

APPLICARE EFFETTI AVANZATI ALLE SCRITTE

1. Innanzitutto l'applicazione di sopra contiene delle scritte diverse da quelle di cui sotto (vedete le prime due figure). Come prima cosa dovete quindi creare tutte le scritte e dividerle nelle due scene.



Fig. 1



Fig. 2

- 2.
3. Vediamo la prima scena: gli effetti da applicare alle varie scritte sono mostrati **qui**, mentre le Azioni da applicare alla scena 1 in Fig. 3 e alla scritta "Skip intro" in Fig. 4.



Fig. 3

```
Layout : Scene_1 | Script |
Add Script ▾ ↑ ↓ ↻ Guid
on (press) {
  getUrl ("main.htm");
}
```

Fig. 4

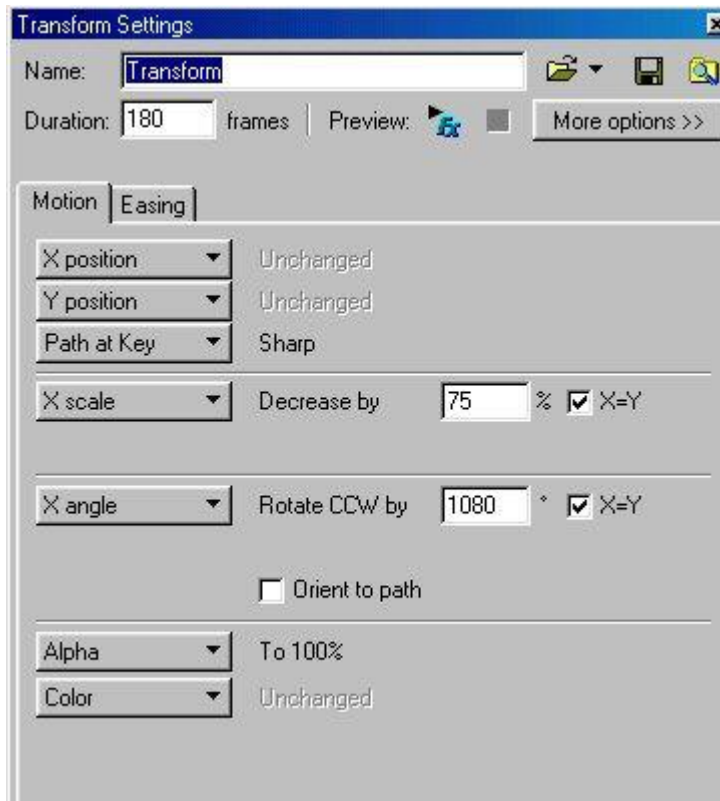


Fig. 5

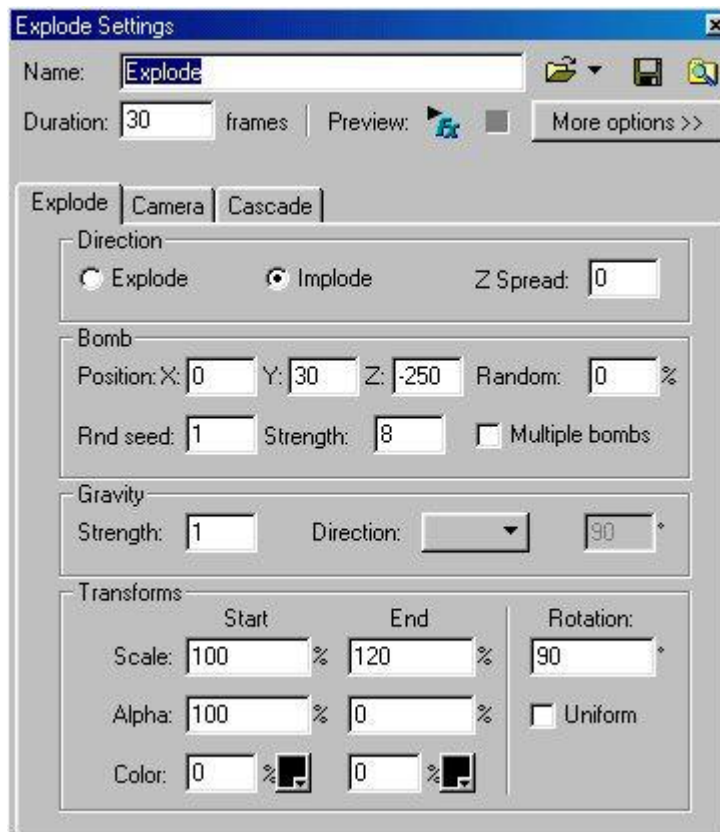


Fig. 6

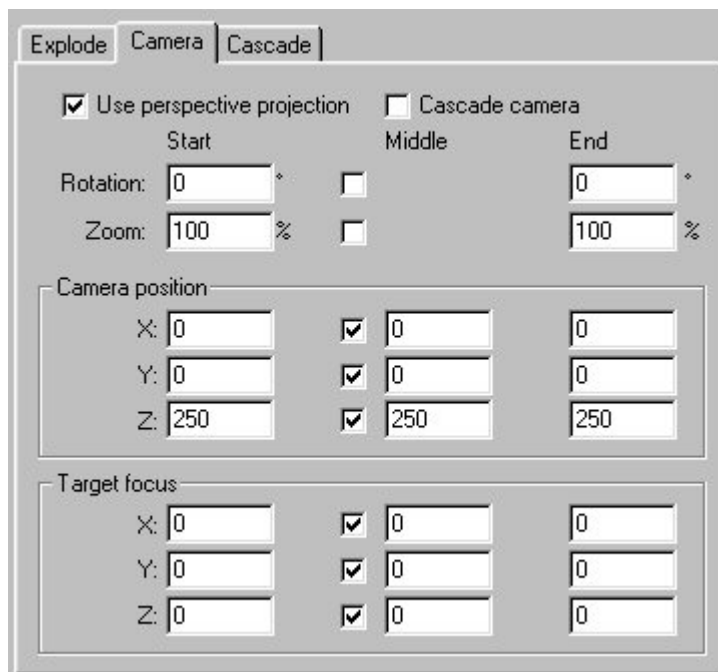


Fig. 7

Di seguito sono riportate le spiegazione delle Azioni `playSound` e `getURL`, mentre lo `SCRIPT (Evento)` on

(press) serve per i link/bottoni (vedi Fig. 8, 9):

Syntax

playSound(soundname, dontplayifloaded [, volume, loop, fadein, fadeout, effect])

Arguments

soundname: Name of imported sound (string).

dontplayifloaded: Boolean. If true, don't play sound if it is already playing.

volume: Volume to play sound, 0 to 100% (default = 100).

loop:: After sound has played, loop this many times. default = 0 (loop audio file one time).

fadein: Boolean. If true, fade in over 1st loop. Default = false.

fadeout: Boolean. If true, fade out during last loop. Default = false.

effect: One of the supported Effect constants, 'None', 'Fade In', 'Fade Out', 'Pan left to right', 'Pan right to left', 'Left channel only' or 'Right channel only'. Default = 'None'.

