Photoshop CS File Formats Specification

ADOBE SYSTEMS INCORPORATED

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

October 2003

Adobe Photoshop File Formats Specification

Copyright 1991–2003 Adobe Systems Incorporated.
All rights reserved.

Portions Copyright 1990–1991, Thomas Knoll.

The information in this document is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Adobe Systems Incorporated. Adobe Systems Incorporated assumes no responsibility or liability for any errors or inaccuracies that may appear in this document. The software described in this document is furnished under license and may only be used or copied in accordance with the terms of such license.

Adobe, After Effects, PhotoDeluxe, Adobe Premiere, Photoshop, Adobe Illustrator, Adobe Type Manager, ATM and PostScript are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries. Apple, Macintosh, and Mac are trademarks of Apple Computer, Inc. registered in the United States and other countries. Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. All other products are the property of their respective owners.

Contents

Preface

Welcome to the Adobe Photoshop® File Format Specification!

This document is the detailed specification of the Adobe Photoshop file format and other pertinent file formats that Adobe Photoshop reads and writes.

Audience

The Adobe Photoshop SDK is a toolkit for C and C++ programmers who wish to write plug–ins for Adobe Photoshop on Macintosh and Windows systems.

This document assumes you are proficient in the C programming language and its tools. The source code files in the SDK are written for Metrowerks CodeWarrior on the Macintosh, and Microsoft Visual C++ on Windows.

You should have a working knowledge of Adobe Photoshop, and understand how plug–in modules work from a user’s viewpoint. This guide assumes you understand Photoshop terminology such as *paths*, *layers* and *masks*. For more information, consult the *Adobe Photoshop User Guide*.

This guide does not contain information on creating plug–in modules for Unix versions of Photoshop. The Photoshop Unix SDK is available on the Photoshop Unix product CD. You must purchase the product CD to obtain the SDK.

What Is In This Document

This document has three chapters:

- Chapter 1, “The Photoshop File Format,” describes the Photoshop (**PSD**) native file format in detail.
- Chapter 2, “Other Document File Formats,” discusses Photoshop’s handling of the **EPS** and **TIFF** file formats, which it can also create and read.
- Chapter 3, “Additional File Formats,” describes the formats of other files used by Photoshop to store information about such items as colors, brushes and so forth.

NOTE: For more information about file formats, you may wish to consult the *Encyclopedia of Graphics File Formats* by James D. Murray & William vanRyper (1994, O’Reilly & Associates, Inc., Sebastopol, CA, ISBN 1–56592–058–9).

SDK Discussion Mailing List

The Adobe Solutions Network (ASN) maintains an electronic mailing list that is used as peer discussion for developers. It is unmoderated and is populated by developers just like yourself, offering peer discussion of software development kit, Adobe plug-ins, and related issues. The mailing list is for discussion of all of the SDKs that fall under the ASN. To join the discussion send an e-mail to:

```
sdk-requests@adobe.com
```

with the subject:

```
SUBSCRIBE
```

and these fields in your message body:

1. Your full name:
2. Business name:
3. Address:
4. City:
5. State:
6. Country:
7. Country code or Zip:
8. Area code and phone number (business is fine):
9. ASN member number:
   `"N/A"` if not a member; `"Info"` if want info.

To un-join the discussion send the same email with `UNSUBSCRIBE` in the subject line.

1 The Photoshop File Format

Introduction

This chapter discusses the Photoshop native file format.

TABLE 1.1 Photoshop file types

OS	Filetype/extension
Mac OS	`8BPS`
Windows	`.PSD`

Large Document Format

The Large Document Format (8BPB/PSB) supports documents up to 300,000 pixels in any dimension. All Photoshop features, such as layers, effects, and filters, are supported by the PSB format. The PSB format is identical to the Photoshop native format in many ways. This document will cover the differences found in the PSB format by donating a **PSB** marker.

Windows

All data is stored in big endian byte order. On the Windows platform, you must byte swap short and long integers when reading or writing.

Mac OS

For cross–platform compatibility, all information needed by Photoshop is stored in the data fork. For interoperability with other Macintosh applications, however, some information is duplicated in resources stored in the resource fork of the file:

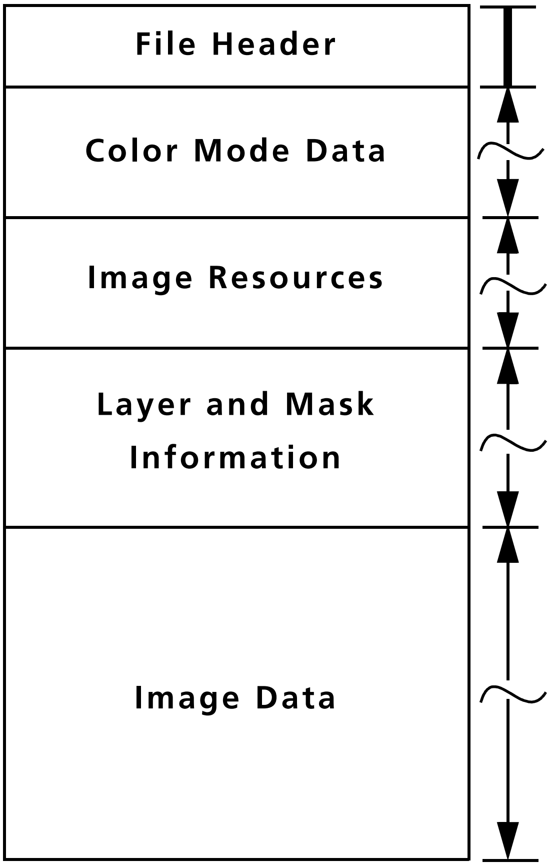
- For compatibility with image cataloging applications, the `'pnot'` resource id 0 contains references to thumbnail, keywords, and caption information stored in other resources.
- The thumbnail picture is stored in a `'PICT'` resource, the keywords are stored in `'STR#'` resource 128 and the caption text is stored in `'TEXT'` resource 128. For more information on the format of these resources see *Inside Macintosh: QuickTime Components* and the *Extensis Fetch Awareness Developer's Toolkit.*
- Photoshop also creates '`icl8`' –16455 and '`ICN#`' –16455 resources containing thumbnail images which will be shown in the Mac OS Finder.

- All of the data from Photoshop's File Info dialog is stored in '**ANPA**' resource 10000. The data in this resource is stored as an IPTC–NAA record 2.For more information on the format of this resource, see the documents in the IPTC folder of the Documentation folder.

Photoshop File Format

The Photoshop file format is divided into five major parts, as shown in Figure 1.1:

Figure 1.1 Photoshop file structure



1. File header (see "File Header Section" on page 10).
2. Color mode data (see "Color Mode Data Section" on page 11)
3. Image resources (see "Image Resources Section" on page 12)
4. Layer and mask information (see "Layer and Mask Information Section" on page 24)
5. Image data (see "Image Data Section" on page 47).

The file header has a fixed length; the other four sections are variable in length.

When writing one of these sections, you should write all fields in the section, as Photoshop may try to read the entire section. Whenever writing a file and skipping bytes, you should explicitly write zeros for the skipped fields.

When reading one of the length-delimited sections, use the length field to decide when you should stop reading. In most cases, the length field indicates the number of bytes, not records, following.

NOTE: The values in “Length” column in all tables are in bytes.

NOTE: All values defined as Unicode strings consist of:

– A 4-byte length field, representing the number of characters in the string (not bytes).
– The string of Unicode values, two bytes per character.

File Header Section

The file header contains the basic properties of the image.

TABLE 1.2 *File header section*

Length	Description
4	Signature: always equal to `'8BPS'`. Do not try to read the file if the signature does not match this value.
2	Version: always equal to 1. Do not try to read the file if the version does not match this value. (****PSB**** version is 2.)
6	Reserved: must be zero.
2	The number of channels in the image, including any alpha channels. Supported range is 1 to 56.
4	The height of the image in pixels. Supported range is 1 to 30,000. (****PSB**** max of 300,000.)
4	The width of the image in pixels. Supported range is 1 to 30,000. **(*PSB**** max of 300,000)
2	Depth: the number of bits per channel. Supported values are 1, 8, and 16.
2	The color mode of the file. Supported values are: Bitmap = 0; Grayscale = 1; Indexed = 2; RGB = 3; CMYK = 4; Multichannel = 7; Duotone = 8; Lab = 9.

Color Mode Data Section

The color mode data section is structured as follows:

TABLE 1.3 Color mode data section

Length	Description
4	The length of the following color data.
Variable	The color data.

Only indexed color and duotone (see the mode field in Table 1.2) have color mode data. For all other modes, this section is just the 4-byte length field, which is set to zero.

- Indexed color images: length is 768; color data contains the color table for the image, in non–interleaved order.
- Duotone images: color data contains the duotone specification (the format of which is not documented). Other applications that read Photoshop files can treat a duotone image as a gray image, and just preserve the contents of the duotone information when reading and writing the file.

Image Resources Section

The third section of the file contains image resources. It starts with a length field, followed by a series of resource blocks.

TABLE 1.4 Image resources section

Length	Description
4	Length of image resource section.
Variable	Image resources (see “Image Resource Blocks”).

Image Resource Blocks

Image resource blocks are the basic building unit of several file formats, including Photoshop’s native file format, JPEG, and TIFF. Image resources are used to store non-pixel data associated with images, such as pen tool paths.

NOTE: They are referred to as resource blocks because they hold data that was stored in the Macintosh’s resource fork in early versions of Photoshop.

The basic structure of image resource blocks is shown in Table 1.5. The last field is the data area, which varies by resource type. The makeup of each resource type is described in the following sections.

TABLE 1.5 Image resource block

Length	Description
2	Signature: **'8BIM'**
2	Unique identifier for the resource. Table 1.6 contains a list of resource IDs used by Photoshop.
Variable	Name: Pascal string, padded to make the size even (a null name consists of two bytes of 0)
4	Actual size of resource data that follows
Variable	The resource data, described in the sections on the individual resource types. It is padded to make the size even.

Image Resource IDs

Image resources use several standard ID numbers, as shown in Table 1.6. Not all file formats use all ID's. Some information may be stored in other sections of the file.

