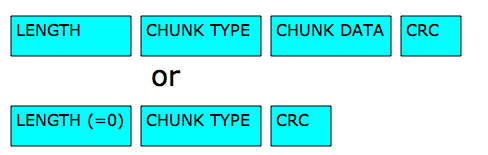
PNG规范(Portable Network Graphics)

为什么要使用PNG？最大一个原因就是它支持透明通道。

PNG是一种无损压缩的位图图形格式

## Chunk Data



|  |  |
| --- | --- |
| Length | A **four-byte** unsigned integer giving the number of bytes in the chunk's data field. The length counts **only** the data field, **not** itself, the chunk type, or the CRC. Zero is a valid length. Although encoders and decoders should treat the length as unsigned, its value shall not exceed 231-1 bytes. |
| Chunk Type | A sequence of **four bytes** defining the chunk type. Each byte of a chunk type is restricted to the decimal values 65 to 90 and 97 to 122. These correspond to the uppercase and lowercase ISO 646 letters (A-Z and a-z) respectively for convenience in description and examination of PNG datastreams. Encoders and decoders shall treat the chunk types as fixed binary values, not character strings. For example, it would not be correct to represent the chunk type **IDAT** by the equivalents of those letters in the UCS 2 character set. Additional naming conventions for chunk types are discussed in 5.4: Chunk naming conventions. |
| Chunk Data | The data bytes appropriate to the chunk type, if any. This field can be of zero length. |
| CRC | A **four-byte** CRC (Cyclic Redundancy Code) calculated on the preceding bytes in the chunk, including the chunk type field and chunk data fields, but **not** including the length field. The CRC can be used to check for corruption of the data. The CRC is always present, even for chunks containing no data.  CRC fields are calculated using standardized CRC methods with pre and post conditioning, as defined by ISO 3309 [ISO-3309] and ITU-T V.42 [ITU-T-V42]. The CRC polynomial employed is：x32 + x26 + x23 + x22 + x16 + x12 + x11 + x10 + x8 + x7 + x5 + x4 + x2 + x + 1 |

**Chunk ordering rules**

|  |  |  |
| --- | --- | --- |
| **Critical chunks**  **(shall appear in this order, except PLTE is optional)** | | |
| **Chunk name** | **Multiple allowed** | **Ordering constraints** |
| **IHDR** | No | Shall be first |
| **PLTE** | No | Before first **IDAT** |
| **IDAT** | Yes | Multiple **IDAT** chunks shall be consecutive |
| **IEND** | No | Shall be last |
| **Ancillary chunks**  **(need not appear in this order)** | | |
| **Chunk name** | **Multiple allowed** | **Ordering constraints** |
| **cHRM** | No | Before **PLTE** and **IDAT** |
| **gAMA** | No | Before **PLTE** and **IDAT** |
| **iCCP** | No | Before **PLTE** and **IDAT**. If the **iCCP** chunk is present, the **sRGB** chunk should not be present. |
| **sBIT** | No | Before **PLTE** and **IDAT** |
| **sRGB** | No | Before **PLTE** and **IDAT**. If the **sRGB** chunk is present, the **iCCP** chunk should not be present. |
| **bKGD** | No | After **PLTE**; before **IDAT** |
| **hIST** | No | After **PLTE**; before **IDAT** |
| **tRNS** | No | After **PLTE**; before **IDAT** |
| **pHYs** | No | Before **IDAT** |
| **sPLT** | Yes | Before **IDAT** |
| **tIME** | No | None |
| **iTXt** | Yes | None |
| **tEXt** | Yes | None |
| **zTXt** | Yes | None |

## Critical chunks

### IHDR Image header

The four-byte chunk type field contains the decimal values

IHDR: 49 48 44 52

The **IHDR** chunk shall be the first chunk in the PNG datastream. It contains:

|  |  |
| --- | --- |
| Width | 4 bytes |
| Height | 4 bytes |
| Bit depth | 1 byte |
| Colour type | 1 byte |
| Compression method | 1 byte |
| Filter method | 1 byte |
| Interlace method | 1 byte |

Width and height give the image dimensions in pixels. They are PNG four-byte unsigned integers. Zero is an invalid value.

