

PARTIGEN 2

Getting Started Guide

Version 2.5

Table of Contents

Emitter Pane	3
<i>Emitter Configuration Controls</i>	<i>4</i>
<i>Particle Configuration Controls</i>	<i>5</i>
<i>Controller Keyframe Graph</i>	<i>5</i>
Library Pane	6
Emitter Usage	7
<i>Using the Emitter Component</i>	<i>7</i>
<i>Custom Particles</i>	<i>8</i>
Bitmap/PixelRenderer Usage	9
<i>Using The BitmapCanvas Component</i>	<i>9</i>
<i>Notes on Using BitmapCanvases</i>	<i>10</i>
Upgrading	11
<i>Interface Changes</i>	<i>11</i>
<i>API Changes</i>	<i>11</i>
<i>Migrating From Partigen v2.0 to v2.5</i>	<i>12</i>
Advanced Developer Tips	13
<i>Keyboard Shortcuts</i>	<i>13</i>

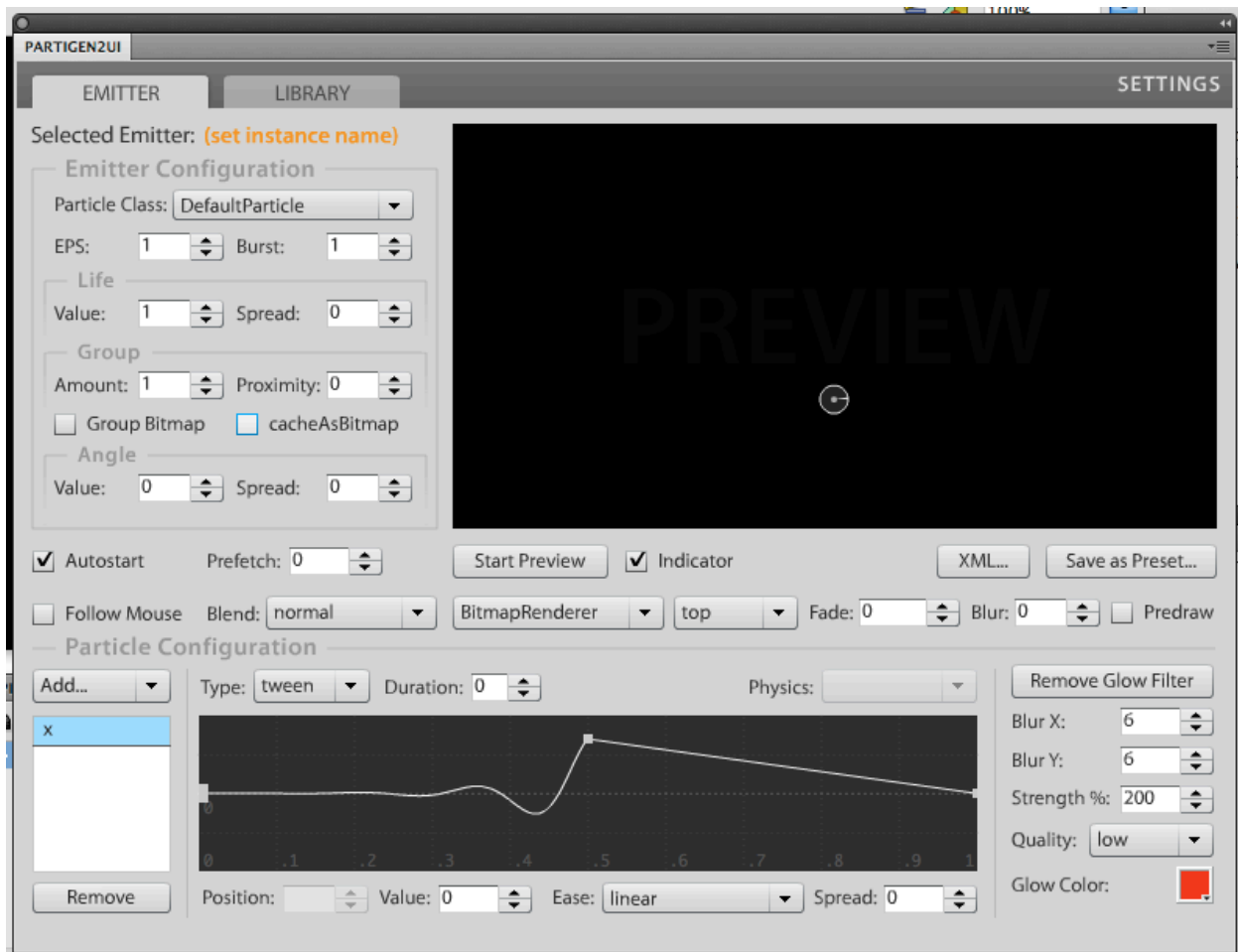
Emitter Pane

The Emitter Pane is the area of the Partigen 2 UI that deals with the configuration of the currently selected emitter.