NOTE: For those resource IDs that have been added since Photoshop 3.0. the entry indicates the version in which they were introduced, e.g. (*Photoshop 6.0).*

TABLE 1.6 Image resource IDs

ID		
Hex	**Decimal**	**Description**
0x03E8	1000	(*Obsolete—Photoshop 2.0 only*) Contains five 2-byte values: number of channels, rows, columns, depth, and mode
0x03E9	1001	Macintosh print manager print info record
0x03EB	1003	(*Obsolete—Photoshop 2.0 only*) Indexed color table
0x03ED	1005	**`ResolutionInfo`** structure See Appendix A in `Photoshop API Guide.pdf`.
0x03EE	1006	Names of the alpha channels as a series of Pascal strings.
0x03EF	1007	**`DisplayInfo`** structure See Appendix A in `Photoshop API Guide.pdf`.
0x03F0	1008	The caption as a Pascal string.
0x03F1	1009	Border information Contains a fixed number (2 bytes real, 2 bytes fraction) for the border width, and 2 bytes for border units (1 = inches, 2 = cm, 3 = points, 4 = picas, 5 = columns).
0x03F2	1010	Background color. See "Color Swatches" on page 68.
0x03F3	1011	Print flags A series of one-byte boolean values (see **Page Setup** dialog): labels, crop marks, color bars, registration marks, negative, flip, interpolate, caption, print flags.
0x03F4	1012	Grayscale and multichannel halftoning information
0x03F5	1013	Color halftoning information
0x03F6	1014	Duotone halftoning information
0x03F7	1015	Grayscale and multichannel transfer function
0x03F8	1016	Color transfer functions

TABLE 1.6 Image resource IDs *(Continued)*

ID		
Hex	**Decimal**	**Description**
0x03F9	1017	Duotone transfer functions
0x03FA	1018	Duotone image information
0x03FB	1019	Two bytes for the effective black and white values for the dot range
0x03FC	1020	*(Obsolete)*
0x03FD	1021	EPS options
0x03FE	1022	Quick Mask information 2 bytes containing Quick Mask channel ID; 1- byte boolean indicating whether the mask was initially empty.
0x03FF	1023	*(Obsolete)*
0x0400	1024	Layer state information 2 bytes containing the index of target layer (0 = bottom layer).
0x0401	1025	Working path (not saved) See "Path resource format" on page 20.
0x0402	1026	Layers group information 2 bytes per layer containing a group ID for the dragging groups. Layers in a group have the same group ID.
0x0403	1027	*(Obsolete)*
0x0404	1028	IPTC-NAA record Contains the **File Info...** information. See the documentation in the `IPTC` folder of the `Documentation` folder.
0x0405	1029	Image mode for raw format files
0x0406	1030	JPEG quality. Private.
0x0408	1032	*(Photoshop 4.0)* Grid and guides information See "Grid and guides resource format" on page 17.
0x0409	1033	*(Photoshop 4.0)* Thumbnail resource for Photoshop 4.0 only See "Thumbnail resource format" on page 19.
0x040A	1034	*(Photoshop 4.0)* Copyright flag Boolean indicating whether image is copyrighted. Can be set via Property suite or by user in **File Info...**

TABLE 1.6 ***Image resource IDs*** *(Continued)*

ID Hex	Decimal	Description
0x040B	1035	*(Photoshop 4.0)* URL Handle of a text string with uniform resource locator. Can be set via Property suite or by user in **File Info...**
0x040C	1036	*(Photoshop 5.0)* Thumbnail resource (supersedes resource 1033) See “Thumbnail resource format” on page 19.
0x040D	1037	*(Photoshop 5.0)* Global Angle 4 bytes that contain an integer between 0 and 359, which is the global lighting angle for effects layer. If not present, assumed to be 30.
0x040E	1038	*(Photoshop 5.0)* Color samplers resource See “Color samplers resource format” on page 20.
0x040F	1039	*(Photoshop 5.0)* ICC Profile The raw bytes of an ICC (International Color Consortium) format profile. See `ICC34.pdf` in the `Documentation` folder and `ICC34.h` in `Sample Code\Common\Includes`.
0x0410	1040	*(Photoshop 5.0)* Watermark One byte.
0x0411	1041	*(Photoshop 5.0)* ICC Untagged Profile 1 byte that disables any assumed profile handling when opening the file. 1 = intentionally untagged.
0x0412	1042	*(Photoshop 5.0)* Effects visible 1-byte global flag to show/hide all the effects layer. Only present when they are hidden.
0x0413	1043	*(Photoshop 5.0)* Spot Halftone 4 bytes for version, 4 bytes for length, and the variable length data.
0x0414	1044	*(Photoshop 5.0)* Document-specific IDs seed number 4 bytes: Base value, starting at which layer IDs will be generated (or a greater value if existing IDs already exceed it). Its purpose is to avoid the case where we add layers, flatten, save, open, and then add more layers that end up with the same IDs as the first set.
0x0415	1045	*(Photoshop 5.0)* Unicode Alpha Names Unicode string (4 bytes length followed by string).
0x0416	1046	*(Photoshop 6.0)* Indexed Color Table Count 2 bytes for the number of colors in table that are actually defined

TABLE 1.6 Image resource IDs *(Continued)*

ID Hex	Decimal	Description
0x0417	1047	*(Photoshop 6.0)* Transparency Index. 2 bytes for the index of transparent color, if any.
0x0419	1049	*(Photoshop 6.0)* Global Altitude 4 byte entry for altitude
0x041A	1050	*(Photoshop 6.0)* Slices See "Slices resource format" on page 22.
0x041B	1051	*(Photoshop 6.0)* Workflow URL Unicode string
0x041C	1052	*(Photoshop 6.0)* Jump To XPEP 2 bytes major version, 2 bytes minor version, 4 bytes count. Following is repeated for count: 4 bytes block size, 4 bytes key, if key = **`'jtDd'`**, then next is a Boolean for the dirty flag; otherwise it's a 4 byte entry for the mod date.
0x041D	1053	*(Photoshop 6.0)* Alpha Identifiers 4 bytes of length, followed by 4 bytes each for every alpha identifier.
0x041E	1054	*(Photoshop 6.0)* URL List 4 byte count of URLs, followed by 4 byte long, 4 byte ID, and Unicode string for each count.
0x0421	1057	*(Photoshop 6.0)* Version Info 4 bytes version, 1 byte **`hasRealMergedData`**, Unicode string: writer name, Unicode string: reader name, 4 bytes file version.
0x0422	1058	*(Photoshop 7.0)* EXIF data 1 See http://www.pima.net/standards/it10/PIMA15740/exif.htm
0x0423	1059	*(Photoshop 7.0)* EXIF data 3 See http://www.pima.net/standards/it10/PIMA15740/exif.htm
0x0424	1060	*(Photoshop 7.0)* XMP metadata File info as XML description. See http://Partners.adobe.com/asn/developer/xmp/main.html
0x0425	1061	*(Photoshop 7.0)* Caption digest 16 bytes: RSA Data Security, MD5 message-digest algorithm

TABLE 1.6 Image resource IDs *(Continued)*

ID Hex	Decimal	Description
0x0426	1062	*(Photoshop 7.0)* Print scale 2 bytes style (0 = centered, 1 = size to fit, 2 = user defined). 4 bytes x location (floating point). 4 bytes y location (floating point). 4 bytes scale (floating point)
0x0428	1064	*(Photoshop CS)* Pixel Aspect Ratio 4 bytes (version = 1), 8 bytes double, x / y of a pixel
0x0429	1065	*(Photoshop CS)* Layer Comps 4 bytes (descriptor version = 16), Descriptor (see "Descriptor structure" on page 57)
0x042A	1066	*(Photoshop CS)* Alternate Duotone Colors 2 bytes (version = 1), 2 bytes count, following is repeated for each count: [Color: 2 bytes for space followed by 4 * 2 byte color component], following this is another 2 byte count, usually 256, followed by Lab colors one byte each for L, a, b *This resource is not read or used by Photoshop.*
0x042B	1067	*(Photoshop CS)* Alternate Spot Colors 2 bytes (version = 1), 2 bytes channel count, following is repeated for each count: 4 bytes channel ID, Color: 2 bytes for space followed by 4 * 2 byte color component *This resource is not read or used by Photoshop.*
0x07D0- 0x0BB6	2000- 2998	Path Information (saved paths) See "Path resource format" on page 20.
0x0BB7	2999	Name of clipping path See "Path resource format" on page 20.
0x2710	10000	Print flags information 2 bytes version (= 1), 1 byte center crop marks, 1 byte (= 0), 4 bytes bleed width value, 2 bytes bleed width scale.

The following sections describe some of the resource formats in more detail.

Grid and guides resource format

Photoshop stores grid and guides information for an image in an image resource block. Each of these resource blocks consists of an initial 16-byte grid and guide header, which is always present, followed by 5-byte blocks of specific guide information for guide direction and location, which are present if there are guides (`fGuideCount` > 0) .

TABLE 1.7 Grid and guide header

Length	Description
4	Version (= 1)
8	Future implementation of document-specific grids (4 bytes horizontal, 4 bytes vertical). Currently, sets the grid cycle to every quarter inch, i.e. 576 for both horizontal & vertical (at 72 dpi, that is 18 * 32 = 576)
4	`fGuideCount`: Number of guide resource blocks (can be 0).

TABLE 1.8 Guide resource block

Length	Description
4	Location of guide in document coordinates. Since the guide is either vertical or horizontal, this only has to be one component of the coordinate.
1 VHSelect	Direction of guide. VHSelect is a system type of **`unsigned char`** where 0 = vertical, 1 = horizontal.

Grid and guide information may be modified using the Property suite. See the Callbacks chapter in `Photoshop API Guide.pdf` for more information.

Thumbnail resource format

Adobe Photoshop (version 5.0 and later) stores thumbnail information for preview display in an image resource block that consists of an initial 28-byte header, followed by a JFIF thumbnail in RGB (red, green, blue) order for both Macintosh and Windows.

NOTE: Adobe Photoshop 4.0 stored the thumbnail information in the same format except the data section is BGR (blue, green, red). The 4.0 format is at resource ID 1033 and the 5.0 format is at resource ID 1036.

TABLE 1.9 Thumbnail resource header

Length	Description
4	Format. 1 = kJpegRGB . Also supports kRawRGB (0).
4	Width of thumbnail in pixels.
4	Height of thumbnail in pixels.
4	Widthbytes: Padded row bytes = (width * bits per pixel + 31) / 32 * 4.
4	Total size = widthbytes * height * planes
4	Size after compression. Used for consistency check.
2	Bits per pixel. = 24
2	Number of planes. = 1
Variable	JFIF data in RGB format. **NOTE:** For resource ID 1033 the data is in BGR format.

Color samplers resource format

Adobe Photoshop (version 5.0 and later) stores color samplers information for an image in an image resource block that consists of an initial 8-byte color samplers header followed by a variable length block of specific color samplers information.

TABLE 1.10 ***Color Samplers header***

Length	Description
4	Version (= 1)
4	Number of color samplers to follow. See Table 1.11.

TABLE 1.11 ***Color Samplers resource block***

Length	Description
8	The vertical and horizontal position of the point (4 bytes each).
2	Color Space: `enum { kDummySpace = -1, kRGBSpace, kHSBSpace, kCMYKSpace, kPantoneSpace, kFocoltoneSpace, kTrumatchSpace,kToyoSpace, kLabSpace, kGraySpace, kWideCMYKSpace,kHKSSpace, kDICSpace, kTotalInkSpace, kMonitorRGBSpace, kDuotoneSpace, kOpacitySpace};`

Path resource format

Photoshop stores the paths saved with an image in an image resource block. These resource blocks consist of a series of 26-byte path point records, so the resource length should always be a multiple of 26.

Photoshop stores its paths as resources of type **8BIM**, with IDs in the range 2000 through 2999. These numbers should be reserved for Photoshop. The name of the resource is the name given to the path when it was saved.

If the file contains a resource of type **8BIM** with an ID of 2999, then this resource contains a Pascal–style string containing the name of the clipping path to use with this image when saving it as an EPS file.

The path format returned by **GetProperty()** call is identical to what is described below. Refer to the IllustratorExport sample plug–in code to see how this resource data is constructed.

Path points

All points used in defining a path are stored in eight bytes as a pair of 32–bit components, vertical component first.

The two components are signed, fixed point numbers with 8 bits before the binary point and 24 bits after the binary point. Three guard bits are reserved in the points to eliminate most concerns over arithmetic overflow. Hence, the range for each component is `0xF0000000` to `0x0FFFFFFF` representing a range of –16 to 16. The lower bound is included, but not the upper bound.

This limited range is used because the points are expressed relative to the image size. The vertical component is given with respect to the image height, and the horizontal component is given with respect to the image width. [`0,0`] represents the top–left corner of the image; [`1,1`] ([`0x01000000,0x01000000`]) represents the bottom–right.

In Windows, the byte order of the path point components are reversed; you should swap the bytes when accessing each 32–bit value.

Path records

The data in a path resource consists of one or more 26-byte records. The first two bytes of each record is a selector to indicate what kind of path it is. For Windows, you should swap the bytes before accessing it as a short.

TABLE 1.12 Path data record types

Selector	Description
0	Closed subpath length record
1	Closed subpath Bezier knot, linked
2	Closed subpath Bezier knot, unlinked
3	Open subpath length record
4	Open subpath Bezier knot, linked
5	Open subpath Bezier knot, unlinked
6	Path fill rule record
7	Clipboard record
8	Initial fill rule record

The first 26-byte path record contains a selector value of 6, path fill rule record. The remaining 24 bytes of the first record are zeroes. Paths use even/odd ruling. Subpath length records, selector value 0 or 3, contain the number of Bezier knot records in bytes 2 and 3. The remaining 22 bytes are unused, and should be zeroes. Each length record is then immediately followed by the Bezier knot records describing the knots of the subpath.

