Digital Negative (DNG) Specification

Version 1.1.0.0

February 2005

ADOBE SYSTEMS INCORPORATED
Corporate Headquarters
345 Park Avenue
San Jose, CA 95110-2704
(408) 536-6000
http://www.adobe.com

Table of Contents

Preface

About This Document

The Digital Negative (DNG) Specification describes a non-proprietary file format for storing camera raw files that can be used by a wide range of hardware and software vendors.

This section contains information about this document, including how it is organized and where to go for additional information.

Audience

This document is intended for developers of hardware and software applications that will generate, process, manage, or archive camera raw files.

How This Document Is Organized

This document has the following sections:

- Chapter 1, "Introduction" explains what digital negatives are, gives an overview of the DNG file format, and discusses the advantages of DNG.
- Chapter 2, "DNG Format Overview" provides an overview of the DNG format, including information on file extensions, SubIFD trees, byte order, masked pixels, defective pixels, metadata, and proprietary data.
- Chapter 3, "Restrictions and Extensions to Existing TIFF Tags" describes tag differences between DNG and the TIFF 6.0 format on which DNG is based.
- Chapter 4, "DNG Tags" lists all DNG-specific tags and describes how they are used.
- Chapter 5, "Mapping Raw Values to Linear Reference Values" specifies DNG's processing model for mapping stored raw sensor values into linear reference values.
- Chapter 6, "Mapping Camera Color Space to CIE XYZ Space" describes DNG's processing model for mapping between the camera color space coordinates (linear reference values) and CIE XYZ coordinates.
- "Appendix A: Compatibility with Previous Versions" documents compatibility between the current and previous DNG versions.

Where to Go for More Information

DNG is an extension of TIFF 6.0 and is compatible with the TIFF-EP standard. See these specifications for more information on TIFF and TIFF-EP:

TIFF 6.0 Specification, Adobe Systems, Inc., 1992-06-03.	http://partners.adobe.com/asn/developer/pdfs/tn/TIFF6.pdf
TIFF/EP Specification, ISO/DIS 12234-2, 2001-10-15.	http://www.iso.org

1 Introduction

The Pros and Cons of Raw Data

Seeking a greater degree of flexibility and artistic control, professional photographers increasingly opt to manipulate raw data from their digital cameras. Unlike JPEG and TIFF formats which store images that have been processed by the camera, camera raw files capture unprocessed or minimally processed data directly from the camera sensor. Because they are analogous to film negatives in a photographer's workflow, camera raw formats are often referred to as "digital negatives."

Camera raw formats offer both advantages and disadvantages. One advantage is increased artistic control for the end user. The user can precisely adjust a range of parameters, including white balance, tone mapping, noise reduction, sharpening and others, to achieve a desired look.

One disadvantage is that unlike JPEG and TIFF files which are ready for immediate use, camera raw files must be processed before they can be used, typically through software provided by the camera manufacturer or through a converter like the Adobe® Camera Raw plug-in for Adobe Photoshop® software.

The challenge for end users and camera vendors alike is that there is no publicly-documented and supported format for storing raw camera data. Every camera manufacturer that supports raw data must create their own proprietary format, along with software for converting the proprietary format into the standard JPEG and/or TIFF formats.

A Standard Format

The lack of a standard format for camera raw files creates additional work for camera manufacturers because they need to develop proprietary formats along with the software to process them. It also poses risks for end users. Camera raw formats vary from camera to camera, even those produced by the same manufacturer. It is not uncommon for a camera manufacturer to terminate support for a discontinued camera's raw format. This means users have no guarantee they will be able to open archived camera raw files in the future.

To address these problems, Adobe has defined a new non-proprietary format for camera raw files. The format, called Digital Negative or DNG, can be used by a wide range of hardware and software developers to provide a more flexible raw processing and archiving workflow. End users may also use DNG as an intermediate format for storing images that were originally captured using a proprietary camera raw format.

The Advantages of DNG

DNG has all the benefits of current camera raw formats; namely, increased flexibility and artistic control. In addition, DNG offers several new advantages over proprietary camera raw formats.

Self-Contained

With the current proprietary camera raw formats, software programs wishing to process camera raw files must have specific information about the camera that created the file. As new camera models are released, software manufacturers (and by extension users) must update their software to accommodate the new camera raw formats.

Because DNG metadata is publicly documented, software readers such as the Adobe Camera Raw plug-in do not need camera-specific knowledge to decode and process files created by a camera that supports DNG. That means reduced software maintenance and a more self-contained solution for end users.

Archival

Camera manufacturers often drop support for a propriety raw format a few years after a camera is discontinued. Without continued software support, users may not be able to access images stored in proprietary raw formats and the images may be lost forever. Since DNG is publicly documented, it is far more likely that raw images stored as DNG files will be readable by software in the distant future, making DNG a safer choice for archival.

TIFF Compatible

DNG is an extension of the TIFF 6.0 format, and is compatible with the TIFF-EP standard. It is possible (but not required) for a DNG file to simultaneously comply with both the Digital Negative specification and the TIFF-EP standard.

2 DNG Format Overview

This section describes the DNG format. As an extension of the TIFF 6.0 format, DNG should follow all the formatting rules for TIFF 6.0. For more information, refer to the TIFF 6.0 specification.

