International Color Consortium ®

Specification ICC.1:2003-09

File Format for Color Profiles (Version 4.1.0)

[REVISION of ICC.1:2001-12]

© 2003 ICC

Copyright notice

Copyright © 2003 International Color Consortium®

Permission is hereby granted, free of charge, to any person obtaining a copy of this Specification (the "Specification") to exploit the Specification without restriction including, without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sublicense, copies of the Specification, and to permit persons to whom the Specification is furnished to do so, subject to the following conditions:

Elements of this Specification may be the subject of intellectual property rights of third parties throughout the world including, without limitation, patents, patent application, utility, models, copyrights, trade secrets or other proprietary rights ("Third Party IP Rights"). Although no Third Party IP Rights have been brought to the attention of the International Color Consortium (the "ICC") by its members, or as a result of the publication of this Specification in certain trade journals, the ICC has not conducted any independent investigation regarding the existence of Third Party IP Rights. The ICC shall not be held responsible for identifying Third Party IP Rights that may be implicated by the practice of this Specification or the permissions granted above, for conducting inquiries into the applicability, existence, validity, or scope of any Third Party IP Rights that are brought to the ICC's attention, or for obtaining licensing assurances with respect to any Third Party IP Rights.

THE SPECIFICATION IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS, IMPLIED, OR OTHERWISE INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, QUIET ENJOYMENT, SYSTEM INTEGRATION, AND DATA ACCURACY. IN NO EVENT SHALL THE ICC BE LIABLE FOR ANY CLAIM, DAMAGES, LOSSES, EXPENSES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, CAUSED BY, ARISING OR RESULTING FROM, OR THAT RELATE TO THE SPECIFICATION OR THE PRACTICE OR OTHER EXPLOITATION OF THE SPECIFICATION. FURTHER, YOU HEREBY AGREE TO DEFEND, INDEMNIFY AND HOLD HARMLESS THE ICC, AND ITS DIRECTORS AND EMPLOYEES, FROM AND AGAINST ANY AND ALL THIRD PARTY CLAIMS, ACTIONS SUITS, DAMAGES, LOSSES, COSTS, EXPENSES, OR OTHER LIABILITIES (INCLUDING REASONABLE ATTORNEY'S FEES AND EXPENSES) THAT WERE CAUSED BY, ARISE OR RESULT FROM, OR RELATE TO, YOUR PRACTICE OR OTHER EXPLOITATION OF THE SPECIFICATION (INCLUDING, WITHOUT LIMITATION, CLAIMS OF INFRINGEMENT).

The above copyright notice, permission, and conditions and disclaimers shall be included in all copies of any material portion of the Specification. Except as contained in this statement, the name "International Color Consortium" shall not be used in advertising or otherwise to promote the use or other dealings in this Specification without prior written authorization from the ICC.

Licenses and trademarks

International Color Consortium and the ICC logo are registered trademarks of the International Color Consortium.

Rather than put a trademark symbol in every occurrence of other trademarked names, we state that we are using the names only in an editorial fashion, and to the benefit of the trademark owner, with no intention of infringement of the trademark.

For additional information on the ICC

Visit the ICC Web site: **http://www.color.org**

© 2001 ICC

© 2001 ICC

© 2001 ICC

© 2001 ICC

© 2003 ICC

© 2001 ICC

© 2001 ICC

0 Introduction

This specification describes the International Color Consortium® profile format. The intent of this format is to provide a cross-platform device profile format. Such device profiles can be used to translate color data created on one device into another device's native color space. The acceptance of this format by operating system vendors allows end users to transparently move profiles and images with embedded profiles between different operating systems. For example, this allows a printer manufacturer to create a single profile for multiple operating systems.

A large number of companies and individuals from a variety of industries participated in very extensive discussions on these issues. Many of these discussions occurred under the auspices of Forschungsgesellschaft Druck e.V. (FOGRA), the German graphic arts research institute, during 1993. The present specification evolved from these discussions and the ColorSync™ 1.0 profile format.

This is a very complex set of issues and the organization of this document strives to provide a clear, clean, and unambiguous explanation of the entire format. To accomplish this, the overall presentation is from a top-down perspective, beginning with a summary overview and continuing down into more detailed specifications to a byte stream description of format.

0.1 Intended audience

This specification is designed to provide developers and other interested parties a clear description of the profile format. A nominal understanding of color science is assumed, such as familiarity with the CIELAB color space, general knowledge of device characterizations, and familiarity with at least one operating system level color management system.

0.2 Organizational description of this specification

This specification is organized into a number of major clauses and annexes. Each clause and subclause is numbered for easy reference. A brief introduction is followed by a detailed summary of the issues involved in this document including: International Color Consortium, device profiles, profile element structure, embedded profiles, registration authority, and color model arbitration.

Clause 1 describes the scope of this specification.

Clause 2 provides the normative references for this specification.

Clause 3 describes the conformance requirements for this specification.

Clause 4 provides general definitions used within this specification.

Clause 5 provides descriptions of notations, symbols and abbreviations used in this specification.

Clause 6 describes the requirements of this specification. 6.1 describes the format header definition. 6.2 describes the tag table. 6.3 provides a top level view of what tags are required for each type of profile classification and a brief description of the algorithmic models associated with these classes. Four additional color transformation formats are also described: DeviceLink, color space conversion, abstract transformations, and named color transforms. 6.4 is a detailed algorithmic and intent description of all of the tagged elements described in the previous clauses. 6.5 provides a byte stream definition of the structures that make up the tags in 6.4.

Annex A describes the color spaces used in this specification. Annex B provides the necessary details to embed profiles into PICT, EPS, TIFF, and JPEG files. Annex C provides a general description of the Post-

© 2003 ICC

Script Level 2 tags used in this specification. Annex D is a paper describing details of the profile connection space. Annex E provides details on the chromaticAdaptationTag. Annex F is a summary of the changes made in the last few revisions of the spec. The C language ICC header file in previous versions of the specification has been removed as an appendix. It is available on the ICC web site as file ICC.3.

0.3 International Color Consortium

Considering the potential impact of this specification on various industries, a consortium has been formed that will administer this specification and the registration of tag signatures and descriptions. The founding members of this consortium are: Adobe Systems Inc., Agfa-Gevaert N.V., Apple Computer, Inc., Eastman Kodak Company, FOGRA (Honorary), Microsoft Corporation, Silicon Graphics, Inc., Sun Microsystems, Inc., and Taligent, Inc. (resigned). These companies have committed to fully support this specification in their operating systems, platforms and applications. The consortium has since been expanded and now has over 60 members. See the ICC web site for information on how to become a member.

0.4 Device profiles

Device profiles provide color management systems with the information necessary to convert color data between native device color spaces and device independent color spaces. This specification divides color devices into three broad classifications: input devices, display devices and output devices. For each device class, a series of base algorithmic models are described which perform the transformation between color spaces. These models provide a range of color quality and performance results. Each of the base models provides different trade-offs in memory footprint, performance and image quality. The necessary parameter data to implement these models is described in the required portions on the appropriate device profile descriptions. This required data provides the information for the color management framework default color management module (CMM) to transform color information between native device color spaces. A representative architecture using these components is illustrated in Figure 1 below.

