

## Flash + ActionScript para interactividad (3)

Written by Ángel Ricardo Puente Pérez  
Wednesday, 18 July 2007 14:16

---

There are no translations available.

Tercera entrega de este curso sobre Flash y ActionScript. Validación de múltiples matrices de datos.

### Flash + ActionScript para interactividad (3)

#### Cuadrado mágico de 3 x 3

En esta ocasión, la novedad fundamental respecto a las dos primeras actividades, consiste en que el botón de validación no va a evaluar una única tira de datos. Va a evaluar las ocho posibilidades diferentes que se corresponden con las soluciones de un **cuadrado mágico** de 3 filas y 3 columnas.

Veamos la animación. Por supuesto, antes de seguir, conviene familiarizarse con ella y **hacerla en su totalidad**

para observar el comportamiento de los botones y de toda la aplicación. Se trata de conseguir las ocho soluciones diferentes. Primero se busca una cualquiera en la parte principal.

Comprobada la solución, se traslada a los cuadros inferiores. Estos cuadros pueden orientarnos también en el encuentro del resto de soluciones pues están ordenados por giros y simetrías.

En [El Tinglado](#) se ha publicado una [variable de esta actividad](#) consistente en cambiar las nueve cifras propuestas por los nueve números enteros que van desde el -4 al 4.

Lógicamente, los cuadrados mágicos, pueden diseñarse con sucesiones aritméticas diferentes...

Y, en [este otro documento](#), hay explicaciones sobre el diseño y explotación de estos cuadrados mágicos de 3 x 3 con animaciones en flash para las búsquedas de soluciones con giros y simetrías.

#### 1.- Independización de los elementos de la animación por capas.

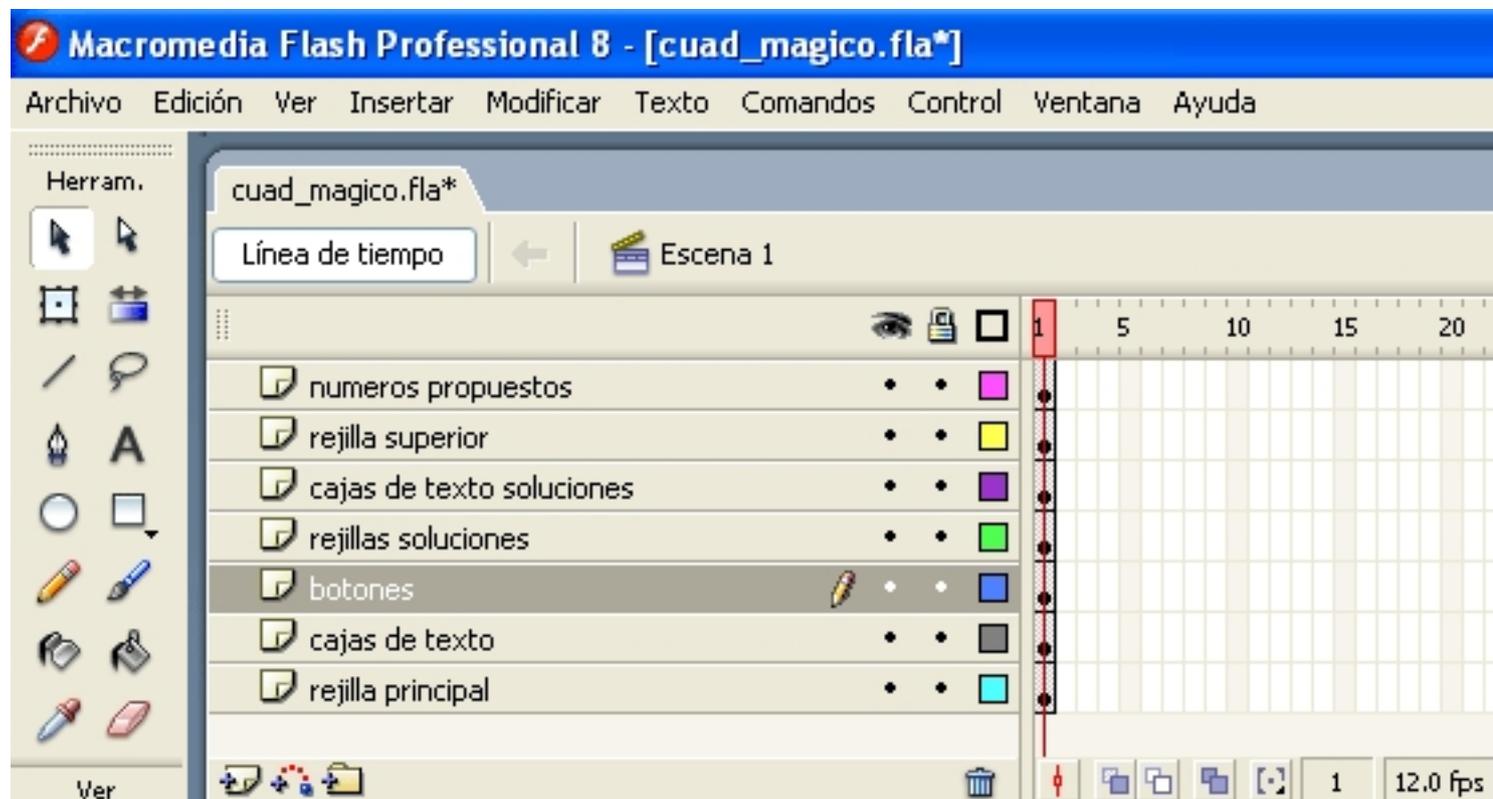
En esta actividad, se han colocado los diferentes elementos, en siete capas distintas. Es muy