In Bezier knot records, the 24 bytes following the selector field contain three path points (described above) for:

1. the control point for the Bezier segment preceding the knot,
2. the anchor point for the knot, and
3. the control point for the Bezier segment leaving the knot.

Linked knots have their control points linked. Editing one point modifies the other to preserve collinearity. Knots should only be marked as having linked controls if their control points are collinear with their anchor. The control points on unlinked knots are independent of each other. Refer to the *Adobe Photoshop User Guide* for more information.

Clipboard records, **`selector=7`**, contain four fixed-point numbers for the bounding rectangle (top, left, bottom, right), and a single fixed-point number indicating the resolution.

Initial fill records, **`selector=8`**, contain one two byte record. A value of 1 means that the fill starts with all pixels. The value will be either 0 or 1.

Slices resource format

Adobe Photoshop 6.0 and later stores slices information for an image in an image resource block. .

TABLE 1.13 Slices header

Length	Description
4	Version (= 6)
4 * 4	Bounding rectangle for all of the slices: top, left, bottom, right of all the slices
Variable	Name of group of slices: Unicode string
4	Number of slices to follow. See Slices resource block in the next table.

TABLE 1.14 Slices resource block

Length	Description
4	ID
4	Group ID
4	Origin
4	Associated Layer ID **NOTE:** Only present if Origin = 1
Variable	Name: Unicode string
4	Type
4 * 4	Left, top, right, bottom positions
Variable	URL: Unicode string
Variable	Target: Unicode string
Variable	Message: Unicode string
Variable	Alt Tag: Unicode string
1	Cell text is HTML: Boolean
Variable	Cell text: Unicode string
4	Horizontal alignment
4	Vertical alignment
1	Alpha color
1	Red
1	Green
1	Blue

Layer and Mask Information Section

The fourth section of a Photoshop file contains information about layers and masks. This section of the document describes the formats of layer and mask records.

NOTE: The complete, merged image data is not stored here; it resides in the last section of the file. See "Image Data Section" on page 47.

Table 1.15 shows the overall structure of this section. If there are no layers or masks, this section is just 4 bytes: the length field, which is set to zero.

TABLE 1.15 Layer and mask information section

Length	Description
4	Length of the layer and mask information section. **(**PSB** length is 8** bytes.)
Variable	Layer info (see Table 1.16 for details).
Variable	Global layer mask info (see Table 1.21 for details).

Table 1.16 shows the high-level organization of the layer information.

TABLE 1.16 Layer info

Length	Description
4	Length of the layers info section, rounded up to a multiple of 2. **(**PSB**** length is 8 bytes.)
2	Layer count. If it is a negative number, its absolute value is the number of layers and the first alpha channel contains the transparency data for the merged result.
Variable	Information about each layer. Table 1.17 describes the structure of this information for each layer.
Variable	Channel image data. Contains one or more image data records (see Table 1.20 for structure) for each layer. The layers are in the same order as in the layer information (previous row of this table).

TABLE 1.17 Layer records

Length	Description
4 * 4	Rectangle containing the contents of the layer. Specified as top, left, bottom, right coordinates
2	Number of channels in the layer
6 * number of channels	Channel information. Six bytes per channel, consisting of: 2 bytes for Channel ID: 0 = red, 1 = green, etc.; –1 = transparency mask; –2 = user supplied layer mask 4 bytes for length of corresponding channel data. (**PSB** 8 bytes for length of corresponding channel data.) See Table 1.20 for structure of channel data.
4	Blend mode signature: **'8BIM'**
4	Blend mode key: '**norm**' = normal, '**dark**' = darken, '**lite**' = lighten, '**hue** ' = hue, '**sat** ' = saturation, '**colr**' = color, '**lum** ' = luminosity, '**mul** ' = multiply, '**scrn**' = screen, '**diss**' = dissolve, '**over**' = overlay, '**hLit**' = hard light, '**sLit**' = soft light, '**diff**' = difference, **'smud'** = exclusion, **'div '** = color dodge, **'idiv'** = color burn, **'lbrn'** = linear burn, **'lddg'** = linear dodge, **'vLit'** = vivid light, **'lLit'** = linear light, **'pLit'** = pin light, **'hMix'** = hard mix
1	Opacity. 0 = transparent ... 255 = opaque
1	Clipping: 0 = base, 1 = non–base

1	Flags:
	bit 0 = transparency protected; bit 1 = visible; bit 2 = obsolete;
	bit 3 = 1 for Photoshop 5.0 and later, tells if bit 4 has useful information;
	bit 4 = pixel data irrelevant to appearance of document

1	Filler (zero)
4	Length of the extra data field (= the total length of the next five fields).
Variable	Layer mask data: See Table 1.18 for structure. Can be 40 bytes, 24 bytes, or 4 bytes if no layer mask.
Variable	Layer blending ranges: See Table 1.19.
Variable	Layer name: Pascal string, padded to a multiple of 4 bytes.

TABLE 1.17 *Layer records* *(Continued)*

Length	Description
Variable	*(Photoshop 4.0, 5.0, 6.0, 7.0)* Series of tagged blocks containing various types of data. See "Additional Layer Information" on page 28 for the list of the types of data that can be included here.

TABLE 1.18 *Layer mask / adjustment layer data*

Length	Name
4	Size of the data: 36, 20, or 0. If zero, the following fields are not present
4 * 4	Rectangle enclosing layer mask: Top, left, bottom, right
1	Default color. 0 or 255
1	Flags. bit 0 = position relative to layer bit 1 = layer mask disabled bit 2 = invert layer mask when blending
2	Padding. Only present if size = 20. Otherwise the following is present
1	Real Flags. Same as Flags information above.
1	Real user mask background. 0 or 255. Same as Flags information above.
4 * 4	Rectangle enclosing layer mask: Top, left, bottom, right.

TABLE 1.19 *Layer blending ranges data*

Length	Name
4	Length of layer blending ranges data
4	Composite gray blend source. Contains 2 black values followed by 2 white values. Present but irrelevant for Lab & Grayscale.
4	Composite gray blend destination range
4	First channel source range
4	First channel destination range
4	Second channel source range
4	Second channel destination range
...	...

TABLE 1.19 ***Layer blending ranges data****(Continued)*

Length	Name
4	Nth channel source range
4	Nth channel destination range

TABLE 1.20 ***Channel image data***

Length	Description
2	Compression. 0 = Raw Data, 1 = RLE compressed, 2 = ZIP without prediction, 3 = ZIP with prediction.
Variable	Image data. If the compression code is 0, the image data is just the raw image data, whose size is calculated as **`(LayerBottom-LayerTop)*`** **`(LayerRight-LayerLeft)`** (from the first field in Table 1.17). If the compression code is 1, the image data starts with the byte counts for all the scan lines in the channel **`(LayerBottom-LayerTop)`**, with each count stored as a two–byte value.(**PSB** each count stored as a four-byte value.) The RLE compressed data follows, with each scan line compressed separately. The RLE compression is the same compression algorithm used by the Macintosh ROM routine PackBits, and the TIFF standard. If the layer's size, and therefore the data, is odd, a pad byte will be inserted at the end of the row. If the layer is an adjustment layer, the channel data is undefined (probably all white.)

TABLE 1.21 ***Global layer mask info***

Length	Description
4	Length of global layer mask info section.
2	Overlay color space (undocumented).
8	4 * 2 byte color components
2	Opacity. 0 = transparent, 100 = opaque.
1	Kind. 0 = Color selected—i.e. inverted; 1 = Color protected;128 = use value stored per layer. This value is preferred. The others are for backward compatibility with beta versions.
Variable	Filler: zeros

Additional Layer Information

There are several types of layer information that have been added in Photoshop 4.0 and later. These exist at the end of the layer records structure (see the last row of Table 1.17). They have the following structure:

TABLE 1.22 Additional layer information

Length bytes)	Description
4	Signature: `'8BIM'`
4	Key: a 4-character code (See individual sections)
4	Length data below, rounded up to an even byte count. (**PSB**, the following keys have a length count of 8 bytes: LMsk, Lr16, Layr, Mt16, Mtrn, Alph.
Variable	Data (See individual sections)

The following sections describe the different types of data available, their keys and their format.

Adjustment layer *(Photoshop 4.0)*

Adjustment layers can have one of the following keys:

- '**`levl`**' = Levels
- '**`curv`**' = Curves
- '**`brit`**' = Brightness/contrast
- '**`blnc`**' = Color balance
- '**`hue `**' = Old Hue/saturation, Photoshop 4.0
- '**`hue2`**' = New Hue/saturation, Photoshop 5.0
- '**`selc`**' = Selective color
- '**`thrs`**' = Threshold
- '**`nvrt`**' = Invert
- '**`post`**' = Posterize

The data for the adjustment layer is the same as the load file formats for each format. See Chapter 3, “Additional File Formats,” for information.

Effects Layer *(Photoshop 5.0)*

The key for the effects layer is **'lrFX'**. The data has the following format:

TABLE 1.23 ***Effects Layer info***

Length	Description
2	Version: 0
2	Effects count: may be 6 (for the 6 effects in Photoshop 5 and 6) or 7 (for Photoshop 7.0)
The next three items are repeated for each of the effects.	
4	Signature: **'8BIM'**
4	Effects signatures: OSType key for which effects type to use: **'cmnS'** = common state (see Table 1.24) **'dsdw'** = drop shadow (see Table 1.25) **'isdw'** = inner shadow (see Table 1.25) **'oglw'** = outer glow (see Table 1.26) **'iglw'** = inner glow (see Table 1.27) **'bevl'** = bevel (see Table 1.28) **'sofi'** = solid fill (*Photoshop 7.0)* (see Table 1.29)
Variable	See appropriate tables.

TABLE 1.24 ***Effects layer, common state info***

Length	Description
4	Size of next three items: 7
4	Version: 0
1	Visible: always true
2	Unused: always 0

TABLE 1.25 *Effects layer, drop shadow and inner shadow info*

Length	Description
4	Size of the remaining items: 41 or 51 (depending on version)
4	Version: 0 (*Photoshop 5.0)* or 2 (*Photoshop 5.5)*
4	Blur value in pixels
4	Intensity as a percent
4	Angle in degrees
4	Distance in pixels
10	Color: 2 bytes for space followed by 4 * 2 byte color component
8	Blend mode: 4 bytes for signature and 4 bytes for key
1	Effect enabled
1	Use this angle in all of the layer effects
1	Opacity as a percent
10	Native color: 2 bytes for space followed by 4 * 2 byte color component

TABLE 1.26 *Effects layer, outer glow info*

Length	Description
4	Size of the remaining items: 32 for Photoshop 5.0; 42 for 5.5
4	Version: 0 for Photoshop 5.0; 2 for 5.5
4	Blur value in pixels.
4	Intensity as a percent
10	Color: 2 bytes for space followed by 4 * 2 byte color component
8	Blend mode: 4 bytes for signature and 4 bytes for the key
1	Effect enabled
1	Opacity as a percent
10	*(Version 2 only)* Native color space. 2 bytes for space followed by 4 * 2 byte color component

TABLE 1.27 Effects layer, inner glow info

Length	Description
4	Size of the remaining items: 33 for Photoshop 5.0; 43 for 5.5
4	Version: 0 for Photoshop 5.0; 2 for 5.5.
4	Blur value in pixels.
4	Intensity as a percent
10	Color: 2 bytes for space followed by 4 * 2 byte color component
8	Blend mode: 4 bytes for signature and 4 bytes for the key
1	Effect enabled
1	Opacity as a percent
Remaining fields present only in version 2	
1	Invert
10	*(Version 2 only)* Native color space. 2 bytes for space followed by 4 * 2 byte color component

TABLE 1.28 Effects layer, bevel info

Length	Description
4	Size of the remaining items (58 for version 0, 78 for version 20
4	Version: 0 for Photoshop 5.0; 2 for 5.5
4	Angle in degrees
4	Strength. Depth in pixels
4	Blur value in pixels.
8	Highlight blend mode: 4 bytes for signature and 4 bytes for the key
8	Shadow blend mode: 4 bytes for signature and 4 bytes for the key
10	Highlight color: 2 bytes for space followed by 4 * 2 byte color component
10	Shadow color: 2 bytes for space followed by 4 * 2 byte color component
1	Bevel style
1	Hightlight opacity as a percent

TABLE 1.28 *Effects layer, bevel info*

Length	Description
1	Shadow opacity as a percent
1	Effect enabled
1	Use this angle in all of the layer effects
1	Up or down
The following are present in version 2 only	
10	Real highlight color: 2 bytes for space; 4 * 2 byte color component
10	Real shadow color: 2 bytes for space; 4 * 2 byte color component

TABLE 1.29 *Effects layer, solid fill (added in Photoshop 7.0)*

Length	Description
4	Size: 34
4	Version: 2
4	Key for blend mode
10	Color space
1	Opacity
1	Enabled
10	Native color space

Type Tool Info *(Photoshop 5.0 and 5.5 only)*

NOTE: Has been superseded in Photoshop 6.0 and beyond by a different structure with the key **'TySh'** (see Table 1.49).

