



# Curso de Javascript

Unidad Didáctica 17: Otras utilidades



Ayuntamiento  
de Vitoria-Gasteiz  
Vitoria-Gasteizko  
Udala

# Índice de contenidos

- Introducción
- Mejorar el rendimiento de las aplicaciones complejas
- Ofuscar el código JavaScript
- Conclusiones



# Introducción

En esta unidad veremos distintas utilidades que nos resultarán útiles a la hora de utilizar Javascript



# Mejorar el rendimiento de las aplicaciones complejas

Cuando se desarrollan aplicaciones complejas, es habitual encontrarse con decenas de archivos JavaScript de miles de líneas de código

Estructurar las aplicaciones de esta forma es correcto y facilita el desarrollo de la aplicación, pero penaliza en exceso el rendimiento de la aplicación



# Mejorar el rendimiento de las aplicaciones complejas

La primera recomendación para mejorar el rendimiento de la aplicación consiste en unir en un único archivo JavaScript el contenido de todos los diferentes archivos JavaScript



# Mejorar el rendimiento de las aplicaciones complejas

En Windows, se puede crear un pequeño programa ejecutable que copia el contenido de varios archivos JavaScript en uno solo:

```
more archivo1.js > archivoUnico.js
```

```
more archivo2.js >> archivoUnico.js
```

```
more archivo3.js >> archivoUnico.js
```

...



# Mejorar el rendimiento de las aplicaciones complejas

La primera instrucción tiene un solo símbolo > para borrar el contenido del archivoUnico.js cada vez que se ejecuta el comando

El resto de instrucciones tienen un símbolo >> para añadir el contenido de los demás archivos al final del archivoUnico.js



# Mejorar el rendimiento de las aplicaciones complejas

En sistemas operativos de tipo Linux es todavía más sencillo unir varios archivos en uno solo:

```
cat archivo1.js archivo2.js archivo3.js >  
    archivoUnico.js
```





# Mejorar el rendimiento de las aplicaciones complejas

La única consideración que se debe tener en cuenta con este método es el de las dependencias entre archivos



# Mejorar el rendimiento de las aplicaciones complejas

Si por ejemplo el archivo1.js contiene funciones que dependen de otras funciones definidas en el archivo3.js, los archivos deberían unirse en este otro orden:

```
cat archivo3.js archivo1.js archivo2.js >  
archivoUnico.js
```



# Mejorar el rendimiento de las aplicaciones complejas

Otra recomendación muy útil para mejorar el rendimiento de la aplicación es la de comprimir el código de JavaScript. Este tipo de herramientas compresoras de código no modifican el comportamiento de la aplicación, pero pueden reducir mucho su tamaño



# Mejorar el rendimiento de las aplicaciones complejas

El proceso de compresión consiste en eliminar todos los espacios en blanco sobrantes, eliminar todos los comentarios del código y convertir toda la aplicación en una única línea de código JavaScript muy larga. Algunos compresores van más allá y sustituyen el nombre de las variables y funciones por nombres más cortos



# Mejorar el rendimiento de las aplicaciones complejas

Closure Compiler es una de las herramientas proporcionadas por Google y que permite compilar nuestro código JavaScript, realizando un análisis del mismo, eliminando código innecesario, re-escribiéndolo y minimizando el código resultante. Además, comprueba nuestro código en busca de errores de sintaxis o mejoras que podamos aplicar



# Ofuscar el código Javascript

El código de las aplicaciones JavaScript, al igual que el resto de contenidos de las páginas web, está disponible para ser accedido y visualizado por cualquier usuario

Con la aparición de las aplicaciones basadas en AJAX, muchas empresas han desarrollado complejas aplicaciones cuyo código fuente está a disposición de cualquier usuario



# Ofuscar el código Javascript

Aunque se trata de un problema casi imposible de solucionar, existen técnicas que minimizan el problema de que se pueda acceder libremente al código fuente de la aplicación

La principal técnica es la de ofuscar el código fuente de la aplicación



# Ofuscar el código Javascript

Los ofuscadores utilizan diversos mecanismos para hacer casi imposible de entender el código fuente de una aplicación

Manteniendo el comportamiento de la aplicación, consiguen ensuciar y dificultar tanto el código que no es mayor problema que alguien pueda acceder a ese código





# Ofuscar el código Javascript

La herramienta Closure Compiler, también ofrece la posibilidad de ofuscar nuestro código JavaScript



# Ofuscar el código Javascript

Al sustituir todos los nombres de las variables y de las funciones por nombres de una sola letra, es prácticamente imposible comprender el código del programa

En ocasiones, también se utilizan ofuscadores de este tipo con el propósito de reducir el tamaño del código fuente



# Ofuscar el código Javascript

Además de aplicaciones comerciales específicamente diseñadas para ofuscar código JavaScript, también se pueden utilizar las herramientas que minimizan el tamaño de los scripts. Eliminando los comentarios y reduciendo el nombre de todas las variables, los programas que minimizan el tamaño de los scripts también consiguen ofuscar su código



# Ofuscar el código Javascript

La aplicación packer gratuita, se puede acceder via web y consigue una excelente compresión del código original



# Ofuscar el código Javascript

También tenemos disponible [jscompress.com](http://jscompress.com) que nos permite comprimir el código Javascript



# Conclusiones

Hemos visto cómo mejorar  
el rendimiento y ofuscar el  
código Javascript



# Datos de Contacto

<http://www.cursosdedesarrollo.com>  
[info@cursosdedesarrollo.com](mailto:info@cursosdedesarrollo.com)

<http://cursosdedesarrollo.com/>



# Licencia



David Vaquero Santiago

Esta obra está bajo una  
Licencia Creative Commons Atribución-  
NoComercial-CompartirIgual 4.0  
Internacional

Derivada de:

<http://www.arkaitzgarro.com/javascript/>

y

<http://javiereguiluz.com/>

<http://cursosdedesarrollo.com/>