## Flash + ActionScript para interactividad (3)

Written by Ángel Ricardo Puente Pérez  
Wednesday, 18 July 2007 14:16

---

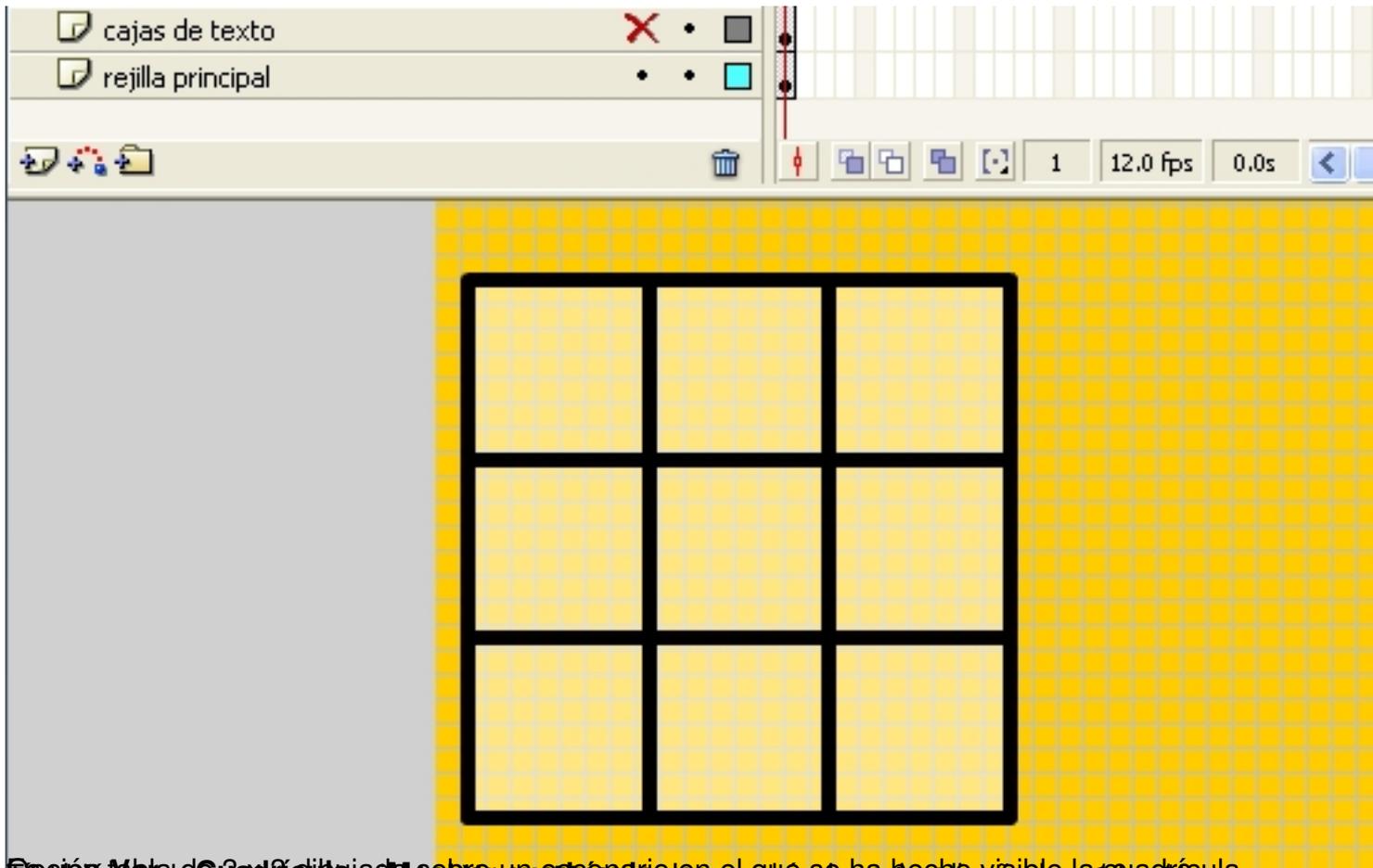
importante hacerlo así. Permite el diseño de un objeto, asegurando lo trabajado en una capa anterior.



