

Webworkers



Francisco Arce
www.pacoarce.com

Webworker

- Un “webworker” es una rutina en javascript que trabaja en el background del navegador sin perjudicar el desempeño del mismo.
- El script puede ejecutar cualquier actividad en paralelo al “webworker”.

Francisco Arce
www.pacoarce.com

Webworker

- IE 10, actual 11
- Chrome 4, actual 42
- Firefox 3.5, actual 37
- Safari 4, actual 8
- Opera 11.5, actual 27

Francisco Arce
www.pacoarce.com



Web Workers - LS

Global

86.93%

Method of running scripts in the background, isolated from the web page

Current aligned

Usage relative

Show all

IE	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Chrome for Android
		31						
		36						
		37						
		38					4.1	
8	31	39					4.3	
9	35	40	7				4.4	
10	36	41	7.1		7.1		4.4.4	
11	37	42	8	27	8.3	8	40	42
TP	38	43		28				
	39	44		29				
	40	45						

Notes

Sub-features (1)

Known issues (0)

Resources (5)

Feedback

No notes

Webworkers

Ejemplo



Francisco Arce
www.pacoarce.com

Webworkers

- Verificar que el navegador soporte webworkers:

```
if(typeof(Worker) !== "undefined") {  
    // Soporta webworkers  
} else {  
    // Navegador chafa...  
}
```

Francisco Arce
www.pacoarce.com

Webworkers

- Creamos una rutina javaScript:

```
var i = 0;  
function contador() {  
    i = i + 1;  
    postMessage(i);  
    setTimeout("contador()",500);  
}  
contador();
```

Francisco Arce
www.pacoarce.com

Webworkers

- La función `postMessage()` manda un mensaje al background.
- Nota: Los webworkers se utilizan preferentemente en programas que requieran muchos recursos del CPU.

Webworkers

- Los objetos webworkers se crean desde una página html.

```
if(typeof(w) == "undefined") {  
    w = new Worker("worker.js");  
}
```

Francisco Arce
www.pacoarce.com

Webworkers

- Podemos recibir mensajes del background con el evento “onmessage”:

```
w.onmessage = function(event){  
  document.getElementById("result").innerHTML =  
  event.data;  
};
```

Francisco Arce
www.pacoarce.com

Webworkers

- El resultado es almacenado en “event.data”.
- El webworker seguirá ejecutándose aunque el script de la página haya terminado.
- Para detenerlo hay que utilizar:
 - `w.terminate();`

Webworkers

- Los webworkers deben de ser escritos en archivos externos js.