PNG-GROUP-DRAFT
MNG
Expires: 30 Mar 1999

PNG Development Group
mpng-list@dworkin.wustl.edu
30 Sep 1998

# JNG (JPEG Network Graphics) Format Version 19980930

## file: draft-jng-19980930.txt

### Extracted from Draft 50, file: draft-mng-19980930.txt

### MNG (Multiple-image Network Graphics) Format Version 19980930

ftp://swrinde.nde.swri.edu/pub/mng/documents/.

## Status of this Memo

This document is an informal draft of the PNG development group.

It is a **proposal**, and the format is subject to change.

Comments on this document can be sent to the PNG specification maintainers at one of the following addresses:

- mpng-list@dworkin.wustl.edu
- png-group@w3c.org
- png-info@uunet.uu.net

Distribution of this memo is unlimited.

At present, the latest version of this document is available on the World Wide Web from

    ftp://swrinde.nde.swri.edu/pub/mng/documents/.

In the case of any discrepancy between this extract and the full MNG specification, the full MNG specification shall take precedence.

## Changes from forty-ninth MNG draft (draft-mng-19980919)

- Various formatting changes to make it look more like the PNG spec.

## Abstract

This document presents the format of a JNG (JPEG Network Graphics) datastream. JNG is a lossy single-image member of the PNG (Portable Network Graphics) format family. It encapsulates a JPEG datastream in PNG-style chunks, along with an optional alpha channel and ancillary chunks that carry color-space information and comments. While JNG is primarily intended as a subformat of the MNG (Multiple-image Network Graphics) format, standalone JNG files are also possible.

# 1 The JPEG Network Graphics (JNG) Format

JNG (JPEG Network Graphics) is the lossy sub-format for MNG objects.

Note: This specification depends on the PNG Portable Network Graphics specification [PNG]. The PNG specification is available at the PNG home page,

```
http://www.cdrom.com/pub/png/
```

A JNG datastream consists of a header chunk (JHDR), JDAT chunks that contain a complete JPEG datastream, and optionally, IDAT chunks that contain a PNG-encoded grayscale image that is to be used as an alpha mask. Such a mask must have the same dimensions as the image itself. The JDAT and IDAT chunks can be interleaved. Some of the PNG ancillary chunks are also recognized in JNG datastreams.

While JNG is primarily intended for use as a sub-format within MNG, a single-image JNG datastream can be written in a standalone file. If so, the first eight bytes of a JNG datastream are

```
139 80 78 74 13 10 26 10
```

(decimal) which is similar to the PNG signature with “\213 J N G” instead of “\211 P N G” in bytes 0-3.

JNG is pronounced “Jing.”

## 1.1 JNG critical chunks

This section specifies the critical chunks that are defined in the JNG format.

### 1.1.1 JHDR JNG header

The format of the JHDR chunk introduces a JNG datastream. It contains:

```
Width:       4 bytes (unsigned integer, range 0..65535).
Height:      4 bytes (unsigned integer, range 0..65535).
Color type:  1 byte
               8: Gray (Y).
              10: Color (YCbCr).
              12: Gray-alpha (Y-alpha).
              14: Color-alpha (YCbCr-alpha).
JDAT sample
  depth:     1 byte
               8: 8-bit samples and quantization tables.
              12: 12-bit samples and quantization
                  tables.
              20: 12-bit image followed by an eight-bit
                  image.
JDAT compression
  method:    1 byte
               8: ISO-10918-1 Huffman-coded baseline JPEG.
JDAT interlace
  method:    1 byte.
               0: Sequential JPEG, single scan.
               8: Progressive JPEG.
IDAT sample
  depth:     1 byte.
               0, 1, 2, 4, 8, or 16.
IDAT compression
  method:    1 byte.
               0: Zlib DEFLATE.
IDAT filter
  method:    1 byte.
               0: Adaptive (see PNG spec).
IDAT interlace
  method:    1 byte.
               0: Not interlaced.
```

The width, height, JDAT_sample_depth, JDAT_compression_method, and JDAT_interlace_method fields are redundant because equivalent information is also embedded in the JDAT datastream. They appear in the JHDR chunk for convenience. Their values must be identical to their equivalents embedded in the JDAT chunk. We use four bytes in the width and height fields for similarity to MNG and PNG, and to leave room for future expansion, even though two bytes would have been sufficient.

When the `color_type` is 8 or 10 (no alpha channel), the last four bytes, which describe the IDAT data, must be set to zero. The `IDAT_sample_depth` must be nonzero when the alpha channel is present.