## 2.- Primera capa. La rejilla principal del cuadrado mágico.

## Flash + ActionScript para interactividad (3)

Written by Ángel Ricardo Puente Pérez  
Wednesday, 18 July 2007 14:16

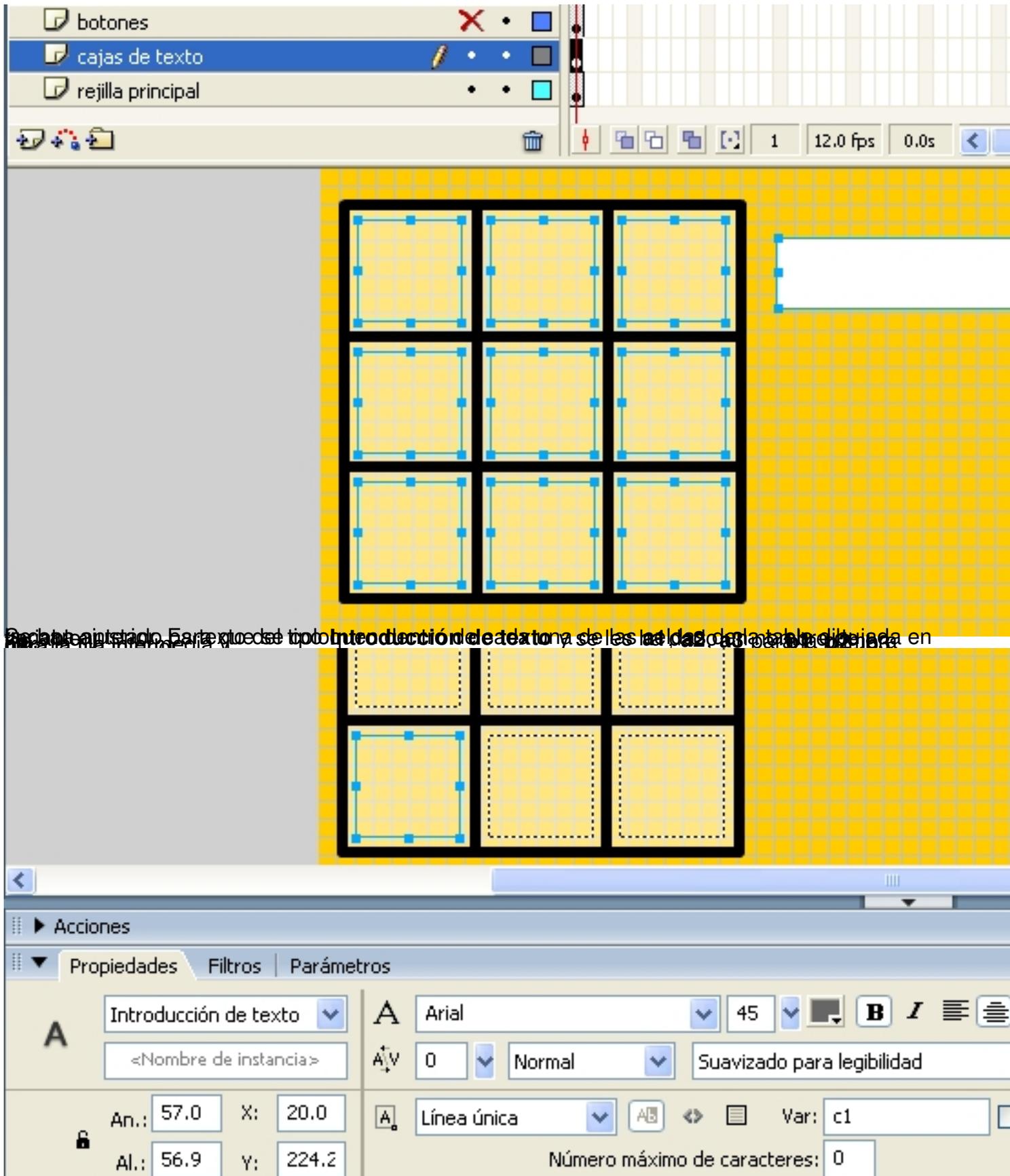


En esta parte de código se muestra un ejemplo en el que se ha hecho visible la cuadrícula  
Segundo paso de código con el que se muestra la cuadrícula para conseguir que los