The following topics are discussed in this section:

- File Extensions
- SubIFD Trees
- Byte Order
- Masked Pixels
- Defective Pixels
- Metadata
- Proprietary Data

File Extensions

The recommended file extension for Digital Negative is ".DNG". Readers should accept either the ".DNG" or ".TIF" extensions for compatibility with TIFF-EP.

SubIFD Trees

DNG recommends the use of SubIFD trees, as described in the TIFF-EP specification. SubIFD chains are not supported.

The highest-resolution and quality IFD should use NewSubFileType equal to 0. Reduced resolution (or quality) thumbnails or previews, if any, should use NewSubFileType equal to 1.

DNG recommends, but does not require, that the first IFD contain a low-resolution thumbnail, as described in the TIFF-EP specification.

Byte Order

DNG readers are required to support either byte order, even for files from a particular camera model. Writers can write either byte order, whichever is easier and/or faster for the writer.

Masked Pixels

Most camera sensors measure the black encoding level using fully-masked pixels at the edges of the sensor. These pixels can either be trimmed before storing the image in DNG, or they can be included in the stored image. If the masked pixels are not trimmed, the area of the non-masked pixels must be specified using the ActiveArea tag.

The black encoding level information extracted from these masked pixels should be used to either pre-compensate the raw data stored in the file or they should be included in the file using the DNG tags for specifying the black level.

This black encoding level information is required even if the masked pixels are not trimmed, to allow DNG readers to process the image without requiring knowledge of the best way to compute the black levels for any given camera model.

Defective Pixels

Defective pixels should be mapped out (interpolated over) before the raw data is stored as DNG.

Metadata

Additional metadata may be embedded in DNG in the following ways:

- Using TIFF-EP or EXIF metadata tags
- Using the IPTC metadata tag (33723)
- Using the XMP metadata tag (700)

Note that TIFF-EP and EXIF use nearly the same metadata tag set, but TIFF-EP stores the tags in IFD 0, while EXIF store the tags in a separate IFD. Either location is allowed by DNG, but the EXIF location is preferred.

Proprietary Data

Camera manufacturers often want to include proprietary data in a raw file for use by their own raw converter. DNG allows proprietary data to be stored using private tags, private IFDs, and/or a private MakerNote.

It is recommended that manufacturers use the DNGPrivateData and MakerNoteSafety tags to ensure that programs that edit DNG files preserve this proprietary data. See Chapter 4, “DNG Tags” on page 17 for more information on the DNGPrivateData and MakerNoteSafety tags.

3 Restrictions and Extensions to Existing TIFF Tags

This section describes the restrictions and extension to the following TIFF tags:

- BitsPerSample
- Compression
- PhotometricInterpretation
- Orientation

BitsPerSample

Supported values are from 8 to 32 bits/sample. The depth must be the same for each sample if SamplesPerPixel is not equal to 1.

If BitsPerSample is not equal to 8 or 16 or 32, then the bits must be packed into bytes using the TIFF default FillOrder of 1 (big-endian), even if the TIFF file itself uses little-endian byte order.

Compression

Two Compression tag values are supported:

- Value = 1: Uncompressed data.
- Value = 7: JPEG compressed data, either baseline DCT JPEG, or lossless JPEG compression.

If PhotometricInterpretation = 6 (YCbCr) and BitsPerSample = 8/8/8, or if PhotometricInterpretation = 1 (BlackIsZero) and BitsPerSample = 8, then the JPEG variant must be baseline DCT JPEG.

Otherwise, the JPEG variant must be lossless Huffman JPEG. For lossless JPEG, the internal width/length/components in the JPEG stream are not required to match the strip or tile's width/length/components. Only the total sample counts need to match. It is common for CFA images to be encoded with a different width, length or component count to allow the JPEG compression predictors to work across like colors.

PhotometricInterpretation

The following values are supported for thumbnail and preview IFDs only:

- 1 = BlackIsZero. Assumed to be in a gamma 2.2 color space.
- 2 = RGB. Assumed to be in the sRGB color space.
- 6 = YCbCr. Used for JPEG encoded preview images.

The following values are supported for the raw IFD, and are assumed to be the camera's native color space:

- 32803 = CFA (Color Filter Array).
- 34892 = LinearRaw.

The CFA PhotometricInterpretation value is documented in the TIFF-EP specification. Its use requires the use of the CFARepeatPatternDim and CFAPattern tags in the same IFD. The origin of the repeating CFA pattern is the top-left corner of the ActiveArea rectangle.

The LinearRaw PhotometricInterpretation value is intended for use by cameras that do not use color filter arrays, but instead capture all color components at each pixel. It can also be used for CFA data that has already been de-mosaiced.

The LinearRaw value can be used in reduced resolution IFDs, even if the raw IFD uses the CFA PhotometricInterpretation value.

Orientation

Orientation is a required tag for DNG. With the Orientation tag present, file browsers can perform lossless rotation of DNG files by modifying a single byte of the file.

DNG readers should support all possible orientations, including mirrored orientations. Note that the mirrored orientations are not allowed by the TIFF-EP specification, so writers should not use them if they want their files be compatible with both specifications.