Partigen 2 is the only particle effects component for Flash that features a large preview of your effect. This allows you to get feedback of your changes in real time, without having to test and export your movie each time you tweak a setting.

You can configure every aspect of an emitter through this panel. Each control represents a corresponding ActionScript method or property, so anything configured in the IDE can be controlled via AS at runtime – 100% dynamically.

For more details, visit the AS3 online API documentation available at our website.



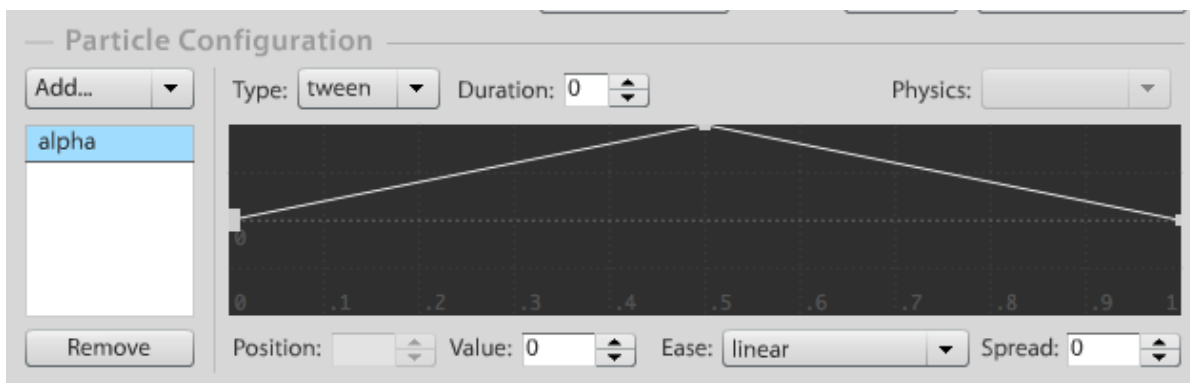
Emitter Configuration Controls

- **Selected Emitter:** this is the emitter currently selected. It will also display a notice if your current selection isn't an emitter.
- **Particle Class:** the current Class of particle you're using. This is a drop-down of all available classes that have been loaded with all Partigen Effect Libraries (PELs).
- **EPS:** how many times to emit per second
- **Burst:** how many particles to create with each emission
- **Life Value:** how many seconds for particles to live
- **Life Spread:** creates a random range of life by adding the life value and this number
- **Group Amount:** how many instances of the particle class to create in a single particle
- **Group Proximity:** a random radius away from the center to position each particle
- **Group Bitmap:** copies all the particles in a group and creates a Bitmap object to use as the particle instead of multiple instances of the particle class
- **cacheAsBitmap:** this sets the cacheAsBitmap property of each particle
- **Angle Value (physics only):** this sets the angle used for any controllers that use physics
- **Angle Spread (physics only):** this creates a random range from the Angle Value

- **Auto Start:** this toggles the emitter starting automatically or not. If not, you will need to start it via ActionScript
- **Follow Mouse:** this attaches the emitter to the mouse at runtime. Should be used for development purposes only.
- **Prefetch:** the amount of seconds to simulate already gone by when first starting the emitter (for effects that require a "build up" time)
- **Blend:** this sets the blendMode of each particle

- **Start/Stop Preview:** this toggles the live emitter preview.
- **Show Indicator:** this shows the emitter indicator on the stage (with angle range). Should be used for development purposes only.
- **XML:** this opens up the XML Configuration window to allow you to generate the current emitter's XML, to apply pasted XML to the current emitter, or edit properties not available as a UI control
- **Save as Preset:** this allows you to name and save the current emitter as a preset in your library.
- **Renderer (drop-down):** selects which renderer to use: Null, Standard, Bitmap, Pixel
- **Order:** the visual ordering or "staging" of particles when they're created