### 3.- Segunda capa. Cajas de texto principales.

## Flash + ActionScript para interactividad (3)

Written by Ángel Ricardo Puente Pérez  
Wednesday, 18 July 2007 14:16



botones  
cajas de texto  
rejilla principal

1 12.0 fps 0.0s

Introducción de texto

Arial 45 Normal Suavizado para legibilidad

An.: 57.0 X: 20.0  
Al.: 56.9 Y: 224.2

Línea única Var: c1  
Número máximo de caracteres: 0

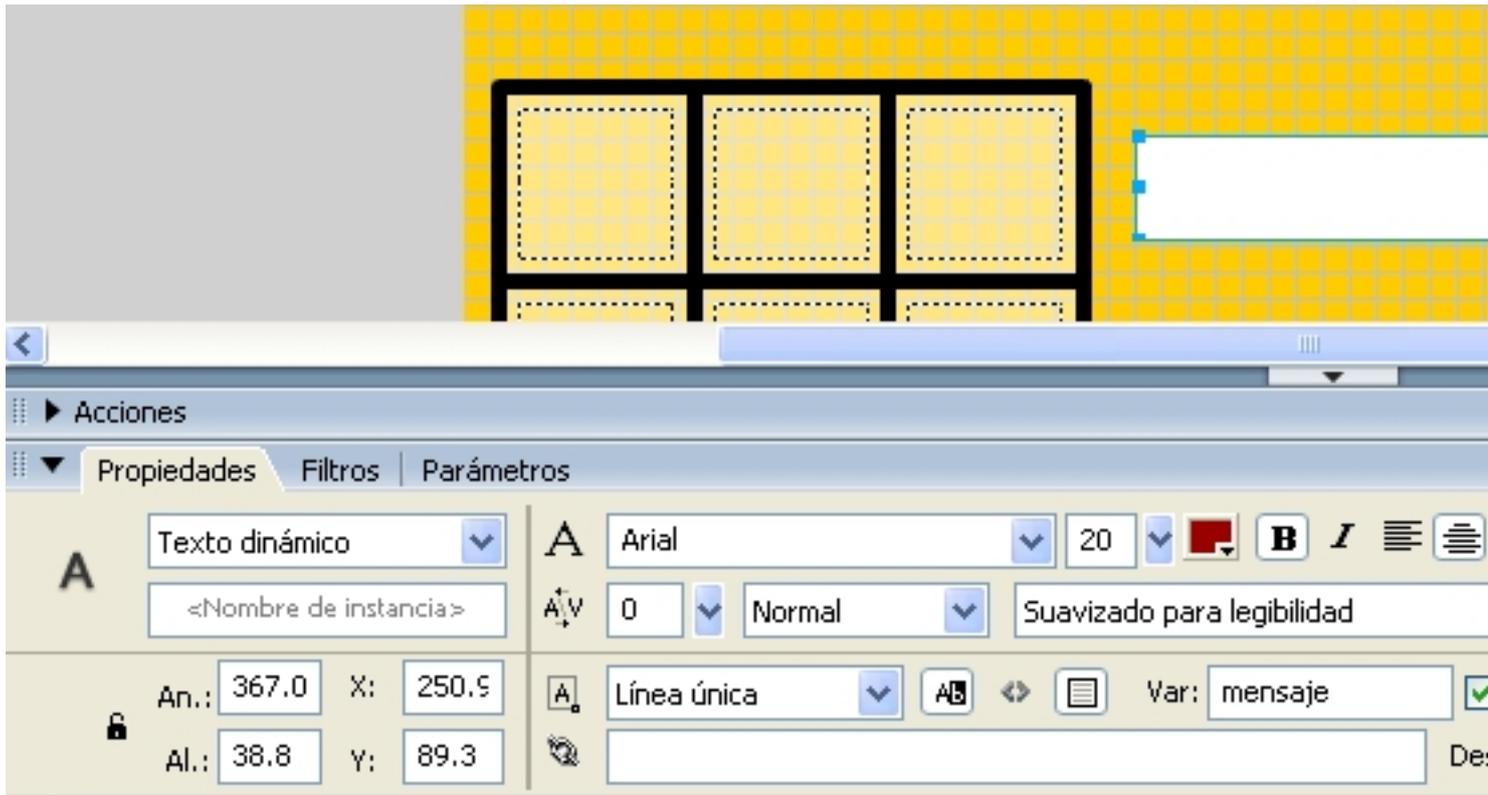
En esta instancia, el texto de tipo **Introducción de texto** se le ha dado una instancia dibujada en

En esta instancia se le ha pasado a dibujado una caja de texto del tipo **Texto dinámico** y, como nombre