Note: Setting the loop value to "1" or "0" will play the sound through one time. Due to a limitation in the Flash player, there is no infinite loop setting. 65535 is the largest number you can have for the loop value and negative loop values will play the sound once.

Note: All of the parameters to the playSound/stopSound must be constant values (either quotes strings, numbers or true/false). You cannot use variables.

Returns

Nothing.

Description

Defines how a sound should be played.

Sample

On Frame 2, play the sound "chatinvt.wav" 5 times at a max volume of 90%.

Fade in on the first loop, fade out on the last loop. Pan from left to right on each loop.

```
onFrame (2) {  
    playSound("ChatInv.t.wav",true,90,5,true,true,'Pan Left to Right');  
}
```

Syntax

getURL(url {,window, method})

Arguments

url: Name of the URL to load (text / string).

window: Optional argument defining which Frame to load URL into. One of "_self", "_parent", "_top" or "_blank".

method: Optional argument consisting of a string defined as either "GET" or "POST" determining whether variables are retrieved using the GET or POST methods. The GET method will append the variables at the end of the URL (used for a small number of variables), and the POST method will send the variables in a separate HTTP header (used for longer strings of variables).

Returns

Nothing.

Description

Function: Loads the specified URL into the specified Frame. Note that Frame is an HTML Frame, not a Movie Frame in this situation.