- **Fade (Bitmap/Pixel):** how much “trailing” (fading) to apply to the particles
- **Blur (Bitmap/Pixel):** how much blurring to apply to the trails (only if fading > 0)
- **Predraw (Bitmap/Pixel):** draws the particle to Bitmap before the fade/blur is applied



Particle Configuration Controls

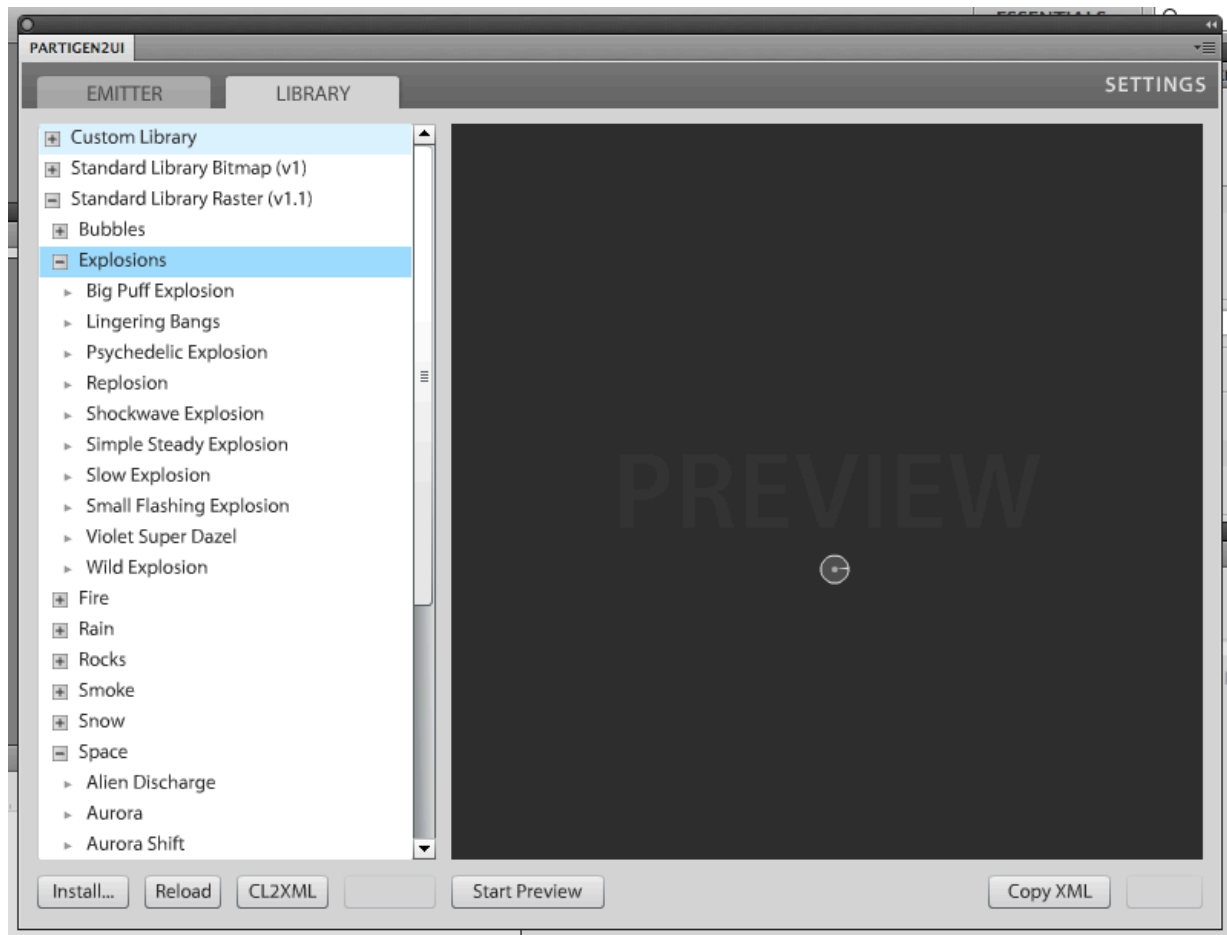
- **Add (drop-down):** this will add a controller for the specified property. The default controller is “Tween”, but you can change this at anytime.
- **Remove:** this removes the currently selected controller property
- **Type:** choose between a “Tween” or “Physics” controller
- **Duration:** how long to set the controller length. 0 Defaults to the life of the particle.
- **Physics:** choose between velocity, acceleration, or friction controllers.

