 200, 210, 220, 230, 240, 250, and 260. There are also small white arrows pointing in various directions, suggesting a sense of rotation or movement.

FRAMEWORK MOOTOOLS

AMAURY PETRINI ET PABLO GABOLDE

SOMMAIRE

- Introduction
- Modules intégrés
- Syntaxe
- Conclusion (Avantages / Inconvénients)

SOMMAIRE

- Introduction
- Modules intégrés
- Syntaxe
- Conclusion (Avantages / Inconvénients)

INTRODUCTION

- Créée en **2006**, par **Valerio Proietti** & les **Mootools developers**
- **Open source**
- **Compact** (Taille totale : 158 ko)
- **Orienté objet**
- **Modulaire** (téléchargement personnalisable)

SOMMAIRE

- Introduction
- **Modules intégrés**
- Syntaxe
- Conclusion (Avantages / Inconvénients)

MODULES INTÉGRÉS

1. Classes:

MooTools Class

A simple MooTools Class example.

```
var Animal = new Class({
  initialize: function(age){
    this.age = age;
  }
});
var Cat = new Class({
  Extends: Animal,
  initialize: function(name, age){
    this.parent(age); // calls initialize method of
    Animal class
    this.name = name;
  }
});
var myCat = new Cat('Micia', 20);
alert(myCat.name); // alerts 'Micia'.
alert(myCat.age); // alerts 20.
```

MODULES INTÉGRÉS

2. Tableaux:

Implement

This method implements a new method to the Array type's prototype.

```
Array.implement('limitTop', function(top){
    for (var i = 0, l = this.length; i < l; i++){
        if (this[i] > top) this[i] = top;
    }
    return this;
});

[1, 2, 3, 4, 5, 6].limitTop(4); // returns [1, 2, 3, 4, 4, 4]
```

SOMMAIRE

- Introduction
- Modules intégrés
- **Syntaxe**
- Conclusion (Avantages / Inconvénients)

SYNTAXE

Manipulation du DOM :

```
// get elements by class
$$('.foo'); // or even: document.getElementsByTagName('.foo');

// selector with different elements
$$('div.foo, div.bar, div.bar a');

// get a single element
document.getElementById('div.foo'); |
```

AJAX :

```
// create a new Class instance
var myRequest = new Request({
  url: 'getMyText.php',
  method: 'get',
  onRequest: function(){
    myElement.set('text', 'loading...');
  },
  onSuccess: function(responseText){
    myElement.set('text', responseText);
  },
  onFailure: function(){
    myElement.set('text', 'Sorry, your request failed :(');
  }
});

// and to send it:
myRequest.send(data);
```

SYNTAXE

Création d'un élément :

```
// the short way
new Element('div#bar.foo');

// using the element constructor
new Element('div', {
  'class': 'foo',
  id: 'bar'
});
```

Evenements :

```
// attach a click event on an element
myElement.addEvent('click', function(){
  alert('clicked!');
});

// attach several events at a time
myElement.addEvents({
  mouseover: function(){
    alert('mouseover');
  },
  click: function(){
    alert('click');
  }
});

// remove a event
myElement.removeEvent(type, fn);
```

SOMMAIRE

- Introduction
- Modules intégrés
- Syntaxe
- Conclusion (Avantages / Inconvénients)

CONCLUSION

- **Avantages :**
 - Gratuit (Open-Source)
 - Utilisé par 5% des sites web ➔ 2° au classement
 - Syntaxe plus proche du JS (simple à apprendre)
- **Faiblesses :**
 - Lenteur au niveau performance (mais téléchargement personnalisable pour garantir une meilleur perf)
 - Nouveauté (version 1.6.0) ➔ Classe « thenable ».