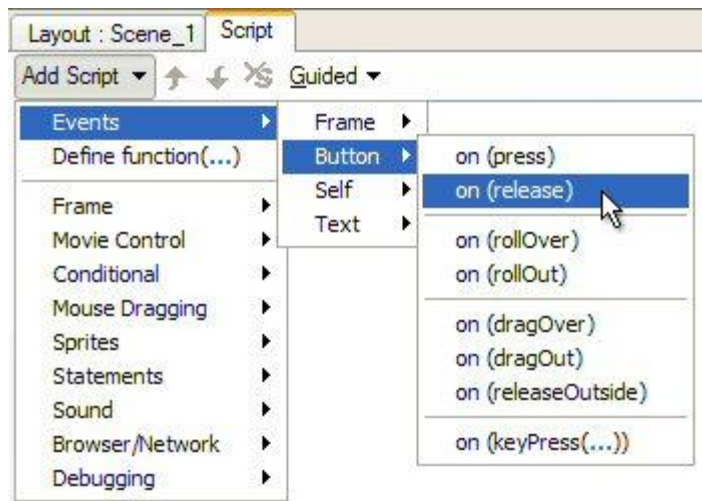


Fig. 8

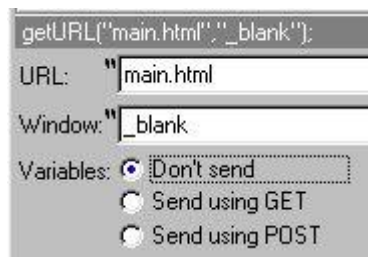


Fig. 9

4. Vediamo ora la scena 2: gli effetti nella Timelinebar sono mostrati **qui**, mentre nelle figure successive i settaggi dei nuovi effetti (*Add Effects* -> *Appear into Position* -> *Scale Letters* (fig. 10,11), *Cartwheel* (o *core Effects* -> *Vortex* in Fig. 12,13,14), *Explode* (fig. 15,16), *3D SPIN* (fig. 17,18,19), *Vortex* (fig. 20,21,22)):

