



Effector Book
Tomo [XXXVIII]
Vol. II

## Kara Effector 3.3

El **Effector Book**, que es una especie de manual o tutorial del uso del **Kara Effector**, está de regreso luego de un tiempo de receso por otras ocupaciones que tuve en el transcurso de este largo año. Desde la publicación del **Tomo 37 (XXXVII)**, el **Kara Effector** ha venido evolucionando de forma exponencial, y por eso la necesidad de tener un material de apoyo que nos guie y ayude a sacarle el mayor provecho a dichas evoluciones.

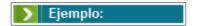
En este **Tomo**, el **38**, veremos las funciones que se han actualizado, en qué consisten y cómo se aplican, y ejemplo de la forma en la que podemos usarla; también veremos la evolución que han tenido las celdas de texto de la **Ventana de Modificación** del **KE**, para que sea aún más simple el poder agregar funciones, tags, entre otras herramientas disponibles para nuestros fx.

# Recursos (KE):

La primera función que me interesa que vean la forma en la que ha evolucionado es la función R (es la versión math.random del KE), que no es otra que aquella que nos permite obtener un número aleatorio o al azar, siguiendo una serie de parámetros o argumentos predeterminados, o por default si así lo queremos.



Recordemos que esta función retorna un número aleatorio que se encuentre entre los argumentos **val1** y **val2**, inclusive, en un paso determinado por el tercer argumento, **Step**.



R(0, 100, 10)

De este modo, la función retornará un número entero entre 0 y 100 de a 10 en 10, es decir que los posibles resultados serían:

0, 10, 20, 30, 40, 50, 60, 70, 80, 90 o 100

Partiendo de este hecho, y de que la función R es una de las más usadas a la hora de desarrollar un efecto, hemos ampliado su rango de operación, para así ampliar nuestras posibilidades, agregando una serie de letras para que la función cumpla con otros requisitos.

Las actualizaciones son 16 en total, y veremos ejemplos de cada una de ellas para comprender en qué consisten:

Rs( Val1, Val2, Step ): la letra "s" es de "signo", es decir que esta función retorna un número aleatorio entre Val1 y Val2, y el signo de dicho número también será asignado de forma aleatoria. Recordemos que los tres argumentos de esta función son opcionales, de manera que si solo ponemos un solo argumento, el random original se generará entre el 1 y dicho argumento y el valor del Step será 1 por



Effector Book Tomo [XXXVIII] Vol. II default. Si ponemos dos argumentos, el random se llevará a cabo entre esos dos valores inclusive, y el **Step** también será 1.



#### Rs(12, 18)

Retornará un número aleatorio entre 12 y 18 con cualquiera de los dos signos, positivo o negativo. Entonces los posibles resultados de este ejemplo serían:

Rr( Val1, Val2, Step ): la letra "r" es de "ratio". La función retornará un número aleatorio que esté entre los primeros dos argumentos inclusive, pero el resultado final dependerá del tamaño del vídeo en dónde se aplique el efecto, es decir que si el vídeo es de 720p el resultado no se modificará, pero si es más pequeño, el resultado disminuirá proporcionalmente; y pasaría lo mismo si el vídeo es de mayor resolución que el ya mencionado.



### Rr(8, 10)

Los tres posibles resultados de este ejemplo en un vídeo de 720 x 1280 px son 8, 9 o 10, pero si el vídeo fuera de 1080 x 1960 px, cualquiera de esos tres resultados se multiplica por el "ratio", es decir, la razón entre 1080 y 720:

O sea que los tres posibles resultados para el vídeo de 1080p vendrían a ser:

- 8 \* 1.5 = 12
- 9 \* 1.5 = 13.5
- 10 \* 1.5 = 15

Esta actualización tiene su aplicación, por ejemplo, al designar aleatoriamente el borde con el tag \bord; ya que un \bord3 para un vídeo de 720p puede que sea muy poco para un vídeo de 1080p, o muy poco para uno de 480p.

