

# Expression 3.0b

Plug In for Photoshop™ and Premiere™

© 1992,1993 Jim Bumgardner (jbum@netcom.com, jbum@aol.com)

## What did I get?

---

You got this file, which contains the documentation, the Expression Photoshop™ filter, and, depending on where you got it from, you may have received a sample Quicktime™ movie or picture that was created with Expression and Adobe Premiere™. Enjoy!

Expression is freeware. The author retains the copyright, including the right to alter or sell the program. You may use and copy Expression freely only on a non-commercial basis. You may not alter or sell this program without the author's permission.

## What is it?

---

Simple Explanation: Expression creates groovy abstract art in Adobe Photoshop, and psychedelic movies in Adobe Premiere. It is also a fun way to explore arithmetic.

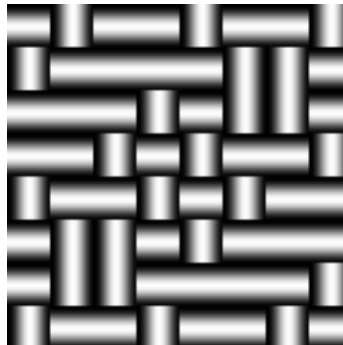


Image created with Expression ("Weave" Preset)

Technical Explanation: Expression is a Photoshop filter that can create images that change over time according to a mathematical formula or *expression* (thus the name).

Possible uses:

- Make psychedelic movies and OpArt.
- Create test patterns galore.
- Make wallpaper.
- Make patterns for textiles / weavings.
- Explore mathematics - zoom into the Mandelbrot set.
- Create animated wipe stencils.
- Create interesting texture and bump maps for 3D graphics.
- Do all kinds of image processing.
- Bring your Quadra to its knees!

---

## How does it work?

---

Expression is a plug-in filter for Adobe Photoshop and Adobe Premiere. It contains a little interpreter that solves an arithmetical expression for each pixel in an image. It changes each pixel to a color that corresponds to the result of the expression. You tell Expression which expressions to solve and how to map the results to a specific color.

When Premiere applies the expression filter to successive frames of a movie, the results can vary according to the frame number, thus creating animation.

---

## What do I need in order to use it?

---

This version of Expression requires a Mac with a 68020 chip and a 68881 fpu. It has been optimized to be fastest on a Quadra, it will run a little slower than it should on an FX or other non-68040 based Mac. Expression works with both Adobe Photoshop and Adobe Premiere.

---

## How do I use it?

---

### With Photoshop...

Place a copy of Expression in the folder where all your other Photoshop filters are. Activate the filter from the Filter > Other sub-menu. The Expression dialog box will appear (see below). Fill in the blanks or select one of the presets. Expression will then modify the selected area accordingly.

### With Premiere...

Expression really comes into its own when used with Adobe Premiere. You can use it to create movies from scratch or to modify existing material. Place a copy of Expression in the folder where your other Premiere plug-ins are kept.

Tip: It is a very good idea to test your expression movie ideas in Photoshop first, to make sure your frames work. Select a small area of an image to process and apply the filter multiple times with Cmd-F. The *t* variable will increase by one each time, showing you how your animation will look.

