



Breizh C@mp
Mix de technologies

ES6 dès aujourd'hui

BABEL

#ES6Today #BzhCmp



#ES6Today #BzhCmp

Grégory Houllier

@ghoullier

Architecte Web, passionné par le Front-End

Co-Fondateur de @RennesJS

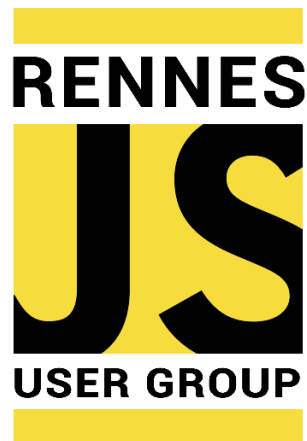


#ES6Today #BzhCmp

@RennesJS

rennesjs.org

meetup.com/RennesJS/



Meetup le dernier jeudi de chaque mois

25/06/2015 à Epitech



#ES6Today #BzhCmp

EcmaScript

Version standardisée de Javascript

Actuellement en version 5.1 (2009)



#ES6Today #BzhCmp

ES6 c'est quoi?

Ensemble de nouvelles APIs et Syntaxes

++productivity && ++expressivity

Finalisé en Juin 2015

Harmony > ES6 > ES2015



#ES6Today #BzhCmp

Les influences



Langage transpilé en Javascript créé fin 2009 par Jérémie Ashkenas (créateur de Backbone et Underscore).

- Classes
- Arrow Functions
- Rest+Spread



#ES6Today #BzhCmp

Les influences



Plateforme basée sur le moteur Javascript V8 de Chrome
créé en 2009

- Dynamisation de l'écosystème Javascript
- Modules CommonJS



#ES6Today #BzhCmp

BABEL c'est quoi?

Un transpileur Javascript

Transforme un code ES6 en code ES5
interprétable sous IE9+, FF19+, CH23+...

Anciennement 6to5, esnext



#ES6Today #BzhCmp

Sous le capot (1/2)

Parser

Convertit le code Javascript en AST

Transformer

Manipule l'AST

Generator

Convertit l'AST en Javascript



#ES6Today #BzhCmp

Sous le capot (2/2)

```
{  
  type: "Program",  
  body: [{  
    type: "VariableDeclaration",  
    kind: "var",  
    declarations: [{  
      type: "VariableDeclarator",  
      id: { type: "Identifier", name: "foo" },  
      params: [],  
      body: [{  
        type: "BlockStatement",  
        body: [{  
          type: "ReturnStatement",  
          argument: { type: "Identifier", name: "bar" }  
        }]  
      }]  
    }]  
  }]  
}
```