Controller Keyframe Graph

The graph displays a visual representation of the keyframes for the controller. You can set the value of a property throughout its entire life. The horizontal axis represents the position percent: .1 = 10% the duration, .5 = half way the duration, 1 = the end value.

- To add a keyframe, right click on the graph and select “Add keyframe here”.
- To remove a keyframe, right click it and select “Remove keyframe”.
- You can drag keyframes around to update their positions and values.
- **Position:** the position (0-1) of the keyframe. The begin and end keyframes can not be moves.
- **Value:** the value for the keyframe
- **Ease:** choose the easing for the keyframe. It’s the easing used from the previous value.
- **Spread:** the spread value to add to the value to create a random range

Library Pane



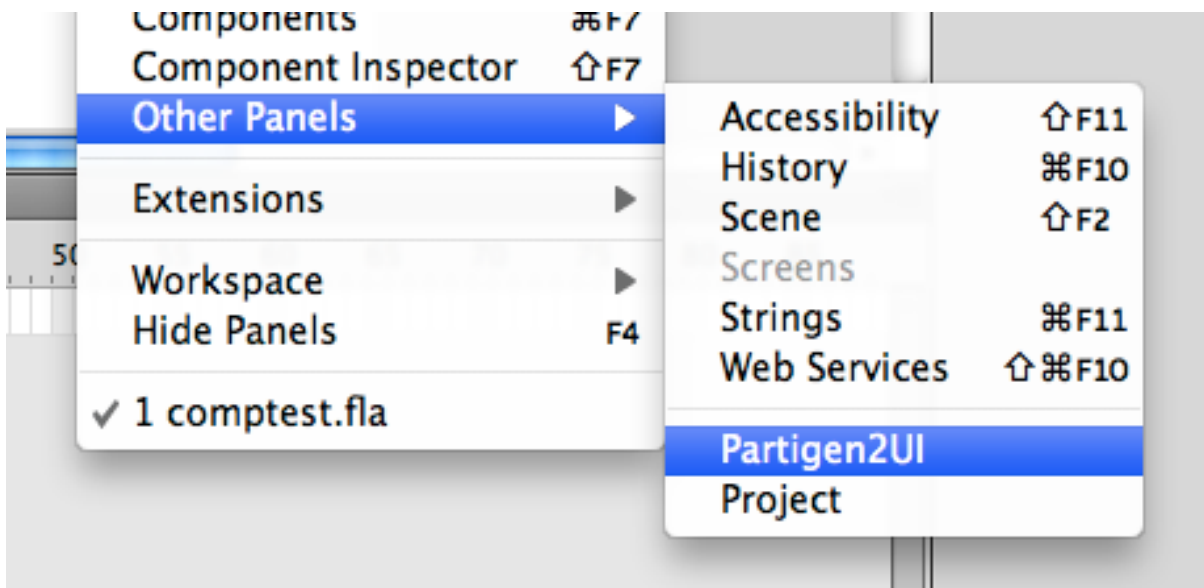
- **Install:** this install new effects – prompts you to select a .pel file, selecting this file will copy it to Partigen’s Library folder, install it, and reload your library.
- **Reload:** this removes all the current presets and cleans the decompressed SWC files out, then loads all the PEL files and rebuilds the library.
- **CL2XML:** this takes all of your Custom Library presets and creates PEL-ready XML and copies it to your clipboard. You can paste this directly into a library.xml file and create your own PEL file without any extra work.
- **Remove:** this removes a custom preset (and is enabled only for custom presets)
- **Start/Stop Preview:** this toggles the preview of the preset
- **Copy XML:** this copies the current preset’s XML configuration to your clipboard.
- **Use:** this applies the current preset to the currently selected emitter

Emitter Usage

Partigen 2 simplifies the process of using emitters and particle effects. No need to “set up” your document or add any ActionScript. It works just how you expect, without any compromises – just drag & drop.