Key is '**tySh**'. Data is as follows:

TABLE 1.30 Type tool Info

Length	Description
2	Version (= 1)
48	6 * 8 double precision numbers for the transform information
Font information	
2	Version (= 6)
2	Count of faces
The next 8 fields are repeated for each count specified above	
2	Mark value
4	Font type data
Variable	Pascal string of font name
Variable	Pascal string of font family name
Variable	Pascal string of font style name
2	Script value
4	Number of design axes vector to follow
4	Design vector value
Style information	
2	Count of styles
The next 10 fields are repeated for each count specified above	
2	Mark value
2	Face mark value
4	Size value
4	Tracking value
4	Kerning value
4	Leading value

TABLE 1.30 Type tool Info

Length	Description
4	Base shift value
1	Auto kern on/off
1	Only present in version <= 5
1	Rotate up/down
Text information	
2	Type value
4	Scaling factor value
4	Sharacter count value
4	Horizontal placement
4	Vertical placement
4	Select start value
4	Select end value
2	Line count, i.e. the number of items to follow.
The next 5 fields are repeated for each item in line count.	
4	Character count value
2	Orientation value
2	Alignment value
2	Actual character as a double byte character
2	Style value
Color information	
2	Color space value
8	4 * 2 byte color component
1	Anti alias on/off

Unicode layer name *(Photoshop 5.0)*

Key is `'luni'`. Data is as follows:

TABLE 1.31 Unicode Layer name

Length	Description
Variable	Unicode string (4 bytes length + string).

Layer ID *(Photoshop 5.0)*

Key is `'lyid'`.

TABLE 1.32 Layer ID

Length	Description
4	Signature: `'8BIM'`
4	Key: `'lyid'`
4	Length: 4
4	ID.

Object-based effects layer info *(Photoshop 6.0)*

Key is `'lfx2'`. Data is as follows:

TABLE 1.33 Object Based Effects Layer info

Length	Description
4	Object effects version: 0
4	Descriptor version (= 16 for Photoshop 6.0).
Variable	Descriptor (see "Descriptor structure" on page 57)

Patterns *(Photoshop 6.0 and CS (8.0))*

This is a list of patterns. Key is `'Patt'` or `'Pat2'`. Data is as follows:

TABLE 1.34 Patterns

Length	Description
The following is repeated for each pattern.	
4	Length of this pattern
4	Version (=1)
4	The image mode of the file. Supported values are: Bitmap = 0; Grayscale = 1; Indexed = 2; RGB = 3; CMYK = 4; Multichannel = 7; Duotone = 8; Lab = 9.
4	Point: vertical, 2 bytes and horizontal, 2 bytes
Variable	Name: Unicode string
Variable	Unique ID for this pattern: Pascal string
Variable	Index color table (256 * 3 RGB values): only present when image mode is indexed color
The following is the pattern data. It is called a *virtual memory array list.*	
4	Version
4	Length
16	Rectangle: top, left, bottom, right
4	Max channels
The following is a *virtual memory array*, repeated for the number of channels in the image mode, not to exceed the max channels.	
4	Boolean indicating whether array is written
4	Length
4	Pixel depth: 1, 8 or 16
1	Compression mode of data to follow. 'Pat2' compression is zip.
Variable	Actual data based on parameters and compression

Annotations *(Photoshop 6.0)*

Key is `'Anno'`. Data is as follows:

TABLE 1.35 Annotations

Length	Description
2	Major version (= 2)
2	Minor version. (= 1)
4	Count of annotations to follow
Following is repeated for each annotation	
4	Length of this annotation
4	Annotation type: either text(`'txtA'`) or sound (`'sndA'`).
1	Is the annotation open
1	Flags.
2	Optional blocks. (=1 for Photoshop 6.0)
16	Rectangle of icon location: top, left, bottom and right.
16	Rectangle of popup locations: top, left, bottom and right
10	2 bytes for space followed by 4 * 2 byte color component
Variable	Pascal string of author’s name aligned to 2 bytes
Variable	Pascal string of name aligned to 2 bytes
Variable	Pascal string of the mod Date aligned to 2 bytes
4	Length of the following 3 fields including this field
4	'`txtC`' or '`sndM`'. Either text or sound
4	Length of the next field
Variable	Actual data for this annotation. The text is an ASCII or Unicode string; the sound annotation is documented in the *PDF Reference*, available at http://Partners.adobe.com/asn/developer/acrosdk/docs.html#filefmtspecs
Variable	Padding to align to multiple of 4 bytes

Blend clipping elements *(Photoshop 6.0)*

Key is `'clbl'`. Data is as follows:

TABLE 1.36 Blend clipping elements

Length	Description
1	Blend clipped elements: boolean
3	Padding

Blend interior elements *(Photoshop 6.0)*

Key is`'infx'`. Data is as follows:

TABLE 1.37 Blend interior elements

Length	Description
1	Blend interior elements: boolean
3	Padding

Knockout setting *(Photoshop 6.0)*

Key is `'knko'`. Data is as follows:

TABLE 1.38 Knockout setting

Length	Description
1	Knockout: boolean
3	Padding

Protected setting *(Photoshop 6.0)*

Key is `'lspf'`. Data is as follows:

TABLE 1.39 Protected setting

Length	Description
4	Protection flags: bits 0 - 2 are used for Photoshop 6.0. Transparency, composite and position respectively.

Sheet color setting *(Photoshop 6.0)*

Key is `'lclr'`. Data is as follows:

TABLE 1.40 Sheet Color setting

Length	Description
4 * 2	Color. Only the first color setting is used for Photoshop 6.0; the rest are zeros

Reference point *(Photoshop 6.0)*

Key is `'fxrp'`. Data is as follows:

TABLE 1.41 Reference point

Length	Description
2 * 8	2 double values for the reference point

Gradient settings *(Photoshop 6.0)*

Key is `'grdm'`. Data is as follows:

TABLE 1.42 Gradient settings

Length	Description
2	Version (=1 for Photoshop 6.0)
1	Is gradient reversed
1	Is gradient dithered
Variable	Name of the gradient: Unicode string, padded
2	Number of color stops to follow
Following is repeated for each color stop	
4	Location of color stop
4	Midpoint of color stop
2	Mode for the color to follow
4 * 2	Actual color for the stop
2	Number of transparency stops to follow
Following is repeated for each transparency stop	

TABLE 1.42 Gradient settings

Length	Description
4	Location of transparency stop
4	Midpoint of transparency stop
2	Opacity of transparency stop
2	Expansion count (= 2 for Photoshop 6.0)
2	Interpolation if length above is non-zero
2	Length (= 32 for Photoshop 6.0)
2	Mode for this gradient
4	Random number seed
2	Flag for showing transparency
2	Flag for using vector color
4	Roughness factor
2	Color model
4 * 2	Minimum color values
4 * 2	Maximum color values
2	Dummy: not used in Photoshop 6.0

Section divider setting *(Photoshop 6.0)*

Key is `'lsct'`. Data is as follows:

TABLE 1.43 Section Divider setting

Length	Description
4	Type. 4 possible values, 0 = any other type of layer, 1 = open “folder”, 2 = closed “folder”, 3 = bounding section divider, hidden in the UI
Following is only present if length = 12	
4	Signature: `'8BIM'`
4	Key. See blend mode keys in Table 1.17.

Channel blending restrictions setting *(Photoshop 6.0)*

Key is `'brst'`. Data is as follows:

TABLE 1.44 Channel blending restrictions setting

Length	Description
Following is repeated length / 4 times.	
4	Channel number that is restricted

Solid color sheet setting *(Photoshop 6.0)*

Key is `'SoCo'`. Data is as follows:

TABLE 1.45 Solid color sheet setting

Length	Description
4	Version (= 16 for Photoshop 6.0)
Variable	Descriptor. Based on the Action file format structure (see “Descriptor structure” on page 57)

Pattern fill setting *(Photoshop 6.0)*

Key is `'PtFl'`. Data is as follows:

TABLE 1.46 Pattern fill setting

Length	Description
4	Version (=16 for Photoshop 6.0)
Variable	Descriptor. Based on the Action file format structure (see “Descriptor structure” on page 57)

Gradient fill setting *(Photoshop 6.0)*

Key is `'GdFl'`. Data is as follows:

TABLE 1.47 Gradient Fill Setting

Length	Description
4 bytes	Version (= 16 for Photoshop 6.0)
Variable	Descriptor. Based on the Action file format structure (see “Descriptor structure” on page 57)

Vector mask setting *(Photoshop 6.0)*

Key is **'vmsk'**. Data is as follows:

Table 1.48 Vector mask setting

Length	Description
4	Version (= 3 for Photoshop 6.0)
4	Flags. bit 1 = invert, bit 2 = not link, bit 3 = disable
The rest of the data is path components, loop until end of the length.	
Variable	Paths. See "Path resource format" on page 20

Type tool object setting *(Photoshop 6.0)*

Note: This supersedes the type tool info in Photoshop 5.0 (see Table 1.30).

Key is **'TySh'**. Data is as follows:

Table 1.49 Type tool object setting

Length	Description
2	Version (=1 for Photoshop 6.0)
6 * 8	Transform: xx, xy, yx, yy, tx, and ty respectively.
2	Text version (= 50 for Photoshop 6.0)
4	Descriptor version (= 16 for Photoshop 6.0)
Variable	Text data (see "Descriptor structure" on page 57)
2	Warp version (= 1 for Photoshop 6.0)
4	Descriptor version (= 16 for Photoshop 6.0)
Variable	Warp data (see "Descriptor structure" on page 57)
4 * 8	left, top, right, bottom respectively.

Foreign effect ID *(Photoshop 6.0)*

Key is **'ffxi'**. Data is as follows:

Table 1.50 Foreign effect ID

Length	Description
4	ID of the Foreign effect.