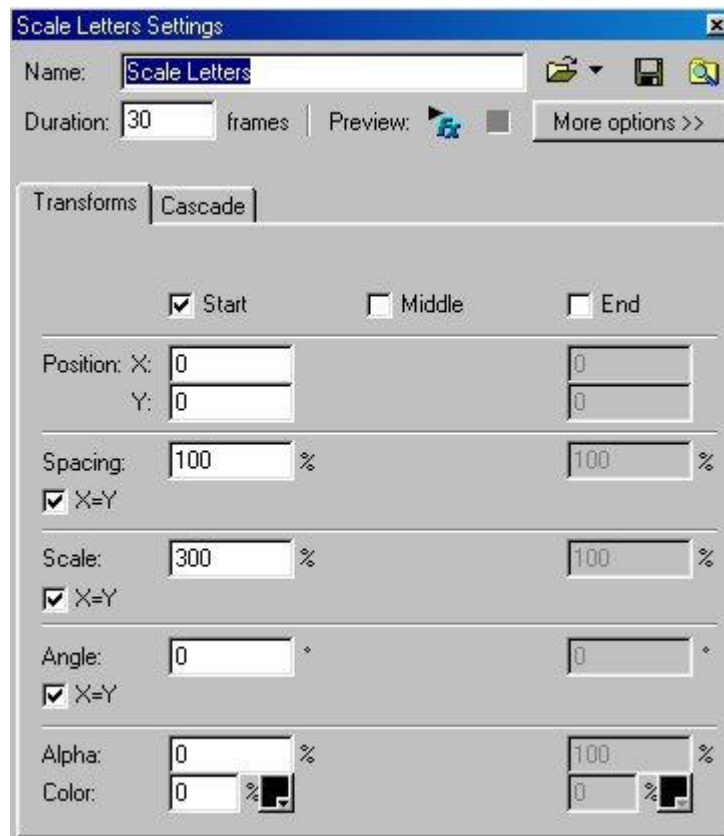


Fig. 10

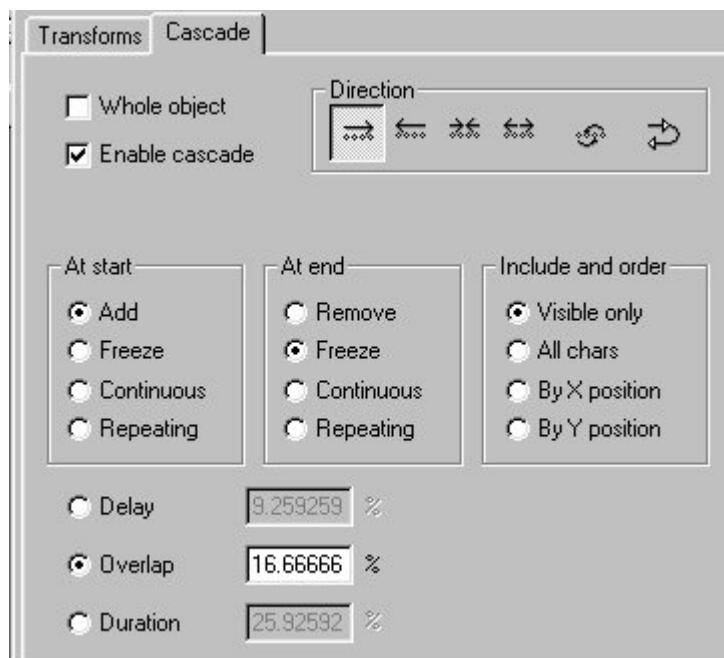


Fig. 11

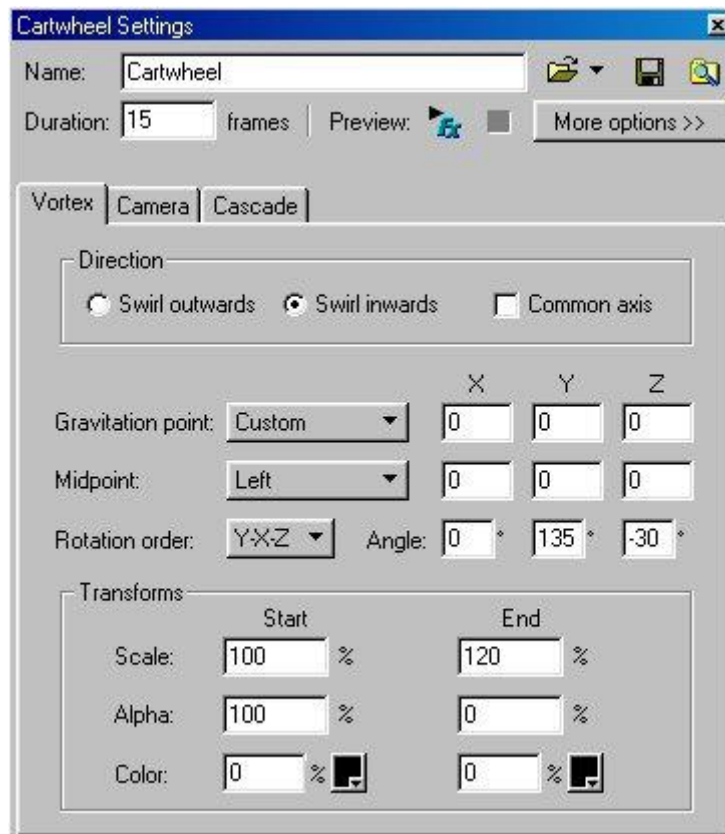


Fig. 12

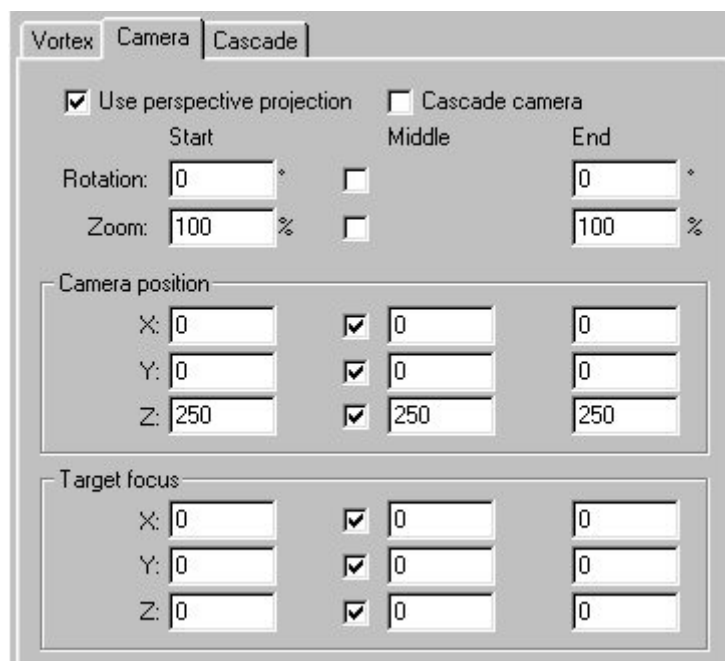