4 DNG Tags

This section describes DNG-specific tags. Note that the tags listed here are not part of the TIFF-EP specification.

DNGVersion

Tag	50706
Type	BYTE
Count	4
Value	See below
Default	Required tag
Usage	IFD 0

Description

This tag encodes the DNG four-tier version number. For files compliant with this version of the DNG specification (1.1.0.0), this tag should contain the bytes: 1, 1, 0, 0.

DNGBackwardVersion

Tag	50707
Type	BYTE
Count	4
Value	See below
Default	DNGVersion with the last two bytes set to zero.
Usage	IFD 0

Description

This tag specifies the oldest version of the Digital Negative specification for which a file is compatible. Readers should not attempt to read a file if this tag specifies a version number that is higher than the version number of the specification the reader was based on.

In addition to checking the version tags, readers should, for all tags, check the types, counts, and values, to verify it is able to correctly read the file.

For more information on compatibility with previous DNG versions, see Appendix A: Compatibility with Previous Versions.

UniqueCameraModel

Tag	50708
Type	ASCII
Count	String length including null
Value	Null terminated string
Default	Required tag
Usage	IFD 0

Description

UniqueCameraModel defines a unique, non-localized name for the camera model that created the image in the raw file. This name should include the manufacturer's name to avoid conflicts, and should not be localized, even if the camera name itself is localized for different markets (see LocalizedCameraModel).

This string may be used by reader software to index into per-model preferences and replacement profiles.

Examples of unique model names are:

- "Canon EOS 300D"
- "Fujifilm FinePix S2Pro"
- "Kodak ProBack645"
- "Minolta DiMAGE A1"
- "Nikon D1X"
- "Olympus C-5050Z"
- "Pentax *istD"
- "Sony F828"

LocalizedCameraModel

Tag	50709
Type	ASCII or BYTE
Count	Byte count including null
Value	Null terminated UTF-8 encoded Unicode string
Default	Same as UniqueCameraModel
Usage	IFD 0

Description

Similar to the UniqueCameraModel field, except the name can be localized for different markets to match the localization of the camera name.

CFAPlaneColor

Tag	50710
Type	BYTE
Count	ColorPlanes
Value	See below
Default	0, 1, 2 (red, green, blue)
Usage	Raw IFD

Description

CFAPlaneColor provides a mapping between the values in the CFAPattern tag and the plane numbers in LinearRaw space. This is a required tag for non-RGB CFA images.

CFALayout

Tag	50711
Type	SHORT
Count	1
Value	See below
Default	1
Usage	Raw IFD

Description

CFALayout describes the spatial layout of the CFA. The currently defined values are:

1 = Rectangular (or square) layout
2 = Staggered layout A: even columns are offset down by 1/2 row
3 = Staggered layout B: even columns are offset up by 1/2 row
4 = Staggered layout C: even rows are offset right by 1/2 column
5 = Staggered layout D: even rows are offset left by 1/2 column

LinearizationTable

Tag	50712
Type	SHORT
Count	N
Value	See below
Default	Identity table (0, 1, 2, 3, etc.)
Usage	Raw IFD

Description

LinearizationTable describes a lookup table that maps stored values into linear values. This tag is typically used to increase compression ratios by storing the raw data in a non-linear, more visually uniform space with fewer total encoding levels.

If SamplesPerPixel is not equal to one, this single table applies to all the samples for each pixel.

See Chapter 5, "Mapping Raw Values to Linear Reference Values" on page 45 for details of the processing model.

BlackLevelRepeatDim

Tag	50713
Type	SHORT
Count	2
Value	Value 0: BlackLevelRepeatRows Value 1: BlackLevelRepeatCols
Default	1, 1
Usage	Raw IFD

Description

This tag specifies repeat pattern size for the BlackLevel tag.

BlackLevel

Tag	50714
Type	SHORT or LONG or RATIONAL
Count	BlackLevelRepeatRows * BlackLevelRepeatCols * SamplesPerPixel
Value	See below
Default	0
Usage	Raw IFD

Description

This tag specifies the zero light (a.k.a. thermal black or black current) encoding level, as a repeating pattern. The origin of this pattern is the top-left corner of the ActiveArea rectangle. The values are stored in row-column-sample scan order.

See Chapter 5, "Mapping Raw Values to Linear Reference Values" on page 45 for details of the processing model.

BlackLevelDeltaH

Tag	50715
Type	SRATIONAL
Count	ActiveArea width
Value	See below
Default	All zeros
Usage	Raw IFD

Description

If the zero light encoding level is a function of the image column, BlackLevelDeltaH specifies the difference between the zero light encoding level for each column and the baseline zero light encoding level.

If SamplesPerPixel is not equal to one, this single table applies to all the samples for each pixel.

See Chapter 5, "Mapping Raw Values to Linear Reference Values" on page 45 for details of the processing model.

BlackLevelDeltaV

Tag	50716
Type	SRATIONAL
Count	ActiveArea length
Value	See below
Default	All zeros
Usage	Raw IFD

Description

If the zero light encoding level is a function of the image row, this tag specifies the difference between the zero light encoding level for each row and the baseline zero light encoding level.

