

Hints and tips

Character and Background design

Always give Shapes and Elements unique names.

This makes things a lot easier when you import Elements into **CelAction2D**. You can, however, name Layers the same if they are in different files and destined for different Actors.

Do not leave Layers with their default names (i.e. Layer 1) as **CelAction2D** can check for unnamed Layers to warn you of potential problems.

Name Shapes and Elements consistently. There are an infinite number of ways to name Shapes and Elements, but here is a naming method which works well, by way of example:

Elements should be named as simply as possible, e.g. Head, Mouth. If there are left and right versions of Elements (such as hands or feet) they should be given numbers or letters to identify them (e.g. Hand 1, Foot A), because when you mirror an Actor the left and right designations may no longer apply.

Shape names should include their Element, Group (if applicable) and position (or phoneme) in Group, e.g. Mouth Sad Ah, Mouth Happy Ee, Hand 1 Clenched, Eyelids Shut, Eyelids Halfopen.

With alternate Shapes, put the Element name first - in the future, **CelAction2D** may give you the option to strip the Element name from the Shape name, and so making it consistent now will ready your Actors to take advantage of this possible feature.

Name your Layers using safe characters only.

Only use the characters A-Z, a-z, 0-9, and the space (), dash (-), plus (+) and underscore () symbols in Layer names. Specifically, do NOT use normal brackets or non-English characters. This ensures that the Layer names will appear as you intended them to when imported into **CelAction2D**. For non-English characters with accents, use the corresponding character without an accent (i.e for 'é' use 'e').

Give the animators notes about the character construction.

If it is not absolutely clear from the character design itself, make sure that detailed instructions are given as to how the character should be built. This will save a lot of time during the construction and animation processes.

Make blinks alternate Shapes.

When characters blink, instead of making a separate Blink Element that can just be vanished, consider making it an alternate Shape, paired with a blank Shape. This way, you get the same single keypress functionality to make a character blink, but you also make it very easy to add a half-blink Shape later without having to rebuild the Actor.

Put camera instructions in the Background file.

When designing backgrounds, you can put camera guides as separate layers to make it absolutely clear where the boundaries are and what the intentions of the designer are. For maximum clarity, put start and end positions for camera moves, and any positions in between if the move is not a straight line.

Illustrator

Put an Align Box in every file.

Because of the way Illustrator stores Layers, you MUST make a special Layer that is larger than the largest Element in the file. This Layer (call it "Align Box" for standardisation purposes) does not get added to the Actor in **CelAction2D**, but it is used as a reference when Shapes are updated. Once you have made the Align Box, you cannot change any Layer in the Illustrator file to go outside it, or updating Shapes will not work correctly. If in doubt, make it very large.

Do not move Layers in the Illustrator file once the Actor has been built.

CelAction2D remembers where Shapes are in each Illustrator file you make, so that if you move the Shapes and then update the Actor, those Shapes will be moved accordingly - you may not always want this! So take care when adjusting the original file. You may change the Layer Order, however, but it will not affect the Display Order if the Actor has already been built.

Don't use Clipping Paths, Masks, Graphs, unexpanded Text, Placed Art, Gradient Meshes or Rasterised images in the Illustrator file.

These functions are not currently supported by **CelAction2D**. Normal gradients and transparent Sublayers can be converted for use.

Don't use CMYK colours, use only RGB.

CMYK conversion is not an exact science, and the quality varies from program to program. So to be safe, always work in RGB for TV, film or web output, and only use CMYK when going straight to print.

Do not use Blending modes.

Blending modes are not supported in **CelAction2D**. You can add them afterwards using Effects.

Photoshop

Put an Align Box in every file.

Because of the way Illustrator stores Layers, you MUST make a special Layer that is larger than the largest Element in the file. This Layer (call it "Align Box" for standardisation purposes) does not get added to the Actor in **CelAction2D**, but it is used as a reference when Shapes are updated. Once you have made the Align Box, you cannot change any Layer in the Illustrator file to go outside it, or updating Shapes will not work correctly. If in doubt, make it very large.

Do not move Layers in the Photoshop file once the Actor has been built.

CelAction2D remembers where Shapes are in each Photoshop file you make, so that if you move the Shapes and then update the Actor, those Shapes will be moved accordingly - you may not always want this! So take care when adjusting the original file. You may change the Layer Order, however, but it will not affect the Display Order if the Actor has already been built.

Do not use the Background Layer.

Using the Background Layer in Photoshop can cause unwanted side effects in **CelAction2D**. If you have used the Background layer, copy it to make a normal Layer, then delete the original Background Layer.

Do not use Styles.

Styles are not supported in **CelAction2D**.

Do not use Blending modes.

Blending modes are not supported in **CelAction2D**. You can add them afterwards using Effects.

CelAction2D

If in doubt, Right Click!

Whenever you want a function but you can't remember where it is, try Right-clicking the mouse to get a context-sensitive menu. More often than not, you will find what you are looking for there.

Do remember, however, that Right-clicking is sensitive to the window you are clicking in, so click in the window containing the data you want to affect. For example, moving an element 10 pixels to the left would require clicking in the Scene View, whereas blanking transforms over 20 frames would be in the Dope Sheet view.

Also note that the context changes whether you have a single frame or multiple frames selected in the Dope Sheet. You can't apply an Algorithm over a single frame, so the option isn't shown. Select 2 or more frames, however, and the option will appear.

Save your work often, using different file names (such as scene1ver01, scene1ver02 etc).

It is important to keep older versions of the file you are working on, at least until you are positive you don't need them anymore. It is easy to make sweeping changes to a **CelAction2D** scene, and even though you have an Undo function, it's easier to keep an older scene for comparison's sake.

We have implemented an AutoVersion button which makes this task much easier.

Exit and restart CelAction2D regularly.

Because **CelAction2D** keeps Actors in memory between scenes, the more scenes you load, the more memory will be filled up, and performance will eventually degrade. By exiting and restarting, you clear **CelAction2D**'s memory and the program will work more efficiently.

Loading more than one Shape file into the Pool simultaneously.

Although you cannot select more than one Shape file at a time when using the Add Elements to Pool dialog box, if you select them in a folder outside of **CelAction2D** and drag them into the Assemble window, it will load all of them simultaneously.

Don't change screen resolution or colour depth while CelAction2D is running.

Although the program will still work, if you are making a preview when the screen mode is changed, black frames may appear, due to certain codecs requiring certain screen modes.

Don't use Reduce Interlace Flicker option when compositing.

Because Reduce Interlace Flicker compromises the quality of an image to make it look better when viewed on a TV, this process should be done *after* all compositing has taken place.

If any applications crash or exit abnormally, it is wise to reboot the machine.

Although Windows XP, Vista and Windows 7 are extremely good at protecting running applications from crashes elsewhere on the machine, large areas of memory and resources may be tied up by the crash, and will not be freed until the machine is rebooted. Therefore if you are experiencing erratic performance after a crash, it is advisable to save your work and reboot.