Using the Emitter Component

1. To get started, simply open your AS3 document and drag the “Emitter” component from the Components Panel (under Partigen 2) in the Flash IDE onto your stage.
2. Open the Partigen 2 Interface by going to Window > Other Panels > Partigen2UI



3. Next, select your emitter on the stage, and the Partigen2UI will automatically detect the emitter and load it's current settings.
4. Once selected, you can configure the emitter how you see fit, or use a preset from the Partigen 2 Library. Presets can all be modified and tweaked to your desire.
5. That's it! You can easily tween the emitter around stage, or use AS3 to control the emitter's settings, including stopping, starting, and much more. Just reference the emitter's instance name in AS and you can control it just like it was made in AS.

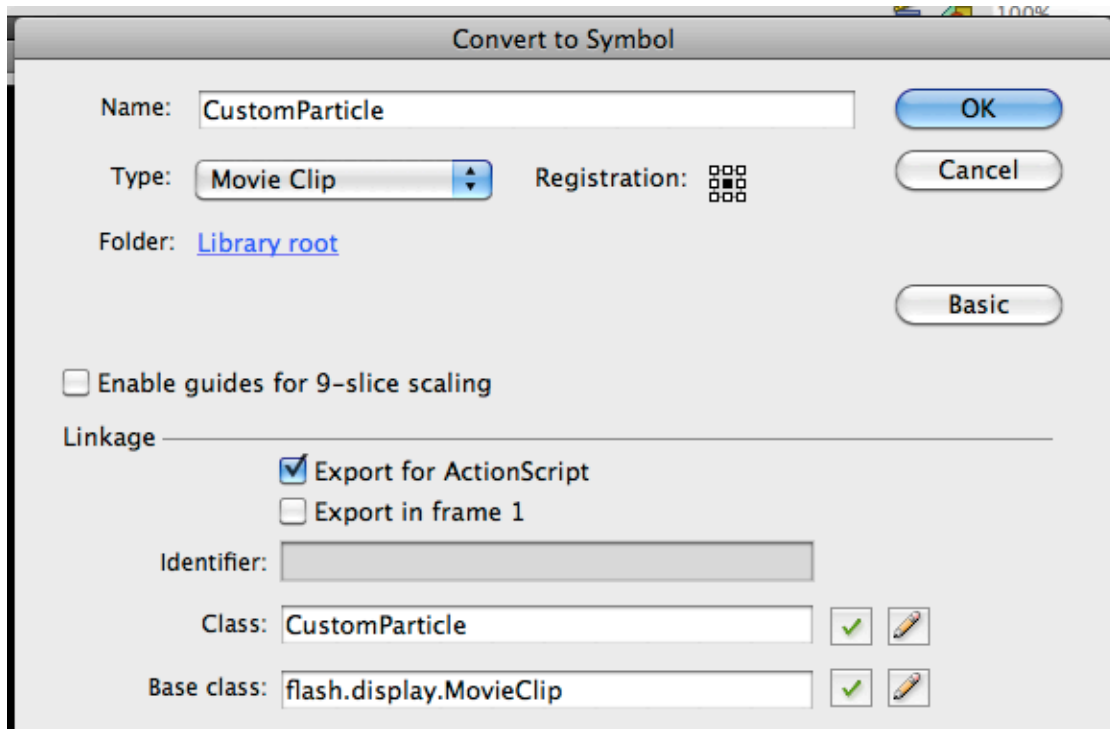
Please see the AS3 API documentation for more information on interacting with Emitters components via ActionScript and creating them through code.

Custom Particles

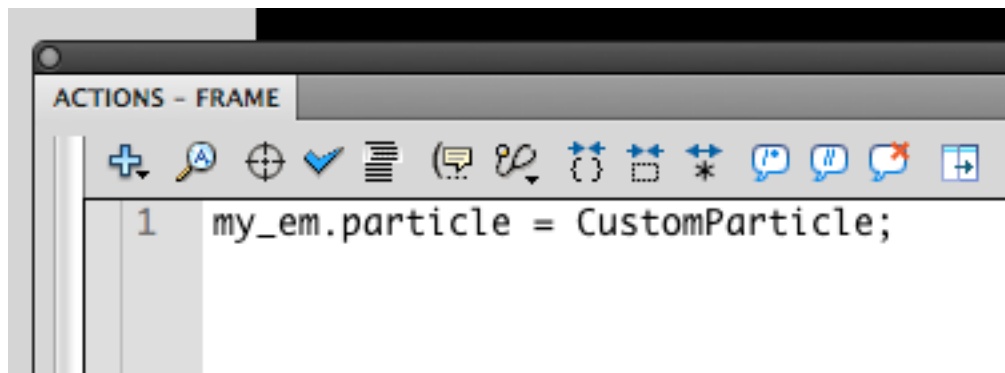
While Partigen 2 comes with an easy way to use preset particles and effects, you can easily use your own. This requires a single line of ActionScript to assign your class.