Layer name source setting *(Photoshop 6.0)*

Key is **'lnsr'**. Data is as follows:

TABLE 1.51 Layer name source setting

Length	Description
4	ID for the layer name

Pattern data *(Photoshop 6.0)*

Key is **'shpa'**. Data is as follows:

TABLE 1.52 Pattern data

Length	Description
4	Version (= 0 for Photoshop 6.0)
4	Count of sets to follow
The following is repeated for the count above.	
4	Pattern signature
4	Pattern key
4	Count of patterns in this set
1	Copy on sheet duplication
3	Padding
The following is repeated for the count of patterns above.	
4	Color handling. Prefer convert = **'conv'**, avoid conversion = **'avod'**, luminance only = **'lumi'**
Variable	Pascal string name of the pattern
Variable	Unicode string name of the pattern
Variable	Pascal string of the unique identifier for the pattern

Metadata setting *(Photoshop 6.0)*

Key is **'shmd'**. Data is as follows:

TABLE 1.53 Metadata setting

Length	Description
4	Count of metadata items to follow
The following is repeated the number of times specified by the count above:	
4	Signature of the data
4	Key of the data
1	Copy on sheet duplication
3	Padding
4	Length of data to follow
Variable	Undocumented data

Layer version *(Photoshop 7.0)*

Key is **'lyvr'**. Data is as follows:

TABLE 1.54 Layer version

Length	Description
4	A 32-bit number representing the version of Photoshop needed to read and interpret the layer without data loss. 70 = 7.0, 80 = 8.0, etc. **NOTE:** The minimum value is 70, because just having the field present in 6.0 triggers a warning. For the future, Photoshop 7 checks to see whether this number is larger than the current version -- i.e., 70 -- and if so, warns that it is ignoring some data.

Transparency shapes layer *(Photoshop 7.0)*

Key is `'tsly'`. Data is as follows:

TABLE 1.55 Transparency shapes layer

Length	Description
1	1: the transparency of the layer is used in determining the shape of the effects. This is the default for behavior like previous versions. 0: treated in the same way as fill opacity including modulating blend modes, rather than acting as strict transparency. Using this feature is useful for achieving effects that otherwise would require complex use of clipping groups.
3	Padding

Layer mask as global mask *(Photoshop 7.0)*

Key is `'lmgm'`. Data is as follows:

TABLE 1.56 Layer mask as global mask

Length	Description
1	1: the layer mask is used in a final crossfade masking the layer and effects rather than being used to shape the layer and its effects. This behavior was previously tied to the link status flag for the layer mask. (An unlinked mask acted like a flag value of 1, a linked mask like 0). For old files that lack this key, the link status is used in order to preserve compositing results.
3	Padding

Vector mask as global mask *(Photoshop 7.0)*

Key is `'vmgm'`. Data is as follows:

TABLE 1.57 Vector mask as global mask

Length	Description
1	Same as in Table 1.56, but applying the vector mask.
3	Padding

Brightness and Contrast

Key is **'brit'**. Data is as follows:

TABLE 1.58 Brightness and Contrast

Length	Description
2	Brightness
2	Contrast
2	Mean value for brightness and contrast
1	Lab color only

Channel Mixer

Key is **'mixr'**. Data is as follows:

TABLE 1.59 Channel Mixer

Length	Description
2	Version (= 1)
2	Monochrome
20	RGB or CMYK color plus constant for the mixer settings. 4 * 2 bytes of color with 2 bytes of constant.

Photo Filter

Key is **'phfl'**. Data is as follows:

TABLE 1.60 Photo Filter

Length	Description
2	Version (= 3)
12	4 bytes each for XYZ color
4	Density
1	Preserve Luminosity

Image Data Section

The last section of a Photoshop file contains the image pixel data. Image data is stored in planar order: first all the red data, then all the green data, etc. Each plane is stored in scan-line order, with no pad bytes,

TABLE 1.61 Image data section

Length	Description
2	Compression method: 0 = Raw image data 1 = RLE compressed the image data starts with the byte counts for all the scan lines (rows * channels), with each count stored as a two–byte value. The RLE compressed data follows, with each scan line compressed separately. The RLE compression is the same compression algorithm used by the Macintosh ROM routine **`PackBits`**, and the TIFF standard. 2 = ZIP without prediction 3 = ZIP with prediction.
Variable	The image data. Planar order = RRR GGG BBB, etc.

2 Other Document File Formats

Photoshop EPS files

The following summarizes the additional information Photoshop writes when creating EPS files:

- Photoshop writes a high–resolution bounding box comment to the EPS file immediately following the traditional EPS bounding box comment. The comment begins with “**`%%HiResBoundingBox`**” and is followed by four numbers identical to those given for the bounding box except that they can have fractional components (i.e., a decimal point and digits after it). The traditional bounding box is written as the rounded version of the high resolution bounding box for compatibility.
- Photoshop writes its image resources out to a block of data stored as follows:

```
%BeginPhotoshop: <length> <hex data>
```

TABLE 2.1 EPS parameters for BeginPhotoshop

Field	Definition
length	Length of the image resource data.
hex data	Image resource data in hexadecimal.

- Photoshop includes a comment in the EPS files it writes so that it is able to read them back in again. Third party programs that write pixel–based EPS files may want to include this comment in their EPS files, so Photoshop can read their files.

  The comment must follow immediately after the **`%%`** comment block at the start of the file. The comment is:

```
%ImageData: <columns> <rows> <depth> <mode> <pad channels> <block size>
<binary/hex> "<data start>"
```

TABLE 2.2 EPS parameters for ImageData

Field	Definition
columns	Width of the image in pixels.
rows	Height of the image in pixels.
depth	Number of bits per channel. Must be 1 or 8.
mode	Image mode. Bitmap/grayscale = 1; Lab = 2; RGB = 3; CMYK = 4.
pad channels	Number of other channels store in the file. Ignored when reading. Photoshop uses this to include a grayscale image that is printed on non-color PostScript printers.
block size	Number of bytes per row per channel. Will be either 1 or formula (below): 1 = Data is interleaved. `(columns*depth+7)/8`=Data is stored in line-interleaved format, or there is only one channel.
binary/ascii	1 = Data is in binary format. 2 = Data is in hex ascii format.
data start	Entire PostScript line immediately preceding the image data. This entire line should not occur elsewhere in the PostScript header code, butit may occur at part of a line.

TIFF files

Table 2.3 describes the standard TIFF (version 6) tags and tag values that Photoshop is able to read and write. Photoshop reads the first Image File Directory (IFD) and writes one IFD per file.

In addition, Photoshop uses a set of tags that are not defined in the TIFF v6 specification to store specific information. See "Photoshop-specific TIFF Tags" on page 53.

See "TIFF Files on Mac OS" on page 53 for information about how TIFF files are stored on Macintosh.

TABLE 2.3 TIFF Tags

Tag	Name	Photoshop reads	Photoshop writes
254	NewSubFileType	Ignored	0
256	ImageWidth	1 to 30000	1 to 30000
257	ImageLength	1 to 30000	1 to 30000
258	BitsPerSample	1, 2, 4, 8, 16 (all same)	1, 8, 16
259	Compression	1 (uncompressed), 2 (CCITT), 5 (LZW), 7 (JPEG), 8 (ZIP), 32773 (PackBits)	1, 5, 7, 8
262	PhotometricInterpretation	0, 1, 2, 3, 5, 8, 9	0 (1–bit), 1 (8–bit), 2, 3,5,8
266	FillOrder	1	No
270	ImageDescription	Printing Caption	Printing Caption
271	EXIF: Make		
272	EXIF: Model		
273	StripOffsets	Yes	Yes
277	SamplesPerPixel	1 to 24	1 to 24
278	RowsPerStrip	Any	Single strip if not compressed, multiple strips if compressed.
279	StripByteCounts	Required if compressed	Yes
282	XResolution	Yes	Yes
283	YResolution	Ignored (square pixels assumed)	Yes
284	PlanarConfiguration	1 or 2	1

TABLE 2.3 TIFF Tags

Tag	Name	Photoshop reads	Photoshop writes
296	ResolutionUnit	2 or 3	2
305	EXIF: Software		
306	EXIF: Date/time		
315	EXIF: Artist		
317	Predictor	1 or 2	1 or 2
320	ColorMap	Yes	Yes
322	TileWidth	Yes	No
323	TileLength	Yes	No
324	TileOffsets	Yes	No
325	TileByteCounts	Required if compressed	No
332	InkSet	1	No
336	DotRange	Yes, if CMYK	Yes
338	ExtraSamples	Ignored (except for count)	Photoshop 5.5 and earlier writes 0. Photoshop 6.0 and later writes 0 or 1 based on the spec.

NOTE: See `Photoshop TIFF.pdf` for additional information about tags 259 and 262.

Photoshop-specific TIFF Tags

TABLE 2.4 Photoshop-specific TIFF tags

Tag	Description
330	**tSubIFD**. Documented in the TIFF-PM6.pdf file as a PageMaker extension
437	JPEG tables. See Photoshop TIFF.pdf for more information.
700	XMP metadata. See http://Partners.adobe.com/asn/developer/xmp/main.html
33723	File information (IPTC–NAA record 2: see the documents in the IPTC folder of the Documentation folder).
34377	Photoshop image resources (see "Image Resources Section" on page 12)
34665	EXIF IFD pointer. See http://www.pima.net/standards/it10/PIMA15740/exif.htm
34675	ICC Profiles (see the ICC34.pdf file from the International Color Consortium in the Documentation folder of the Photoshop SDK)
34853	EXIF GPS info. See http://www.pima.net/standards/it10/PIMA15740/exif.htm
37724	**tImageSourceData**. Begins with the null-terminated string "Adobe Photoshop Document Data Block", (**PSB** "Adobe Photoshop Document Data V0002"), followed by data of various types. See Photoshop TIFF.pdf for a list .
50255	**tAnnotations**. See Table 1.35 for details.

TIFF Files on Mac OS

For cross–platform compatibility, all information in a Macintosh TIFF file is stored in the data fork. For interoperability with other Mac OS applications, however, some information is duplicated in resources stored in the resource fork of the file.

- For compatibility with image cataloging applications, the **'pnot'** resource id 0 contains references to thumbnail, keywords, and caption information stored in other resources.
- The thumbnail picture is stored in a **'PICT'** resource, the keywords are stored in **'STR#'** resource 128 and the caption text is stored in **'TEXT'** resource 128. For more information on the format of these resources see *Inside Macintosh: QuickTime Components* and the *Extensis Fetch Awareness Developer's Toolkit.*
- All of the data from Photoshop's **File Info** dialog is stored in **'ANPA'** resource 10000.
- **STR '** resource -16396 contains a string indicating the application that created the TIFF file.
- Photoshop also creates **'icl8'** –16455 and **'ICN#'** –16455 resources containing thumbnail images which are shown in the Mac OS Finder.

3 Additional File Formats

In addition to documents that the user creates in Adobe Photoshop (discussed in Chapter 1, “The Photoshop File Format”), there are a number of additional files used by Photoshop to store information about such items as colors, brushes, and so forth. These are known as *load files.*

This chapter describes the format of each load file. Some of the files can saved by the user; others are load only, as indicated in the sections.

Each file has a unique file type and file extension associated with it. Photoshop for Macintosh recognizes either, but does not require the use of the extension. In the file dialogs, Photoshop for Windows looks for files with the given file extension automatically; this can be overridden.

Under Mac OS, all information is stored in the data forks of Photoshop’s load files. The files are completely interchangable with Windows or any other platform.

NOTE: Consistent byte ordering is required across platforms when reading and writing load files. Photoshop stores multi–byte values with the high–order bytes first, (big–endian), as on Mac OS., which is the opposite of Windows’ standard byte order.. For more information, see “Macintosh and Windows development” in chapter 2 of `Photoshop API Guide.pdf`.

NOTE: All values defined as Unicode strings consist of:

- A 4-byte length field, representing the number of characters in the string (not bytes).
- The string of Unicode values, two bytes per character.

Actions

Actions are accessed by means of the Actions palette. The object effects use the actions mechanism to output information to the PSD file format.