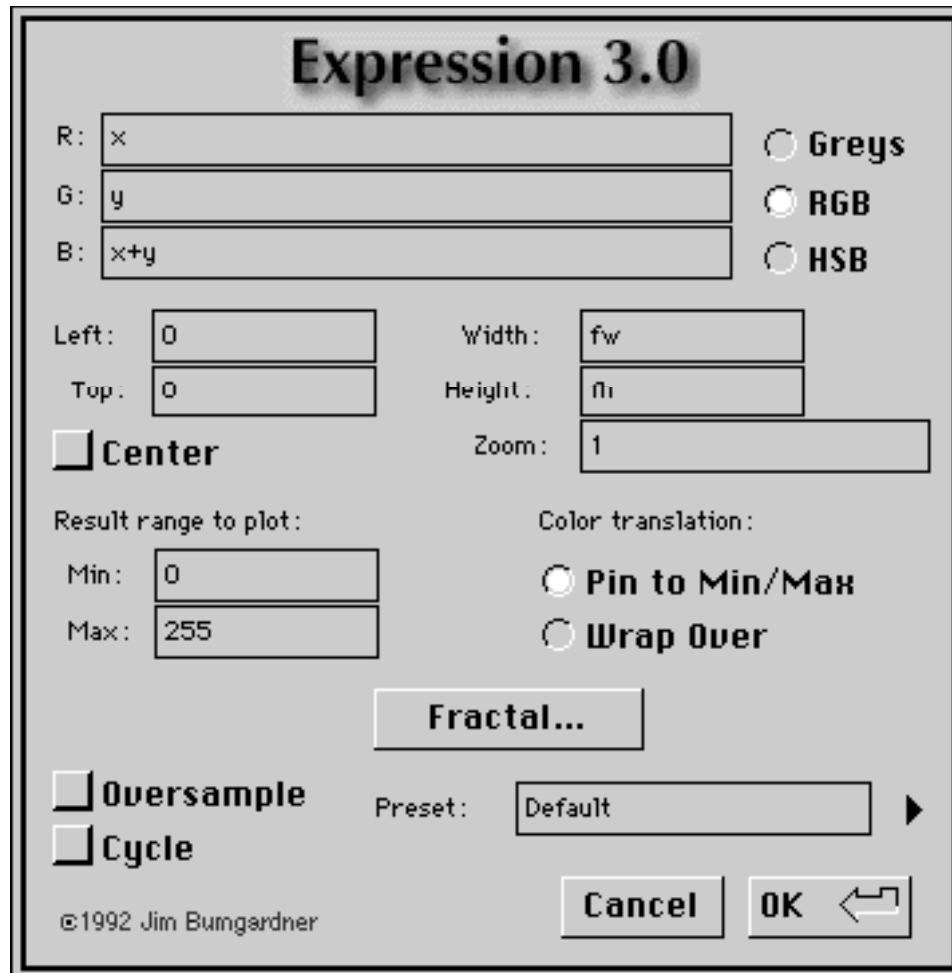
To create a movie from scratch, create a black matte in the Project window using the "Add Matte..." menu command. This will give Expression some raw material to work with. Drag the matte into the construction window in video track A or B and stretch it the duration you would like your movie to be. I usually create movies that are just a few seconds long, because they take a long time to generate. Select the matte track by clicking on it and then select the "Filters..." menu command. Select Expression from the list of filters on the left and hit the "<< Add >>" button. The Expression dialog box will appear (see below). Fill in the blanks or select one of the presets. Later, when you select "Preview" or "Make Movie...", Premiere will apply the Expression filter to each frame. This will significantly add to the length of time it takes to output each frame, depending on the complexity of the expression. Mandelbrot and Julia sets take especially long to generate. It took 14 hours to generate a 2.5 minute fractal zoom movie on a Quadra 900 (The results were well worth it though!)

To modify an existing movie, use the movie as the source material for the filter, instead of using a black matte. By judicious use of the R, G and B variables, you can do a number of colorization effects.

### **Using Expression with other programs...**

Expression can be used with any program that uses Photoshop filters. For example it can be used to make movies with **deBabelizer**, and any other program which can apply Photoshop filters to the frames in a movie. You may find this more convenient than Premiere, since the whole process can be scripted.

## The Expression Dialog Box



### R, G, and B (RGB Mode)

### H, S, and B (HSB Mode)

These fields are where you specify the expression you want to solve for each pixel. A single expression is solved for the red, green and blue components of each pixel. If the Grays radio button is selected, only the first expression is solved, and the result is copied to the green and blue. If the HSB radio button is selected, the three expressions modify hue, saturation and brilliance. The RGB and HSB modes work in Photoshop only when the image is in RGB mode, otherwise, you are operating only on a single channel, and you will get gray levels only.

These expressions can use most C language operators (see the reference guide), floating point constants, and some predefined variables (see the reference guide). For example, if the R field contains the expression  $(r+10)$ , you will lighten the red component of the picture. The expression  $\text{int}(x/20)*40$  will create vertical stripes.

**Grays**  
**RGB**  
**HSB**

When RGB is selected, the three expressions operate on the red, green and blue components of each pixel.

When Grays is selected, Expression only calculates the first of the three expressions. The resultant picture will be gray scale.

When HSB is selected, the three expressions are interpreted as hue, saturation and brilliance (instead of red, green and blue).