## Flash + ActionScript para interactividad (3)

Written by Ángel Ricardo Puente Pérez  
Wednesday, 18 July 2007 14:16

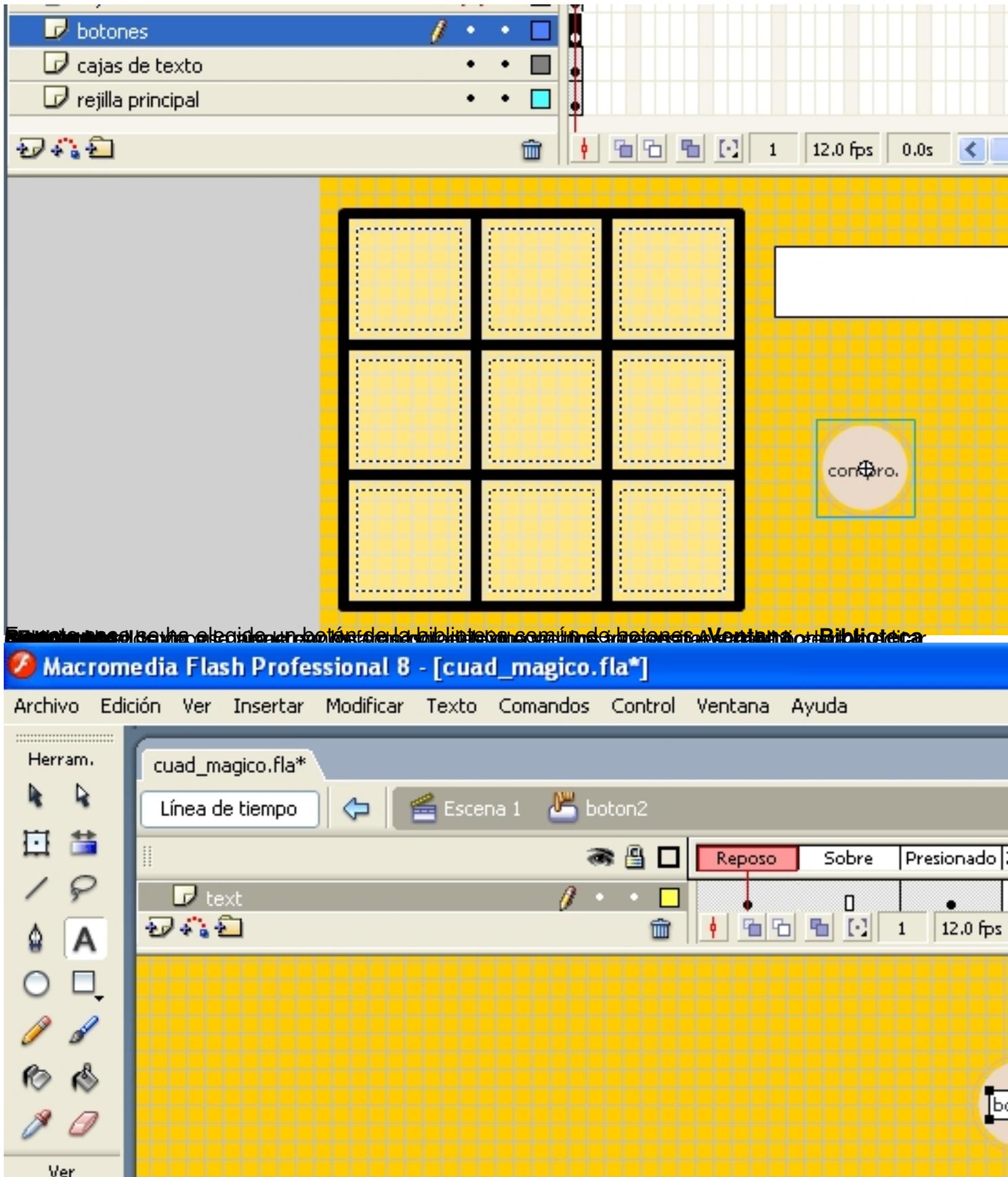
---



### 4.- Tercera capa. Botones.

## Flash + ActionScript para interactividad (3)

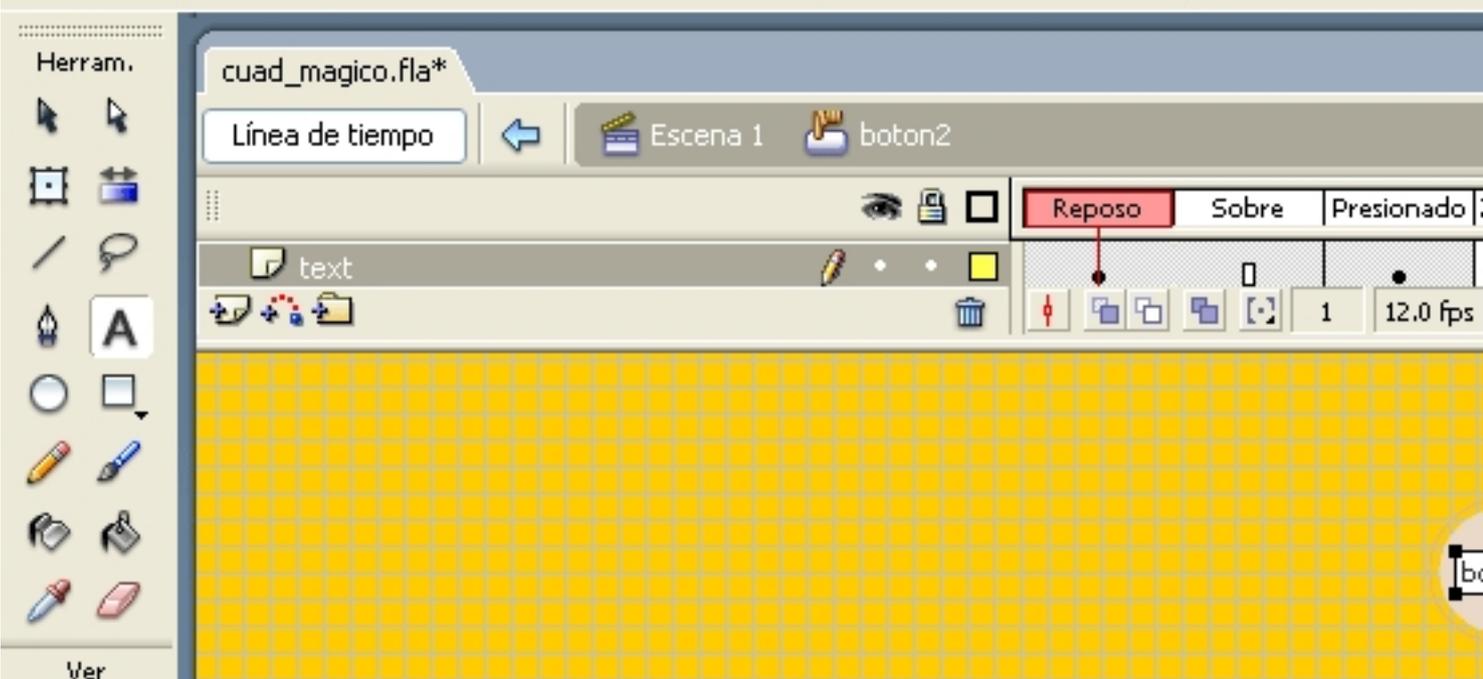
Written by Ángel Ricardo Puente Pérez  
Wednesday, 18 July 2007 14:16



En este caso se ha elegido un botón de la biblioteca común de botones. **Ventana > Biblioteca**

