

Putting Things Together

257522_PixelRule.tiff ↗

Now that you know what it uses, here's how to use it. For examples of any of these steps, see the source to *GraphicsWorkshop*.

1. Initialize the `ControlLoader` object by telling it to load an instance of "Bitmap" tools. This will cause the program to search for converters, as well as link in the `ImageControl` object for later instantiations. It should only be called once.
2. When ready, create an open panel, but to run it, call the *runOpenPanel* method in the `NXBitmapImageRepControl` object. This will show the open panel for all file types found in Step 1. It will return the same values as the normal Open Panel found in the NeXTStep Reference, Volume 1.
3. Once you have from 1 to n filenames, pass these names, one at a time, to the *openAndReturnImage* method. This will return an `NXBitmapImageRep` for each image.
4. Manipulate the image any way you'd like.
5. Now that you're ready to save the image, create a save panel, but run it via the *runSavePanel* method. This will bring up a panel, with the images current type linked into the code. Should the user select a new type, that converter will be linked, it's custom view, if

any, displayed, and it's file extension added to the bitmaps filename. If it returns YES, then you can move onto step 6, otherwise, the user canceled the save.

- 5a. The programmer can also get parameter information from the converter object itself at this point. However, this is discouraged in applications using the NeXTStep interface, as every converter can have it's "own" set of parameters and using the set and get methods only ties the program down to a few or one specific converter.
6. Call the *saveImage* method to save the bitmap. The correct converter was already linked, so make sure you don't call the *handleLink* message before the *saveImage* message as this will eradicate the user's customization selections.
7. That's it.