AST

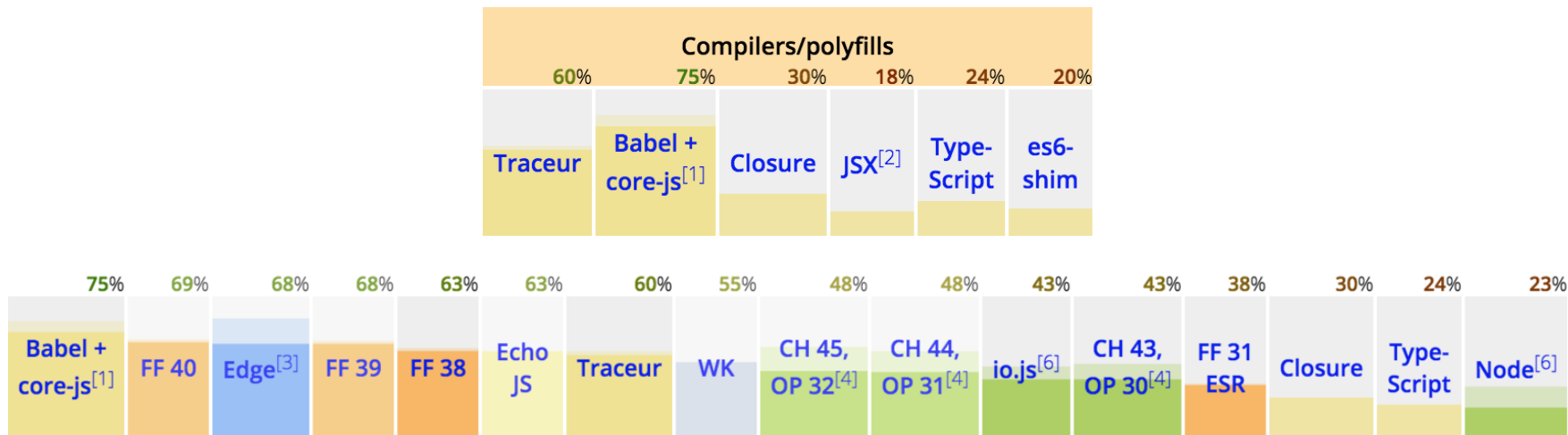
AST: Abstract Synthax Tree



#ES6Today #BzhCmp

Pourquoi choisir *BABEL*?

kangax.github.io/compat-table/es6/





#ES6Today #BzhCmp

Qui utilise *BABEL*?

babeljs.io/users/





#ES6Today #BzhCmp

BABEL



☐ Experimental ☐ Loose mode ☒ Evaluate

```
1 class View {
2   constructor(name) {
3     this.name = name;
4   }
5   render() {
6     return ``;
7   }
8 }
9
10 class HomeView extends View {
11   constructor(name) {
12     super(name);
13   }
14   render() {
15     return `<section class="View View--Home"></section>`;
16   }
17 }
```

```
1 "use strict";
2
3 var _get = function get(_x, _x2, _x3) { var _again = true; _
4
5 var _createClass = (function () { function defineProperties(
6
7 function _inherits(subClass, superClass) { if (typeof superC
8
9 function _classCallCheck(instance, Constructor) { if (!(insti
10
11 var View = (function () {
12   function View(name) {
13     _classCallCheck(this, View);
14
15     this.name = name;
16   }
17
18   _createClass(View, [{
19     key: "render",
20     value: function render() {
21       return "";
22     }
23   }]);
24
25   return View;
26 })());
27
```



#ES6Today #BzhCmp

Démarrer avec *BABEL*

```
# ProTip™ .bashrc
export PATH=$PATH:./node_modules/.bin
```

CLI

```
# Installation locale
npm install babel -S
```

```
# Transpiler vos sources ES6 en ES5
babel src/main.js -o dist/main.js
```

```
# Executer vos sources ES6
babel-node src/server.js
```



#ES6Today #BzhCmp

Débugger avec *BABEL*

```
# Install dependencies  
npm i -S browserify babelify watchify
```

CLI

```
# Browserify + debug  
browserify src/main.js -t babelify --debug --outfile dist/bundle.js
```

```
# Watching  
watchify src/main.js -t babelify --debug --outfile dist/bundle.js
```

Démo



#ES6Today #BzhCmp

Les nouveautés ES6

Question de timing on ne verra pas tout ...



#ES6Today #BzhCmp

Classes

"Enfin!!! Moi qui avait rien compris aux prototypes ça tombe bien"

Et non ... c'est juste du sucre
syntactique à l'approche prototypal



#ES6Today #BzhCmp

Classes

```
function Talk(date, subject) {  
  this.date = date;  
  this.subject = subject;  
}  
Talk.prototype.getDate = function getDate() {  
  return this.date;  
};  
Talk.prototype.getSubject = function getSubject() {  
  return this.subject;  
};  
Talk.isValid = function isValid(talk) {  
  return null !== talk.subject;  
};
```

ES5



#ES6Today #BzhCmp

Classes

```
function JsTalk() {  
  Talk.prototype.constructor.call(this, Date.now(), 'JS');  
}  
JsTalk.prototype = Object.create(Talk.prototype);
```

ES5



#ES6Today #BzhCmp

Classes

Utilité: ★★☆☆

Support:

- Babel: **25/30**
- Browsers: FF 39+, (Chrome, Edge, io.js) w/ flag

Limitations:

- Pas de propriétés static
- Attention à la chaine de résolution des prototypes
- Pas de hoisting



#ES6Today #BzhCmp

Arrow Functions





#ES6Today #BzhCmp

Arrow Functions

```
// Verbose
var getTitle = (item) => {
  return item.title;
};

// Implicit return
var getTitle = (item) => item.title;

// Optional parenthesis
var getTitle = item => item.title;
```

ES6



#ES6Today #BzhCmp

Arrow Functions

```
function clicker(node) {  
  this.counter = 0;  
  node.addEventListener((event) => {  
    // Lexical binding sur le context parent  
    ++this.counter;  
    arguments[0] === event; // false  
    arguments[0] === node; // true  
  });  
}
```

ES6



#ES6Today #BzhCmp

Arrow Functions

Utilité: ★★☆☆

Support:

- Babel: **9/11**
- Browsers: FF 39+

Limitations:

- Ce ne sont pas des constructeurs (pas de new)
- .call .apply et .bind ne change pas le this



#ES6Today #BzhCmp

Modules

"Plus d'excuses pour pourrir le scope global"

Fortement inspiré de CommonJS



#ES6Today #BzhCmp

Modules

```
// ./controller/home.js
```

ES6

```
export default class HomeController {  
  ...  
}
```

```
// ./main.js
```

```
import HomeController from './controller/home.js';
```



#ES6Today #BzhCmp

Modules

```
// ./utils.js  
  
export function mixin() {  
  ...  
}  
  
export const API_VERSION = 1;
```

```
// ./main.js  
  
import * as utils from './utils';  
  
utils.API_VERSION; // 1
```

ES6



#ES6Today #BzhCmp

Modules

```
// ./utils.js
```

```
function mixin() {  
  ...  
}
```

```
const API_VERSION = 1;
```

```
// Une instruction pour les exporter tous  
export { mixin, API_VERSION };
```

ES6



#ES6Today #BzhCmp

Modules

Utilité: ★★★★★

Support:

- Babel: **5/6***
- Browsers: N/A

Limitations:

- Nécessite un environnement CommonJS (Node, Browserify, ...) même avec Babel



#ES6Today #BzhCmp

Template Strings

"Fini la concaténation de chaines !!"

backtick, un nouveau delimiter pour ne plus avoir à choisir entre simple et double quotes



#ES6Today #BzhCmp

Template Strings

```
// Template simple
function text(messages) {
  return `you have ${ messages } message${ messages !== 1 ? `s`
}

// Template avancé
function html(list) {
  return `<ul>${list.map((item) => {
    return `<li>${item.value}</li>`;
  })}</ul>`;
}
```

ES6



#ES6Today #BzhCmp

Template Strings

```
// Evaluation
var welcome = `Welcome ${user} at Breizhcamp!`;
// user is not defined

// Multi lines
var multilines =
  `Header

  Content

  Footer`;
```

ES6



#ES6Today #BzhCmp

Template Strings

Et côté performance?

Testing in Chrome 43.0.2357.81 on OS X 10.10.3		
	Test	Ops/sec
Lodash	<pre>lodashTemplate({ name1: 'Joe', name2: 'Richard', name3: 'Larry' });</pre>	636,758 ±1.34% 93% slower
es6 template	<pre>es6template('Joe', 'Richard', 'Larry');</pre>	9,076,827 ±2.37% fastest



#ES6Today #BzhCmp

Template Strings

Utilité: ★★★★★

Support:

- Babel: **3/3**
- Browsers: Chrome 43+, FF 38+, Edge, io.js



#ES6Today #BzhCmp

Object Literals

"Don't repeat yourself"

Propriétés raccourcies,
méthodes, propriétés calculées



#ES6Today #BzhCmp

Object Literals

```
function todo() {}
```

ES6

```
var context = {  
  // Raccourci  
  todo,  
  // Propriété calculée  
  ['uniq' + Date.now()]: 'test',  
  // Méthode  
  toString() {  
    return 'another value';  
  }  
};
```