If you use Expression in Photoshop on a grayscale image (or any multi-channel image other than RGB), it will behave as if you have the Grays button checked.

**Left, Top, Width, Height**

These expressions are solved once for each frame. They are used to calculate what the x and y variables will contain for each pixel. For example, if left and top are set to (0,0) and the width and height are set to (10000,10000), and you are rendering an image that is 100 pixels across, then for each pixel on the first scanline, x and y will be set to (0,0), (100,0), (200,0) etc.

To make the x and y values correspond to pixel coordinates, set left and top to (0,0) and set width height to (fw, fh).

Idea: You can make an image pan to the right by incorporating t (time) into left and right. By incorporating t (time) into width and height, you can create different zooming effects. For example, the following settings, pan diagonally while zooming out:

left: t*5	width: 160*(1.1**t)
top: t*5	height: 120*(1.1**t)

**Center**

This checkbox causes the image to be centered around the coordinates specified by left and top. Otherwise, left and top will correspond to the top lefthand corner of the image.

**Zoom**

This specifies a zoom factor which is recomputed once per frame. This can be used to create very dramatic effects in QuickTime movies. The zoom is applied to the width and height after they have been recomputed. For example, if the initial width and height are (100,100), and the zoom factor is 1.1, the widths and heights of each frame will be (100,100) (110,110), (121,121), (133.1, 133.1) etc. A zoom factor of 1.0 will result in no zooming. Zoom factors less than 1.0 will zoom in, and zoom factors greater than 1.0 will zoom out. Normally I use Zoom in conjunction with Center to zoom into the center of the picture. Most of the fractal presets perform zooming.

You can compute how fast the movie will zoom in by using the following formula:

$$r = Z^t$$

where  $r$  equals the image ratio - the width or height of the zoomed image compared to the original width and height.

$Z$  is the zoom factor used in the Expression dialog, and

$t$  is the frame number (starting from 0).

Tip: If you want to see where the fractal movies are zooming into from Photoshop, temporarily set the width and height to a small value, such as 0.0004.

### **Result Range to Plot:**

#### **Min and Max**

These fields are used to determine the range of results that are mapped to the intensity of each color component (which can only show 256 discrete values). If Min is 0 and Max is 255, there is a direct 1-1 correspondence. If Max is 1000, then the result will be scaled down so that the range 0-1000 can be displayed as 0-255 levels of intensity.

For example, if you are mapping the expression  $\sin(x)$ , the resultant values will range from -1 to +1. In this case, you should set min to -1 and max to 1 so the sine wave will appear as a fluctuation from black to white.

#### **Pin to Min/Max**

##### **Wrap Over**

If the results of the expression go outside of the min-max range, this setting determines how it is dealt with (much like the AddPin and AddOver operations in Quickdraw). If "Pin to Min/Max" is selected the value is pinned to 0 or 255. If "Wrap Over" is selected, the value wraps over. The wrap over setting can be used to create some interesting moire effects (try "x\*y" with wrap turned on for a nice example).

#### **Fractals...**

This brings up the Fractal Navigation dialog. More on this below.

#### **Cycle**

This checkbox can be used to force a cheesy color cycling effect on each frame. After computing the color as 0-255, Expression will cycle the value by computing  $v=(v+t) \bmod 256$ . You can get more precise control over color cycling by incorporating the  $t$  variable into your R G and B expressions. The Mandel Cycle preset shows an example of this method.

#### **OverSample (new for version 2.2)**

This checkbox causes Expression to antialias the output by averaging the results of four samples for each pixel. This improves the appearance of fractals but causes Expression to take four times longer. *Zzzzzzzz*.