### 1.1.2 JDAT JNG image data

A JNG datastream must contain one or more JDAT chunks, whose data, when concatenated, forms a complete JNG baseline JPEG datastream.

A JNG baseline JPEG is a baseline JPEG as defined by JPEG Part 1 (ISO IS 10918-1), using only JFIF-compatible component interpretations, with a few additional restrictions that reflect limitations of many existing JPEG implementations.

- Baseline JPEG restrictions

  A baseline JPEG according to Part 1 is DCT-based (lossy) sequential JPEG, using Huffman entropy encoding, with the following further restrictions:

  - Eight-bit and twelve-bit sample precision are both allowed, but this specification only requires minimal JNG decoders to process datastreams with eight-bit sample precision. When `JDAT_sample_depth=20`, the datastream contains a twelve-bit image followed by an eight-bit image, with a JSEP chunk between them. Viewers must display only one of them; the eight-bit image is only for use by viewers that cannot decode twelve-bit images.
  - Quantization table precision must be the same as the sample precision.
  - Huffman code tables can have table numbers 0 and 1 only.

  The SOF marker type for baseline JPEG is SOF0.

  JDAT datastreams must always follow "interchange JPEG" rules: all necessary quantization and Huffman tables must be included in the datastream, no tables can be omitted.

- JFIF-compatible restrictions

  The image data is always stored left-to-right, top-to-bottom, ie, only the default SPIFF orientation is permissible.

  The encoded data shall have one of the two colorspace interpretations allowed by the JFIF specification:

  - Grayscale: a single component representing luminance, ranging from 0 for black to 255 for white (or 0 to 4095 when dealing with 12-bit data). This component shall have JPEG component identifier 1.
  - YCbCr: three components representing luminance, chroma blue, and chroma red, in that order. The components shall be assigned JPEG component identifiers 1, 2, 3 respectively. YCbCr is defined as a linear transformation from RGB color space:

    ```
    Y = Luma_red*R + Luma_green*G + Luma_blue*B
    Cb = (B - Y) / (2 - 2*Luma_blue) + Half_scale
    Cr = (R - Y) / (2 - 2*Luma_red)  + Half_scale
    ```

    By convention, the luminance coefficients are always those defined by CCIR Recommendation 601-1:

    ```
    Luma_red   = 0.299
    Luma_green = 0.587
    Luma_blue  = 0.114
    ```

The constant Half_scale is 128 when dealing with 8-bit data, 2048 for 12-bit data. With these equations, Y, Cb, and Cr all have the same range as R, G, and B: 0 to 255 for 8-bit data, 0 to 4095 for 12-bit data.

The JFIF convention for YCbCr differs from typical digital television practice in that no headroom/footroom is reserved: the coefficient values range over the full available 8 or 12 bits.

Intercomponent sample alignment shall be such that the first (upper leftmost) samples of each component share a common upper left corner position. This again differs from common digital TV practice, in which the first samples share a common center position. The JFIF convention is simpler to visualize: subsampled chroma samples always cover an integral number of luminance sample positions, whereas with co-centered alignment, chroma samples only partially overlap some luminance samples.

- Additional JNG restrictions

  JNG imposes three additional restrictions not found in the text of either JPEG Part 1 or the JFIF specification:

  - The sampling factors for YCbCr images must be one of these sets:
    - 1h1v,1h1v,1h1v (also called 4:4:4 or 1x1 sampling)
    - 2h1v,1h1v,1h1v (also called 4:2:2 or 2x1 sampling)
    - 2h2v,1h1v,1h1v (also called 4:2:0 or 2x2 sampling)
    - 1h2v,1h1v,1h1v (also called 2:4:2 or 1x2 sampling)

    In other words, the chroma components may be downsampled 2:1 or 1:2 horizontally or vertically relative to luminancey, or they may be left full size. These four sampling ratios are the only ones supported by a wide spectrum of implementations (1x2 is relatively new, and is usually the result of a lossless rotation of a 2x1 sampling).

    For grayscale images, the sampling factors are irrelevant according to a strict reading of JPEG Part 1. Hence decoder authors should accept any sampling factors for grayscale. However, we recommend that encoders always emit sampling factors 1h1v for grayscale, since some decoders have been observed to malfunction when presented with other sampling factors.
  - There must be only one scan in an image: that is, YCbCr images must be fully interleaved. There is little advantage to be gained by encoding a baseline image in multiple scans, and many baseline decoders do not support multiple scans at all.
  - The DNL (Define Number of Lines) marker is prohibited. The image height must always be specified accurately in the SOFn marker and in the JHDR chunk.

