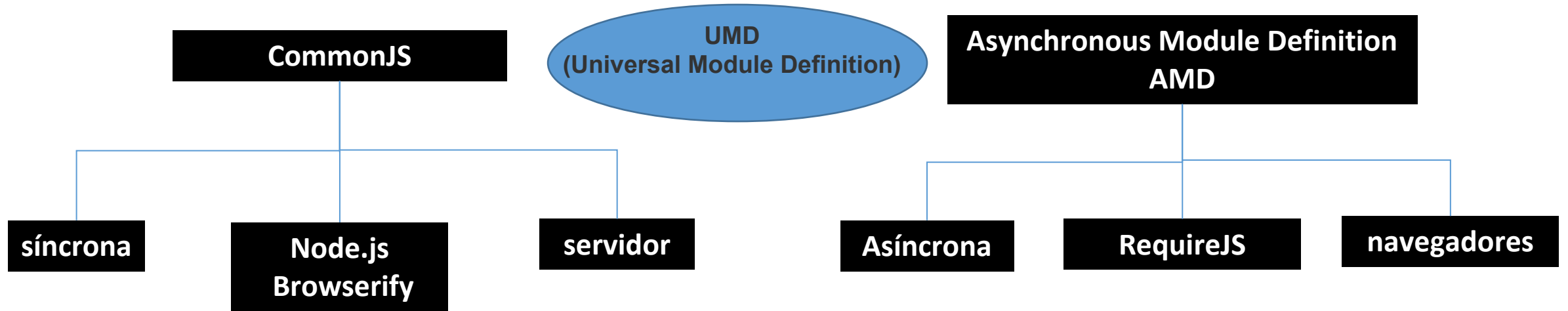# MÓDULOS EN ES6

Ronald Pérez Álvarez

Luiz Quelves da Silva

Módulos | Uso de Hacks (ES5)

Módulos | export e import

Export

```
export var numeros = {
  numero1: 5,
  numero2: 2,
};

export {resta,multiplicación};

export {división as div};

export default function sumar(numero1, numero2) {

  return numero1 + numero;
}

function resta(numero1, numero2) {

  return numero1 - numero;
}

function multiplicación(numero1, numero2) {

  return numero1 * numero;
}

function división (numero1,numero2){
    return numero1/numero2;
}
```

import

```
import suma from 'operaciones';

import {resta} from 'operaciones';

import {multiplicacion as mult} from 'operaciones';

import {numero1, numero2} from 'operaciones';


console.log(sumar(2,2));
console.log(resta(2,2));
console.log(mult(2,2));
console.log(numero1+numero2);

/*if (Math.random()) {
    import 'foo'; // SyntaxError
}

// You can't even nest `import` and `export`
// inside a simple block:
{
    import 'foo'; // SyntaxError
}*/
```

Módulos | import

```
import name from "module-name";
import { member } from "module-name";
import { member as alias } from "module-name";
import { member1 , member2 } from "module-name";
import { member1 , member2 as alias2 , [...] } from "module-name";
import name , { member [ , [...] ] } from "module-name";
import "module-name";
```

```html
<script type="module">
    import $ from 'lib/jquery';
    var x = 123;

    // The current scope is not global
    console.log('$' in window); // false
    console.log('x' in window); // false

    // `this` still refers to the global object
    console.log(this === window); // true
</script>
```

Fuente:

https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/Sentencias/import
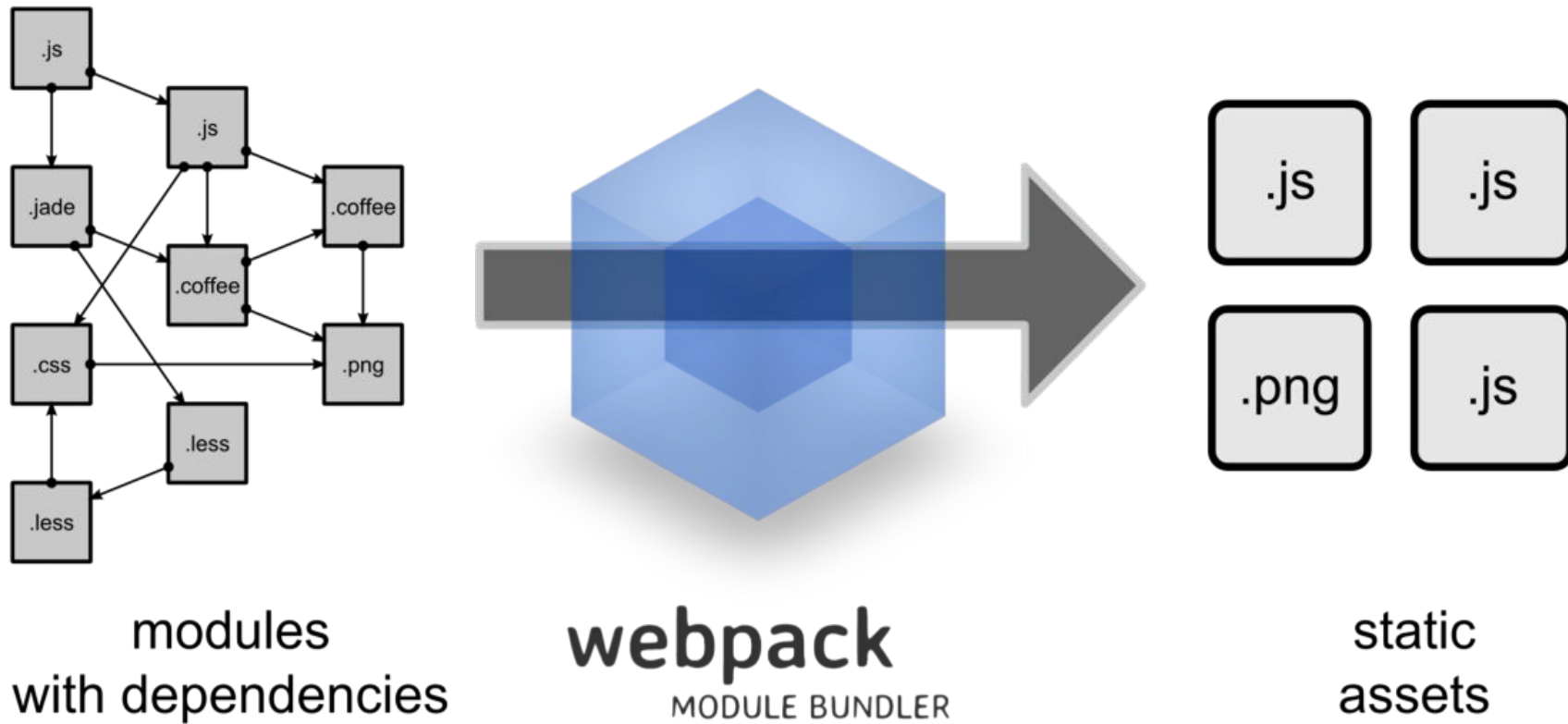
Módulos | export

```
export { name1, name2, …, nameN };
export { variable1 as name1, variable2 as name2, …, nameN };
export let name1, name2, …, nameN; // also var
export let name1 = …, name2 = …, …, nameN; // also var, const

export default expression;
export default function (…) { … } // also class, function*
export default function name1(…) { … } // also class, function*
export { name1 as default, … };

export * from …;
export { name1, name2, …, nameN } from …;
export { import1 as name1, import2 as name2, …, nameN } from …;
```

Fuente: https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/Sentencias/export

Módulos | webpack

modules
with dependencies

webpack
MODULE BUNDLER

static
assets

Módulos | referencias

- (Descripción general)   https://www.mozilla-hispano.org/es6-en-detalle-modulos/

- (Uso export) http://www.cristalab.com/tutoriales/uso-de-modulos-en-javascript-con-ecmascript-6-c114342l/

- (Descripción general, Sintaxis) https://carlosazaustre.es/blog/ecmascript-6-el-nuevo-estandar-de-javascript/

- (Módulos ES6 y ES5) http://datyayu.xyz/es6-modulos/

- (Repositorio UMD) https://github.com/umdjs/umd

- (export e import ,sintaxis) https://developer.mozilla.org/es/docs/Web/JavaScript/Referencia/Sentencias

- (webpack, configuración) https://medium.com/@sergiodxa/compilando-el-frontend-con-webpack-d251f7a632ec#.56flsqsw7

- (Módulos, Sintaxis) http://exploringjs.com/es6/ch_modules.html