Fig. 13

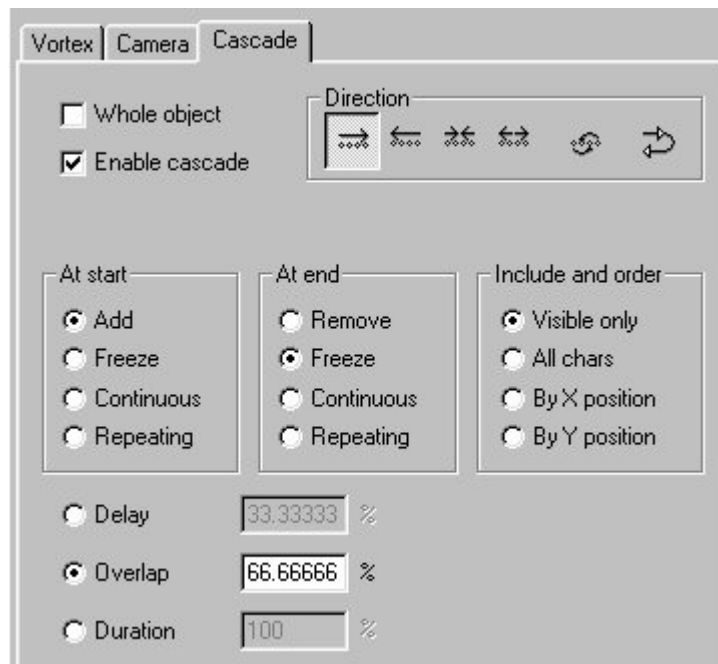


Fig. 14

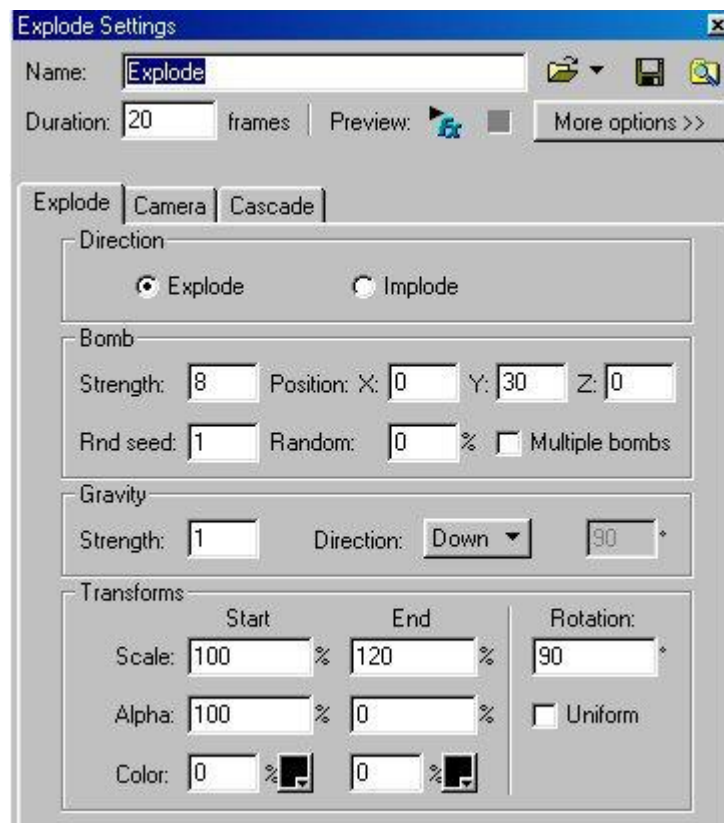


Fig. 15

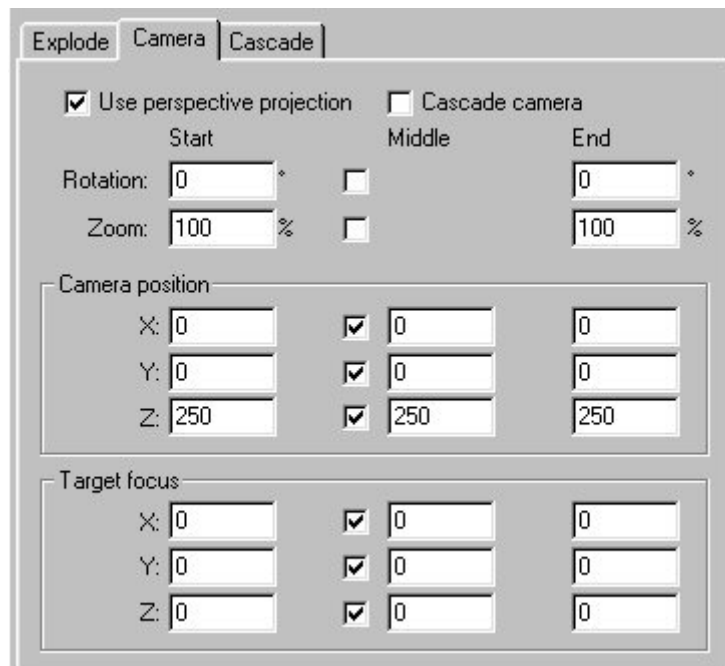


Fig. 16