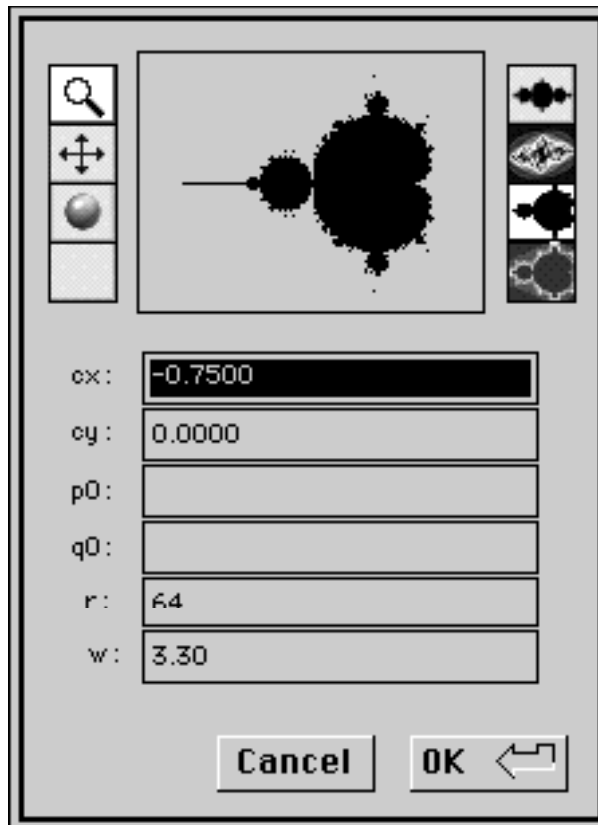
### **Preset**

This popup menu allows you to select (or create) preset expressions. Previously saved popups will appear on the menu (A ResEdit TMPL is included to allow you to add or edit presets from within ResEdit). You can add and delete presets from Expression by entering the name of the Preset you wish to add or delete, and then use the "Add" or "Delete" menu items. These items are disabled if no name has been typed in.

Tip: You can change Expression's default settings by modifying the preset labelled "default".



## The Fractal Navigation Dialog Box



Fractal Movie making has been greatly simplified with Expression 3.0, also the time to render fractals has been sped up. You can specify a zoom-point for a fractal movie using the Fractal Navigation dialog box.

Note: Before hitting the "Fractal" button to go into the Fractal Navigator, select one of the fractal presets from the presets menu or manually enter a fractal expression into the R,G,B fields. You need to load an expression into the R,G,B fields which uses either the MAND() or JULIA() functions. These expressions determine how the fractal will be colored. The fractal navigator only controls the left,top,width,height and recursion fields on the main dialog - it is up to you to provide the R,G,B fields.

## Fractal Navigator Dialog Items

### Zoom Tool:

When this tool is selected, you can zoom into the currently displayed fractal by clicking on the part you want to zoom into. The *cx*, *cy*, and *w* fields will be updated to reflect the new coordinates.

### Pan Tool:

When this tool is selected you can scroll up, down, left and right by clicking on the appropriate area. The *cx* and *cy* fields will be updated to reflect the new coordinates.

### Render Modes:

These tools change the rendering mode - you can render either a Mandelbrot or Julia set. Either set can be rendered black and white or with the equipotential fields in shades of gray.

### *cx*:

This field corresponds to the "left" field in the main dialog box. It is the coordinate of the center of the preview window.

### *cy*:

This field corresponds to the "top" field in the main dialog box. It is the Y coordinate of the center of the preview window.

### *p0*:

### *q0*:

For Julia sets, these coordinates determine the overall shape of the julia set. Values inside the area of the mandelbrot set will produce "connected" shapes. Values outside the area of the mandelbrot set will produce dusts. The most interesting looking shapes result when *p0* and *q0* are close to the border of the mandelbrot set.

Tip: You can create shape changing julia set movies by animating the *p0* and *q0* values using *t* (time).

### *r*:

This expression determines how much **recursion** is performed in calculating fractals. It is recomputed once per frame. If the result of the expression is less than 64, 64 will be used. Most of the fractal presets use an expression which causes recursion to deepen as you zoom in. For some fractals, you may want to lessen or deepen the recursion.

### *w*:

This expression gets copied to the width and height fields when you hit the OK button. It reflects how far you are "zoomed in" to the fractal. To create a zooming-in movie, zoom in as far as you like here, then reset the width to 4 before computing each frame.

---

**Expression Syntax**

---

Expressions look pretty much like 'C' expressions with the exception of the **\*\*** operator and the **?** and **π** variables. Look at the presets to see some examples.

**Operators**

The following operators are available, ordered by precedence:

Note: New Operators to version 3.0 are in bold.