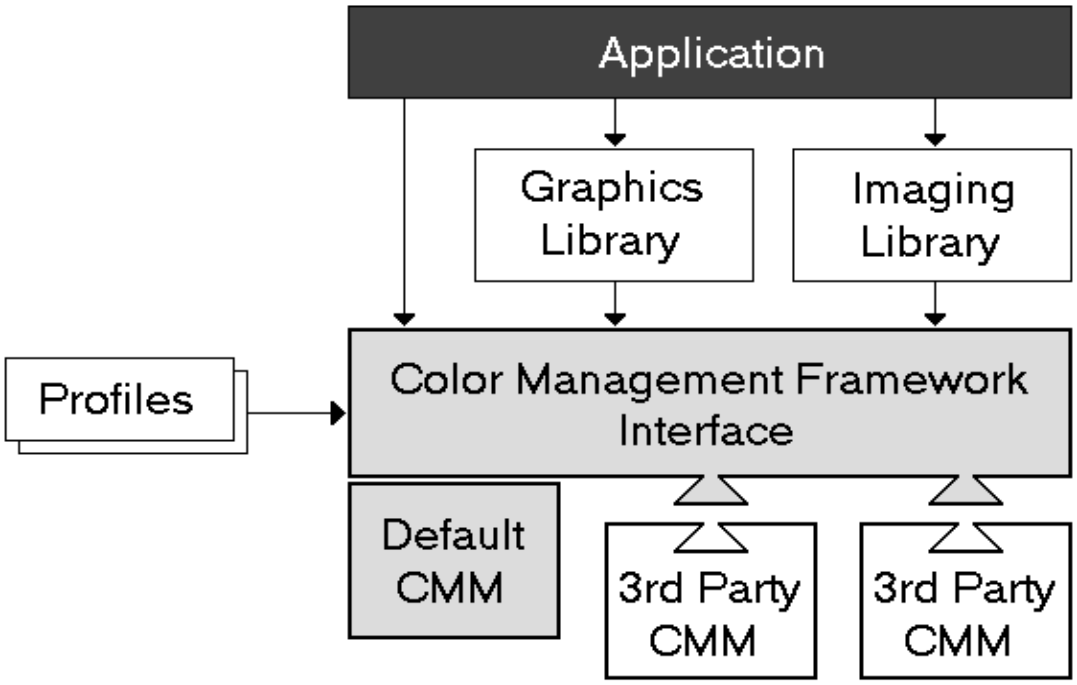

Figure 1 — Color management architecture

© 2003 ICC

0.5 Profile element structure

The profile structure is defined as a header followed by a tag table followed by a series of tagged elements that can be accessed randomly and individually. This collection of tagged elements provides three levels of information for developers: required data, optional data and private data. An element tag table provides a table of contents for the tagging information in each individual profile. This table includes a tag signature, the beginning address offset and size of the data for each individual tagged element. Signatures in this specification are defined as a four-byte hexadecimal number. This tagging scheme allows developers to read in the element tag table and then randomly access and load into memory only the information necessary to their particular software application. Since some instances of profiles can be quite large, this provides significant savings in performance and memory. The detailed descriptions of the tags, along with their intent, are included later in this specification.

The required tags provide the complete set of information necessary for the default CMM to translate color information between the profile connection space and the native device space. Each profile class determines which combination of tags is required. For example, a multi-dimensional lookup table is required for output devices, but not for display devices.

In addition to the required tags for each device profile, a number of optional tags are defined that can be used for enhanced color transformations. Examples of these tags include calibration support, and others. In the case of required and optional tags, all of the signatures, an algorithmic description, and intent are registered with the International Color Consortium.

Private data tags allow CMM developers to add proprietary value to their profiles. By registering just the tag signature and tag type signature, developers are assured of maintaining their proprietary advantages while maintaining compatibility with this specification. However, the overall philosophy of this format is to maintain an open, cross-platform standard, therefore the use of private tags should be kept to an absolute minimum.

0.6 Embedded profiles

In addition to providing a cross-platform standard for the actual disk-based profile format, this specification also describes the convention for embedding these profiles within graphics documents and images. Embedded profiles allow users to transparently move color data between different computers, networks and even operating systems without having to worry if the necessary profiles are present on the destination systems. The intention of embedded profiles is to allow the interpretation of the associated color data. Embedding specifications are described in Annex B of this document.

0.7 Registration authority

This specification requires that signatures for CMM type, device manufacturer, device model, profile tags and profile tag types be registered to insure that all profile data is uniquely defined. The registration authority for these data is the ICC Technical Secretary. See the ICC Web Site (www.color.org) for contact information.

If and when this document becomes an International Standard this registration responsibility must be brought into conformance with ISO procedures. These procedures are being investigated on behalf of ICC and TC130.

© 2003 ICC

0.8 Redundant data arbitration

There are several methods of color rendering described in the following structures that can function within a single CMM. If data for more than one method are included in the same profile, the following selection algorithm should be used by the software implementation.

For profile types input, display, output, or color space, the priority of the tag usage for each rendering intent is:

1. BToA0Tag, BToA1Tag, BToA2Tag, AToB0Tag, AToB1Tag, or AToB2Tag designated for the rendering intent
2. BToA0Tag or AToB0Tag
3. TRC's (redTRCTag, greenTRCTag, blueTRCTag, or grayTRCTag) and colorants (redMatrixColumnTag, greenMatrixColumnTag, blueMatrixColumnTag)

The available valid tag with the lowest number defines the transform.

1 Scope

This specification defines the data necessary to describe the color characteristics used to input, display, or output images, and an associated file format for the exchange of this data.

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this specification. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this specification are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of ISO and IEC maintain registers of currently valid International Standards.

CIE Publication 15.2-1986, "Colorimetry, Second Edition"

CIE Publ. 122-1996 The Relationship between Digital and Colorimetric Data for Computer-Controlled CRT Displays

EBU Tech. 3213-E: EBU standard for chromaticity tolerances for studio monitors

EC/CD 61966-2.1: Colour measurement and management in Multimedia systems and equipment - Part 2.1: Colour management in multimedia systems - Default RGB colour space - sRGB

IEC/CD 61966-3: Colour measurement and management in multimedia systems and equipment - Part 3: Equipment using cathode ray tubes

ISO 5-1:1984, "Photography -- Density measurements -- Part 1: Terms, symbols and notations"

ISO 5-2:1991, "Photography -- Density measurements -- Part 2: Geometric conditions for transmission density"

ISO 5-4:1995, "Photography -- Density measurements -- Part 4: Geometric conditions for reflection density"

ISO/IEC 646:1991, "Information technology -- ISO 7-bit coded character set for information interchange"

© 2003 ICC

ISO 3664:1975, “Photography -- Illumination conditions for viewing colour transparencies and their reproductions”

ISO/IEC 8824-1:1995, “Information technology -- Abstract Syntax Notation One (ASN.1): Specification of basic notation”

ISO/IEC 10918-1:1994, “Information technology -- Digital compression and coding of continuous-tone still images: Requirements and guidelines”

ISO/DIS 12234, “Photography -- Electronic still picture cameras -- Removeable memory (TIFF/EP)”

ISO/FDIS 12639, “Graphic Technology -- Prepress digital data exchange -- Tag image file format for image technology (TIFF/IT)”

ISO 12641:1997, “Graphic technology -- Prepress digital data exchange -- Colour targets for input scanner calibration”

ISO 12642:1996, “Graphic technology -- Prepress digital data exchange -- Input data for characterization of 4-colour process printing”

ISO 13655:1996, “Graphic technology -- Spectral measurement and colorimetric computation for graphic arts images”

ITU-R BT.709-2, Parameter values for the HDTV standards for production and international programme exchange

PICT Standard Specifications, published by Apple Computer, Inc.

PostScript Language Reference Manual, Third Edition, Adobe Systems Inc.

SMPTE RP 145-1994: SMPTE C Color Monitor Colorimetry

TIFF 6.0 Specification, published by Adobe Systems Incorporated.

3 Conformance

Any color management system, application, utility or device driver that is in conformance with this specification shall have the ability to read the profiles as they are defined in this specification. Any profile-generating software and/or hardware that is in conformance with this specification shall have the ability to create profiles as they are defined in this specification. ICC conforming software will use the ICC profiles in an appropriate manner.