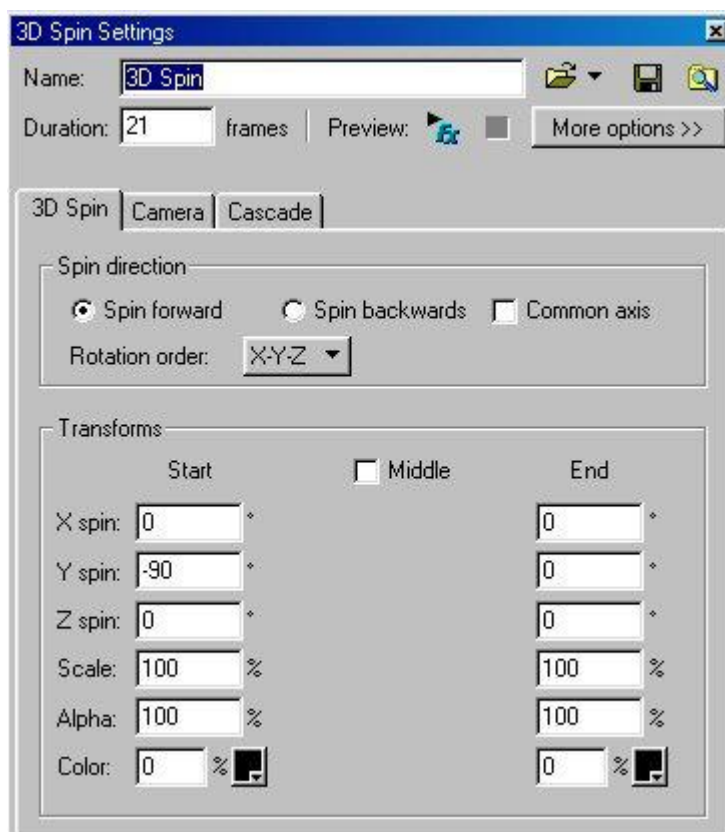


Fig. 17

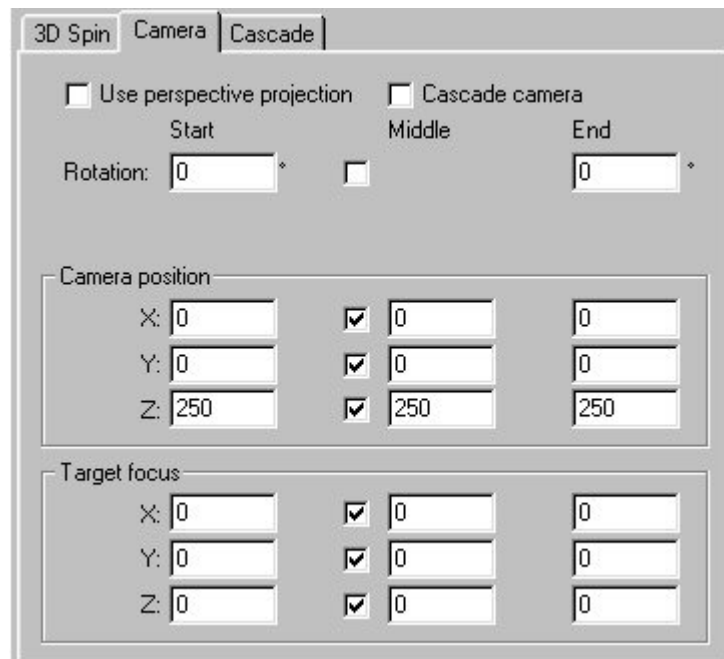


Fig. 18

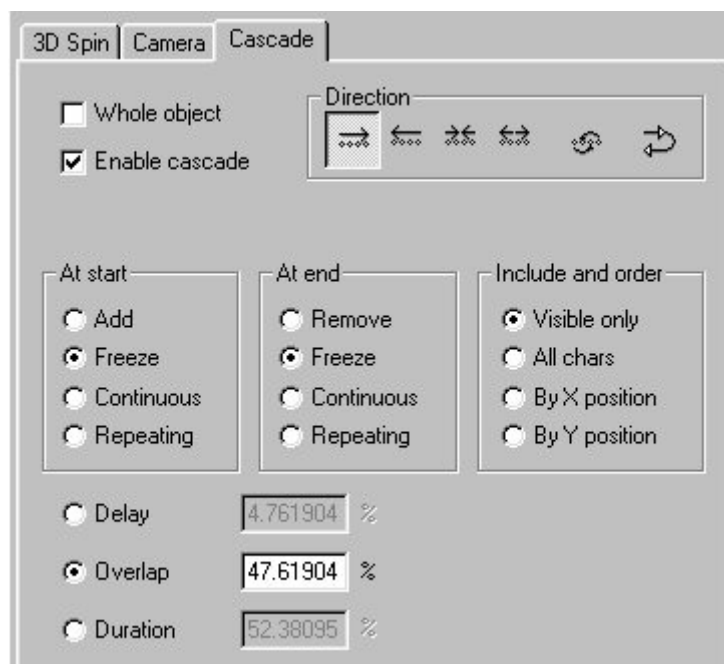


Fig. 19

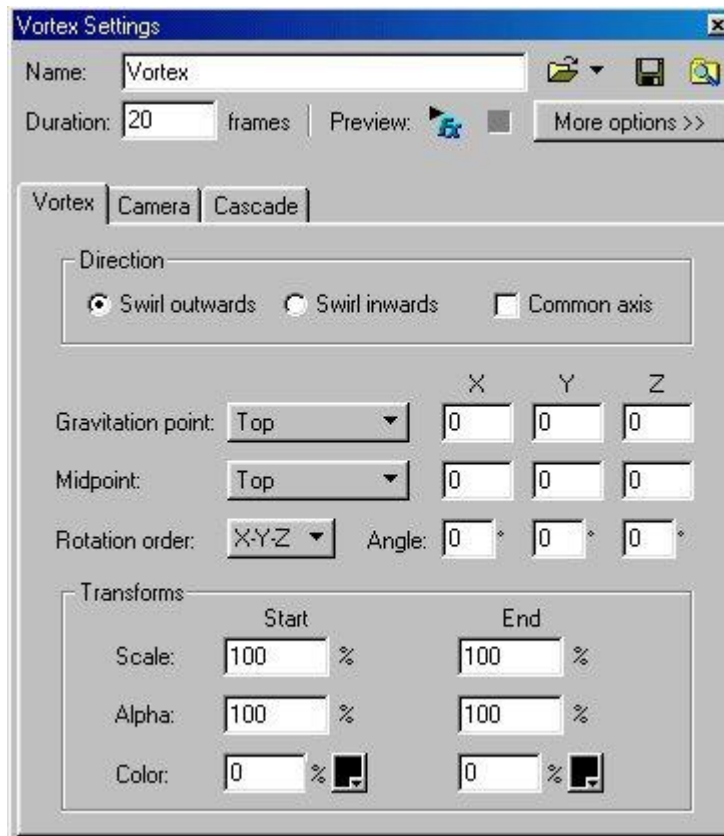