If SamplesPerPixel is not equal to one, this single table applies to all the samples for each pixel.

See Chapter 5, “Mapping Raw Values to Linear Reference Values” on page 45 for details of the processing model.

WhiteLevel

Tag	50717
Type	SHORT or LONG
Count	SamplesPerPixel
Value	See below
Default	(2 ** BitsPerSample) - 1
Usage	Raw IFD

Description

This tag specifies the fully saturated encoding level for the raw sample values. Saturation is caused either by the sensor itself becoming highly non-linear in response, or by the camera's analog to digital converter clipping.

See Chapter 5, “Mapping Raw Values to Linear Reference Values” on page 45 for details of the processing model.

DefaultScale

Tag	50718
Type	RATIONAL
Count	2
Value	Value 0: DefaultScaleH Value 1: DefaultScaleV
Default	1.0, 1.0
Usage	Raw IFD

Description

DefaultScale is required for cameras with non-square pixels. It specifies the default scale factors for each direction to convert the image to square pixels. Typically these factors are selected to approximately preserve total pixel count.

For CFA images that use CFALayout equal to 2, 3, 4, or 5, such as the Fujifilm SuperCCD, these two values should usually differ by a factor of 2.0.

BestQualityScale

Tag	50780
Type	RATIONAL
Count	1
Value	see below
Default	1.0
Usage	Raw IFD

Description

For some cameras, the best possible image quality is not achieved by preserving the total pixel count during conversion. For example, Fujifilm SuperCCD images have maximum detail when their total pixel count is doubled.

This tag specifies the amount by which the values of the DefaultScale tag need to be multiplied to achieve the best quality image size.

DefaultCropOrigin

Tag	50719
Type	SHORT or LONG or RATIONAL
Count	2
Value	Value 0: DefaultCropOriginH Value 1: DefaultCropOriginV
Default	0, 0
Usage	Raw IFD

Description

Raw images often store extra pixels around the edges of the final image. These extra pixels help prevent interpolation artifacts near the edges of the final image.

DefaultCropOrigin specifies the origin of the final image area, in raw image coordinates (i.e., before the DefaultScale has been applied), relative to the top-left corner of the ActiveArea rectangle.

DefaultCropSize

Tag	50720
Type	SHORT or LONG or RATIONAL
Count	2
Value	Value 0: DefaultCropSizeH Value 1: DefaultCropSizeV
Default	ImageWidth, ImageLength
Usage	Raw IFD

Description

Raw images often store extra pixels around the edges of the final image. These extra pixels help prevent interpolation artifacts near the edges of the final image.

DefaultCropSize specifies the size of the final image area, in raw image coordinates (i.e., before the DefaultScale has been applied).

CalibrationIlluminant1

Tag	50778
Type	SHORT
Count	1
Value	See below
Default	0 (unknown)
Usage	IFD 0

Description

The illuminant used for the first set of color calibration tags (ColorMatrix1, CameraCalibration1, ReductionMatrix1). The legal values for this tag are the same as the legal values for the LightSource EXIF tag.

See Chapter 6, “Mapping Camera Color Space to CIE XYZ Space” on page 47 for details of the color-processing model.

CalibrationIlluminant2

Tag	50779
Type	SHORT
Count	1
Value	See below
Default	None
Usage	IFD 0

Description

The illuminant used for an optional second set of color calibration tags (ColorMatrix2, CameraCalibration2, ReductionMatrix2). The legal values for this tag are the same as the legal values for the CalibrationIlluminant1 tag; however, if both are included, neither is allowed to have a value of 0 (unknown).

See Chapter 6, “Mapping Camera Color Space to CIE XYZ Space” on page 47 for details of the color-processing model.

ColorMatrix1

Tag	50721
Type	SRATIONAL
Count	ColorPlanes * 3
Value	See below
Default	None
Usage	IFD 0

Description

ColorMatrix1 defines a transformation matrix that converts XYZ values to reference camera native color space values, under the first calibration illuminant. The matrix values are stored in row scan order.

The ColorMatrix1 tag is required for all non-monochrome DNG files.

See Chapter 6, “Mapping Camera Color Space to CIE XYZ Space” on page 47 for details of the color-processing model.

ColorMatrix2

Tag	50722
Type	SRATIONAL
Count	ColorPlanes * 3
Value	See below
Default	None
Usage	IFD 0

Description

ColorMatrix2 defines a transformation matrix that converts XYZ values to reference camera native color space values, under the second calibration illuminant. The matrix values are stored in row scan order.

See Chapter 6, “Mapping Camera Color Space to CIE XYZ Space” on page 47 for details of the color-processing model.

CameraCalibration1

Tag	50723
Type	SRATIONAL
Count	ColorPlanes * ColorPlanes
Value	See below
Default	Identity matrix
Usage	IFD 0

Description

CameraClalibration1 defines a calibration matrix that transforms reference camera native space values to individual camera native space values under the first calibration illuminant. The matrix is stored in row scan order.

This matrix is stored separately from the matrix specified by the ColorMatrix1 tag to allow raw converters to swap in replacement color matrices based on UniqueCameraModel tag, while still taking advantage of any per-individual camera calibration performed by the camera manufacturer.