4 Definitions

For the purposes of this specification, the following definitions shall apply:

4.1
aligned
A data element is aligned with respect to a data type if the address of the data element is an integral multiple of the number of bytes in the data type.

© 2003 ICC

4.2
ASCII string
A sequence of bytes, each containing a graphic character from ISO/IEC 646, the last character in the string being a NULL (character 0/0).

4.3
big-endian
Addressing the bytes within a 16, 32 or 64-bit value from the most significant to the least significant, as the byte address increases.

4.4
bit position
Bits are numbered such that bit 0 is the least significant bit.

4.5
byte
An 8-bit unsigned binary integer.

4.6
byte offset
The number of bytes from the beginning of a field.

4.7
fixed point representation
A method of encoding a real number into binary by putting an implied binary point at a fixed bit position. See Table 3 in 5.3.3 for an example.

Many of the tag types contain fixed point numbers. Several references can be found (MetaFonts, etc.) illustrating the preferability of fixed point representation to pure floating point representation in very structured circumstances.

4.8
NULL
The character coded in position 0/0 of ISO/IEC 646.

4.9
long word
A 32-bit quantity.

4.10
profile connection space (PCS)
An abstract color space used to connect the source and destination profiles. See A.1 for a full description.

4.11
rendering intent
A particular gamut mapping style or method of converting colors in one gamut to colors in another gamut. See Annex A for a more complete description of the rendering intents used in ICC profiles.

4.12
signature
An alphanumerical 4-byte value, registered with the ICC. Shorter values are padded at the end with 20h bytes.

© 2003 ICC

5 Notation, symbols and abbreviations

5.1 Notations

All numeric values in this specification are expressed in decimal, unless otherwise indicated. A letter “h” is suffixed to denote a hexadecimal value.

Literal strings are denoted in this specification by enclosing them in double quotation marks.

5.2 Symbols and abbreviations

The following symbols and abbreviations are used within this specification with the meanings indicated:

ANSI	American National Standards Institute
CIE	*Commission Internationale de l’Éclairage* (International Commission on Illumination)
CLUT	color Lookup Table (multidimensional)
CMM	color Management Module
CMS	color Management System
CMY	Cyan, Magenta, Yellow
CMYK	Cyan, Magenta, Yellow, Key (black)
CRD	color Rendering Dictionary
CRT	Cathode-Ray Tube
EPS	Encapsulated PostScript
ICC	International Color Consortium
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization
LCD	Liquid Crystal Display
LUT	Lookup Table
PCS	Profile Connection Space
PPD	PostScript Printer Description
RGB	Red, Green, Blue
TIFF	Tagged Image File Format
TRC	Tone Reproduction Curve

© 2003 ICC

5.3 Basic numeric types

5.3.1 dateTimeNumber

A 12-byte value representation of the time and date. The actual values are encoded as 16-bit unsigned integers.

Table 1 — dateTimeNumber

Byte Offset	Content	Encoded as...
0..1	number of the year (actual year, e.g. 1994)	uInt16Number
2..3	number of the month (1-12)	uInt16Number
4..5	number of the day of the month (1-31)	uInt16Number
6..7	number of hours (0-23)	uInt16Number
8..9	number of minutes (0-59)	uInt16Number
10..11	number of seconds (0-59)	uInt16Number

All the dateTimeNumber values in a profile shall be in Coordinated Universal Time (UTC, also known as GMT or ZULU Time). Profile writers are required to convert local time to UTC when setting these values. Programs that display these values may show the dateTimeNumber as UTC, show the equivalent local time (at current locale), or display both UTC and local versions of the dateTimeNumber.

5.3.2 response16Number

This type is used to associate a normalized device code with a measurement value.

Table 2 — response16Number

Byte Offset	Content	Encoded as...
0..1	16-bit number encoding the interval [DeviceMin to DeviceMax] with DeviceMin encoded as 0000h and DeviceMax encoded as FFFFh	uInt16Number
2..3	reserved, must be zero	
4..7	measurement value	s15Fixed16Number

© 2003 ICC

5.3.3 s15Fixed16Number

This type represents a fixed signed 4-byte/32-bit quantity which has 16 fractional bits. An example of this encoding is:

Table 3 — s15Fixed16Number

-32768,0	80000000h
0	00000000h
1,0	00010000h
32767 + (65535/65536)	7FFFFFFFh

5.3.4 u16Fixed16Number

This type represents a fixed unsigned 4-byte/32-bit quantity which has 16 fractional bits. An example of this encoding is:

Table 4 — u16Fixed16Number

0	00000000h
1,0	00010000h
65535 + (65535/65536)	FFFFFFFFh

5.3.5 u8Fixed8Number

This type represents a fixed unsigned 2-byte/16-bit quantity which has 8 fractional bits. An example of this encoding is:

Table 5 — u8Fixed8Number

0	0000h
1,0	0100h
255 + (255/256)	FFFFh

5.3.6 uInt16Number

This type represents a generic unsigned 2-byte/16-bit quantity.

5.3.7 uInt32Number

This type represents a generic unsigned 4-byte/32-bit quantity.

5.3.8 uInt64Number

This type represents a generic unsigned 8-byte/64-bit quantity.

5.3.9 uInt8Number

This type represents a generic unsigned 1-byte/8-bit quantity.

© 2003 ICC

5.3.10 XYZNumber

This type represents a set of three fixed signed 4-byte/32-bit quantities used to encode CIEXYZ tristimulus values where byte usage is assigned as follows:

Table 6 — XYZNumber

Byte Offset	Content	Encoded as...
0..3	CIE X	s15Fixed16Number
4..7	CIE Y	s15Fixed16Number
8..11	CIE Z	s15Fixed16Number

All XYZNumbers (other than those specifying luminance) are scaled such that Y is specified over the range of 0 to 1,0. Tristimulus values must be non-negative.

© 2003 ICC

5.3.11 Seven-bit ASCII

Table 7 — Hexadecimal

Hexadecimal															
00	nul	01	soh	02	stx	03	etx	04	eot	05	enq	06	ack	07	bel
08	bs	09	ht	0a	nl	0b	vt	0c	np	0d	cr	0e	so	0f	si
10	dle	11	dc1	12	dc2	13	dc3	14	dc4	15	nak	16	syn	17	etb
18	can	19	em	1a	sub	1b	esc	1c	fs	1d	gs	1e	rs	1f	us
20	sp	21	!	22	“	23	#	24	$	25	%	26	&	27	‘
28	(	29	)	2a	*	2b	+	2c	,	2d	-	2e	.	2f	/
30	0	31	1	32	2	33	3	34	4	35	5	36	6	37	7
38	8	39	9	3a	:	3b	;	3c	<	3d	=	3e	>	3f	?
40	@	41	A	42	B	43	C	44	D	45	E	46	F	47	G
48	H	49	I	4a	J	4b	K	4c	L	4d	M	4e	N	4f	O
50	P	51	Q	52	R	53	S	54	T	55	U	56	V	57	W
58	X	59	Y	5a	Z	5b	[	5c	\	5d	]	5e	^	5f	_
60	`	61	a	62	b	63	c	64	d	65	e	66	f	67	g
68	h	69	i	6a	j	6b	k	6c	l	6d	m	6e	n	6f	o
70	p	71	q	72	r	73	s	74	t	75	u	76	v	77	w
78	x	79	y	7a	z	7b	{	7c	|	7d	}	7e	~	7f	del

