canBeCompressedUsing:
getCompression:andFactor:
setCompression:andFactor:

Checking unpacked data handling

+ setUnpackedImageDataAcceptable:
+ isUnpackedImageDataAcceptable

Archiving

canBeCompressedUsing:

(BOOL)canBeCompressedUsing:(int)compression

Tests whether the receiver can be compressed by compression type. Compression types are define
method returns YES if the receiver's data matches compression for example, if compression is
NX_TIFF_COMPRESSION_CCITTFAX3, then the data must be one bit-per-sample and one sam
NO if the data doesn't match compression or if compression is unsupported.

(void)getCompression:(int *)compression andFactor:(float *)factor

Returns by reference the receiver's compression type and compression factor. Use this method to
compression type for the source image data. compression represents the compression type used or
corresponds to one of the values returned by the class method getTIFFCompressionTypes:count:.
between 0.0 and 255.0, with 0.0 representing no compression.

setCompression:andFactor:

(void)setCompression:(int)compression andFactor:(float)factor

Sets the receiver's compression type and compression factor. compression is one of the supported
listed in the getTiffCompressionTypes:count: class method description. factor is a compression fa
0 (no compression) and 255.0 (maximum compression).

When an NXBitmapImageRep is created, the instance stores the compression type and factor for th
the data is subsequently saved, writeTIFF: tries to use the stored compression type and factor. Use
the compression type and factor.

getCompression:andFactor:

getCompression:andFactor:, setCompression:andFactor:, writeTIFF:usingCompression:andFacto

255.0, with higher values yielding greater compression but also greater information loss. The compression schemes are discussed briefly in the class description, above.