See Chapter 6, "Mapping Camera Color Space to CIE XYZ Space" on page 47 for details of the color-processing model.

CameraCalibration2

Tag	50724
Type	SRATIONAL
Count	ColorPlanes * ColorPlanes
Value	See below
Default	Identity matrix
Usage	IFD 0

Description

CameraCalibration2 defines a calibration matrix that transforms reference camera native space values to individual camera native space values under the second calibration illuminant. The matrix is stored in row scan order.

This matrix is stored separately from the matrix specified by the ColorMatrix2 tag to allow raw converters to swap in replacement color matrices based on UniqueCameraModel tag, while still taking advantage of any per-individual camera calibration performed by the camera manufacturer.

See Chapter 6, "Mapping Camera Color Space to CIE XYZ Space" on page 47 for details of the color-processing model.

ReductionMatrix1

Tag	50725
Type	SRATIONAL
Count	3 * ColorPlanes
Value	See below
Default	None
Usage	IFD 0

Description

ReductionMatrix1 defines a dimensionality reduction matrix for use as the first stage in converting color camera native space values to XYZ values, under the first calibration illuminant. This tag may only be used if ColorPlanes is greater than 3. The matrix is stored in row scan order.

See Chapter 6, “Mapping Camera Color Space to CIE XYZ Space” on page 47 for details of the color-processing model.

ReductionMatrix2

Tag	50726
Type	SRATIONAL
Count	3 * ColorPlanes
Value	See below
Default	None
Usage	IFD 0

Description

ReductionMatrix2 defines a dimensionality reduction matrix for use as the first stage in converting color camera native space values to XYZ values, under the second calibration illuminant. This tag may only be used if ColorPlanes is greater than 3. The matrix is stored in row scan order.

See Chapter 6, “Mapping Camera Color Space to CIE XYZ Space” on page 47 for details of the color-processing model.

AnalogBalance

Tag	50727
Type	RATIONAL
Count	ColorPlanes
Value	See below
Default	All 1.0
Usage	IFD 0

Description

Normally the stored raw values are not white balanced, since any digital white balancing will reduce the dynamic range of the final image if the user decides to later adjust the white balance; however, if camera hardware is capable of white balancing the color channels before the signal is digitized, it can improve the dynamic range of the final image.

AnalogBalance defines the gain, either analog (recommended) or digital (not recommended) that has been applied the stored raw values.

See Chapter 6, “Mapping Camera Color Space to CIE XYZ Space” on page 47 for details of the color-processing model.

AsShotNeutral

Tag	50728
Type	SHORT or RATIONAL
Count	ColorPlanes
Value	See below
Default	None
Usage	IFD 0

Description

AsShotNeutral specifies the selected white balance at time of capture, encoded as the coordinates of a perfectly neutral color in linear reference space values. The inclusion of this tag precludes the inclusion of the AsShotWhiteXY tag.

AsShotWhiteXY

Tag	50729
Type	RATIONAL
Count	2
Value	See below
Default	None
Usage	IFD 0

Description

AsShotWhiteXY specifies the selected white balance at time of capture, encoded as x-y chromaticity coordinates. The inclusion of this tag precludes the inclusion of the AsShotNeutral tag.

BaselineExposure

Tag	50730
Type	SRATIONAL
Count	1
Value	See below
Default	0.0
Usage	IFD 0

Description

Camera models vary in the trade-off they make between highlight headroom and shadow noise. Some leave a significant amount of highlight headroom during a normal exposure. This allows significant negative exposure compensation to be applied during raw conversion, but also means normal exposures will contain more shadow noise. Other models leave less headroom during normal exposures. This allows for less negative exposure compensation, but results in lower shadow noise for normal exposures.

Because of these differences, a raw converter needs to vary the zero point of its exposure compensation control from model to model. BaselineExposure specifies by how much (in EV units) to move the zero point. Positive values result in brighter default results, while negative values result in darker default results.

BaselineNoise

Tag	50731
Type	RATIONAL
Count	1
Value	See below
Default	1.0
Usage	IFD 0

Description

BaselineNoise specifies the relative noise level of the camera model at a baseline ISO value of 100, compared to a reference camera model.

Since noise levels tend to vary approximately with the square root of the ISO value, a raw converter can use this value, combined with the current ISO, to estimate the relative noise level of the current image.

BaselineSharpness

Tag	50732
Type	RATIONAL
Count	1
Value	See below
Default	1.0
Usage	IFD 0

Description

BaselineSharpness specifies the relative amount of sharpening required for this camera model, compared to a reference camera model. Camera models vary in the strengths of their anti-aliasing filters. Cameras with weak or no filters require less sharpening than cameras with strong anti-aliasing filters.

BayerGreenSplit

Tag	50733
Type	LONG
Count	1
Value	See below
Default	0
Usage	Raw IFD

Description

BayerGreenSplit only applies to CFA images using a Bayer pattern filter array. This tag specifies, in arbitrary units, how closely the values of the green pixels in the blue/green rows track the values of the green pixels in the red/green rows.

A value of zero means the two kinds of green pixels track closely, while a non-zero value means they sometimes diverge. The useful range for this tag is from 0 (no divergence) to about 5000 (quite large divergence).