© 2003 ICC

Table 8 — Decimal

Decimal															
0	nul	1	soh	2	stx	3	etx	4	eot	5	enq	6	ack	7	bel
8	bs	9	ht	10	nl	11	vt	12	np	13	cr	14	so	15	si
16	dle	17	dc1	18	dc2	19	dc3	20	dc4	21	nak	22	syn	23	etb
24	can	25	em	26	sub	27	esc	28	fs	29	gs	30	rs	31	us
32	sp	33	!	34	“	35	#	36	$	37	%	38	&	39	‘
40	(	41	)	42	*	43	+	44	,	45	-	46	.	47	/
48	0	49	1	50	2	51	3	52	4	53	5	54	6	55	7
56	8	57	9	58	:	59	;	60	<	61	=	62	>	63	?
64	@	65	A	66	B	67	C	68	D	69	E	70	F	71	G
72	H	73	I	74	J	75	K	76	L	77	M	78	N	79	O
80	P	81	Q	82	R	83	S	84	T	85	U	86	V	87	W
88	X	89	Y	90	Z	91	[	92	\	93	]	94	^	95	_
96	`	97	a	98	b	99	c	100	d	101	e	102	f	103	g
104	h	105	i	106	j	107	k	108	l	109	m	110	n	111	o
112	p	113	q	114	r	115	s	116	t	117	u	118	v	119	w
120	x	121	y	122	z	123	{	124	|	125	}	126	~	127	del

© 2003 ICC

6 Requirements

An ICC profile shall include the following elements, in the order shown below in Figure 2, as a single file.

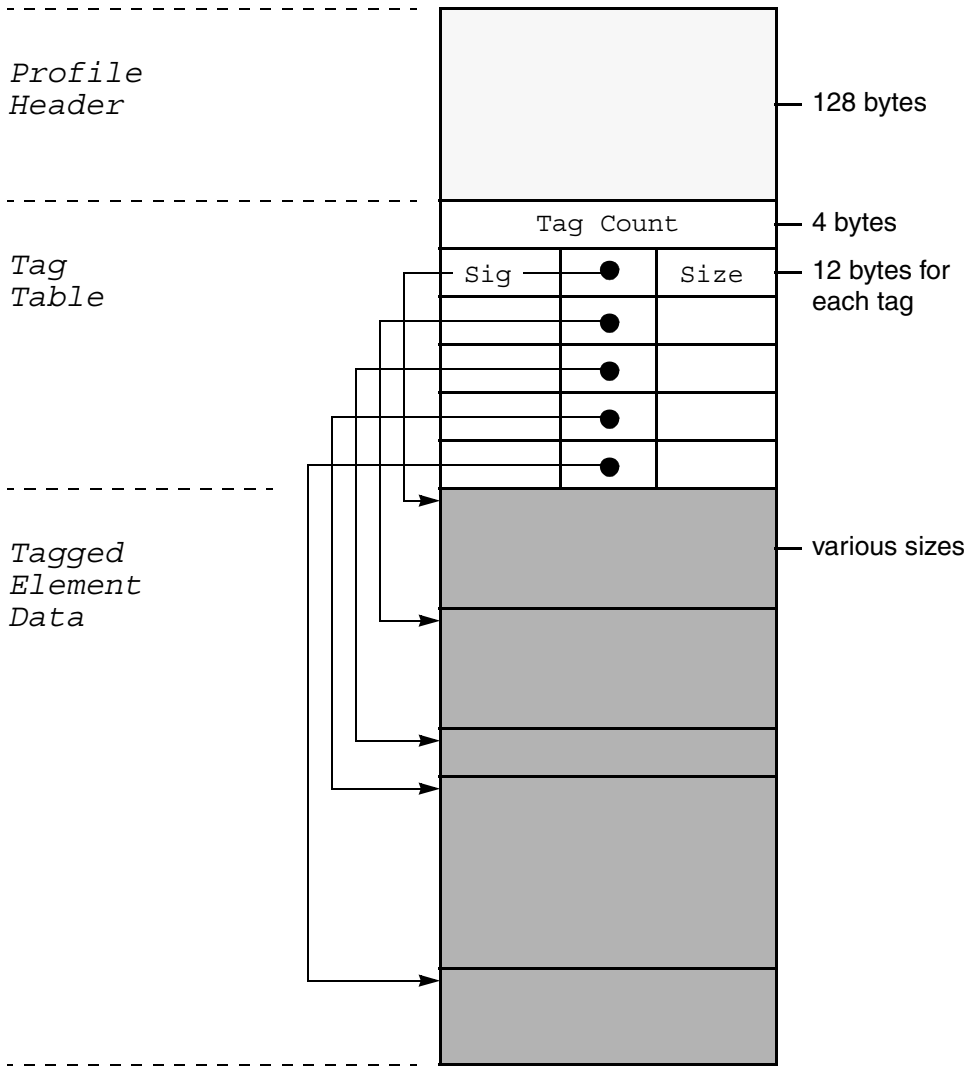

Figure 2 — Profile Map

First, the 128-byte file header as defined in 6.1.

Second, the tag table as defined in 6.2.

Third, the tagged element data in accordance with the requirements of 6.3, 6.4 and 6.5.

There are additional requirements on the structure of the profile.

1) The first tagged element data must immediately follow the tag table.
2) All tagged element data, including the last, must be padded by no more than three following pad bytes to reach a 4-byte boundary.
3) All pad bytes must be NULL.

© 2003 ICC

4) The profile size value in the header of the profile must be the exact size obtained by combining the profile header, the tag table, and the tagged element data, including the pad bytes for the last tag. This implies that the length must be a multiple of four.

The above restrictions result in two key benefits. First, the likelihood of two profiles which contain the same tag data yet have different checksum values is reduced. Second, all profiles are reduced to a minimum size.

The information necessary to understand and create the tagged element data is arranged within this specification as follows. Each class, and subclass, of device (e.g.: input, RGB) requires the use of specific tags and allows other optional tags. These relationships are described in 6.3. Tags themselves are described in 6.4. However tag descriptions draw upon a series of commonly used “tag types” which are defined in 6.5. The definition of the basic number types used for data encoding are found in 5.3.

All profile data must be encoded as big-endian.

All color spaces used in this specification shall be in accordance with Annex A.

6.1 Header description

The profile header provides the necessary information to allow a receiving system to properly search and sort ICC profiles. Table 9 gives the byte position, content and encoding of the profile header.

This header provides a set of parameters at the beginning of the profile format. For color space conversion and abstract profiles, the device profile dependent fields are set to zero if irrelevant. Having a fixed length header allows for performance enhancements in the profile searching and sorting operations.

Table 9 — Header (Part 1 of 2)

Byte Offset	Content	Encoded as...
0..3	Profile size	uInt32Number
4..7	CMM Type signature	see below
8..11	Profile version number	see below
12..15	Profile/Device Class signature	see below
16..19	Color space of data (possibly a derived space) [i.e. “the canonical input space”]	see below
20..23	Profile Connection Space (PCS) [i.e. “the canonical output space”]	see below
24..35	Date and time this profile was first created	dateTimeNumber
36..39	‘acsp’ (61637370h) profile file signature	
40..43	Primary Platform signature	see below
44..47	Flags to indicate various options for the CMM such as distributed processing and caching options	see below
48..51	Device manufacturer of the device for which this profile is created	see below
52..55	Device model of the device for which this profile is created	see below

© 2003 ICC

Table 9 — Header (Part 2 of 2)

56..63	Device attributes unique to the particular device setup such as media type	see below
64..67	Rendering Intent	see below
68..79	The XYZ values of the illuminant of the Profile Connection Space. This must correspond to D50. It is explained in more detail in A.1.	XYZNumber
80..83	Profile Creator signature	see below
84..99	Profile ID	see below
100..127	28 bytes reserved for future expansion - must be set to zeros	

6.1.1 Profile size

The total size of the profile in bytes.

6.1.2 CMM Type signature

Identifies the preferred CMM to be used. The signatures must be registered in order to avoid conflicts (see 0.7). If no CMM is preferred, this field should be set to zero.

