

Q: My application is a simple paint program. The user opens a TIFF image, then scribbles into it, and finally saves the new image as a TIFF file. However, the changes made by the user aren't saved into the TIFF file—it contains the original image. Why?

A: This occurs if you open the TIFF file like this:

```
image = [[NXImage alloc] initWithFile:fileName];
```

NXImage will have two representations—the file, and the cache. NXImage will treat the cache as a transitory image, and the file as its "best representation." The cache is the off-screen window to which the user's scribbles are drawn. When asked to write out the image, NXImage writes out its best representation of the image—which is the actual TIFF file residing on disk—thus ignoring completely the changes made to the image. To get around this you must fake out NXImage by forcing the cache to be the best representation of the image.

The following code snippet illustrates what you must do:

```
/* When the user opens the image */
rep = [[NXBitmapImageRep alloc] initWithFile:fileName];
[rep getSize:&imageSize];

image = [[NXImage alloc] initWithSize:&imageSize];

if ([image useCacheWithDepth:d] && [image lockFocus]) {
    [rep draw];
    [image unlockFocus];
}
[rep free];
```

This code sample initialized an `NXBitmapImageRep` from the file containing the opened image. The `NXImage` is initialized from this representation. Now the `NXImage` does not have a file which can serve as its best representation. It only has the cache. Thus when you tell `NXImage` to **writeTIFF:** the cache with all of the user's scribbles is written out properly.

QA786

Valid for 1.0, 2.0, 3.0