Fig. 20

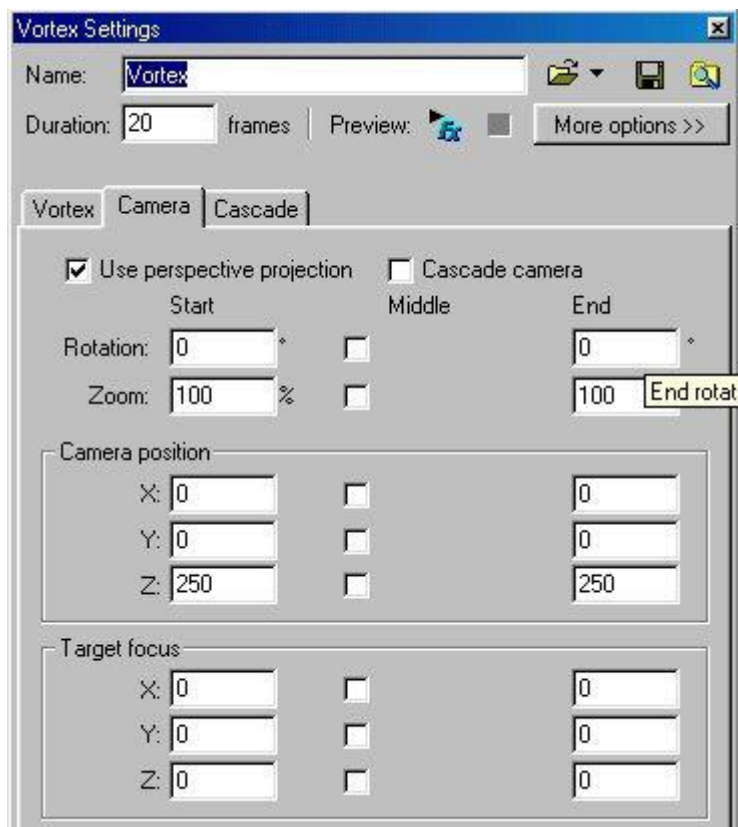


Fig. 21

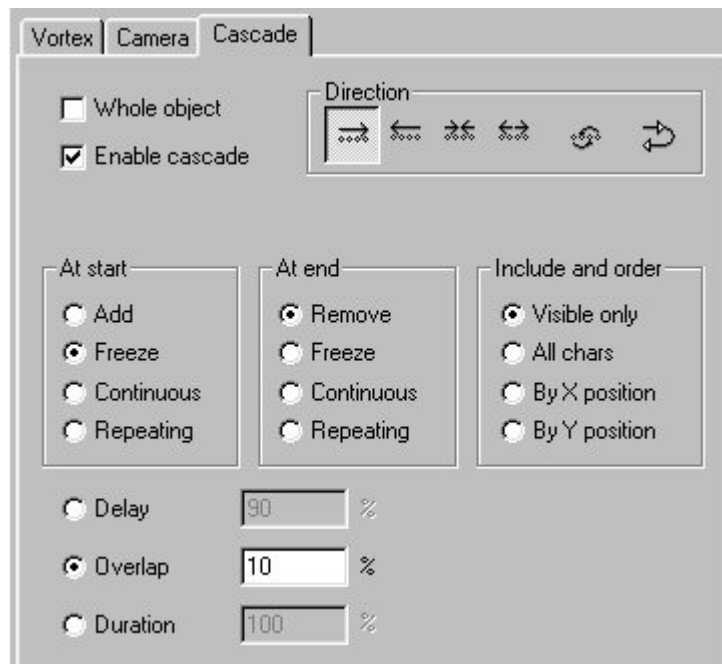


Fig. 22

5. Ora guardiamo le Azioni da mettere alla seconda scena (Fig. 23) e alle due scritte "Skip intro" e "Click to enter" (Fig. 24):



Fig. 21

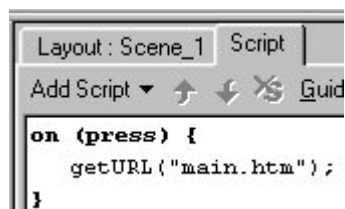


Fig. 22

6. A tal punto non rimane altro che salvare ed esportare l'applicazione.