TABLE 3.1 Action file types

OS	Filetype/extension
Mac OS	`8BAC`
Windows	`.ATN`

Each action file comprises an *action set*. The format of the action file is described in the table below:

TABLE 3.2 Action file format

Length	Description
4	Version (= 16)
Variable	Unicode string: action set name
1	Boolean: true if set is expanded for the Actions palette
4	Number of actions in action set
The following is repeated for each action in the set	
2	Index of action
1	Boolean: true if Shift key needed for keyboard shortcut
1	Boolean: true if Command key needed for keyboard shortcut
2	Color index information
Variable	Unicode string: action name
1	Boolean: true if action is expanded in the Actions palette
4	Number of items in action
The following is repeated for each item	
1	Boolean: true if action is expanded in the Actions palette
1	Boolean: true if action is enabled
1	Boolean: true if dialogs should be displayed
1	Options for displaying dialogs

TABLE 3.2 Action file format

Length	Description
4	Identifier: **`'TEXT'`** or **`'long'`**
Variable	Event: if identifier is **`'TEXT'`**,4 bytes of length followed by the string; if identifier is **`'long'`**, 4 bytes of itemID
Variable	Dictionary name: 4 bytes of length followed by the string
4	-1 if a descriptor follows or 0 for none.
Variable	Descriptor: see Descriptor structure (Table 3.3) for details

TABLE 3.3 Descriptor structure

Length	Description
Variable	Unicode string: name from classID
Variable	classID: 4 bytes (length), followed either by string or (if length is zero) 4-byte classID
4	Number of items in descriptor
The following is repeated for each item in descriptor	
Variable	Key: 4 bytes (length) followed either by string or (if length is zero) 4-byte key

4	Type: OSType key
	`'obj ' = Reference`
	`'Objc' = Descriptor`
	`'VlLs' = List`
	`'doub' = Double`
	`'UntF' = Unit float`
	`'TEXT' = String`
	`'enum'= Enumerated`
	`'long' = Integer`
	`'bool' = Boolean`
	`'GlbO' = GlobalObject same as Descriptor`
	`'type' or GlbC'= Class`
	`'alis' = Alias`

Variable	Item type: see the tables below for each possible type

TABLE 3.4 Reference Structure

Length	Description
4	Number of items
The following is repeated for each item in reference	

4	OSType key for type to use:
	`'prop' = Property`
	`'Clss' = Class`
	`'Enmr' = Enumerated Reference`
	`'rele' = Offset`
	`'Idnt' = Identifier`
	`'indx' = Index`
	`'name' =Name`

Variable	Item type: see the tables below for each possible Reference type

TABLE 3.5 Property Structure

Length	Description
Variable	Unicode string: name from classID
Variable	classID: 4 bytes (length), followed either by string or (if length is zero) 4-byte classID
Variable	KeyID: 4 bytes (length), followed either by string or (if length is zero) 4-byte keyID

TABLE 3.6 Unit float structure

Length	Description
4	Units the following value is in. One of the following: **`'#Ang'`** = angle: base degrees **`'#Rsl'`** = density: base per inch **`'#Rlt'`** = distance: base 72ppi **`'#Nne'`** = none: coerced. **`'#Prc'`** = percent: ged unit value **`'#Pxl'`** = pixels: tagged unit value
8	Actual value (double)

TABLE 3.7 ***Double structure***

Length	Description
8	Actual value (double)

TABLE 3.8 ***Class structure***

Length	Description
Variable	Unicode string: name from classID
Variable	ClassID: 4 bytes (length), followed either by string or (if length is zero) 4-byte classID

TABLE 3.9 ***String structure***

Length	Description
Variable	String value as Unicode string

TABLE 3.10 ***Enumerated reference***

Length	Description
Variable	Unicode string: name from ClassID.
Variable	ClassID: 4 bytes (length), followed either by string or (if length is zero) 4-byte classID
Variable	TypeID: 4 bytes (length), followed either by string or (if length is zero) 4-byte typeID
Variable	enum: 4 bytes (length), followed either by string or (if length is zero) 4-byte enum

TABLE 3.11 ***Offset structure***

Length	Description
Variable	Unicode string: name from ClassID
Variable	ClassID: 4 bytes (length), followed either by string or (if length is zero) 4-byte classID
4	Value of the offset

TABLE 3.12 ***Boolean structure***

Length	Description
1	Boolean value

TABLE 3.13 *Alias structure*

Length	Description
4	Length of data to follow
Variable	`FSSpec` for Macintosh or a handle to a string to the full path on Windows

TABLE 3.14 *List structure*

Length	Description
4	Number of items in the list
The following is repeated for each item in list	
4	OSType key for type to use. See Table 3.3 for types.
Variable	See the tables above for each possible type

TABLE 3.15 *Integer*

Length	Description
4	Value

TABLE 3.16 *Enumerated descriptor*

Length	Description
Variable	Type: 4 bytes (length), followed either by string or (if length is zero) 4-byte typeID
Variable	Enum: 4 bytes (length), followed either by string or (if length is zero) 4-byte enum

Arbitrary Map

Arbitrary Map files are accessed by means of the Curves dialog (*load only*).

TABLE 3.17 Arbitrary map file types

OS	Filetype/extension
Mac OS	`8BLT`
Windows	`.AMP`

There is no version number written in the file.

The files are an even multiple of 256 bytes long. Each 256 bytes is a lookup table, where:

- The first byte of the table corresponds to byte zero of the image.
- The last byte of the table corresponds to byte 255 of the image.
- A `NULL` table that has no effect on an image is a linear table of bytes from 0 to 255.

If the file has one table, it is applied to the image's channels according to these priorities:

- If the image has a master composite channel, the table is applied to it. If not, then:
- If the image has a single active channel, the table is applied to it. If not, then:
- If the image has no composite channel and more than one active channel, the table is not applied.

If the file has exactly three tables, it is applied to the image's channels according to these priorities:

- The tables are assumed to represent RGB lookups. They are applied to the first three channels in the image, leaving the master composite untouched. Or:
- If the image has a single active channel, the tables are converted to grayscale and the result is applied to the active channel. Or:
- The first table is treated as a master. The remaining tables are applied to the image channels in turn (second table is applied to first channel, third table is applied to second channel, etc.).

Single active channels

Photoshop handles single active channels in a special fashion. When saving a map applied to a single channel, only one table is written to the file. Similarly, when reading a file for application to a single active channel, the master table is the one that will be used on that channel. This allows easy application of a single file to both composite and grayscale images.

Brushes

Brushes settings files are loaded and saved in Photoshop's Brushes palette, as well as the Preset Manager dialog. They are typically stored in the `Brushes` sub–folder of the `Presets` folder.

TABLE 3.18 Brushes file types

OS	Filetype/extension
Mac OS	**`8BBR`**
Windows	**`.ABR`**

TABLE 3.19 Brushes file format

Length	Description
2	Version (= 1)
2	Number of brushes in the remainder of the file
Variable	Data for each brush. See Table 3.20 for details.

TABLE 3.20 Brush components

Length	Description
2	Type of brush. Two types are currently supported: 1 = Elliptical, computed brush: created with the **New Brush** command 2 = Sampled brush: created from selected image data using the **Define Brush** command
4	Number of bytes in the remainder of the brush definition. Photoshop uses this information to skip over brush types that it doesn't understand.
Variable	Data. Computed brush data is always 14 bytes (see Table 3.21); sampled brush data varies in size depending on the image data that makes up the brush tip (see Table 3.22).

TABLE 3.21 Computed brush parameters

Length	Description
4	Miscellaneous. Ignored
2	Spacing: 0...999 where 0 = no spacing
2	Diameter: 1...999
2	Roundness: 0...100

TABLE 3.21 Computed brush parameters

Length	Description
2	Angle: -180...180
2	Hardness: 0...100

TABLE 3.22 Sampled brush parameters

Length	Description
4	Miscellaneous. Ignored.
2	Spacing: 0...999 where 0 = no spacing
1	0 = no anti-aliasing when applied; 1 = anti-alias when applied. Brushes with sampled data taller or wider than 32 pixels are never anti-aliased.
8	Rectangle: Four short integers giving the bounds of the sampled data in the order: top, left, bottom, right.
16	Rectangle: same as above, but in four long integers.
2	Depth of the sample data. Always 8.
Variable	Image data. If the bounds are taller than 16384 pixels, the data is broken into 16384-line chunks. Each chunk is streamed as shown in Table 3.23.

TABLE 3.23 Sampled brush image data structure

Length	Description
2	Compression. 0: data is just the raw image data. 1: data starts with the byte counts for all the scan lines (equal to the number of rows, as described by the bounds), with each count stored as a two–byte value. The RLE compressed data follows, with each scan line compressed separately. The RLE compression is the same compression algorithm used by the Macintosh ROM routine `PackBits`, and the TIFF standard.
Variable	The brush tip image data is a single plane of grayscale data, stored in scanline order, with no pad bytes.

CMYK Setup

CMYK settings files are accessed in Photoshop's Color Settings dialog *(load only)*.

Table 3.24 CMYK file types

OS	Filetype/extension
Mac OS	`8BIC`
Windows	`.API`

Table 3.25 CMYK setup file format

Length	Description
2	Version (= 7)
27*2	Nine sets of three short integers specifying th\e `xyY` (CIE) values for the inks and their combinations. The inks are specified in the order cyan, magenta, yellow, magenta–yellow (red), cyan–yellow (green), cyan–magenta (blue), cyan–magenta–yellow, followed by the white and black points. Each set is written in the order `xyY` where: `x` = 0...10000, representing 0.0...1.0000. `y` = 1...10000, representing 0.0001...1.0000. `Y` = 0...20000, representing 0.00...200.00.
2	Dot gain. Short integer from -10...40, representing –10%...40%.
1	Use curves. = 1 if curves table present.
1	Filler: zero
13*4*2	Only present if "use curves" = 1. 4 sets of 13 short integers specifyting the cyan, magenta, yellow, and black curve percentages from the Dot Gain Curves dialog. 0...1000, representing 0.0...100.0 %
Variable	Separation setup: see Table 3.26

TABLE 3.26 Separation file format

Length	Description
2	Version (= 300)
2	Separation type. 0 = UCR separations; 1 = GCR separations
2	Blank ink limit (0...100)
2	Total ink limit (200...400)
2	Undercolor addition for GCR separations (0...100)
Variable	Black generation (spline) curve detailed in Table 3.27. See also the Curves data format in Table 3.41.

TABLE 3.27 Black generation curve data structure

Length	Description
2	Number of points in curve (2...19)
2* number of points	Each curve point is a pair of short integers where the first number is the output value (vertical coordinate on the Black Generation dialog graph) and the second is the input value. All coordinates have range 0 to 255. A `NULL` curve (no change to image data) is represented by the following five–number, ten–byte sequence in a file: `2 0 0 255 255.` **NOTE:** The black generation curve and the UCA limit must both be present even if the separation type is set to UCR (= 0).

Color Books

Color book files *(Photoshop 7.0)* are automatically loaded by Photoshop; they cannot be saved or loaded via a menu item. You can place custom color books into the `Presets\Color Books` folder. Use the **Custom** button on the Adobe color picker to access them.

TABLE 3.28 Color book file types

OS	Filetype/extension
Mac OS	**`8BCB`**
Windows	**`.ACB`**

TABLE 3.29 Color book file format

Length	Description
4	Signature: **`8BCB`**
2	Version (=1)
2	Vendor ID. Existing IDs: 3000 (ANPA), 3001 (Focoltone), 3002 (PantoneCoated), 3003 (PantoneProcess), 3004 (PantoneProSlim), 3005 (PantoneUncoated), 3006 (Toyo), 3007 (Trumatch), 3008 (HKSE), 3009 (HKSK), 3010 (HKSN), 3011 (HKSZ), 3012 (DIC), 3020 (PantonePastelCoated), 3021 (PantonePastelUncoated), 3022 (PantoneMetallic)
Variable	Unicode string: title
Variable	Unicode string: prefix
Variable	Unicode string: postfix
Variable	Unicode string: description
2	Number of colors (<= 8000)
2	Colors per page (<= 9)
2	Key color page; must be less than or equal to colers per page
2	Color type. 0 = RGB; 2 = CMYK; 7 = Lab
The following are repeated for the number of colors	
Variable	Unicode string: name
6	Unique key for the color
4	Color values: 4 bytes for CMYK; 3 bytes for RGB and Lab

Color Table

Color Table files are accessed using the Colors palette *(load only)*.