6.1.3 Profile version

Profile version number where the first 8 bits identify the major revision and the next 8 bits identify the minor revision and bug fix revision. The major and minor revision are set by the International Color Consortium and will match up with editions of this specification.

The current profile version number is "4.1.0" (encoded as 04000000h).

The encoding is such that:

Table 10 — Profile version

Byte Offset	Content
0	Major Revision in Binary-Coded Decimal
1	Minor Revision & Bug Fix Revision in each nibble in Binary-Coded Decimal
2	reserved, must be set to 0
3	reserved, must be set to 0

A major revision change will only happen when a profile specification change requires that all CMMs be upgraded in order to correctly use the profile. For example, the addition of new required tags would cause the major revision to change. A minor version change will occur when new profiles can still be used by existing CMMs. For example, the addition of new optional tags would cause the minor revision to change, because existing CMMs will be able to process the profiles correctly while ignoring the new tags.

© 2003 ICC

6.1.4 Profile/Device Class signature

There are three basic classes of device profiles: Input, Display and Output profiles.

Within each of these classes there can be a variety of subclasses, such as RGB scanners, CMYK scanners and many others. These basic classes have the following signatures:

Table 11 — Device class

Device Class	Signature	Hex Encoding
Input Device profile	‘scnr’	73636E72h
Display Device profile	‘mntr’	6D6E7472h
Output Device profile	‘prtr’	70727472h

In addition to the three basic device profile classes, four additional color processing profiles are defined. These profiles provide a standard implementation for use by the CMM in general color processing or for the convenience of CMMs which may use these types to store calculated transforms. These four profile classes are: DeviceLink, color space conversion, abstract, and named color profiles.

DeviceLink profiles provide a mechanism in which to save and store a series of device profiles and non-device profiles in a concatenated format as long as the series begins and ends with a device profile. This is useful for workflows where a combination of device profiles and non-device profiles are used repeatedly.

ColorSpace Conversion profiles are used as a method for CMMs to convert between different non-device color spaces.

The Abstract color profiles provide a generic method for users to make subjective color changes to images or graphic objects by transforming the color data within the PCS.

Named Color profiles can be thought of as sibling profiles to device profiles. For a given device there would be one or more device profiles to handle process color conversions and one or more named color profiles to handle named colors. There might be multiple named color profiles to account for different consumables or multiple named color vendors.

These profiles have the following signatures:

Table 12 — Profile class

Profile Class	Signature	Hex Encoding
DeviceLink profile	‘link’	6C696E6Bh
ColorSpace Conversion profile	‘spac’	73706163h
Abstract profile	‘abst’	61627374h
Named Color profile	‘nmcl’	6E6D636Ch

© 2003 ICC

6.1.5 Color Space signature

The encoding is such that:

Table 13 — Color space signature

Color Space	Signature	Hex Encoding
XYZData	‘XYZ ’	58595A20h
labData	‘Lab ’	4C616220h
luvData	‘Luv ’	4C757620h
YCbCrData	‘YCbr ’	59436272h
YxyData	‘Yxy ’	59787920h
rgbData	‘RGB ’	52474220h
grayData	‘GRAY’	47524159h
hsvData	‘HSV ’	48535620h
hlsData	‘HLS ’	484C5320h
cmykData	‘CMYK’	434D594Bh
cmyData	‘CMY ’	434D5920h
2colorData	‘2CLR’	32434C52h
3colorData (if not listed above)	‘3CLR’	33434C52h
4colorData (if not listed above)	‘4CLR’	34434C52h
5colorData	‘5CLR’	35434C52h
6colorData	‘6CLR’	36434C52h
7colorData	‘7CLR’	37434C52h
8colorData	‘8CLR’	38434C52h
9colorData	‘9CLR’	39434C52h
10colorData	‘ACLR’	41434C52h
11colorData	‘BCLR’	42434C52h
12colorData	‘CCLR’	43434C52h
13colorData	‘DCLR’	44434C52h
14colorData	‘ECLR’	45434C52h
15colorData	‘FCLR’	46434C52h

© 2003 ICC

6.1.6 Profile Connection Space signature

The encoding is such that:

Table 14 — Profile connection space signature

Profile Connection Color Space	Signature	Hex Encoding
XYZData	‘XYZ ’	58595A20h
labData	‘Lab ’	4C616220h

When the profile is a DeviceLink profile, the Profile Connection Space Signature can be any of the signatures in the Color Space Signatures table. (See 6.1.5)

6.1.7 Primary Platform signature

Signature to indicate the primary platform/operating system framework for which the profile was created.

The encoding is such that:

Table 15 — Primary platform signature

Primary Platform	Signature	Hex Encoding
Apple Computer, Inc.	‘APPL’	4150504Ch
Microsoft Corporation	‘MSFT’	4D534654h
Silicon Graphics, Inc.	‘SGI ’	53474920h
Sun Microsystems, Inc.	‘SUNW’	53554E57h
Taligent, Inc.	‘TGNT’	54474E54h

If there is no primary platform, this field should be set to zero.

6.1.8 Profile flags

Flags to indicate various hints for the CMM such as distributed processing and caching options. The least-significant 16 bits are reserved for the ICC.

The encoding is such that:

Table 16 — Profile flags

Flags	Bit Position
Embedded Profile (0 if not embedded, 1 if embedded in file)	0
Profile cannot be used independently from the embedded color data (set to 1 if true, 0 if false)	1

6.1.9 Device manufacturer and model signatures

The signatures for various manufacturers and models are listed in a separate document (ICC Signatures). New signatures must be registered with the ICC (see 0.7).

© 2003 ICC

6.1.10 Attributes

Attributes unique to the particular device setup such as media type. The least-significant 32 bits of this 64-bit value are reserved for the ICC.

The encoding is such that (with “on” having value 1 and “off” having value 0):

Table 17 — Header attributes

Attribute	Bit Position
Reflective (off) or Transparency (on)	0
Glossy (off) or Matte (on)	1
Positive (off) or negative (on) media	2
Color (off) or black & white (on) media	3

Notice that bits 0, 1, 2, and 3 describe the media, not the device. For example, a profile for a color scanner that has been loaded with black & white film will have bit 3 set on, regardless of the colorspace of the scanner (see 6.1.5).

If the media is not inherently "color" or "black & white" (such as the paper in an inkjet printer), the reproduction takes on the property of the device. Thus, an inkjet printer loaded with a color ink cartridge can be thought to have "color" media.

6.1.11 Rendering intent

Perceptual, media-relative colorimetric, saturation and ICC-absolute colorimetric are the four intents required to be supported. The least-significant 16 bits are reserved for the ICC.

The encoding is such that:

Table 18 — Header rendering intents

Rendering Intent	Value
Perceptual	0
Media-Relative Colorimetric	1
Saturation	2
ICC-Absolute Colorimetric	3

The rendering intent specifies the style of reproduction which should be used (or, in the case of a DeviceLink profile, was used) when this profile is (was) combined with another profile. In a sequence of profiles, it applies to the combination of this profile and the next profile in the sequence and not to the entire sequence. Typically, the user or application will set the rendering intent dynamically at runtime or embedding time. Therefore, this flag may not have any meaning until the profile is used in some context, e.g in a DeviceLink or an embedded source profile.

The field is a uInt32Number in which the least-significant 16 bits are used to encode the rendering intent. The most significant 16 bits should be set to 0.

© 2003 ICC

6.1.12 Profile Creator signature

Identifies the creator of the profile. The signatures are from the group of signatures used for the device manufacturer field.