• Rd( Val1, Val2, Step ): la letra "d" es de "décima". La función retorna un valor entre los dos primeros argumentos, pero redondeados al primer decimal.



### Rd(-1, 2)

La función retornaría un número entre -1 y 2, pero tendría en cuenta los números redondeados al primer decimal, o sea que los resultados serían:

Rc( Val1, Val2, Step ): la letra "c" es de "centésima". Es similar a la actualización anterior, pero con la diferencia que el valor retornado estará redondeado al segundo decimal, es decir que la cantidad de opciones posibles se amplían 100 veces más que en la función normal.



### Effector Book Tomo [XXXVIII] Vol. II

 Rm( Val1, Val2, Step ): la letra "m" es de "milésima". Redondea el número a la tercera cifra decimal.

Para las siguientes actualizaciones, lo único que debemos hacer es combinar las letras aprendidas en las actualizaciones anteriores de la función **R**, de la siguiente forma:

Rsr o Rrs: Random + signo + ratio Rdr: Random + décima + ratio Random + centésima + ratio Rcr: Random + milésima + ratio Rmr: Random + décima + signo Rds: Rcs: Random + centésima + signo Random + milésima + signo Rms: Rdrs: Random + décima + ratio + signo Random + centésima + ratio + signo Rcrs: Random + milésima + ratio + signo Rmrs:

Y la última de las actualizaciones de la función **R** es una que nos permite ingresarle una **tabla** y obtendremos uno de sus elementos seleccionados de forma aleatoria:

• Re( tabla ): la letra "e" es de "elemento". Retorna un elemento seleccionado aleatoriamente de la tabla que le ingresemos como su único argumento. Tiene la ventaja que también lo puede hacer en tablas que no sean indexadas:

```
Ejemplo:

my_table = {
    [1] = 4,
    [2] = "\\fry-45",
    Size = Rr(80, 100),
    80,
    -12
}

Re( my_table )
```

Así, de este modo, la función podría retornar cualquiera de los cinco elementos de la tabla "my\_table", es decir: 4, \\fry-45, Rr(80, 100), 80 o -12

Con todas estas modificaciones, creo que quedan cubiertas muchas o gran parte de todas las posibilidades a la hora de generar un valor aleatorio. Lo que viene de acá en más es poner en práctica y experimentar con lo aprendido e ir desarrollando nuestros propios efectos y estilos.

El tema que veremos a continuación, considero que es uno de los más importantes para todos aquellos que les gusta desarrollar y llevar a cabo aquellos efectos que se les vienen en mente o que se presentan en los diferentes vídeos y que desean de alguna manera poder "imitarlos" para sus propios proyectos. El tema consiste es una serie de abreviaciones, convenciones y un par de trucos más que el **KE** tiene en sus librerías para que lo que antes eran simples tags, ahora se conviertan en funcionen, y por qué no, en efectos en sí mismos.

### Effector Book Tomo [XXXVIII] Vol. II



Parte de esta herramienta ya la empezamos a ver en el Tomo anterior, y no está de más que los recordemos y ver en qué consistía cada uno de ellos:

	tags abreviados		
	tag	Ejemplo (valor constante)	resultado
1	\\fscxy	"\\fscxy147"	\fscx147\fscy147
2	\\frxy	"\\frxy-23"	\frx-23\fry-23
3	\\frxz	"\\frxz12"	\frx12\frz12
4	\\fryz	"\\fryz345"	\fry345\frz345
5	\\frxyz	"\\frxyz-360"	\frx-360\fry-360\frz-360

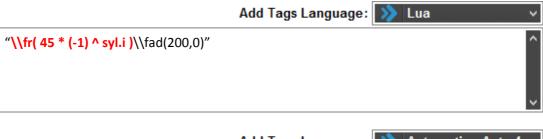
Estas mismas cinco abreviaciones de tags también las podemos usar con la función **R** y todas sus nuevas actualizaciones que recién vimos:

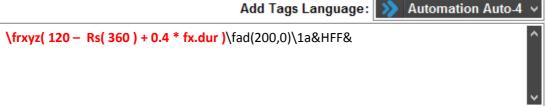
	tags abreviados	
	tag	Ejemplo (valor aleatorio)
1	\\fscxy	"\\fscxyR( 100, 200, 10 )"
2	\\frxy	"\\frxyRs( 20, 50 )"
3	\\frxz	"\\frxzRcs( -125, 215 )"
4	\\fryz	"\\fryzRd( 22, 43 )"
5	\\frxyz	"\\frxyzRs( 360 )"

Ahora también podemos usar, y no solo con estos cinco tags sino también con todos, operaciones y/o funciones, con solo añadir un paréntesis después de los mismos:



- "\\fscxy( 120 + Rs( 40 ) )"
- "\\fr( 45 \* (-1) ^ syl.i )"









Effector Book
Tomo [XXXVIII]
Vol. II

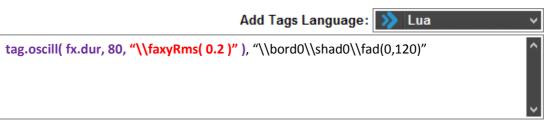
Entonces, cualquier tag lo podemos usar de las anteriores tres formas que acabamos de ver:

- Valor constante
- Valor aleatorio
- Operación y/o función

Esto nos ahorrará un poco de trabajo y hace que sea más simple el uso de cada uno de ellos. Y con el fin de seguir ahorrando tiempo y trabajo, el **KE** ya consta de otras abreviaciones que podemos usar de las tres formas ya mencionadas. Las primeras tres nuevas abreviaciones básicas que veremos son las siguientes:

	nuevos tags abreviados	
	tag	equivalencia
1	\\faxy	\fax + \fay
2	\\xybord	\xbord + \ybor
3	\\xyshad	\xshad + \yshad





Las anteriores ocho abreviaciones son las consideradas "básicas", y a partir de ellas se abre un mundo de posibilidades ante nosotros, que espero poder explicarlas todas y no olvidar ninguna por el camino. Empecemos viendo un resumen de las abreviaciones vista hasta este momento y unas nuevas que ya merecen ser mencionadas:

	tags abreviados	
	tag	equivalencia
1	\\fscxy	\fscx + \fscy
2	\\frxy	\frx + \fry
3	\\frxz	\frx + \frz
4	\\fryz	\fry + \frz
5	\\frxyz	frx + fry + frx
6	\\faxy	\fax + \fay
7	\\xybord	\xbord + \ybord
8	\\xyshad	\xshad + \yshad

La primera abreviación (\\fscxy) agrupa los dos tags de porcentajes de tamaño, y cuando la usamos con un valor aleatorio, cada uno de los dos tags adquiere valores diferentes entre ellos, según los parámetros ingresados, ejemplo:

• \\fscxyR( 80, 120 ) → \fscx104\fscy86





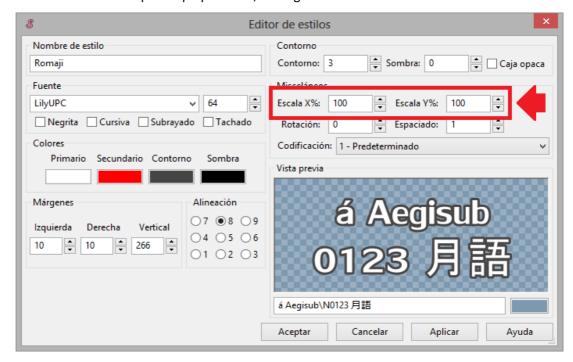
Effector Book
Tomo [XXXVIII]
Vol. II

En el ejemplo anterior, 104 y 86 son posibles resultados aleatorios de la función R, cuyo valor mínimo sería 80 y valor máximo 120. El ejemplo anterior no garantiza que el valor aleatorio sea el mismo para ambos tags, y para que ello suceda, es necesario agregar la letra "i" al tag justo antes de la función R:



\\fscxyiRc( 40, 60 ) → \fscx47.18\fscy47.18

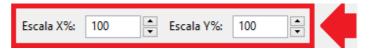
Lo que nos quiere indicar el ejemplo anterior es que el valor aleatorio que arroje la función Rc para el tag \fscx, será el mismo que para el tag \fscy, según los parámetros de la función random. Esta habilidad que le da la letra "i" a la abreviación \fscxy garantiza que la variación de tamaño de objeto karaoke sea siempre proporcional, siempre que los valores de escala en el Estilo de la línea a la que le apliquen el fx, sean iguales entre sí:

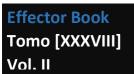


Para el caso en que estos dos valores no sean iguales entre sí en un Estilo, veremos las siguientes cuatro adaptaciones de los tags de escala de tamaño:

- \\fscxr
- \\fscyr
- \\fscxyr
- \\fscxyir

La letra "r" hace referencia al "ratio" o "razón" de la variación respecto a los valores de escala en un determinado Estilo de Línea



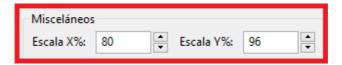








Supongamos que para cierto Estilo de Línea los valores de escala son los siguientes:



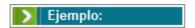
#### **Entonces:**

•	\\fscx <b>r</b> 1 = \\fscx80	porque 80 x 1 = 80
•	\\fscy <b>r</b> 1 = \\fscy96	porque 96 x 1 = 96
•	\\fscx <b>r</b> 0.5 = \\fscx40	porque 80 x 0.5 = 40
•	\\fscyr1.2 = \\fscy115.2	porque 96 x 1.2 = 115.2
•	$\fscxr2.5 = \fscx200$	porque 80 x 2.5 = 200
•	$\fscxyr1.5 = \fscx120\fscy144$	porque 80 x 1.5 = 120 y 96 x 1.5 = 144
•	$\frac{1}{1} Scxyr0.4 = \frac{32}{fscy38.4}$	porque 80 x 0.4 = 32 y 96 x 0.4 = 38.4

Así que el valor que coloquemos después de la "r" se multiplicará por el valor de escala original del Estilo de Línea al que le apliquemos el fx, entonces, si por ejemplo queremos que ambas escalas aumenten al doble de su valor original, lo único que tenemos que poner sería:

\\fscxyr2

Si queremos que ambas escalas se modifiquen aleatoriamente de forma proporcional usamos \fscxyir



•  $\ \$  \\fscxyirRd(1, 2)  $\rightarrow$  \\fscxyr1.6  $\rightarrow$  \\fscxr1.6\\fscyr1.6  $\rightarrow$  \\fscx128\\fscy153.6

Para el próximo Tomo, seguiremos viendo más de las abreviaciones de los tags que podemos usar en el **Kara Effector** y que hará que hacer efectos sea una tarea aún más sencilla de lo que podríamos llegar a imaginarnos, así que estén pendiente de futuras entregas para que estén al tanto de todas la novedades.

Es todo por ahora para el **Tomo XXXVIII**. Intenten poner en práctica todos los ejemplos vistos y no olviden descargar la última actualización disponible del **Kara Effector 3.3** y visitarnos en la **Web Oficial**, lo mismo que en los canales de **YouTube** para descargar los nuevos Efectos o dejar algún comentario. Pueden visitarnos y dejar su comentario en nuestra página de **Facebook**:

- www.karaeffector.com
- www.facebook.com/karaeffector
- www.youtube.com/user/victor8607
- www.youtube.com/user/NatsuoKE
- www.youtube.com/user/karalaura2012