As of Partigen version 2.5, you no longer need to extend the Particle base class. **You can now use any class** that inherits the DisplayObject (i.e. any MovieClip Symbol in your library exported for ActionScript with a Class name).

Here's an example of what that would look like (just like creating anything else):



The last step is to set the custom particle to the emitter. Setting the particle is the same as if you created your emitter with ActionScript, or with a component. Simply set the “particle” property of your emitter to the class name of your custom particle.



Bitmap/PixelRenderer Usage

With Partigen 2.5, you can now select which renderer you'd like to use through the interface, as well as through XML. This lets presets contain renderer information, allowing incredible new bitmap-based effects to be save, shared, and installed without any ActionScript needed.

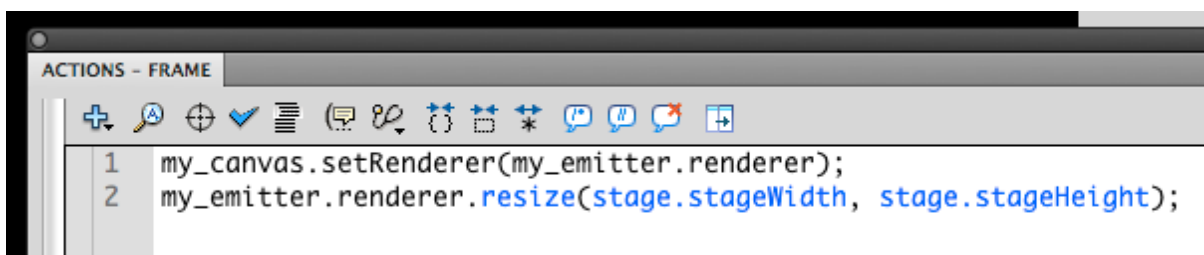
When using either BitmapRenderers or PixelRenderers, you'll need to use a BitmapCanvas to display the effect on the screen, since unlike the StandardRenderer that use Flash's DisplayList, these use BitmapData to renderer particles.

The BitmapCanvas class is an easy way to show these complex effects without worrying about lots of configuring or what's going on in the background.

Using The BitmapCanvas Component

If your emitter is using a Bitmap or Pixel renderer, you'll need to use a BitmapCanvas to actually display your effects on the screen. While creating this via ActionScript is easy, using a component on the stage allows you even more design-time control unavailable anywhere else.

1. To get started, let's assume you already have an emitter component on your stage that's using a BitmapRenderer. If you test your movie now, you won't see any particles – this is normal.
2. Drag the “BitmapCanvas” component from the Components Panel (under Partigen 2) in the Flash IDE onto your stage and give it an instance name.
3. Set the x and y positions to 0, aligning it to the upper left of your stage. This will ensure the rendering of particles aligns with your emitter.
4. Deselect everything and click on the first keyframe. Open the Actions Panel, here we require 2 lines of AS: 1 to tell our BitmapCanvas which emitter to use, and 1 to set the size of the renderer.
5. First, call “setRenderer” on the instance of your BitmapCanvas component, passing your emitter instance's renderer variable (my_emitter.renderer).
6. Then call “resize” on your emitter's renderer, passing the stage with & height.



Notes on Using BitmapCanvases

- Multiple BitmapCanvas components can be used per emitter/renderer to duplicate effects and increase performance by avoiding duplicating entire emitters
- You only need to call `resize` once per renderer, or when your stage resizes – the BitmapCanvas components will automatically detect the renderer's resize
- Filters and `blendModes` can be applied to BitmapCanvases via `ActionScript` or nesting them in `MovieClips`
- You can even rotate, move, and tween BitmapCanvas components to make quick changes to the direction of effects without changing any `ParticleControllers`
- You can turn off the indicator through the `Component Inspector`
- Positioning the BitmapCanvas other than 0,0 will offset the effect from the location of the emitter
- The renderer for a BitmapCanvas component can be changed dynamically at runtime
- You can improve the performance of `BitmapRenderer`-based effects by calling its `resize()` method to set a smaller area if you don't need the entire stage

Upgrading

This section covers everything you need to know if you're upgrading from a previous version of the Partigen plugin to version 2.5. If this is the first time using Partigen, you may safely skip over this section.