**Bit depth** is a single-byte integer giving the number of bits per sample or per palette index (not per pixel). Valid values are 1, 2, 4, 8, and 16, although not all values are allowed for all colour types.

**Allowed combinations of colour type and bit depth**

|  |  |  |  |
| --- | --- | --- | --- |
| **PNG image type** | **Colour type** | **Allowed bit depths** | **Interpretation** |
| Greyscale | 0 | 1, 2, 4, 8, 16 | Each pixel is a greyscale sample |
| Truecolour | 2 | 8, 16 | Each pixel is an R,G,B triple |
| Indexed-colour | 3 | 1, 2, 4, 8 | Each pixel is a palette index; a **PLTE** chunk shall appear. |
| Greyscale with alpha | 4 | 8, 16 | Each pixel is a greyscale sample followed by an alpha sample. |
| Truecolour with alpha | 6 | 8, 16 | Each pixel is an R,G,B triple followed by an alpha sample. |

**Compression method** is a single-byte integer that indicates the method used to compress the image data. **Only** compression method 0 (deflate/inflate compression with a sliding window of at most 32768 bytes) is defined in this International Standard. All conforming PNG images shall be compressed with this scheme.

**Filter method** is a single-byte integer that indicates the preprocessing method applied to the image data before compression. **Only** filter method 0 (adaptive filtering with five basic filter types) is defined in this International Standard.

**Interlace method** is a single-byte integer that indicates the transmission order of the image data. Two values are defined in this International Standard: 0 (no interlace) or 1 (Adam7 interlace).

### PLTE(pallete chunk)

The **PLTE** chunk contains from 1 to 256 palette entries, each a three-byte series of the form:

|  |  |
| --- | --- |
| Red | 1 byte |
| Green | 1 byte |
| Blue | 1 byte |

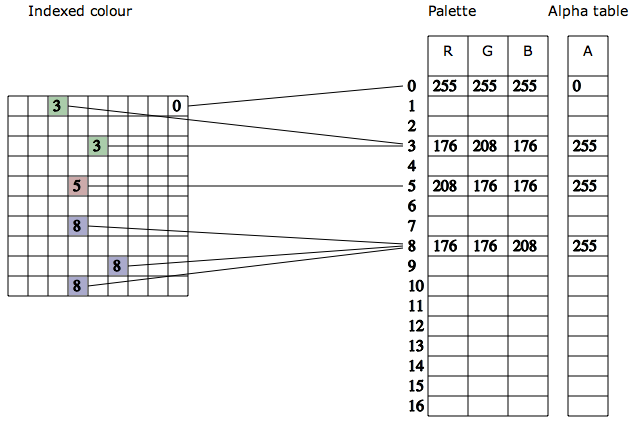
This chunk shall appear for colour type 3, and may appear for colour types 2 and 6; it shall not appear for colour types 0 and 4. There shall not be more than one **PLTE** chunk.

### IDAT(Image Data)

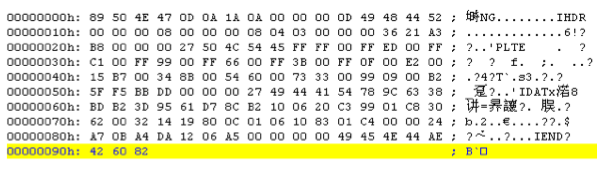
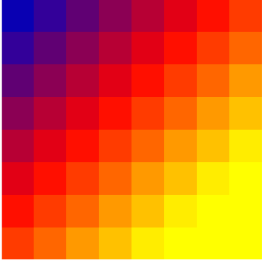
The **IDAT** chunk contains the actual image data which is the output stream of the compression algorithm.

### IEND

The **IEND** chunk marks the end of the PNG datastream. The chunk's data field is empty.



Example:



参考：

Portable Network Graphics (PNG) Specification (Second Edition)

<http://www.w3.org/TR/2003/REC-PNG-20031110/>

pngout

<http://advsys.net/ken/utils.htm> window/dos

<http://www.jonof.id.au/kenutils> linux/bsd

<http://tinypng.org/>