LinearResponseLimit

Tag	50734
Type	RATIONAL
Count	1
Value	See below
Default	1.0
Usage	IFD 0

Description

Some sensors have an unpredictable non-linearity in their response as they near the upper limit of their encoding range. This non-linearity results in color shifts in the highlight areas of the resulting image unless the raw converter compensates for this effect.

LinearResponseLimit specifies the fraction of the encoding range above which the response may become significantly non-linear.

CameraSerialNumber

Tag	50735
Type	ASCII
Count	String length including null
Value	Null terminated string
Default	None
Usage	IFD 0

Description

CameraSerialNumber contains the serial number of the camera or camera body that captured the image.

LensInfo

Tag	50736
Type	RATIONAL
Count	4
Value	Value 0: Minimum focal length in mm. Value 1: Maximum focal length in mm. Value 2: Minimum (maximum aperture) f-stop at minimum focal length. Value 3: Minimum (maximum aperture) f-stop at maximum focal length.
Default	none
Usage	IFD 0

Description

LensInfo contains information about the lens that captured the image. If the minimum f-stops are unknown, they should be encoded as 0/0.

ChromaBlurRadius

Tag	50737
Type	RATIONAL
Count	1
Value	Chroma blur radius in pixels
Default	See below
Usage	Raw IFD

Description

ChromaBlurRadius provides a hint to the DNG reader about how much chroma blur should be applied to the image. If this tag is omitted, the reader will use its default amount of chroma blurring.

Normally this tag is only included for non-CFA images, since the amount of chroma blur required for mosaic images is highly dependent on the de-mosaic algorithm, in which case the DNG reader's default value is likely optimized for its particular de-mosaic algorithm.

AntiAliasStrength

Tag	50738
Type	RATIONAL
Count	1
Value	Relative strength of the camera's anti-alias filter
Default	1.0
Usage	Raw IFD

Description

AntiAliasStrength provides a hint to the DNG reader about how strong the camera's anti-alias filter is. A value of 0.0 means no anti-alias filter (i.e., the camera is prone to aliasing artifacts with some subjects), while a value of 1.0 means a strong anti-alias filter (i.e., the camera almost never has aliasing artifacts).

Note that this tag overlaps in functionality with the BaselineSharpness tag. The primary difference is the AntiAliasStrength tag is used as a hint to the de-mosaic algorithm, while the BaselineSharpness tag is used as a hint to a sharpening algorithm applied later in the processing pipeline.

DNGPrivateData

Tag	50740
Type	BYTE
Count	Length of private data block in bytes
Value	See below
Default	None
Usage	IFD 0

Description

DNGPrivateData provides a way for camera manufacturers to store private data in the DNG file for use by their own raw converters, and to have that data preserved by programs that edit DNG files.

The private data must follow these rules:

- *The private data must start with a null-terminated ASCII string identifying the data.* The first part of this string must be the manufacturer's name, to avoid conflicts between manufacturers.
- *The private data must be self-contained.* All offsets within the private data must be offsets relative to the start of the private data, and must not point to bytes outside the private data.
- *The private data must be byte-order independent.* If a DNG file is converted from a big-endian file to a little-endian file, the data must remain valid.

MakerNoteSafety

Tag	50741
Type	SHORT
Count	1
Value	0 (unsafe) or 1 (safe)
Default	0
Usage	IFD 0

Description

MakerNoteSafety lets the DNG reader know whether the EXIF MakerNote tag is safe to preserve along with the rest of the EXIF data. File browsers and other image management software processing an image with a preserved MakerNote should be aware that any thumbnail image embedded in the MakerNote may be stale, and may not reflect the current state of the full size image.

A MakerNote is safe to preserve if it follows these rules:

- *The MakerNote data must be self-contained.* All offsets within the MakerNote must be offsets relative to the start of the MakerNote, and must not point to bytes outside the MakerNote.
- *The MakerNote data must be byte-order independent.* Moving the data to a file with a different byte order must not invalidate it.

ShadowScale

Tag	50739
Type	URATIONAL
Count	1
Value	See below
Default	1.0
Usage	IFD 0

Description

This tag is used by Adobe Camera Raw to control the sensitivity of its "Shadows" slider.

RawDataUniqueID

Tag	50781
Type	BYTE
Count	16
Value	See below
Default	Optional
Usage	IFD 0

Description

This tag contains a 16-byte unique identifier for the raw image data in the DNG file. DNG readers can use this tag to recognize a particular raw image, even if the file's name or the metadata contained in the file has been changed.

If a DNG writer creates such an identifier, it should do so using an algorithm that will ensure that it is very unlikely two different images will end up having the same identifier.

OriginalRawFileName

Tag	50827
Type	ASCII or BYTE
Count	Byte count including null
Value	Null terminated UTF-8 encoded Unicode string
Default	Optional
Usage	IFD 0

Description

If the DNG file was converted from a non-DNG raw file, then this tag contains the file name of that original raw file.

OriginalRawFileData

Tag	50828
Type	UNDEFINED
Count	Byte count of embedded data
Value	See below
Default	Optional
Usage	IFD 0

Description

If the DNG file was converted from a non-DNG raw file, then this tag contains the compressed contents of that original raw file.

