Getting started with the Advanced CSS Animator

In this article, Nancy will introduce you to the DMXzone's latest extension, the Advanced CSS Animator. She will introduce you to the extension and take you through 3 very simple examples of how you can use it to create some wonderful CSS Animation. And, unlike CSS animation that only works in a few browsers, the Advanced CSS Animator has seamlessly integrated jQuery functions so they work in all major browsers.

CSS Animation

In the early days of the web, most animation was done with Animated GIFs. They provided a little bit of movement but due, in part, to the 255 color limitation on a GIF, the animations didn't always look that great. Along came a little program called Future Splash and their now famous "Baby on the Beach". Now THAT was animation.   Future Splash was quickly purchased by Macromedia and Flash was born. Unfortunately, Flash was all too often overdone by animators and would-be animators who put bandwidth-consuming splash pages on everything. Flash began to get a bad name when in reality, it wasn't the program, but rather the use of the program that deserved the heat. Flash has since evolved to do a lot more than animation and Flash is truly an amazing tool, but what is really stealing the spotlight lately is CSS Animation with the coming of CSS3.

As you would expect, the biggest problem with CSS3 at the moment is browser adoption. Google Chrome, Opera (Labs version) and Safari 4 support it fairly well due to the Webkit and Mozilla engines they utilize. All well and good, but who is going to spend weeks developing applications for 10% of the public? Firefox support is coming by way of beta at the moment but not as of this writing in the release version. Internet Explorer does not support CSS3 yet but has promised some support in IE9, purported to be available around the end of this year.

Meet the DMXzone Advanced CSS Animator

The DMXzone creative gurus put their heads together over this dilemma and came up with a solution that will work whether you are running Chrome Latest or Internet Explorer 6! Enter the Advanced CSS Animator, a Dreamweaver CS5 extension so versatile that it can determine what user selections will work with just CSS and what selections need a little help to work in the browsers we're stuck with today. Where needed, the CSS gets a little help from jQuery scripts that allow you to easily upload them to a Scripts Library so there is nothing to stop your effects from working in whatever browser your users might be viewing your content.

The **Advanced CSS Animator** installs like any other Dreamweaver extension, by either double clicking on the file to open the Extension Manager or opening the Extension Manager through the main computer menu or from within Dreamweaver at either Commands/Manage Extensions, Help/Manage Extensions or the Extend Dreamweaver button at the top of the program interface window. If you use the second way, then just click Install on the Extension Manager interface and navigate to where you saved the Advanced CSS Animator and it will install the extension for you.

Restart Dreamweaver to get the extension recognized by the program and you're ready to go. Since it's important to me that this article show you the WOW factor, I'm going to take you through a few demonstrations and explain how the extension works as we go along. With this article, I have included a zip file with the completed examples used here and a page called template.html to get you started. It's not a real Dreamweaver template, but rather a very simple file with a div that defines a box that you will animate in different ways in each demonstration.

Like any Dreamweaver Extension, the **Advanced CSS Animator** requires an open document so you will need to create a new HTML document in order to activate the Behaviors Panel to gain access to the extension. As with Dreamweaver CS4, the Behaviors Panel is coupled with the Tag Inspector panel. Once you activate that panel and select Behaviors, clicking the **+** sign will reveal the dropdown list of Behaviors available and DMXzone is listed. If you have other DMXzone behaviors installed, you will see a menu listing all of them, but if not, then just Advanced CSS Animator. Clicking on that choice will reveal the interface pictured below.

The Main Interface of the Advanced CSS Animator

The first thing we notice is two tabs labelled CSS and Presets. The **CSS** tab allows you to select CSS properties in three groups, Basic CSS Properties, Advanced CSS Properties or Custom property, which allows you to create your own custom settings. The **Presets** tab is shown on the right with the Built In Presets showing and a folder below it where you can store Custom Presets that you create. Let's see how this works, ok?

Example 1: Move It!

Time to take this thing for a spin and kick the tires a bit. The name of the extension is **Advanced CSS Animator**, so let's **MOVE IT**! Use either template.html from the included files or if you prefer to make your own from scratch, make a new HTML document in Dreamweaver with the following code:

```
<!DOCTYPE HTML>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Untitled Document</title>
<style type="text/css">
#box1 {
      margin: 5px;
      padding: 5px;
      float: left;
      height: 60px;
      width: 200px;
      border: solid #f00 2px;
}
</style>

</head>


<body>
<div id="box1">my box</div>
</body>
</html>
```

NOTE: It should be noted that, while I used the HTML5 doctype in these examples, HTML 4.01 or XHTML doctypes will work also.