- Recommended progressive JPEG subset

  For JNG progressive JPEG datastreams, the JPEG process is progressive Huffman coding (SOF marker type SOF2) rather than baseline (SOF0). All JNG-compliant decoders must support full progression, including both spectral-selection and successive-approximation modes, with any sequence of scan progression parameters allowed by the JPEG Part 1 standard.

  Otherwise, all the restrictions listed above apply, except these:

- Multiple-scan support is obviously required for progressive JPEG.
- Huffman table numbers up to 3 (the full JPEG limit) may be used, since the baseline two-table limit is unlikely to be needed by any decoder that can handle progressive JPEG.

We require full progression support since relatively little code savings can be achieved by subsetting the JPEG progression features. In particular, successive approximation offers significant gains in the visual quality of early scans. Omitting successive-approximation support from a decoder does not save nearly enough code to justify restricting JNG progressive encoders to spectral selection only.

No particular progressive scan sequence is specified or recommended by this specification. Not enough experience has been gained with progressive JPEG to warrant making such a recommendation. To allow for future experimentation with scan sequences, decoders are expected to handle any JPEG-legal sequence. Again, the code savings that might be had by making restrictive assumptions are too small to justify a limitation.

When the JSEP chunk is present, both images must be progressive if one of them is progressive.

JDAT chunks are like PNG IDAT chunks in that there may be multiple JDAT chunks, the data from which are concatenated to form a single datastream that can be sent to the decompressor. No chunks are permitted among the sequence of JDAT chunks, except for interleaved IDAT chunks. The ordering requirements of other ancillary chunks are the same with respect to JDAT as they are in PNG with respect to the IDAT chunk.

### 1.1.3 IDAT JNG alpha data

This chunk is exactly like the IDAT chunk in a PNG grayscale image, except that it is interpreted as an alpha mask to be applied to the image data from the JDAT chunks. The alpha channel, if present, can have sample depths 1, 2, 4, 8, or 16. The IDAT chunks can be interleaved with the JDAT chunks. No other chunk type can appear among the sequence of IDAT and JDAT chunks. No other chunk type can appear between the sequences of IDAT and JDAT chunks when they are not interleaved. The samples in the IDAT must be presented in noninterlaced order, left to right, top to bottom. As in PNG, zero means fully transparent and `2^IDAT_sample_depth-1` means fully opaque.

The IDAT chunks must precede the JSEP chunk, if the JSEP chunk is present. Minimal viewers that skip the 12-bit JDAT chunks must read the IDAT chunks and apply the alpha samples to the 8-bit image that is contained in the JDAT chunks that follow the JSEP chunk.

### 1.1.4 IEND End of JNG datastream

The JNG IEND chunk is identical to its counterpart in PNG. Its data length is zero, and it serves to mark the end of the JNG datastream.

### 1.1.5 JSEP 12-bit/8-bit image separator

The JSEP chunk is empty.

A JSEP chunk must appear between the JDAT chunks of a 12-bit datastream and those of an 8-bit datastream, when `JDAT_sample_depth=20` in the JHDR chunk. The 12-bit datastream must appear first. Both images must have the same width, height, color type, compression method, and interlace type. Viewers can choose to display one or the other image, but not both.

## 1.2 JNG ancillary chunks

Some PNG ancillary chunks can also appear in JNG datastreams, and are used for the same purposes as described in the PNG specification:

If the bKGD chunk is present, it must be written as if it were written for a PNG datastream with sample_depth=8. It has one 2-byte entry for grayscale JNGs and three 2-byte entries for color JNGs. The first (most significant) byte of each entry must be 0.

The following chunks have exactly the same meaning and have the same format as given in the PNG specification: cHRM, gAMA, iCCP, sRGB, pHYs, oFFs, tEXt, tIME, and zTXt.

The PNG PLTE, hIST, pCAL, sBIT, and tRNS chunks are not defined in JNG.

When cHRM, gAMA, iCCP, or sRGB are present, they provide information about the colorspace of the decoded JDAT image, and they have no effect on the decoded alpha samples from the IDAT chunks.

The chunk copying and ordering rules for JNG are the same as those in PNG, except for the fact that JDAT and IDAT chunks can be interleaved.

# Author's Address

Glenn Randers-Pehrson
611 Rivershore Court
Edgewood, MD 21040-3603

Phone: (410) 278-6554

EMail: randeg @ alumni.rpi.edu

*End of JNG Specification. This document expires on 30 March 1999*