<u>Operator</u>	<u>Precedence</u>	<u>Meaning</u>
!	1	Logical NOT (unary)
~	1	Bitwise NOT (unary)
**	2	Raise to power
*	3	Multiplication
/	4	Division
%	5	Modulo
+	6	Addition
-	6	Subtraction
&	7	Bitwise AND
	7	Bitwise OR
^	7	Bitwise XOR
= or ==	8	Equality
!= or <>	9	Inequality
<	9	Less Than
<=	9	Less Than or Equal
>	9	Greater Than
>=	9	Greater Than or Equal
<b>&amp;&amp;</b>	<b>10</b>	<b>Logical AND</b>
<b>  </b>	<b>10</b>	<b>Logical OR</b>

**Variables**

The following variables are available:

Note: New variables to version 3.0 are in bold.

x	Expression horizontal Coordinate	
y	Expression vertical Coordinate	
w	Expression image width	
h	Expression image height	
px	Horizontal pixel coordinate	
py	Vertical pixel coordinate	
pw	Image width in pixels	
ph	Image height in pixels	
fw	Filter rectangle width (corresponds to photoshop selection rectangle)	
fh	Filter rectangle height (corresponds to photoshop selection rectangle)	height)
<b>cx</b>	<b>The x coordinate of the center of the image.</b>	
<b>cy</b>	<b>The y coordinate of the center of the image.</b>	
<b>d</b>	<b>Distance from the center of the image</b>	
<b>a</b>	<b>Angle of the vector emanating from the center of the image, expressed in radians (0-2<math>\pi</math>).</b>	
r	Input Red Component (0-255)	
g	Input Green Component (0-255)	
b	Input Blue Component (0-255)	
<b>l</b>	<b>Input Luminance Component (HSV) 0-255</b>	
<b>c</b>	<b>Input Hue component (HSV) 0-255</b>	
<b>s</b>	<b>Input Saturation component (HSV) 0-255</b>	
?	Random Number $r$ where $0 \leq r < 1$	
$\pi$	Pi (3.14159)	
t	Time (frame number)	

In Photoshop, the  $t$  value will increase by one each time you invoke the filter using Cmd-F. In Premiere, the  $t$  value will increase by one for each frame that is generated. To reset  $t$  to zero, tweak the parameters by re-initializing the filter.

New for version 3.0: You can now specify  $r, g, b, l, c, s$  as offsets from the current position, for example  $r[+1, -3]$ . This allows you to do image processing tasks which require information from neighboring pixels. Because of slowness of processing large images in Photoshop, there is currently a limit on Y offsets - you can't go more than 4 in either direction.

**Functions**

The following functions are available:

<b>int(n)</b>	<b>cos(n)</b>	<b>cosh(n)</b>
<b>sqrt(n)</b>	<b>sin(n)</b>	<b>sinh(n)</b>
<b>log(n)</b>	<b>tan(n)</b>	<b>tanh(n)</b>
<b>log10(n)</b>	<b>acos(n)</b>	
<b>fabs(n)</b>	<b>asin(n)</b>	
<b>exp(n)</b>	<b>atan(n)</b>	

These functions correspond to their standard C equivalents. If you aren't familiar with C, most of these are available on most spreadsheets.

**sin2(n), cos2(n), sin3(n), cos3(n)**

These functions compute sine squared ( $\sin(x) * \sin(x)$ ), cosine squared, sine cubed ( $\sin(n) * \sin(n) * \sin(n)$ ) and cosine cubed, respectively. The cubed functions work nicely for shading fractals, since they provide more gradation in the mid range, where the eye is more sensitive. See Mandel Color 2 for a nice example.

**mand() - modified for version 3.0**

This function returns an integer in the range 0-n, where n is the maximum number of iterations (see below). If  $\text{mand}(x,y)$  returns 0, the coordinates x and y are considered to be inside the Mandelbrot set, otherwise the number represents the number of iterations it took before the complex expression  $z = z^2 + c$  went outside the bounds of 2, and can be thought of as the "distance" to the Mandelbrot set. Using this number to color the pixels in the image results in some very beautiful images.

The maximum number of iterations is recomputed for each frame using the expression in the "Recurse" field. It is typically computed as

$$\text{pixelwidth} * -\log_{10}(\text{expression width})$$

In other words, the more you zoom in, the deeper the recursion is, and the longer it takes to compute each frame.