The document contains one div and the CSS for this div makes it into a box 200px x 60px with a red border so you can see the box clearly. In its native state, the box is floated to the left. We're going to use the Advanced CSS Animator to make it move to the right when the box encounters the mouse and move back to its original position when the mouse moves away from the box. **Save** template.html as boxmove.html to keep the original document free of other code.
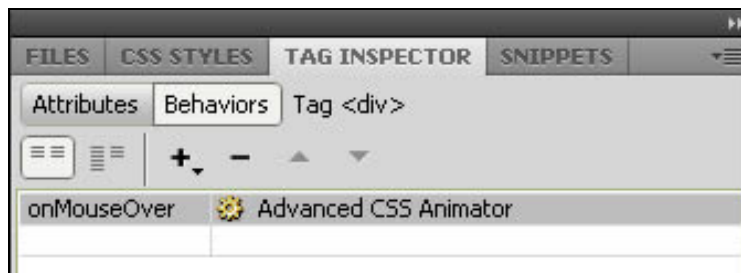
In your new document, select the box div and open the Advanced CSS Animator by **clicking** on the Behaviors Panel and selecting DMXZone/Advanced CSS Animator from the dropdown menu. In the Advanced CSS Animator dialogue box, select the CSS tab and dropdown Advanced CSS Properties. Select the TranslateX option and **click the right arrow button** to move the TranslateX behavior to the box on the right. You will notice that the Animate From column currently says Auto Detect. We want to tell the behavior where the box is sitting right now to establish a starting point. If we put 0 in the From box and then click in the Used Properties box, we can see that Auto Detect is now 0, meaning that box1 is already to the left of the document at the 0 position on the X axis. Now we want to move it so specify a value in the To input box (I used 500 to move the box 500 pixels to the right). Click again in the Used Properties box and make sure that the To value is now what you requested.

This completes the Used Properties dialogue, but you also will notice some options under Animation Properties below the Used Properties box. The first is Duration, or how long you want the transition to take. By default, the transition is 1000 or 1 second. You can add to that to make it move slower or enter a smaller number to get a faster animation. For right now, let's leave that at 1000 or 1 second. The other option under Animation Properties is Easing. If you click on the dropdown list next to Easing, you will see a LOT of built in

transitions already built for you. The default is swing, but you can make it bounce in, bounce out, act like a piece of elastic or many more options as the object moves.   Hours of playing for you! For now, let's leave it at the default, **swing**, and click **OK**.

There is another button for Advanced Animations but we'll leave that for another article. As soon as you click okay, you will see the item in the Tag Inspector. Be sure to check the event that triggers the CSS Animator because you'll want to make sure it's an event you can control or the behavior won't work. I selected OnMouseOver for the event so when I put the mouse on the box, it will perform the animation.

Save your document and **click Live View** in Dreamweaver CS5 to see it happen. Vroom .. isn't that cool?

The Advanced CSS Animator is activated with the onMouseOver Event
in Example 1.

We can make it better though. A one-way animation isn't probably what you are going to use in real life so let's add another animation and trigger that one OnMouseOut so the second animation will be performed when the mouse leaves the box. **Click** on the box or select the div tag in the Document Inspector and open the Advanced CSS Animator again. The CSS Tab is probably already selected so once again, drop down the Advanced CSS Properties and select TranslateX and **click the right arrow** to move the TranslateX behaviour over to the Used Properties dialogue. But this time, select 500 for the From input box and 0 for the To input box. Leave the transition time at 1000 (1 second) and the Easing on swing. Click **OK** and set the Event box in the Tag Inspector to onMouseOut. **Save** the document.

Now when you click Live View in Dreamweaver CS5, the box will move to the right when you put the mouse over the box, but it will also move back to its original position when you take the mouse away. Forward, back. Forward back. Got a headache yet?

Okay, you think. This is very cute, but what can I DO with this? Actually, you can do a lot. Slide a box containing a detailed explanation of a term used in your document and use the onClick event to activate it. Use onMouseOut or a second click to make it move back out of the way. Use this technique to slide a second menu from behind a primary menu and use either onMouseOver or onClick to reveal the menu which slides to the right of the first one. These are a couple of examples, but the possibilities are endless.
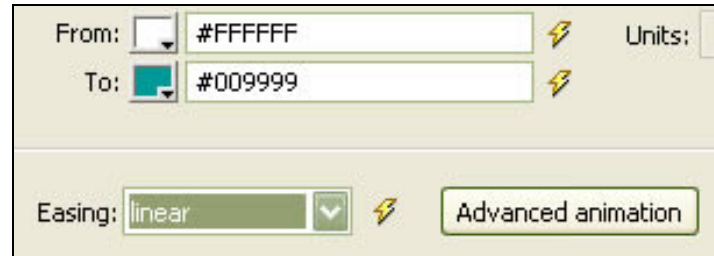
Example 2: Background Changes

Let's go back to the template.html document and save another document with our basic box in it. This time, instead of moving the box, let's change the way it looks. Select the box and open the Advanced CSS Animator. Currently we have a white box with a red border. This isn't very exciting so let's make the box change to blue-green when we mouse over the box. To do that, select the box and open the Advanced CSS Animator. The behavior we want is found on the CSS tab in the Advanced CSS Properties menu and the colors submenu. **Click** on background-color and **click** the right arrow to move it to the Used Properties dialogue box.