**Macromedia Flash Professional 8 - [cuad\_magico fla\*]**

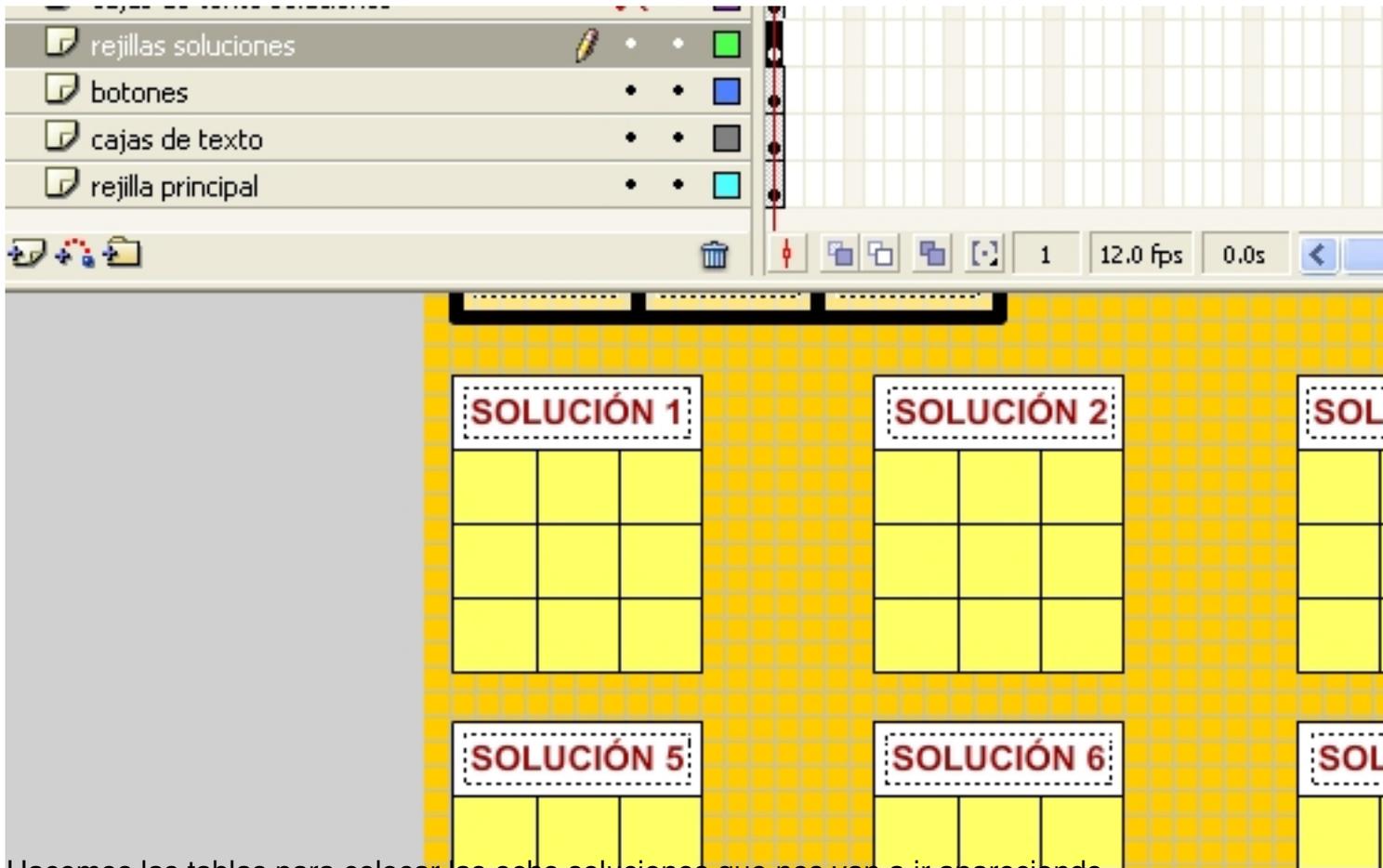
Archivo Edición Ver Insertar Modificar Texto Comandos Control Ventana Ayuda