If the result of this expression is less than 64, than max iterations is forced to 64.

An optimization has been added that allows you to use  $\text{MAND}(x,y)$  multiple times in the R, G and B expressions without paying a performance penalty. Basically, the program remembers the arguments and result of the last call, so that if MAND is called twice with the same arguments, it can return the same value without recomputing it.

For version 3.0, this function has been sped up, and you don't have to pass in X and Y coordinates - it automatically uses the left,top values in the dialog.

**man3()**

This function is identical in features to the mand() function, but uses the formula  $z^3 + c$ . This function is more computationally expensive than the mand() function, and produces images that are "busier".

**manc() or mcpm() - New for version 3.0**

This function is identical in features to the mand() function, but produces continuous instead of discrete values for the equipotential curves outside the mandelbrot set.

**julia()**

This function is identical in features to the mand() function, but computes the result for a julia set using the initial values of p and q.

The mand(), man3() and julia() functions are extremely computationally expensive, because they may do thousands of iterations to compute the color for each pixel. For version 2.2, the mand and julia functions have been optimized to provide about a 37% improvement on a Quadra 950.

For version 3.0, this function has been sped up, and you don't have to pass in X and Y coordinates - it automatically uses the cx,cy,p0,q0 values in the dialog.

## New Functions

### **Fib(n)**

Returns the nth number in the fibonacci series.

### **Prime(n)**

Returns 1 if n is a prime number, 0 if not.

### **Noise(x,y)**

Returns a random number which is unique for each x,y pair. If x and y are non-integers, the value will be interpolated. Returned value is in the range 0 to 1.

### **Turb(x,y)**

Similar to Noise() but produces a 1/f fractal noise by summing Noise() values at different scales. Returned value is in the range -1 to 1 (so you can use it to cause natural looking fluctuations in a sine wave). You can use the Turb function to create realistic looking clouds, rock and wood textures.

### **Cond(condition, aresult, bresult)**

If condition is true, return aresult, else return bresult. This is similar to the ?: operator in the C language. See the Solarize and Weave presets for examples.

### **Angle(x,y)**

Returns the angle of the vector emanating from the center of the image and passing through x and y. Returned values range from 0 to  $2\pi$ .

### **Dist(x,y)**

Returns the distance from x,y to the center of the image.

---

## Legal Mumbo-Jumbo

---

Expression is freeware. The author retains the copyright, including the right to alter or sell the program. You may use and copy Expression freely only on a non-commercial basis. You may not alter or sell this program without the author's permission. THIS INCLUDES "SHAREWARE" COLLECTIONS - PLEASE CONTACT THE AUTHOR IF YOU WISH TO INCLUDE EXPRESSION IN A COLLECTION OF SHAREWARE / PD SOFTWARE WHICH IS BEING SOLD.

Expression is being distributed free to the public by the author over AppleLink, America Online and the Internet. If you would like to upload Expression to another public network, please contact the author at [wmm.tech@applelink.apple.com](mailto:wmm.tech@applelink.apple.com) (Applelink: WNM.TECH, AOL: JBum).

Photoshop and Premiere are trademarks of Adobe Systems Incorporated. Buy your copies now!

QuickTime is a trademark of Apple Computer, Inc.

## **Revision History**

---

**Note:** If you are upgrading from an older (pre 3.0) version of Expression, don't re-use the filter settings that Premiere saves in your Project file. To re-use an old premiere project, remove the filtered mattes and add new ones. Expression will crash if Premiere attempts to pass it the saved settings from an older version of Expression.

### **9/7/93 - Version 3.0b**

This version contains some bug fixes that prevent Expression from crashing in Premiere and NIH Image.

Another bug was fixed that was affecting the presets which modify an existing image, such as solarize and invert (the routines that sample pixels weren't working properly).

### **3/27/93 - Version 3.0**

Added fractal navigation dialog.

Changed the appearance of the dialogs a bit.

Made arguments to Man, Man3 and Julia functions implied, rather than explicitly passed in.

Added MANC (alias MCPM) continous potential method mandelbrot set function.

Sped up floating point to integer conversions on the Quadra. This makes some filters twice as fast on the Quadra, but it does slow down the fx and other 68030,68020 machines - Sorry! Perhaps in the future I will ship two versions.