6.1.13 Profile ID

This field is optional but should be used to record a checksum value which if used shall be generated using the MD5 fingerprinting method as defined in RFC 1321. The entire profile, based on the size field in the header, shall be used to calculate the ID after the values in the 6.1.8 Profile flags field (Bytes 44 to 47), 6.1.11 Rendering intent field (Bytes 64 to 67), and 6.1.13 Profile ID field (Bytes 84 to 99) fields in the profile header have been temporarily replaced with zeros. If a profile ID has not been calculated the value of the field shall be set to zero.

6.2 Tag table definition

The tag table acts as a table of contents for the tags and tag element data in the profiles. The first four bytes contain a count of the number of tags in the table itself. The tags within the table are not required to be in any particular order.

Each tag signature in the tag table must be unique; a profile cannot contain more than one tag with the same signature.

Individual tag structures within the Tag Table:

Table 19 — Tag table structure

Byte Offset	Content	Encoded as...
0..3	Tag Signature	
4..7	Offset to beginning of tag data	uInt32Number
8..11	Element Size	uInt32Number

6.2.1 Tag signature

A four-byte value registered with the ICC (see 0.7).

6.2.2 Offset

The address of the tag data element. This is the number of bytes from the beginning of the profile data stream (i.e. the offset to the first byte in the profile header is 0). For profiles that are not embedded in images, this number is the same as the file offset.

All tag data is required to start on a 4-byte boundary (relative to the start of the profile data stream) so that a tag starting with a 32-bit value will be properly aligned without the tag handler needing to know the contents of the tag. This means that the least-significant two bits of each offset must be zero.

© 2003 ICC

6.2.3 Element size

The number of bytes in the tag data element. The element size must be for the actual data and must not include any padding at the end of the tag data. An element may have any size (up to the limit imposed by the 32-bit offsets).

6.3 Required tags for profiles

This clause provides a top level view of what tags are required for each type of profile classification and a brief description of the algorithmic models associated with these classes. A general description for each tag is included in this clause.

Note that these descriptions assume two things; every profile contains a header, and may include additional tags beyond those listed as required in this clause. The explicitly listed tags are those which are required in order to comprise a legal profile of each type.

In general, multi-dimensional tables refer to lookup tables with more than one input component.

The intent of requiring tags with profiles is to provide a common base level of functionality. If a custom CMM is not present, then the default CMM will have enough information to perform the requested color transformations. The particular models implied by the required data are also described below. While this data might not provide the highest level of quality obtainable with optional data and private data, the data provided is adequate for sophisticated device modeling.

In the following tables, the term "undefined" means that the use of the tag in that situation is not specified by the ICC. The ICC recommends that such tags not be included in profiles. If the tag is present, its use is implementation dependent. In general, the BToAxTags represent the inverse operation of the AToBxTags.

Note that AToB1Tag and BToA1Tag are used to provide both of the colorimetric intents.

Note that tags may reference the same tag data but that this may not be suitable in many cases.

If an optional transformation tag is not present, see Section 0.8.

© 2003 ICC

The interpretation of some tags are context dependent. This dependency is described below in Table 20

Table 20 — Profile type/profile tag and defined rendering intents

Profile Class	**AToB0Tag**	**AToB1Tag**	**AToB2Tag**	**TRC/ matrix**	**BToA0Tag**	**BToA1Tag**	**BToA2Tag**
Input	Device to PCS: perceptual	Device to PCS: colorimetric	Device to PCS: saturation	colorimetric	PCS to Device: perceptual	PCS to Device: colorimetric	PCS to Device: saturation
Display	Device to PCS: perceptual	Device to PCS: colorimetric	Device to PCS: saturation	colorimetric	PCS to Device: perceptual	PCS to Device: colorimetric	PCS to Device: saturation
Output	Device to PCS: perceptual	Device to PCS: colorimetric	Device to PCS: saturation	colorimetric	PCS to Device: perceptual	PCS to Device: colorimetric	PCS to Device: saturation
Color-Space	ColorSpace to PCS: perceptual	ColorSpace to PCS: colorimetric	ColorSpace to PCS: saturation	undefined	PCS to Color-Space: perceptual	PCS to Color-Space: colorimetric	PCS to Color-Space: saturation
Abstract	PCS to PCS	undefined	undefined	undefined	undefined	undefined	undefined
DeviceLink	Device1 to Device2 rendering intent defined according to Table 9	undefined	undefined	undefined	undefined	undefined	undefined
Named Color	undefined	undefined	undefined	undefined	undefined	undefined	undefined

6.3.1 Input Profile

This profile represents input devices such as scanners and digital cameras.

6.3.1.1 Monochrome Input Profiles

Table 21 — Monochrome input profile required tags

Tag Name	**General Description**
profileDescriptionTag	Structure containing invariant and localizable versions of the profile name for display
grayTRCTag	Gray tone reproduction curve (TRC)
mediaWhitePointTag	Media XYZ white point
copyrightTag	Profile copyright information
chromaticAdaptationTag	Converts XYZ color from the actual illumination source to PCS illuminant. Required only if the actual illumination source is not D50.

The mathematical model implied by this data is:

$$connection = \text{grayTRC}[device] \quad (1)$$

This represents a simple tone reproduction curve adequate for most monochrome input devices. The *connection* values in this equation should represent the achromatic channel of the profile connection space in the range of 0 to 1,0 where 0 represents black and 1,0 represents white. The PCS value is derived by

© 2003 ICC

multiplying the D50 white point by the normalized TRC value between 0 and 1,0. If the inverse of this is desired, then the following equation is used:

$$device = \text{grayTRC}^{-1}[connection] \tag{2}$$

AToB0Tag, AToB1Tag, AToB2Tag, BToA0Tag, BToA1Tag, BToA2Tag may be included in monochrome profiles. If these are present, their usage shall be as defined in Table 20.

6.3.1.2 Three-component Matrix-based Input Profiles

This profile type is often used with devices whose nominal color space is RGB. .

Table 22 — Three-component matrix-based input profile required tags

Tag Name	General Description
profileDescriptionTag	Structure containing invariant and localizable versions of the profile name for display
redMatrixColumnTag	The first column in the matrix used in TRC/matrix transforms (This column is combined with the linear red channel during the matrix multiplication)
greenMatrixColumnTag	The second column in the matrix used in TRC/matrix transforms (This column is combined with the linear green channel during the matrix multiplication)
blueMatrixColumnTag	The third column in the matrix used in TRC/matrix transforms (This column is combined with the linear blue channel during the matrix multiplication)
redTRCTag	Red channel tone reproduction curve
greenTRCTag	Green channel tone reproduction curve
blueTRCTag	Blue channel tone reproduction curve
mediaWhitePointTag	Media XYZ white point
copyrightTag	Profile copyright information
chromaticAdaptationTag	Converts XYZ color from the actual illumination source to PCS illuminant. Required only if the actual illumination source is not D50.