The contents of this tag always use the big-endian byte order.

The tag contains a sequence of data blocks. Future versions of the DNG specification may define additional data blocks, so DNG readers should ignore extra bytes when parsing this tag. DNG readers should also detect the case where data blocks are missing from the end of the sequence, and should assume a default value for all the missing blocks.

There are no padding or alignment bytes between data blocks. The sequence of data blocks is:

1. Compressed data fork of original raw file.
2. Compressed Mac OS resource fork of original raw file.
3. Mac OS file type (4 bytes) of original raw file.
4. Mac OS file creator (4 bytes) of original raw file.
5. Compressed data fork of sidecar ".THM" file.
6. Compressed Mac OS resource fork of sidecar ".THM" file.
7. Mac OS file type (4 bytes) of sidecar ".THM" file.
8. Mac OS file creator (4 bytes) of sidecar ".THM" file.

If the Mac OS file types or creator codes are unknown, zero is stored.

If the Mac OS resource forks do not exist, they should be encoded as zero length forks.

Each fork (data or Mac OS resource) is compressed and encoded as:

> ForkLength = first four bytes. This is the uncompressed length of this fork. If this value is zero, then no more data is stored for this fork.

From ForkLength, compute the number of 64K compression blocks used for this data (the last block is usually smaller than 64K):

ForkBlocks = Floor ((ForkLength + 65535) / 65536)

The next (ForkBlocks + 1) 4-byte values are an index into the compressed data. The first ForkBlock values are offsets from the start of the data for this fork to the start of the compressed data for the corresponding compression block. The last value is an offset from the start of the data for this fork to the end of the data for this fork.

Following this index is the ZIP compressed data for each 64K compression block.

ActiveArea

Tag	50829
Type	SHORT or LONG
Count	4
Value	See below
Default	0, 0, ImageLength, ImageWidth
Usage	Raw IFD

Description

This rectangle defines the active (non-masked) pixels of the sensor. The order of the rectangle coordinates is: top, left, bottom, right.

MaskedAreas

Tag	50830
Type	SHORT or LONG
Count	4 * number of rectangles
Value	See below
Default	None
Usage	Raw IFD

Description

This tag contains a list of non-overlapping rectangle coordinates of fully masked pixels, which can be optionally used by DNG readers to measure the black encoding level.

The order of each rectangle's coordinates is: top, left, bottom, right.

If the raw image data has already had its black encoding level subtracted, then this tag should not be used, since the masked pixels are no longer useful.

Note that DNG writers are still required to include estimate and store the black encoding level using the black level DNG tags. Support for the MaskedAreas tag is not required of DNG readers.

AsShotICCProfile

Tag	50831
Type	UNDEFINED
Count	Length of ICC profile in bytes
Value	See below
Default	Optional
Usage	IFD 0

Description

This tag contains an ICC profile that, in conjunction with the AsShotPreProfileMatrix tag, provides the camera manufacturer with a way to specify a default color rendering from camera color space coordinates (linear reference values) into the ICC profile connection space.

The ICC profile connection space is an output referred colorimetric space, whereas the other color calibration tags in DNG specify a conversion into a scene referred colorimetric space. This means that the rendering in this profile should include any desired tone and gamut mapping needed to convert between scene referred values and output referred values.

DNG readers that have their own tone and gamut mapping controls (such as Adobe Camera Raw) will probably ignore this tag pair.

AsShotPreProfileMatrix

Tag	50832
Type	SRATIONAL
Count	3 * ColorPlanes or ColorPlanes * ColorPlanes
Value	See below
Default	Identity matrix
Usage	IFD 0

Description

This tag is used in conjunction with the AsShotICCProfile tag. It specifies a matrix that should be applied to the camera color space coordinates before processing the values through the ICC profile specified in the AsShotICCProfile tag.

The matrix is stored in the row scan order.

If ColorPlanes is greater than three, then this matrix can (but is not required to) reduce the dimensionality of the color data down to three components, in which case the AsShotICCProfile should have three rather than ColorPlanes input components.

CurrentICCProfile

Tag	50833
Type	UNDEFINED
Count	Length of ICC profile in bytes
Value	See below
Default	Optional
Usage	IFD 0

Description

This tag is used in conjunction with the CurrentPreProfileMatrix tag.

The CurrentICCProfile and CurrentPreProfileMatrix tags have the same purpose and usage as the AsShotICCProfile and AsShotPreProfileMatrix tag pair, except they are for use by raw file editors rather than camera manufacturers.

CurrentPreProfileMatrix

Tag	50834
Type	SRATIONAL
Count	3 * ColorPlanes or ColorPlanes * ColorPlanes
Value	See below
Default	Identity matrix
Usage	IFD 0

Description

This tag is used in conjunction with the CurrentICCProfile tag.

The CurrentICCProfile and CurrentPreProfileMatrix tags have the same purpose and usage as the AsShotICCProfile and AsShotPreProfileMatrix tag pair, except they are for use by raw file editors rather than camera manufacturers.