#ES6Today #BzhCmp

Object Literals

Utilité: ★☆☆☆

Support:

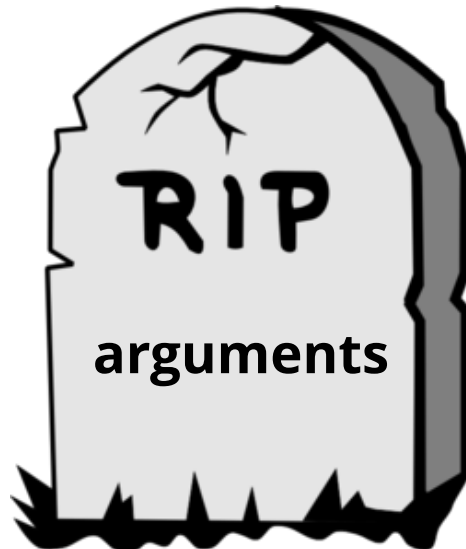
- Babel: **6/6**
- Browsers: Chrome 44+, FF 38+, Edge, io.js



#ES6Today #BzhCmp

Rest, Spread, Default

"Une gestion des arguments aux petits oignons"





#ES6Today #BzhCmp

Rest

```
function log(level, ...messages) {  
  console.log(`${level} ${messages.length} messages`);  
}  
  
log('warn');  
//> warn 0 messages  
  
log('debug', 'hello', 'breizhcamp', 2015);  
//> debug 3 messages
```

ES6



#ES6Today #BzhCmp

Spread

```
var args = [  
  'click',  
  function (event) { console.log(event); },  
  false  
];  
  
document.addEventListener.apply(document, args);
```

ES5



#ES6Today #BzhCmp

Default

```
function img(src, height = 100, width = 100) {  
  return `}  
  
img('avatar.png');  
//>   
  
img('player.png', 50, 50);  
//> 
```

ES6



#ES6Today #BzhCmp

Rest, Spread, Default

Utilité: ★★☆☆

Support:

- Babel: **19/21**
- Browsers: FF 38+, Edge

Limitations:

- Rest n'est pas aussi puissant qu'en Coffee



#ES6Today #BzhCmp

Destructuring

"Prenez une aspirine..."

Extraction des variables d'un objet



#ES6Today #BzhCmp

Destructuring

```
// Object Destructuring
var node = document.querySelector('input[type=password]');
var { top, right, left, bottom } = getComputedStyle(node);

// Array Destructuring
var pattern = /(\d\d)\.(\d\d)\.(\d\d\d\d)/;
var value = '07/05/2015';
var [all, day, month, year] = pattern.exec(value);
// ou
var [, day, , year] = pattern.exec(value);
```

ES6



#ES6Today #BzhCmp

Destructuring

```
// Object Destructuring
var node = document.querySelector('input[type=password]');

console.log(node)
//> {
  name: 'mypassword',
  value: '1234',
  style: {
    top: 0,
    ...
  }
  ...
}
var { name, value: password, style: { top } } = node;
```

ES6

```
var name = node.name;
var password = node.value;
var top = node.style.top;
```

ES5



#ES6Today #BzhCmp

Destructuring

```
// Parameters Destructuring + Default
function open({ src, readonly = true }) {
  console.log(src, readonly);
}

let src = '/tmp/user.json';

open({src, readonly: false});
//> '/tmp/user.json', false

open({src});
//> '/tmp/user.json', true
```

ES6



#ES6Today #BzhCmp

Destructuring

Utilité: ★★☆☆

Support:

- Babel: **27/30**
- Browsers: Partiel dans FF 38+

Remarques:

- Fortement intégré dans la syntaxe des modules



#ES6Today #BzhCmp

Generators

"Les machines à états, c'est mon dada"

```
function* () {} // Return Generator
```




#ES6Today #BzhCmp

Generators

```
function* workflow(initial = 0) {  
  var step = initial;  
  console.log('step', step);  
  yield 'step' + step;  
  
  ++step;  
  console.log('step', step);  
  yield 'step' + step;  
}
```

```
var w = workflow(1);
```

```
w.next(); // {"done":false, "value":"step1"}  
w.next(); // {"done":false, "value":"step2"}  
w.next(); // {"done":true}
```

