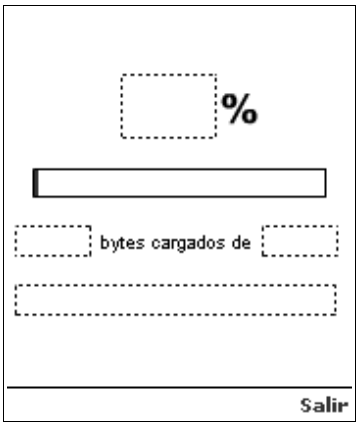
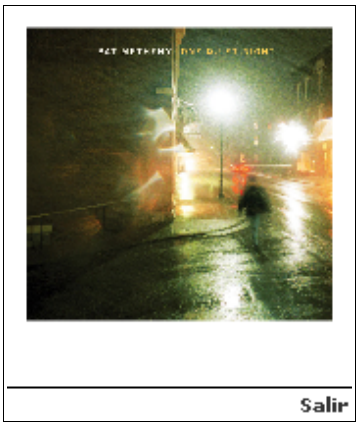
Preloader

Cálculo del tamaño de la aplicación y tasa de transferencia

Los preloaders son indicadores del estado de carga de cualquier objeto en general –- incluida la aplicación. Antiguamente el cálculo se hacía en función del contenido de los fotogramas. Esto no es un método exacto, pues si nuestra aplicación tiene 200 frames y en el frame 5 está el 50% del peso del archivo total, el preloader marcaría un 2.5% al llegar a ese fotograma, y se quedaría en ese porcentaje hasta que dicho frame fuera cargado totalmente. La forma correcta es basarse en el tamaño del archivo, esto es, los bytes que ocupe el fichero SWF.

Para esta práctica vamos a utilizar tres indicadores visuales del estado de carga de la aplicación, a saber: el porcentaje cargado, una barra que irá creciendo en función del porcentaje de carga y los bytes del archivo. Como extra se calculará la tasa de transferencia y el tiempo restante, que aparecerán en un campo de texto llamado `tasa_txt`. Estos serán los dos fotogramas de la aplicación: el primero para la precarga y el segundo para mostrar el contenido, en este caso una imagen de 153 KB (para que se vea bien el progreso de descarga).





Vamos con el primer fotograma, que es precisamente lo interesante de esta práctica. El campo de texto que mostrará el porcentaje de carga lo llamaremos `porcentaje`. Debajo creamos un rectángulo sin contorno, solamente el relleno, y lo transformamos en el movieClip `barra`, y lo escalamos horizontalmente al 1%. Debajo ubicaremos tres campos de texto, los de los extremos serán dinámicos, llamados `cargados` y `totales`. Debajo añadiremos posteriormente el cálculo del tiempo de descarga y la tasa de transferencia, que aparecerá en `tasa_txt`.

El código inicial de este primer fotograma es conocido:

```
fscommand2("FullScreen", true);

var teclas:Object = new Object();

teclas.onKeyDown = function() {

if (Key.getCode() == ExtendedKey.SOFT2) →
fscommand2("Quit");

};

Key.addListener(teclas);
```

Opciones de publicación

Dado que el emulador de dispositivos no puede simular el tiempo de descarga de la aplicación, hay que comentar este bloque anterior y publicar la aplicación como Flash 8, no como Flash Lite 2.x.

Cálculo de la precarga

Necesitamos crear un bucle que se actualice constantemente hasta tener la aplicación totalmente cargada.

```
this.onEnterFrame = function():Void {

 barra._xscale =                          →
Math.round(_root.getBytesLoaded()*100     →
/_root.getBytesTotal());

 cargados.text = _root.getBytesLoaded();

 totales.text = _root.getBytesTotal();

 porcentaje.text = barra._xscale;

};
```

Como la imagen la vamos a importar desde la biblioteca, la precarga la hacemos sobre la aplicación (`_root`), pero la forma correcta de hacerlo sería cargando la imagen externamente mediante la clase MovieClipLoader, la cual permite registrar muchos (y prácticos) eventos; por ejemplo: `onLoadError`, `onLoadProgress`, `onLoadComplete`.

Tasa de transferencia y tiempo restante

Para terminar de aderezar la práctica, a continuación se muestran los cálculos que hay que añadir dentro del loop `onEnterFrame` que definíamos anteriormente:

```
var time:Number = getTimer()/1000;

var bps:Number =                         →
Math.round(_root.getBytesLoaded()/time);

var bytesLeft:Number =                   →
_root.getBytesTotal() -                  →
```

```
_root.getBytesLoaded();

var segLeft:Number =                    →
Math.ceil(bytesLeft/bps);

var minLeft:Number =                    →
Math.floor(segLeft/60);

var hrsLeft:Number =                    →
Math.floor(minLeft/60);

minLeft -=(hrsLeft*60);

segLeft -=((hrsLeft * 60) + minLeft) * 60;

var kbps:Number =                       →
(Math.floor((bps/1024)*10))/10;

tasa_txt.text = "quedan "+hrsLeft+"h:"+ →
minLeft+"m:"+segLeft+"seg. a "+kbps+    →
"kbps";
```

Ya sólo falta importar la imagen al escenario y ubicarla en la posición que deseemos. En este segundo fotograma deberemos detener la reproducción de la aplicación y borrar el loop `onEnterFrame`:

```
stop();

delete this.onEnterFrame;
```