Interface Changes

There have been numerous changes to the interface and components. Here's a list of some major changes to be aware of:

- The interface window has been increased, as well as the preview areas, and the emitter XML pane has been greatly increased to allow easier editing
- There are 2 new options available in the Settings Pane: *enable undo* & *enable cache*
 - You can now undo and redo changes made to the emitter through the native Flash undo/redo/history methods
 - The Partigen2UI library now uses proprietary caching technology to speed up the load time of the interface considerably
- There is now controls for Glow filters on particles
- Many new interface controls and the ability to choose renderers
- The render to parent parameter on the Emitter component has been removed
- A new component is available: BitmapCanvas
- Particle components don't inherit partigen.particles.Particle anymore, as well as the removal of predefined blendmodes in the particle components

API Changes

There have also been many updates to the core AS3 engine that powers Partigen. Most features and changes have been made to be forwards compatible, but there are a few important API changes that can effect your current movies to not work properly.

- BitmapRenderers no longer accept a Bitmap object, and instead are now used with BitmapCanvases to display effects
- Emitter.groupAmount now needs to be at least 1 (0 will display nothing)
- Pooling has been overhauled, and the default pool for emitters is BasicPool
- Emitter.start() now accepts the 'prefetch' parameter first
- Emitter.renderer.start() should no longer be called – the starting/stopping of rendering is now handled automagically, and is very efficient
- Pools now require the particleBaseClass in the constructor

Migrating From Partigen v2.0 to v2.5

If you have old projects that use a previous version of Partigen 2.0, you can benefit from the new features and performance enhancements of v2.5.

Migrating old projects to Partigen 2.5 is ridiculously easy thanks to intelligent helpers in the Partigen2UI that automatically update your old presets to the current format, as well as automatically reloading the new libraries, and even migrating old emitter configurations to latest and greatest format as soon as you select them.

There's significant changes to the XML configuration in this latest release, and luckily for you, Partigen makes the update absolutely seamless.

That being said, there are a few things you'll have to do on your own:

- Update any AS3 conflicts with the new API changes as noted in the previous section
- You need to update your current Emitter components in your FLA library by dragging the current one from the Components Panel (be sure to select "Replace" when prompted that it already exists)
- **Note:** be sure you're also using the same AS3 classes included with this release. Using newer or older releases may cause incompatibilities.

Advanced Developer Tips

These are some advanced tips for developers that most users won't need on a daily basis, yet provide power-users with some extras.

- Opening the XML Config pane in the UI and applying a blank text field will reset the emitter to its default settings
- Using the XML config pane, you can add in your own filters (besides Glow) to your emitters settings, and these will even be saved through presets and PELs. Just add an XML node for the filter you want to use as a child of <Filters />

Look at the Glow filter XML for an example of what this looks like. Hint: you can use: `com.desuade.utils.XMLHelper.objectToXML(my_filter_object)`

- Using the CL2XML feature, you can paste the XML into a file called "library.xml", zip it up, and rename the zip file "Exported_Custom_Library.pel" (be sure to remove the .zip extension, if using OS X, you may have to view the file's info and change it there).

If you want to name it different, change the name in the library.xml file and mirror the name changes to the file name. They must match, and spaces turn to underscores. If there's any custom particles used, you can include each as a .SWC file inside the zip alongside of library.xml.

Keyboard Shortcuts

There are also various keyboard shortcuts for the interface, giving you access to some behind-the-scenes magic. These are for development and debugging purposes and are not considered supported features.

- Show developer log – combo: SL
- Clear developer log – combo: CL
- Load all PELs – combo: LP (*hold for 2 seconds*)
- Unload all PELs – combo: UP (*hold for 2 seconds*)
- Clear Library cache – combo: RC (*hold for 2 seconds*)
- Remove all custom presets – combo: RP (*hold for 4 seconds*)

Note: the way Flash handles keyboard shortcuts through the SWF interface is sometimes a bit random, and you may have to click inside the Partigen2UI a few times or on a UI control for Flash to detect your keystrokes.