This model describes transformation from device color space to PCS (see Annex A for PCS description). The transformation is based on three non-interdependent per-channel tone reproduction curves to convert between non-linear and linear RGB values and a 3x3 matrix to convert between linear RGB values and relative XYZ values. The mathematical model implied by this data is:

$$\begin{aligned} linear_r &= \text{redTRC}[device_r] \\ linear_g &= \text{greenTRC}[device_g] \\ linear_b &= \text{blueTRC}[device_b] \end{aligned} \tag{3}$$

© 2003 ICC

$$\begin{bmatrix} connection_X \\ connection_Y \\ connection_Z \end{bmatrix} = \begin{bmatrix} redMatrixColumn_X & greenMatrixColumn_X & blueMatrixColumn_X \\ redMatrixColumn_Y & greenMatrixColumn_Y & blueMatrixColumn_Y \\ redMatrixColumn_Z & greenMatrixColumn_Z & blueMatrixColumn_Z \end{bmatrix} \begin{bmatrix} linear_r \\ linear_g \\ linear_b \end{bmatrix} \quad (4)$$

This represents a simple linearization followed by a linear mixing model. The three tone reproduction curves linearize the raw values with respect to the luminance (Y) dimension of the CIEXYZ encoding of the profile connection space. The 3x3 matrix converts these linearized values into XYZ values for the CIEXYZ encoding of the profile connection space. The inverse model is given by the following equations:

$$\begin{bmatrix} linear_r \\ linear_g \\ linear_b \end{bmatrix} = \begin{bmatrix} redMatrixColumn_X & greenMatrixColumn_X & blueMatrixColumn_X \\ redMatrixColumn_Y & greenMatrixColumn_Y & blueMatrixColumn_Y \\ redMatrixColumn_Z & greenMatrixColumn_Z & blueMatrixColumn_Z \end{bmatrix}^{-1} \begin{bmatrix} connection_X \\ connection_Y \\ connection_Z \end{bmatrix} \quad (5)$$

$$\begin{aligned} device_r &= \text{redTRC}^{-1}[1](linear_r > 1) \\ device_r &= \text{redTRC}^{-1}[linear_r](0 \le linear_r \le 1) \\ device_r &= \text{redTRC}^{-1}[0](linear_r < 0) \end{aligned} \quad (6)$$

$$\begin{aligned} device_g &= \text{greenTRC}^{-1}[1](linear_g > 1) \\ device_g &= \text{greenTRC}^{-1}[linear_g](0 \le linear_g \le 1) \\ device_g &= \text{greenTRC}^{-1}[0](linear_g < 0) \end{aligned} \quad (7)$$

$$\begin{aligned} device_b &= \text{blueTRC}^{-1}[1](linear_b > 1) \\ device_b &= \text{blueTRC}^{-1}[linear_b](0 \le linear_b \le 1) \\ device_b &= \text{blueTRC}^{-1}[0](linear_b < 0) \end{aligned} \quad (8)$$

Only the CIEXYZ encoding of the profile connection space can be used with matrix/TRC models. This profile may be used for any device which has a three-component color space suitably related to XYZ by this model. An AToB0Tag must be included if the CIELAB encoding of the profile connection space is to be used.

© 2003 ICC

Additional multidimensional tags (AToB0Tag, AToB1Tag, AToB2Tag, BToA0Tag, BToA1Tag, BToA2Tag) may also be included. If these are present, their usage shall be as defined in Table 20.

An additional multidimensional gamut tag (gamutTag) may be included. The usage of this tag is identical as in output profiles (Section 6.3.3.2).

6.3.1.3 N-component LUT-based Input Profiles

Table 23 — N-component LUT-based input profile required tags

Tag Name	General Description
profileDescriptionTag	Structure containing invariant and localizable versions of the profile name for display
AToB0Tag	Device to PCS: 8-bit or 16-bit data
mediaWhitePointTag	Media XYZ white point
copyrightTag	Profile copyright information
chromaticAdaptationTag	Converts XYZ color from the actual illumination source to PCS illuminant. Required only if the actual illumination source is not D50.

The AToB0Tag represents a device model that can use a multi-dimensional lookup table. The model is described in detail in the tag descriptions and the general lookup table tag element structures.

Additional multidimensional tags (AToB1Tag, AToB2Tag, BToA0Tag, BToA1Tag, BToA2Tag) may also be included. If these are present, their usage shall be as defined in Table 20.

In addition, a gamutTag may be included. The usage of this tag is identical as in output profiles (Section 6.3.3.2).

6.3.2 Display Profile

This profile represents display devices such as monitors.

6.3.2.1 Monochrome Display Profiles

Table 24 — Monochrome display profile required tags

Tag Name	General Description
profileDescriptionTag	Structure containing invariant and localizable versions of the profile name for display
grayTRCTag	Gray tone reproduction curve
mediaWhitePointTag	Media XYZ white point
copyrightTag	Profile copyright information
chromaticAdaptationTag	Converts XYZ color from the actual illumination source to PCS illuminant. Required only if the actual illumination source is not D50.

The mathematical model implied by this data is:

© 2003 ICC

$$connection = \text{grayTRC}[device] \tag{9}$$

This represents a simple tone reproduction curve adequate for most monochrome display devices. The *connection* values in this equation should represent the achromatic channel of the profile connection space in the range of 0 to 1,0 where 0 represents black and 1,0 represents white. The PCS value is derived by multiplying the D50 white point by the normalized TRC value between 0 and 1,0. If the inverse of this is desired, then the following equation is used:

$$device = \text{grayTRC}^{-1}[connection] \tag{10}$$

AToB0Tag, AToB1Tag, AToB2Tag, BToA0Tag, BToA1Tag, BToA2Tag may be included in monochrome profiles. If these are present, their usage shall be as defined in Table 20.

6.3.2.2 Three-component Matrix-based Display Profiles

This profile type is often used with devices whose nominal color space is RGB. .

Table 25 — Three-component matrix-based display profile required tags

Tag Name	General Description
profileDescriptionTag	Structure containing invariant and localizable versions of the profile name for display
redMatrixColumnTag	The first column in the matrix used in TRC/matrix transforms (This column is combined with the linear red channel during the matrix multiplication)
greenMatrixColumnTag	The second column in the matrix used in TRC/matrix transforms (This column is combined with the linear green channel during the matrix multiplication)
blueMatrixColumnTag	The third column in the matrix used in TRC/matrix transforms (This column is combined with the linear blue channel during the matrix multiplication)
redTRCTag	Red channel tone reproduction curve
greenTRCTag	Green channel tone reproduction curve
blueTRCTag	Blue channel tone reproduction curve
mediaWhitePointTag	Media XYZ white point
copyrightTag	Profile copyright information
chromaticAdaptationTag	Converts XYZ color from the actual illumination source to PCS illuminant. Required only if the actual illumination source is not D50.

This model describes transformation from device color space to PCS (see Annex A for PCS description). The transformation is based on three non-interdependent per-channel tone reproduction curves to convert between non-linear and linear RGB values and a 3x3 matrix to convert between linear RGB values and relative XYZ values. The mathematical model implied by this data is:

© 2003 ICC

$$\begin{aligned} linear_r &= \text{redTRC}[device_r] \\ linear_g &= \text{greenTRC}[device_g] \\ linear_b &= \text{blueTRC}[device_b] \end{aligned} \tag{11}$$

$$\begin{bmatrix} connection_X \\ connection_Y \\ connection_Z \end{bmatrix} = \begin{bmatrix} redMatrixColumn_X & greenMatrixColumn_X & blueMatrixColumn_X \\ redMatrixColumn_Y & greenMatrixColumn_Y & blueMatrixColumn_Y \\ redMatrixColumn_Z & greenMatrixColumn_Z & blueMatrixColumn_Z \end{bmatrix} \begin{bmatrix} linear_r \\ linear_g \\ linear_b \end{bmatrix} \tag{12}$$

This represents a simple linearization followed by a linear mixing model. The three tone reproduction curves linearize the raw values with respect to the luminance (Y) dimension of the CIEXYZ encoding of the profile connection space. The 3x3 matrix converts these linearized values into XYZ values for the CIEXYZ encoding of the profile connection space. The inverse model is given by the following equations:

$$\begin{bmatrix} linear_r \\ linear_g \\ linear_b \end{bmatrix} = \begin{bmatrix} redMatrixColumn_X & greenMatrixColumn_X & blueMatrixColumn_X \\ redMatrixColumn_Y & greenMatrixColumn_Y & blueMatrixColumn_Y \\ redMatrixColumn_Z & greenMatrixColumn_Z & blueMatrixColumn_Z \end{bmatrix}^{-1} \begin{bmatrix} connection_X \\ connection_Y \\ connection_Z \end{bmatrix} \tag{13}$$

$$\begin{aligned} device_r &= \text{redTRC}^{-1}[1](linear_r > 1) \\ device_r &= \text{redTRC}^{-1}[linear_r](0 \le linear_r \le 1) \\ device_r &= \text{redTRC}^{-1}[0](linear_r < 0) \end{aligned} \tag{14}$$

$$\begin{aligned} device_g &= \text{greenTRC}^{-1}[1](linear_g > 1) \\ device_g &= \text{greenTRC}^{-1}[linear_g](0 \le linear_g \le 1) \\ device_g &= \text{greenTRC}^{-1}[0](linear_g < 0) \end{aligned} \tag{15}$$

© 2003 ICC

$$
\begin{aligned}
device_b &= \text{blueTRC}^{-1}[1](linear_b > 1) \\
device_b &= \text{blueTRC}^{-1}[linear_b](0 \le linear_b \le 1) \\
device_b &= \text{blueTRC}^{-1}[0](linear_b < 0)
\end{aligned}
\tag{16}
$$

Only the CIEXYZ encoding of the profile connection space can be used with matrix/TRC models. This profile may be used for any device which has a three-component color space suitably related to XYZ by this model. An AToB0Tag must be included if the CIELAB encoding of the profile connection space is to be used.