TABLE 3.30 Color table file types

OS	Filetype/extension
Mac OS	`8BCT`
Windows	`.ACT`

There is no version number written in the file. The file is exactly 76

long, and contains 256 RGB colors:

- The first color in the table is index zero.
- There are three bytes per color in the order Red, Green, Blue.

If loaded into the Colors palette, the colors will be installed in the color swatch list as RGB colors.

Color Swatches

Color swatch files are loaded and saved in Photoshop's Color Swatches palette. These are typically stored in the `Color Swatches` sub–directory in the `Presets` directory.

TABLE 3.31 Color swatches file types

OS	Filetype/extension
Mac OS	`8BCO`
Windows	`.ACO`

TABLE 3.32 Color swatches file format

Length	Description
2	Version (=1)
2	Count of colors in the file.
count *10	Colors. Each color is 10 bytes, as described in Table 3.33.
At the end of a version 1 file is the version 2 information.	
2	Version (= 2)
2	Count of colors in the file. The next two fields are repeated for each count.
count *10	Colors. Each color is 10 bytes, as described in Table 3.33.
Variable	Unicode string: color name.

TABLE 3.33 Color structure

Length	Description
2	The color space the color belongs to (see Table 3.34).
8	Four short unsigned integers with the actual color data. If the color does not require four values, the extra values are undefined and should be written as zeros. See Table 3.34.

TABLE 3.34 Color space IDs

Color ID	Description
0	RGB. The first three values in the color data are *red*, *green*, and *blue*. They are full unsigned 16–bit values as in Apple's **`RGBColor`** data structure. Pure red = 65535, 0, 0.
1	HSB. The first three values in the color data are *hue*, *saturation*, and *brightness*. They are full unsigned 16–bit values as in Apple's **`HSVColor`** data structure. Pure red = 0,65535, 65535.
2	CMYK. The four values in the color data are *cyan*, *magenta*, *yellow*, and *black*. They are full unsigned 16–bit values. **NOTE:** 0 = 100% ink. For example, pure cyan = 0,65535,65535,65535.
7	Lab. The first three values in the color data are *lightness*, *a chrominance*, and *b chrominance*. Lightness is a 16–bit value from 0...10000. Chrominance components are each 16–bit values from –12800...12700. Gray values are represented by chrominance components of 0. Pure white = 10000,0,0.
8	Grayscale. The first value in the color data is the gray value, from 0...10000.

Photoshop allows the specification of custom colors, such as those colors that are defined in a set of custom inks provided by a printing ink manufacturer. These colors can be stored in the Colors palette and streamed to and from load files. The details of a custom color's color data fields are not public and should be treated as a black box.

Table 3.35 gives the color space IDs currently defined by Photoshop for some custom color spaces.

TABLE 3.35 Custom color spaces

Color ID	Name
3	Pantone matching system
4	Focoltone colour system
5	Trumatch color
6	Toyo 88 colorfinder 1050
10	HKS colors

Contours

Contour settings files *(Photoshop 6.0)* are loaded and saved in Photoshop's Layer Effects dialog.

TABLE 3.36 Contour file types

OS	Filetype/extension
Mac OS	`8BFS`
Windows	`.SHC`

TABLE 3.37 Contour file format

Length	Description
4	Type (= `'8BFS'`)
2	Version (= 1)
4	Count of contours
The following is repeated for each contour	
4	Version (= 1 or 2)
Variable	Unicode string: contour name
Variable	version 1 or 2 data follows. See Table 3.38 for version 1 and Table 3.39 for version 2.

TABLE 3.38 Contours Version 1

Length	Description
2	Count of points
4* count	For each point: 4 bytes of point data (2 bytes vertical, 2 bytes horizontal_
4	Minimum input range
4	Maximum input range

TABLE 3.39 ***Contours Version 2***

Length	Description
2	Count of points
4 * count	For each point: Point data (2 bytes vertical, 2 bytes horizontal)
1 * count	For each point: boolean indicating whether the point is continuous
4	Min input range
4	Max input range

Curves

Curves settings files are loaded in Photoshop's Curves dialog and Black Generation curve dialog (from within Separation Setup Preferences). Curves files can also be loaded into any of Photoshop's transfer function dialogs, such as the Duotone Curve dialog from within Duotone Options, and Print transfer dialog. Curves are saved as `.ATF` and `.ACV` files.

NOTE: When loaded into a transfer function dialog, only the first curve in a Curves file is used.

TABLE 3.40 Curves file types

OS	Filetype/extension
Mac OS	`8BSC`
Windows	`.CRV`

TABLE 3.41 Curves file format

Length	Description
2	Version (= 1 or = 4)
2	Count of curves in the file.
The following is the data for each curve specified by count above	
2	Count of points in the curve (short integer from 2...19)
point count * 4	Curve points. Each curve point is a pair of short integers where the first number is the output value (vertical coordinate on the Curves dialog graph) and the second is the input value. All coordinates have range 0 to 255. See also "Null curves" below.

Null curves

A `NULL` curve (no change to image data) is represented by the following five–number, ten–byte sequence in a file:

```
2 0 0 255 255
```

Displaying ink percentages

Photoshop allows the option of displaying ink percentages instead of pixel values; this is a display option only and the internal data is unchanged, with 100% ink equal to image data of 0 and 0% ink equal to image data of 255.

Curves data order

1. The first curve is a master curve that applies to all the composite channels (RGB) when in composite image mode.
2. The remaining curves apply to the active channels in order: curve two applies to channel one, curve three applies to channel two, etc., up until curve 17, which applies to channel 16.

Indexed color

The exception to the normal order, and the reason there are up to 19 curves, is when the mode is Indexed color. In this case:

1. The first curve is a master curve.
2. The next three curves are created for the Red, Green, and Blue portions of the image's color table, and they are applied to the first channel.
3. The remaining curves apply to any remaining alpha channel that is active: for instance, if channel two is active, curve five applies to it; if channel three is active, curve six applies to it, etc., up until curve 19, which applies to channel 16.

Single active channels

Photoshop handles single active channels in a special fashion. When saving the curves applied to a single channel, the settings are stored into the master curve, at the beginning of the file. Similarly, when reading a curves file for application to a single active channel, the master curve is the one that will be used on that channel. This allows easy application of a single file to both RGB and grayscale images.

Additional information

At the end of the Version 1 file is the following information:

Extra level record info marker **'Crv '**

TABLE 3.42 Extra curves marker

Length	Description
4	= **'Crv '** for extra curve information
2	Version (= 4)
2	Count of items to follow.
The following is the data for each curve specified by count above	
2	Before each curve is a channel index.
2	Count of points in the curve (short integer from 2...19)

TABLE 3.42 Extra curves marker

Length	Description
point count * 4	Curve points. Each curve point is a pair of short integers where the first number is the output value (vertical coordinate on the Curves dialog graph) and the second is the input value. All coordinates have range 0 to 255. See also "Null curves" below.

Custom Kernel

Kernel settings files are loaded and saved in Photoshop’s Custom Filter dialog..

TABLE 3.43 Custom kernel file types

OS	Filetype/extension
Mac OS	`8BCK`
Windows	`.ACF`

TABLE 3.44 Custom filter structure

Length	Description

50	Weights.
	The first 25 values are the custom weights from –999...999, applied to pixels offset from each pixel by [-2,-2] to [2,2]. The values progress through horizontal offsets first, as follows:
	`{[-2,-2],[-1,-2],[ 0,-2],[ 1,-2],[ 2,-2],`
	`[-2,-1],[-1,-1],[ 0,-1],[ 1,-1],[ 2,-1],`
	`[-2, 0],[-1, 0],[ 0, 0],[ 1, 0],[ 2, 0],`
	`[-2, 1],[-1, 1],[ 0, 1],[ 1, 1],[ 2, 1],`
	`[-2, 2],[-1, 2],[ 0, 2],[ 1, 2],[ 2, 2]}`

27*2	Ink colors. Nine sets of three short integers specifying the `xyY` (CIE) values for the inks and their combinations. The inks are specified in the order cyan, magenta, yellow, magenta–yellow (red), cyan–yellow (green), cyan–magenta (blue), cyan–magenta–yellow, followed by the white and black points. Each set is written in the order `xyY` where: `x` = 0...10000, representing 0.0...1.0000. `y` = 1...10000, representing 0.0001...1.0000. `Y` = 0...20000, representing 0.00...200.00.
2	Scale. Short integer from 1...9999.
2	Offset. Short integer from –9999...9999.

Duotone Options

Duotone settings files are loaded and saved in the Duotone Options dialog..

TABLE 3.45 Duotone file types

OS	Filetype/extension
Mac OS	`8BDT`
Windows	`.ADO`

TABLE 3.46 Duotone file format

Length	Description
2	Version (= 1)
2	**`Count`**. Number of plates in duotone spec (short integer). 1 = Monotone; 2 = Duotone; 3 = Tritone; 4 = Quadtone.
4*10	Four ink colors, regardless of the number of plates. The contents of the colors beyond the last plate specified by Count are undefined. Each color is 10 bytes and described in Table 3.47. It is identical to the format in a Colors load file.
4*64	Four ink names, regardless of the number of plates. Each name is streamed as a Pascal-style string with a length byte followed by the string name. Names may not be more than 63 characters. Each name is padded to occupy 64 bytes, including the length byte. Any names beyond the last plate specified by `Count` should be empty, size = 0.
4*28	Four ink curves, regardless of the number of plates. Described in Table 3.48.
2	Dot gain (= 20). Kept for compatability with Photoshop 2.0. Ignored.
11*10	Eleven overprint colorscolors, regardless of the number of plates. The number of defined overprints depends on **`Count`**. Monotones = no overprint colors. Duotones = one overprint color. Tritones = four overprint colors. Quadtones = 11 overprint colors. The contents of the colors beyond the last defined overprint are undefined. Each color is 10 bytes and described in Table 3.47. It is identical to the format in a Colors load file.

TABLE 3.47 Duotone color structure

Length	Description
2	The color space the color belongs to (see Table 3.34).
8	Four short unsigned integers with the actual color data. If the color does not require four values to specify, the extra values are undefined and should be written as zeros.

TABLE 3.48 Ink curves structure

Length	Description
26	Transfer curve: Array of 13 short integers from 0...1000 representing 0.0...100.0. All but the first and last value may be –1, representing no point on the curve. Any curves beyond the last plate should be equal to the `NULL` curve. A `NULL` transfer curve looks like this: 0, –1, –1, –1, –1, –1, –1, –1, –1, –1, –1, –1, 1000.
2	Override (= 0). Short integer for compatibility. Ignored by Photoshop 3.0 and higher.

Halftone Screens

Halftone Screens settings files are loaded and saved in Photoshop's Halftone Screens dialog (available from **Edit > Print with Preview** in Photoshop 7, or **Page Setup** or **Print Options** in previous versions).

TABLE 3.49 Halftone screen file types

OS	Filetype/extension
Mac OS	`8BHS`
Windows	`.AHS`

TABLE 3.50 Halftone screens file format

Length	Description
2	Version (= 5)
4*18	Four screen descriptions. See Table 3.51.
Variable	For every screen that has a custom spot function, the PostScript function text is written here, one after the other, with no header information, in the same order as the screen settings. The size of each custom spot is the absolute value of its negative shape code.

