

MartinView

MartinView is a port of an X-Windows program called xmartin written by Ed Kubaitis at UIUC which I have included with its man page as a tribute and source. I saw this running on a local machine and said "that's cool, that should be on a NeXT", and where else but on BackSpace.

Why the name MartinView you ask? Well, taken from the xmartin source code:

[The] Hopalong [algorithm] was attributed to Barry Martin of Aston University (Birmingham, England) by A. K. Dewdney in the 9/86 Scientific American. The Ranf portable random number generator is based on work by D. H. Lehmer, Knuth, David Sachs(Fermilab) and Curt Canada(NCSA).

Installation Instructions

This directory contains the source for MartinView, a BackSpace screen-saver module that works with BackSpace v3.0. This module needs to be installed so that BackSpace can find it, but by default, make will not install it. The following will install the module in your **~/Library/BackSpaceViews** directory:

```
make install
```

BackSpace will actually find the module if it is in one of the following places:

```
~/Library/BackSpaceViews,  
its "app wrapper" folder  
/LocalLibrary/BackSpaceViews
```

You can specify an alternate installation directory like this:

```
make install INSTALLDIR=/LocalLibrary/BackSpaceViews
```

Inspector Panel Options

The first thing to notice is that I added so many options, that I needed to use a separate panel for the Inspector instead of the default box view. Thanks for this option Sam!
I wanted to be as true to the original xmartin source and offer as many features as possible.

Many features have Auto switch buttons which determine whether their values are determined automatically (in some cases meaning randomly) or whether they are set to a specific user defined value.

The **Function** pop-up button specifies which of the four algorithms is in use. With Auto on, one of the four functions is chosen randomly based on percentages specified in MartinView.m.

The **Points** box determines the **maximum number of in-range points** to use (those that fall within the coordinates of the view and the **maximum total points** (includes all points calculated). Both are necessary to insure proper completion of one set because given the random nature of the parameters you sometimes get results outside the view's range. With Auto turned on, the maximum values are calculated as a percentage of the size of the view.

The **Hopalong Parameters** are specified by three floating point values and six boolean values basically determining signs. See the `-oneStep` method in MartinView.m for more specifics on what these values do for each algorithm. With Auto turned on, their values are randomly determined.

You can specify a **seed perturbation** value and how often this perturbation should be applied. With Auto turned on these values are determined randomly. A **magnification factor** can be

specified for the entire set and the Auto option always returns a value of 4 for the martin2 function, and 1 for the other functions. An overall **displacement** vector can be specified in the default coordinate system of the view as well.

The **flush buffer interval** specifies the number of pixels to compute before flushing them to the screen. The default flush buffer size is 400 pixels in one call to NXRectFillList, with a maximum value of 900. I think any amount higher than 900 or so may cause a slow down with respect to time/pixel. So the only essential usefulness of this option is to slow down the drawing of the fractal by setting the interval to a low number. The original xmartin code allowed for the creation of the image off-screen and a complete flush to the screen at one time, but that is no fun, and I think this is a good example to show those groups of individuals who say "The NeXT?, but isn't it slow".

Color change interval specifies the interval within which the color should change. With Auto on, the interval is a certain percentage of the view size.

For those B&W users out there, there is a pop-up button that specifies whether **RGB** colors or shades of **Gray** should be used. With **RandomColor** on, a completely random color or shade of gray is used each time, otherwise a source coded array of 14 colors is used whose order is fixed.

Given the randomness of the parameters, every so often you get a picture that you know just isn't going to turn out to be something spectacular or fill the whole screen. Press the **New One** button and get on with your life! Some of the results are so cool I wanted to do a screen grab, so I added a **Pause** button so that you would have all the time in the world to screen grab what you want.

Finally, if you want to show this off to your friends and/or want to use MartinView as a screen-saver and want spectacular results every time, the **Remember** button and **use file** switch button are for you. The **Remember** button appends all of the current parameter values and Function to a file in your home directory called `~/.martinView`. Hit the Remember button while one of your favorite parameter sets is running, and when the **use file** switch is on, MartinView will cycle through all of the sets you have stored in the file `~/.martinView`. The numbers are stored in the file in a straightforward way and each set begins with the line "MartinView" so it is easy to edit the file and remove any sets that are getting old.

I have included the file `".martinView"` containing some nice fractal parameter settings I have found. Just copy this file directly into your home directory and choose **use file** to see these settings. Besides storing complete sets in a file, MartinView also stores five of the options in the Defaults database for BackSpace.app. Yes, you could store all of the values, but I am not sure

of the usefulness of storing them all. The following parameters are stored in the Defaults database:

Parameters stored in Defaults database:

The Function and whether it is randomly chosen or not (**MartinFunction**)

Maximum total points (**MartinMaxTot**)

Maximum in-range points (**MartinMaxIn**)

The color change interval (**MartinColorInt**)

Whether to use random color choices (**MartinRandomCol**)

Whether to run in Color or B&W (**MartinColorMode**)

Whether to use the file ~/.martinView (**MartinUseFile**)

Thanks to...

Ed Kubaitis, at UIUC for the X-Windows code on xmartin from which my port was made.

and of course, **Sam Streeper** for BackSpace and some useful suggestions for improving this module.

Any suggestions, comments or extremely interesting sets of parameters for MartinView would be greatly appreciated.

Jeffrey Adams
Email: jeffa@wri.com
NeXTmail accepted