Added NOISE(x,y) function.

Added TURB(x,y) function.

Added COND(cond,a,b) function. Simplified some presets, (such as Solarize) using COND function.

Added ANGLE(x,y) and DIST(x,y) functions.

Added FIB() and PRIME() for fun.

Added D and A variables - polar coordinates.

Added cx and cy variables - center of screen coords.

Added c (color) and s (saturation) variables.

Switched to a slower, more accurate luminence calculation for l.

Added Logical AND and Logical OR operators (&& and ||).

Added offset capabilities to r,g,b,c,s,l. For example, to see the color of a neighboring pixel, you can say r[+1,-2]



Made equality (=) precedence higher than other comparison ops.

Fixed parser to treat unary + properly. As in  $r[+1,-1]$

Fixed bug in which Expression wouldn't run in some third party products because it was calling the Photoshop HostProc.

Fixed bug in which certain expressions, such as  $X*Y+SIN(X)$  didn't compute properly because parse got confused about precedence.

Fixed bug in which  $2*\pi$  in width field didn't get loaded in properly. This probably caused other wierd memory corruptions.

Added new presets to illustrate new features.

### **10/28/92 - Version 2.2**

The mandelbrot and julia set functions have been optimized for a 37% speed improvement (this was timed on a Quadra 950).

Non-fractal expressions have been optimized to be 4-15% faster depending on complexity.

Added Oversampling option, which averages four samples for each pixel. This makes Expression take 4 times longer than usual, but produces higher quality images for continuous expressions.

Added Recurse setting, to allow you to make fractal recursion deeper or shallower.

More presets have been added, including posterize, solarize, invert, harden, soften and more fractals.

Added Sin2 (sine squared), Cos2, Sin3 (sine cubed) and Cos3 functions. Sine cubed is useful for coloring fractals, since it provides more gradations in the middle part of the luminence spectrum, where the eye is more sensitive. See Mandel Color 2 for a nice example.

You can now use the constant  $\pi$  (option-p).

Fixed a bug in which < and > were reversed.

Added <=, >=, !=, <> and == operators.

**9/21/92 Version 2.1**

Now works correctly with programs that use an Alpha Channel (specifically Premiere 2.0). A black alpha channel is output, if necessary.

Added zero-check on modulo operator to prevent divide by zero errors. If an expression resolves to  $x \% 0$ , the result is zero.

Replaced Grayscale check box with Gray/RGB/HSB Modes.

Changed random number to be 0 - 1 instead of -32k - +32k. This range is easier to use.

Added more presets, including "default" preset, which you can modify to change the default settings.

Tweaked dialog items a bit for better editing of longer numbers.

Increased compiled expression size to 512b.

Added MAN3() function for a variant on the mandelbrot set ( $z^3 + c$ )

**5/28/92 Version 2.0  
(Major changes since version 1.0)**

Expression now includes functions for rendering Julia Sets and the Mandelbrot set. These have been used to generate some beautiful QuickTime movies of colorful fractal zooms.

Previously, Expression always computed x and y as being the pixel coordinates. Now, x and y scan a virtual space with dimensions you specify. You indicate the coordinates of the top,left corner, and the width and height.

A zoom factor expression allows you to modify the width and height of each frame, which makes a zoom effect possible.

The center option makes the top,left values correspond to the center of the image. This simplifies specifying a point in the Mandelbrot set to zoom into.

Expression has some new presets to show off its new fractal zooming capabilities. It takes about six hours to generate a 40 second fractal zoom movie on a Quadra 900.

**Minor changes since version 1.0**

Added variables px,py,pw,ph,fw,fh.

Expression now supports the unary negation operator, e.g.  $2 * -(5*6)$ . Previously, only subtraction was supported e.g.  $2*(0-(5*6))$

Fixed a bug in number parsing to allow .123 with no leading zero.

Added grayscale option to speed up calculations on grayscale images. This tells Expression that it doesn't need to recompute green and blue, but can copy the red result into the green and blue pixels.

Expression now works correctly with grayscale images in Photoshop.

Applelink:	WNM.TECH
America Online:	JBum
Internet	jbum@netcom.com

Happy Expressing!