ES6



#ES6Today #BzhCmp

Generators

```
var app = require('koa')();  
// x-response-time  
app.use(function *(next){  
  var start = new Date;  
  yield next;  
  var ms = new Date - start;  
  this.set('X-Response-Time', ms + 'ms');  
});  
// logger  
app.use(function *(next){  
  var start = new Date;  
  yield next;  
  var duration = new Date - start;  
  console.log(`${this.method} ${this.url} - ${duration}`);  
});  
// response  
app.use(function *(){  
  this.body = 'Hello World';  
});  
app.listen(3000);
```



#ES6Today #BzhCmp

Generators

Utilité: ★★☆☆

Support:

- Babel: **20/21**
- Browsers: Partiel (FF 38+, Chrome 43+, Node 0.12)



#ES6Today #BzhCmp

Block declaration

"let, const is the new var"

Limiter la portée d'une variable
Eviter les collisions de nommage



#ES6Today #BzhCmp

Block declaration

```
if (true) {  
  let first = 0;  
}  
console.log(first, second);  
//> first is undefined
```

```
if (false) {  
  function test() {}  
}  
console.log(test);  
//> test is undefined
```

```
const VERSION = 1.1;  
VERSION = 2.0;  
//> "VERSION" is read-only
```

ES6



#ES6Today #BzhCmp

Block declaration

Utilité: ★★☆☆

Support:

- Babel: **14/18**
- Browsers: Partiel (FF 38+, Chrome 43+, Node 0.12)

Remarques:



#ES6Today #BzhCmp

Les autres nouveautés

*"Promise, Set, Map, WeakMap, Symbol,
Proxy, for...of"*



#ES6Today #BzhCmp

Promise

```
function after(delay = 1000) {  
  return new Promise((resolve, reject) => {  
    setTimeout(resolve, delay);  
  });  
}  
  
after(100).then(() => {  
  console.log('exectued after 100ms');  
});
```

ES6



#ES6Today #BzhCmp

Set, Map, WeakMap

```
// Map fonctionne comme un dictionnaire
var map = new Map();
map.set('key', 'value');

// WeakMap permet d'utiliser un objet comme clé
var map = new WeakMap();
var key = new Date();
map.set(key, 'value');

// Set stock uniquement des valeurs
var set = new Set([1, 2, 3, 2, 3, 4]);
set.values()
//> [1, 2, 3, 4]
```

ES6



#ES6Today #BzhCmp

Symbol

```
var Api = (function() {  
  const secret = Symbol('secret');  
  const API_SECRET = 'ThisIsSecret';  
  return class Api {  
    constructor() {  
      this[secret] = API_SECRET;  
    }  
    static isValid(Api) {  
      return Api[secret] === API_SECRET;  
    }  
  }  
})();  
  
var api = new Api();  
console.log(Api.isValid(api)); // true  
console.log(Api.secret); // undefined
```

ES6



#ES6Today #BzhCmp

Proxy

```
// Proxying a normal object
var target = {};
var handler = {
  get: (receiver, name) => {
    return `Hello, ${name}!`;
  }
};
var p = new Proxy(target, handler);
p.world === "Hello, world!";

// Proxying a function object
var target = function () { return "I am the target"; };
var handler = {
  apply: (receiver, ...args) => {
    return "I am the proxy";
  }
};
var p = new Proxy(target, handler);
p() === "I am the proxy";
```

ES6



#ES6Today #BzhCmp

Quelques liens

<http://babeljs.io>

<https://github.com/babel>

<http://git.io/es6features>

<https://kangax.github.io/compat-table/es6/>

<http://www.2ality.com>

<http://es6-features.org>

<https://speakerdeck.com/sebmck/javascript-transformation-jsconf-2015>



#ES6Today #BzhCmp

Questions?





#ES6Today #BzhCmp

Bonus: Aperçu d'ES7

*"Async functions, Object.observe,
autobind"*