TABLE 3.51 Halftone screen parameter structure

Length	Description
4	Ink's screen frequency, in lines per inch. Binary fixed point value ;16 bits representing the integer and fractional parts from 1.0...999.999.
2	Units for the screen frequency. Lines per inch = 1; lines per centimeter = 2. Only affects display, not screen frequency.
4	Angle for screen. Binary fixed point value with 16 bits representing the integer and fractional parts from -180.0000 ... 180.0000, measured in degrees.
2	Code representing the shape of the halftone dots. 0 = Round; 1 = Ellipse; 2 = Line; 3 = Square; 4 = Cross; 6 = Diamond. Negative numbers represent custom shapes; the absolute value is the size in bytes of the custom spot function described in Table 3.50.
4	= 0. Not currently used by Photoshop.
1	Boolean. 1 = Use accurate screens; 0 = Use other.
1	Boolean. 1 = Use printer's default screens; 0 = Use other.

Hue/Saturation

Hue/Saturation settings files are loaded and saved in Photoshop's Hue/Saturation dialog/

TABLE 3.52 Hue/saturation file types

OS	Filetype/extension
Mac OS	`8BHA`
Windows	`.AHV`

TABLE 3.53 Hue/saturation file format

Length	Description
2	Version (= 2)
1	0 = Use settings for hue-adjustment; 1 = Use settings for colorization.
1	Padding byte; must be present but is ignored by Photoshop.
6	Colorization. Photoshop 5.0: The actual values are stored for the new version. Hue is -180...180, Saturation is 0...100, and Lightness is -100...100. Photoshop 4.0: Three short integers Hue, Saturation, and Lightness from –100...100. The user interface represents hue as –180...180, saturation as 0...100, and Lightness as -100...1000, as the traditional HSB color wheel, with red = 0.
6	Master hue, saturation and lightness values.
6 sets of the following 14 bytes (4 range values followed by 3 settings values)	
8: range values	For RGB and CMYK, those values apply to each of the six hextants in the HSB color wheel: those image pixels nearest to red, yellow, green, cyan, blue, or magenta. These numbers appear in the user interface from –60...60, however the slider will reflect each of the possible 201 values from –100...100. For Lab, the first four of the six values are applied to image pixels in the four Lab color quadrants, yellow, green, blue, and magenta. The other two values are ignored (= 0). The values appear in the user interface from –90 to 90.
6:settings values	

Levels

Levels settings files are loaded and saved in the Levels dialog.

Table 3.54 Levels file types

OS	Filetype/extension
Mac OS	`8BLS`
Windows	`.ALV`

Table 3.55 Levels file format

Length	Description
2	Version (= 2)
29 * 10	29 sets of level records, each level containing 5 short integers (see Table 3.56).

Table 3.56 Level record structure

Length	Description
2	Input floor (0...253)
2	Input ceiling (2...255)
2	Output floor (0...255). Matched to input floor.
2	Output ceiling (0...255)
2	Gamma. Short integer from 10...999 representing 0.1...9.99. Applied to all image data.

Level record sets order

The first set of levels is the master set that applies to all of the composite channels (RGB) when in composite image mode.

The remaining sets apply to the active channels individually; set two applies to channel one, the set three to channel two, etc., up until set 25, which applies to channel 24.

Sets 28 and 29 are reserved and should be set to zeros.

Indexed color

The exception to the normal order is when the mode is Indexed:

- The first set is a master set.
- The next three sets are created for the Red, Green, and Blue portions of the image's color table, and they are applied to the first channel.
- The remaining sets apply to any remaining alpha channels that are active: for instance, if channel two is active, set five applies to it; if channel three is active, set six applies to it, etc., up until channel 27, which applies to channel 24.
- Sets 28 and 29 are reserved and should be set to zeros.

Single active channels

Photoshop handles single active channels in a special fashion. When saving the levels applied to a single channel, the settings are stored into the master set, at the beginning of the file. Similarly, when reading a levels file for application to a single active channel, the master levels are the ones that will be used on that channel. This allows easy application of a single file to both RGB and grayscale images.

Photoshop CS (8.0) Additional information

At the end of the Version 2 file is the following information:

Extra level record info marker **'Lvls'**

TABLE 3.57 Extra levels marker

Length	Description
4	= **'Lvls'** for extra level information
2	Version (= 3)
2	Count of total level record structures. Subtract the legacy number of level record structures, 29, to determine how many are remaining in the file for reading.
Variable	Additianol level records according to count. Table 3.56

Monitor Setup

NOTE: This format has been superseded by ICC profiles. See `ICC34.pdf` for details.

Monitor settings files are accessed in Photoshop's Color Settings dialog, via the **Edit** menu *(load only)*.

TABLE 3.58 Monitor setup file types

OS	Filetype/extension
Mac OS	**`8BMS`**
Windows	**`.AMS`**

TABLE 3.59 Monitor setup file format

Length	Description
2	Version (= 2.)
2	Gamma. Short integer from 75...300 representing 0.75...3.00.
2*2	White point. Two short integers as CIE chromaticity coordinates: `x,y`. `x` = 0...10000 representing 0.0...1.0000. `y` = 1...10000 representing 0.0001...1.0000.
6*2	Phosphors. Three sets of two integers giving `x,y` coordinates of the red, green, and blue phosphors. `x` = 0...10000 representing 0.0...1.0000. `y` = 1...10000 representing 0.0001...1.0000. In the order *red* `x`, *red* `y`; *green* `x`, *green* `y`; *blue* `x`, *blue* `y`.

Replace Color/Color Range

Replace Color settings files are loaded and saved in the Color Range dialog (available via the **Select** menu).

TABLE 3.60 Replace color/Color range file types

OS	Filetype/extension
Mac OS	`8BXT`
Windows	`.AXT`

TABLE 3.61 Replace color/Color range file format

Length	Description
2	Version (= 1)
2	Short integer indicating what space the color components are in. 7 = Lab color, 8 = grayscale. No other values are supported.
6	Component ranges. Six unsigned byte values representing the range of colors within which a pixel's color must fall to be considered selected for color replacement, or color range selecting. Described in Table 3.62.
2	Fuzziness. Short integer from 0...200 controlling how colors close to selected colors are affected.
6	Transform settings. When used with Replace Color: Three short integers from –100...100. Described in Table 3.63. When used with Color Range: Writes zeros into the three short integers and ignores.

TABLE 3.62 Component range structure

Length	Description
1	if Lab (color space = 7): low endpoint of `L` value if grayscale (color space = 8): low endpoint of gray range

1	if Lab: high endpoint of `L` value
	if grayscale: 0

1	if Lab: low endpoint of `a` chrominance value
	if grayscale: 0

1	if Lab: high endpoint of `a` chrominance value
	if grayscale: 0

1	if Lab: low endpoint of `b` chrominance value if grayscale: low endpoint of gray range
1	if Lab: high endpoint of `b` chrominance value if grayscale: high endpoint of gray range

TABLE 3.63 Replace color transform settings

Length	Description
2	Hue change. Short integer from –100...100.
2	Saturation change. Short integer from –100...100.
2	Lightness change Short integer from –100...100. .

Selective Color

Selective Color settings files are loaded and saved in Photoshop's Selective Color dialog.

***TABLE 3.64** Selective color file types*

OS	Filetype/extension
Mac OS	`8BSV`
Windows	`.ASV`

***TABLE 3.65** Selective color file format*

Length	Description
2	Version (= 1)
2	Correction method.. 0 = Apply color correction in relative mode; 1 = Apply color correction in absolute mode.
80	Ten eight-byte plate correction records, described in Table 3.66. The first record is ignored by Photoshop and is reserved for future use. It should be set to all zeroes. The rest of the records apply to specific areas of colors or lightness values in the image, in the following order: reds, yellows, greens, cyans, blues, magentas, whites, neutrals, blacks.

***TABLE 3.66** Plate correction structure*

Length	Description
2	Amount of cyan correction. Short integer from –100...100.
2	Amount of magenta correction. Short integer from –100...100.
2	Amount of yellow correction. Short integer from –100...100.
2	Amount of black correction. Short integer from –100...100.

Separation Tables

NOTE: This format has been superseded by ICC profiles. See `ICC34.pdf` for details.

Separation Table files are accessed in the Separation Tables dialog *(load only)*.

TABLE 3.67 Separation table file types

OS	Filetype/extension
Mac OS	`8BST`
Windows	`.AST`

Format:

1. If the size of the file is `33 * 33 * 33 * 4`, then the file consists only of a Lab–>CMYK table as currently documented.
2. If the size of the file is `( 33 * 33 * 33 + 256 ) * 3`, then the file consists only of a CMYK–>Lab table as currently documented.
3. Otherwise, the file has the format listed in Table 3.68.

TABLE 3.68 Separation table file format

Length	Description
2	Version (= 300)
1	Boolean. True if contains Lab–>CMYK table.
1	Boolean. True if contains CMYK–>Lab table.
33*33*33*4	If file contains Lab–>CMYK table, this section contains CMYK colors for 33*33*33 Lab colors. The CMYK colors are written in interleaved order, one byte each ink. 0 = 100%, 255 = 0%. See “Generating Lab source colors” below.
(33*33*33 +256)*3	If file contains CMYK–>Lab table, this section contains Lab colors for 33*33*33+256 CMYK colors. The Lab colors are written in interleaved order, one byte per component. See “Generating CMYK source colors” below.
1	Boolean. True if gamut table follows.
1	If entry above is **`false`**, this byte will not be present. If **`true`**, this byte should be set to 1 for compatibility.

TABLE 3.68 Separation table file format

Length	Description
(((33*33*33L)+7)>>3) if gamut table present, zero otherwise	Gamut table, if present. The gamut table is a bit table indexed in the same way as the Lab->CMYK table with the high bit of the first byte at index 0. See "Testing for bits in the gamut table" below.

Generating Lab source colors

The Lab colors that are the source colors can be generated from the `Lab->CMYK` table with the following routine:

```
for (i = 0; i< 33; i++)
     for (j = 0; j < 33; j++)
            for (n = 0; n < 33; n++)
            {
                   L = Min (i * 8, 255);
                   a = Min (j * 8, 255);
                   b = Min (n * 8, 255);
            }
```

Generating CMYK source colors

The CMYK colors that are the source colors can be generated from the `CMYK->Lab` table with the following routine:

```
for (i = 0; i< 33; i++)
     for (j = 0; j < 33; j++)
            for (n = 0; n < 33; n++)
            {
                   c = Min (i * 8, 255);
                   m = Min (j * 8, 255);
                   y = Min (n * 8, 255);
                   k = 255;
            }
for (i = 0; i < 256; i++)
{
     c = 255;
     m = 255;
     y = 255;
     k = i;
}
```

Testing for bits in the gamut table

To test the bit at `bitIndex`, use table:

```
([bitIndex >> 3] & (0x0080 >> (bitIndex & 0x07))) != 0.
```

`bitIndex` itself is calculated in the same way you would calculate an index into the `Lab->CMYK` table.

A result of `1` indicates that the color is in gamut and `0` indicates that it is out of gamut.

Transfer Function

Transfer Function settings files are accessed *(load only)* in Photoshop's Duotone Curve dialog from within Duotone Options and Transfer Function dialogs (available from **Edit > Print with Preview** in Photoshop 7, or **Page Setup** or **Print Options** in previous versions).Transfer Function files can also be loaded into any of Photoshop's curves dialogs, such as the Curves color adjustment dialog.

TABLE 3.69 Transfer function file types

OS	Filetype/extension
Mac OS	**`8BTF`**
Windows	**`.ATF`**

TABLE 3.70 Transfer function file format

Length	Description
2	Version (= 4)
112 (= 28*4)	Four transfer functions , described in Table 3.71. **NOTE:** The file always contains four functions. When writing the printer transfer functions for grayscale images, for instance, Photoshop writes four copies of the single transfer function specified in the user interface.

TABLE 3.71 Transfer function structure

Length	Description
26	Curve. Array of 13 short integers from 0...1000 representing 0.0...100.0. All but the first and last value may be –1, representing no point on the curve. Any curves beyond the last plate should be equal to the `NULL` curve. A `NULL` transfer curve looks like this: `0, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, 1000.`
2	Boolean. 0 = Let printer supply curve; 1 = Override printer's default transfer curve.