The Advanced CSS Animator interface with background-color selected.

We already have a white box on the page so our From color will be white which can be easily selected with the color picker next to the From: input box. We want to change it to a greenish blue, so I have selected the

swatch that matches #009999 as the To: color. This time I want a linear transition, so I will select linear from the Easing dropdown (it's right above the default swing) and click okay. If I wanted to transition slower, I could change the Duration to 2000, for example. If I wanted a very quick transition, I might change the Duration to 300. But we'll leave it at the default 1000 or 1 second for this example. Click **OK**.

The input boxes with our colors selected.

In the Tag Inspector, we need to remember to set the event from the default onFocus to a mouse event so we can see this happening. I have selected onMouseOver and indeed when I check the transition in Live View (or in a browser), the background of the box changes when the mouse is over it. Again, this is okay, but it would be better if it could change back when the mouse is removed, right? Easy to do; just select the box again, select the Advanced CSS Animator from the Behaviors panel and repeat the steps for the transition, but this time, the From color is #009999 and the To color is #FFFFFF or white. I left the easing as linear and the default Duration of 1000. Click **OK** and set the Event to OnMouseOut in the Tag Inspector so the color now changes to green-blue when the mouse is over the box and back to white when the mouse is removed.

Now view your document in Live View. Click on the box and it will change to the greenish blue color. Take your mouse off the box and it will change back to white. See how that works? Are you getting the idea? Let's try one more. This time, we're going to make the box bigger when we click on it. Ready?

Example 3: Watch it Grow!!

Open template.html one more time and this time, save the document as scale.html. Click on the box to select it and open the Advanced CSS Animator. On the CSS tab, go to Advanced CSS Properties and dropdown the CSS Transforms group. Select "ScaleX" and the right arrow to move it to the Used Properties box. Again, you have the same From and To box and this time, you will put **1** in the From box and **2** in the To box in order to get it to scale the width to twice its size. Now we want to add another transform so this time, select "ScaleY" and the right arrow to move it to the Used Properties box. Again, you have the same From and To box and this time, you will put **1** in the From box and **2** in the To box in order to get it to scale the height by the same amount.

Applying the Scale properties to scale the box to twice its size

Specify what you want for the Easing property and the Duration and click **OK**. Back in the Tag Inspector, choose the Event to trigger the animation. I used onMouseOver, but you could use onClick or whatever you like. Check the result in Live View or in your browser. Very nice, but the results of this transition dictate even more the necessity of having a way to get back to the initial state, so select the box one more time and open the Advanced CSS Animator again and choose CSS/Advanced CSS Properties/CSS Transforms and choose Scale X and Scale Y as we did before. This time, select **2** in the From box and **1** in the To box and select the Easing and Duration values you wish to use. Click **OK** and select the inverse trigger event in the Tag Inspector (onMouseOut if you used onMouseOver the first time, etc). Try Live view or your browser to see what happens. Depending on the event you used, you should have the box and its contents grow to twice the size when the mouse is over the box (or the box is clicked) and it should revert to its original size and position when the mouse is off the box.

What Now?

Obviously, there is a lot more you can do with the CSS Animator than I can show you in one article. This time, you have learned the basics of the Advanced CSS Animator. I am very impressed with how easy and how quick it is to produce some pretty cool effects on your web pages. I can't imagine how anyone moving into CSS3 wouldn't want this in their extension arsenal because not only can you quickly produce CSS3 effects, but you have the assurance of cross browser compatibility thanks to the DMXzone's clever use of JavaScript to help out the older browsers. Attached to this article is a zip file with the sample pages used in the examples. Take a look at them in your favorite browser (or Internet Explorer) and see for yourself how you can have CSS3 effects in the browsers of today with the DMXzone Advanced CSS Animator!

In my next article, I'll show you how to use the Advanced CSS Animator in your every day work, whether you design for Fortune 500 corporations or make children's games. Until then, take a look at some more examples of this magnificent tool and get a copy so you can make some cool stuff all your own!

Check out some cool demo's at:

http://www.dmxzone.com/demo/dmxAnimator/effects/popup.html
http://www.dmxzone.com/demo/dmxAnimator/effects/dropin.html
http://www.dmxzone.com/demo/dmxAnimator/effects/zoom.html
http://www.dmxzone.com/demo/dmxAnimator/effects/fadein.html
http://www.dmxzone.com/demo/dmxAnimator/effects/slide_out_menu.html
http://www.dmxzone.com/demo/dmxAnimator/effects/rainbow.html
http://www.dmxzone.com/demo/dmxAnimator/effects/splash.html

http://www.dmxzone.com/demo/dmxAnimator/sideSlideMenu/demo-test.html