5 Mapping Raw Values to Linear Reference Values

The section describes DNG's processing model for mapping stored raw sensor values into linear reference values. Linear reference values encode zero light as 0.0, and the maximum useful value (limited by either sensor saturation or analog to digital converter clipping) as 1.0. If SamplesPerPixel is greater than one, each sample plane should be processed independently.

The processing model follows these steps:

- Linearization
- Black Subtraction
- Rescaling
- Clipping

Linearization

The first step is to process the raw values through the look-up table specified by the LinearizationTable tag, if any. If the raw value is greater than the size of the table, it is mapped to the last entry of the table.

Black Subtraction

The black level for each pixel is then computed and subtracted. The black level for each pixel is the sum of the black levels specified by the BlackLevel, BlackLevelDeltaH and BlackLevelDeltaV tags.

Rescaling

The black subtracted values are then rescaled to map them to a logical 0.0 to 1.0 range. The scale factor is the inverse of the difference between the value specified in the WhiteLevel tag and the maximum computed black level for the sample plane.

Clipping

The rescaled values are then clipped to a 0.0 to 1.0 logical range.

6 Mapping Camera Color Space to CIE XYZ Space

This section describes DNG's processing model for mapping between the camera color space coordinates (linear reference values) and CIE XYZ coordinates.

One or Two Color Calibrations

DNG provides for one or two sets of color calibration tags, each set optimized for a different illuminant.

If both sets of color calibration tags are included, the raw converter can interpolate between the color calibrations based on the color temperature selected by the user, or can allow the user to select between the two calibrations.

If two calibrations are included, it is recommended that one of the calibrations be for a low color temperature illuminant (e.g., Standard-A), and the second calibration illuminant be a higher color temperature illuminant (e.g., D55 or D65). This combination has been found to work well for wide range of real-world digital camera images.

XYZ to Camera Transform

The transform from XYZ coordinates to camera color space coordinates is defined by a sequence of matrix operations. This transform is inverted to create a transform from camera color space coordinates to XYZ coordinates.

Let n be the dimensionality of the camera color space (usually 3 or 4).

Let M be the n-by-3 matrix defined by the ColorMatrix tag.

Let C be the n-by-n matrix defined by the CameraCalibration tag.

Let G be the n-by-n matrix, which is zero except for the diagonal entries, which are defined by the AnalogBalance tag.

Then the XYZ to camera space matrix is: A = G * C * M.

Camera to XYZ Transform

Raw converters actually need the inverse of this matrix: B = Inverse (A), which is needed to convert camera color space coordinates to XYZ coordinates.

If n = 3, then A is a 3-by-3 matrix, and has a unique inverse.

If n > 3, then A is an n-by-3 matrix, and there are an infinite number of possible inverses. The optional ReductionMatrix tag provides hint to the raw converter on how to invert this matrix.

Let R be the 3-by-n matrix defined by the ReductionMatrix tag.

Then the suggested inverse is: B = Inverse (R * A) * R.

If the ReductionMatrix tag is not included, the usual inversion method is to use R = Transpose (A), which is known as the pseudo inverse.

Appendix A: Compatibility with Previous Versions

This appendix documents only the differences between this and previous versions of the DNG specification that are relevant to compatibility. Differences that are not relevant to compatibility (e.g., new optional tags that DNG readers are not required to support) are not documented in this appendix.

This information is useful in enabling DNG readers to correctly read DNG files with older version numbers. It also helps determine what version DNG writers can include in the DNGBackwardVersion tag.

The first version of the DNG speciation that was published was version 1.0.0.0.

Compatibility Issue 1: ActiveArea Tag

The ActiveArea tag was added to the DNG specification in version 1.1.0.0. Previous versions of the DNG specification do not support storing masked pixels.

DNG writers should set the DNGBackwardVersion to a minimum of 1.1.0.0 if the masked pixels are stored in the DNG file.

Compatibility Issue 2: 16-bit Lossless JPEG Encoding

The Lossless JPEG encoder/decoder used by Adobe applications to read and write DNG files before version 1.1.0.0 incorrectly deviated from the JPEG specification when dealing with 16-bit data. Since both the encoder and decoder deviated in the same way, no data was lost; however the data stream did not exactly match the data stream specified in the Lossless JPEG specification.

Because the vast majority of DNG 1.0.0.0 files using 16-bit Lossless JPEG encoding were created by Adobe applications, it is strongly recommended that software that reads or writes DNG files with version numbers less than 1.1.0.0 incorporate this deviation. Software that reads or writes DNG files with version 1.1.0.0 or later can safely assume that the Lossless JPEG stream is fully compliant with the Lossless JPEG specification.

Description of deviation

Lossless JPEG encodes the difference between a predicted value and the actual value for each pixel. With 16-bit data, these differences are computed modulo 16-bits, so the range of possible differences is -32768 to + 32767. Two values are stored for the difference. First the number of bits required to store the difference (encoded via a Huffman code), and then the actual difference.

Using the difference encoding scheme in the Lossless JPEG specification, only one difference value would take 16-bits to store: -32768. The Lossless JPEG specification special cases this difference bit length, and since there is only one possible difference value it does not bother to use any bits to store the actual difference.

In earlier versions of DNG the special case logic is not present, and the difference value for -32768 is stored in the compressed data stream as with all other difference bit lengths.