### 5.- Cuarta capa. Rejilla soluciones.

## Flash + ActionScript para interactividad (3)

Written by Ángel Ricardo Puente Pérez  
Wednesday, 18 July 2007 14:16

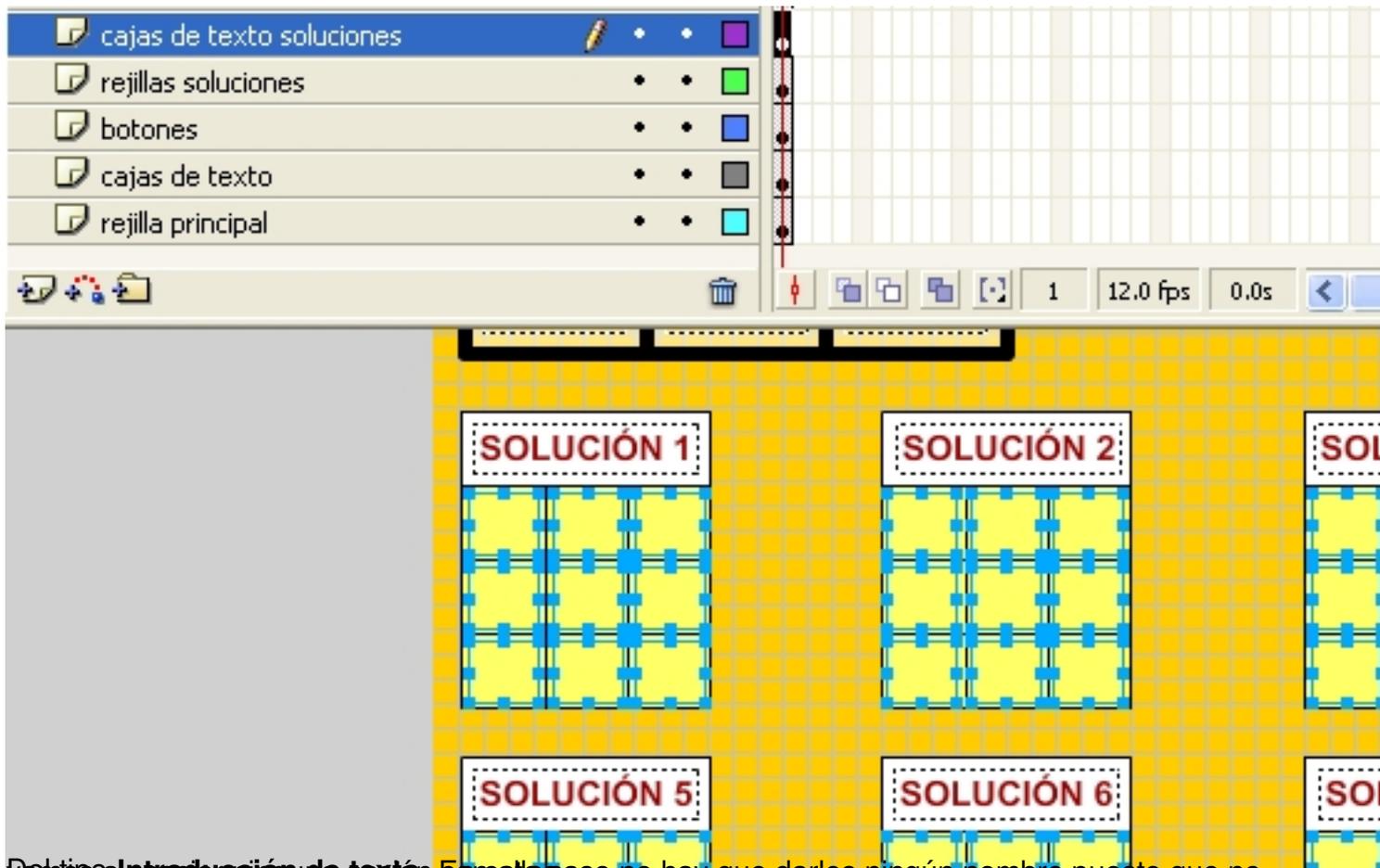


Hacemos las tablas para colocar las ocho soluciones que nos van a ir apareciendo.

### 6.- Quinta capa. Cajas de texto de las soluciones.

## Flash + ActionScript para interactividad (3)

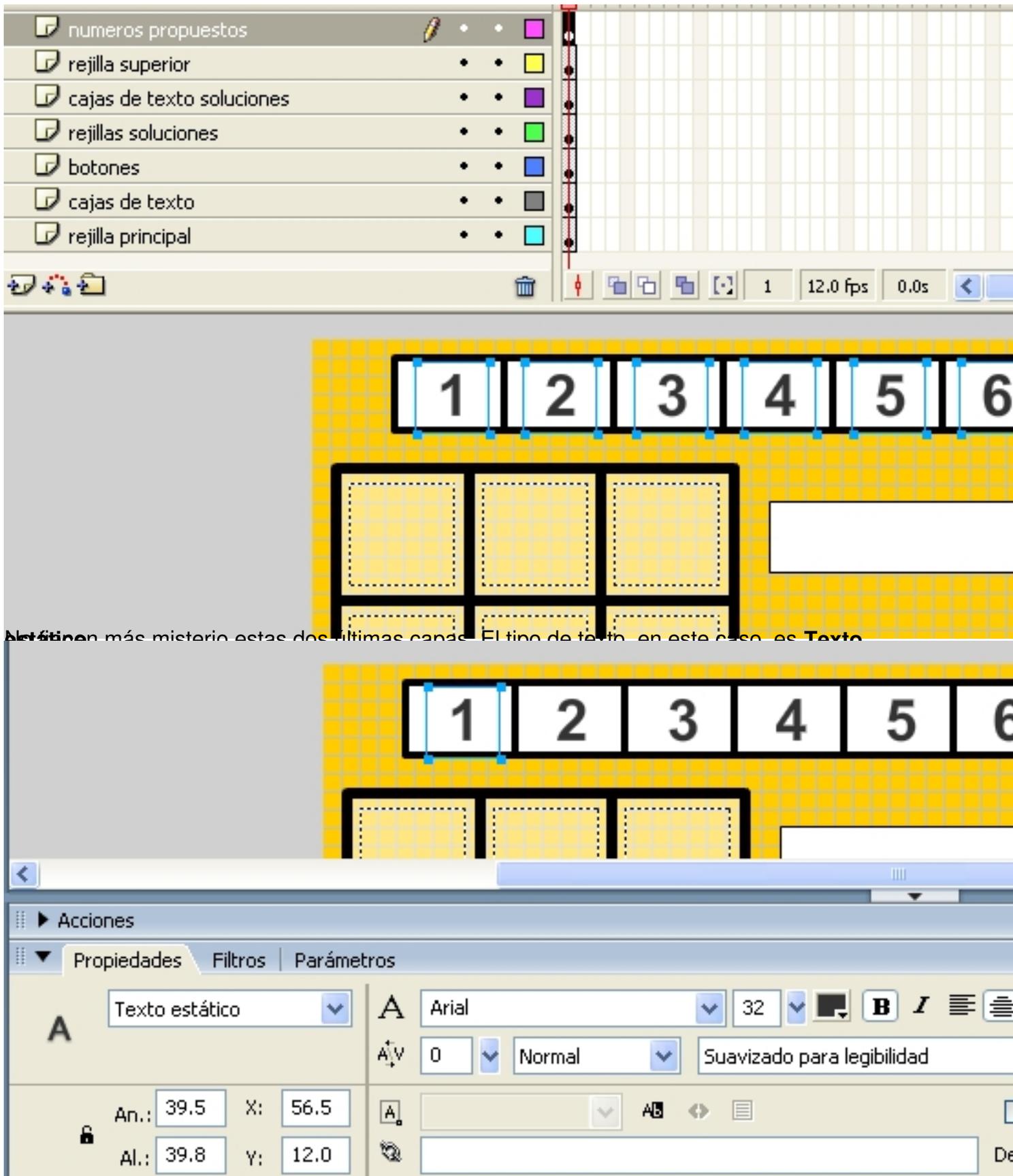
Written by Ángel Ricardo Puente Pérez  
Wednesday, 18 July 2007 14:16



De ahí se introduce el texto. En este caso no hay que darles ningún nombre puesto que no  
**7.- Dos últimas capas. Caja para los números propuestos y los propios números.**

## Flash + ActionScript para interactividad (3)

Written by Ángel Ricardo Puente Pérez  
Wednesday, 18 July 2007 14:16



The screenshot displays the Adobe Flash IDE interface. At the top, a library panel lists several assets: 'numeros propuestos', 'rejilla superior', 'cajas de texto soluciones', 'rejillas soluciones', 'botones', 'cajas de texto', and 'rejilla principal'. The main workspace shows a yellow grid background with a row of six numbered boxes (1-6) and a 2x3 grid of dashed boxes below them. A text input field is visible on the right. The Properties panel at the bottom is open to the 'Propiedades' tab, showing the text type as 'Texto estático', font as 'Arial', size as '32', and alignment as 'Suavizado para legibilidad'. The coordinates for the text are An.: 39.5, X: 56.5, Al.: 39.8, and Y: 12.0.

Atención más misterio estas dos últimas celdas. El tipo de texto en este caso es **Texto**

### 8.- Código del botón Comprobar.

## Flash + ActionScript para interactividad (3)

Written by Ángel Ricardo Puente Pérez  
Wednesday, 18 July 2007 14:16

---

Es lo más difícil de la actividad.

Tenemos que hacer que evalúe cualquiera de las ocho posibilidades que tenemos de soluciones. Para ello, primero tendremos que escribirlas y asociar el nombre de la variable de la caja de **Introducción de texto** con el valor que se debe introducir.

Escribimos aquí las soluciones tal como se han diseñado en la aplicación:

SOLUCIÓN 1		
4	3	8
9	5	1
2	7	6

SOLUCIÓN 2		
2	9	4
7	5	3
6	1	8

SOLUCIÓN 3		
6	7	2
1	5	9
8	3	4

SOLUCIÓN 4		
8	3	4
3	5	9
4	7	2

SOLUCIÓN 5		
2	7	6
9	5	1
4	3	8

SOLUCIÓN 6		
6	1	8
7	5	3
2	9	4

SOLUCIÓN 7		
8	3	4
1	5	9
6	7	2

SOLUCIÓN 8		
4	7	2
3	5	9
8	3	4

Pues bien, tendremos que coger el botón **comprobar**, seleccionarlo y darle a **Acciones** para escribir esto:

```
1 on (release) (  
2     if (a1 == 4 && a2 == 3 && a3 == 8 && b1 == 9 && b2 == 5 && b3 == 1 && c1 == 7  
3         mensaje = "Fenomenal. Solución número 1";  
4     ) else if (a1 == 2 && a2 == 9 && a3 == 4 && b1 == 7 && b2 == 5 && b3 == 3  
5         mensaje = "Fenomenal. Solución número 2";  
6     ) else if (a1 == 6 && a2 == 7 && a3 == 2 && b1 == 1 && b2 == 5 && b3 == 9  
7         mensaje = "Fenomenal. Solución número 3";  
8     ) else if (a1 == 8 && a2 == 1 && a3 == 6 && b1 == 3 && b2 == 5 && b3 == 7  
9         mensaje = "Fenomenal. Solución número 4";  
10    ) else if (a1 == 2 && a2 == 7 && a3 == 6 && b1 == 9 && b2 == 5 && b3 == 1  
11        mensaje = "Fenomenal. Solución número 5";  
12    ) else if (a1 == 6 && a2 == 1 && a3 == 8 && b1 == 7 && b2 == 5 && b3 == 3  
13        mensaje = "Fenomenal. Solución número 6";  
14    ) else if (a1 == 8 && a2 == 3 && a3 == 4 && b1 == 1 && b2 == 5 && b3 == 9  
15        mensaje = "Fenomenal. Solución número 7";  
16    ) else if (a1 == 4 & a2 == 9 & a3 == 2 & b1 == 3 & b2 == 5 & b3 == 7 & c1 == 1  
17        mensaje = "Fenomenal. Solución número 8";  
18    ) else (  
19        mensaje = "Hay algún error";  
20    )  
21 }
```

Puedes clickear sobre la imagen para verla un poco más grande.

### 9.- Código del botón Borrar.

Este botón ya lo hemos visto en anteriores ocasiones. Se trata, simplemente de este código:

## Flash + ActionScript para interactividad (3)

Written by Ángel Ricardo Puente Pérez  
Wednesday, 18 July 2007 14:16

```
1 on (release) {  
2     a1 = "";  
3     a2 = "";  
4     a3 = "";  
5     b1 = "";  
6     b2 = "";  
7     b3 = "";  
8     c1 = "";  
9     c2 = "";  
10    c3 = "";  
11    mensaje = "";  
12 }
```

~~mensaje = "";~~ ~~Se cierra el botón con on()~~ ~~mensaje de éxito o error.~~

### 10.- Código del botón Pista.

Se trata de introducir la variable que es fija en las ocho soluciones. Es la posición del centro. En este caso ocupada por el número central en la progresión aritmética del 1 al 9, es decir, el 5.

```
1 on (release) {  
2     b2 = 5;  
3 }  
4
```

#### código explicación

on (release) {	Quando liberemos el botón...
b2 = 5;	En la caja <b>b2</b> se colocará el 5.
}	Se cierra el <b>on()</b> .