Additional multidimensional tags (AToB0Tag, AToB1Tag, AToB2Tag, BToA0Tag, BToA1Tag, BToA2Tag) may also be included. If these are present, their usage shall be as defined in Table 20.

An additional multidimensional gamut tag (gamutTag) may be included. The usage of this tag is identical as in output profiles (Section 6.3.3.2).

6.3.2.3 N-Component LUT-Based Display Profiles

Table 26 — N-component LUT-based display profile required tags

Tag Name	General Description
profileDescriptionTag	Structure containing invariant and localizable versions of the profile name for display
AToB0Tag	Device to PCS: 8-bit or 16-bit data: intent of 0
BToA0Tag	PCS to Device space: 8-bit or 16-bit data: intent of 0
mediaWhitePointTag	Media XYZ white point
copyrightTag	Profile copyright information
chromaticAdaptationTag	Converts XYZ color from the actual illumination source to PCS illuminant. Required only if the actual illumination source is not D50.

The AToB0Tag and the BToA0Tag represent a device model that can use a multi-dimensional lookup table. The model is described in detail in the tag descriptions and the general lookup table tag element structures.

Additional multidimensional tags (AToB1Tag, AToB2Tag, BToA1Tag, BToA2Tag) may also be included. If these are present, their usage shall be as defined in Table 20.

An additional multidimensional gamut tag (gamutTag) may be included. The usage of this tag is identical as in output profiles (Section 6.3.3.2).

6.3.3 Output Profile

This profile represents output devices such as printers and film recorders.

© 2003 ICC

6.3.3.1 Monochrome Output Profiles

Table 27 — Monochrome output profile required tags

Tag Name	General Description
profileDescriptionTag	Structure containing invariant and localizable versions of the profile name for display
grayTRCTag	Gray tone reproduction curve
mediaWhitePointTag	Media XYZ white point
copyrightTag	Profile copyright information
chromaticAdaptationTag	Converts XYZ color from the actual illumination source to PCS illuminant. Required only if the actual illumination source is not D50.

The mathematical model implied by this data is:

$$connection = \text{grayTRC}[device] \tag{17}$$

This represents a simple tone reproduction curve adequate for most monochrome output devices. The *connection* values in this equation should represent the achromatic channel of the profile connection space in the range of 0 to 1,0 where 0 represents black and 1,0 represents white. The PCS value is derived by multiplying the D50 white point by the normalized TRC value between 0 and 1,0. If the inverse of this is desired, then the following equation is used:

$$device = \text{grayTRC}^{-1}[connection] \tag{18}$$

AToB0Tag, AToB1Tag, AToB2Tag, BToA0Tag, BToA1Tag, BToA2Tag may be included in monochrome profiles. If these are present, their usage shall be as defined in Table 20.

NOTE The output values are the control values and not the "K" (black) values.

© 2003 ICC

6.3.3.2 Color Output Profiles

Table 28 — Color output profile required tags

Tag Name	General Description
profileDescriptionTag	Structure containing invariant and localizable versions of the profile name for display
AToB0Tag	Device to PCS: 8-bit or 16-bit data: intent of 0
BToA0Tag	PCS to Device space: 8-bit or 16-bit data: intent of 0
gamutTag	Out of Gamut: 8-bit or 16-bit data
AToB1Tag	Device to PCS: 8-bit or 16-bit data: intent of 1
BToA1Tag	PCS to Device space: 8-bit or 16-bit data: intent of 1
AToB2Tag	Device to PCS: 8-bit or 16-bit data: intent of 2
BToA2Tag	PCS to Device space: 8-bit or 16-bit data: intent of 2
mediaWhitePointTag	Media XYZ white point
colorantTableTag	Colorants used in the profile, required if Color Space Signature is xCLR (e.g., 3CLR)
copyrightTag	Profile copyright information
chromaticAdaptationTag	Converts XYZ color from the actual illumination source to PCS illuminant. Required only if the actual illumination source is not D50.

The intent values used in Table 28 directly correlate to the rendering intent values defined in Table 18.

Each of the first three intents in Table 18 are associated with a specific tag. The fourth intent, ICC-absolute colorimetry, is obtained by using the media-relative colorimetric intent tag and the mediaWhitePointTag as described in Annex A.

It is permissible to reference the same tag data for all of these intents and to use the media-relative colorimetric intent tag when ICC-absolute colorimetry is specified.

The AToB0Tag, BToA0Tag, AToB1Tag, BToA1Tag, AToB2Tag, and BToA2Tag represent a device model that can use a multi-dimensional lookup table. The model is described in detail in the tag descriptions and the general lookup table tag element structures.

The colorantTableTag (Section 6.4.14) is a required tag to specify the names and XYZ or L*a*b* values of the colorants for the xCLR multi-dimensional color spaces, as these names are not otherwise implicit in the choice of the color space.

© 2003 ICC

6.3.4 Additional Profile Formats

6.3.4.1 DeviceLink Profile

This profile represents a one-way link or connection between devices. It does not represent any device model nor can it be embedded into images.

Table 29 — DeviceLink profile required tags

Tag Name	General Description
profileDescriptionTag	Structure containing invariant and localizable versions of the profile name for display
AToB0Tag	Actual transformation parameter structure; 8-bit or 16-bit data
profileSequenceDescTag	An array of descriptions of the profile sequence
colorantTableTag	Colorants used in the profile, required if Color Space Signature is xCLR (e.g., 3CLR)
copyrightTag	Profile copyright information

The single AToB0Tag may contain data for any one of the four possible rendering intents. The rendering intent used is indicated in the header of the profile.

The AToB0Tag represents a device model that can use a multi-dimensional lookup table. The model is described in detail in the tag descriptions and the general lookup table tag element structures.

The DeviceLink profile is a pre-evaluated transform that cannot be undone.

The color space of data in the DeviceLink profile will be the same as the color space of the data of the first profile in the sequence. The profile connection space will be the same as the color space of the data of the last profile in the sequence.

The colorantTableTag (Section 6.4.14) is a required tag to specify the names and XYZ or L*a*b* values of the colorants for the xCLR multi-dimensional color spaces, as these names are not otherwise implicit in the choice of the color space.

© 2003 ICC

6.3.4.2 ColorSpace Conversion Profile

This profile provides the relevant information to perform a color space transformation between the non-device color spaces and the PCS. It does not represent any device model. ColorSpace Conversion profiles may be embedded in images.

Table 30 — ColorSpace conversion profile required tags

Tag Name	General Description
profileDescriptionTag	Structure containing invariant and localizable versions of the profile name for display
BToA0Tag	Inverse transformation parameter structure; 8-bit or 16-bit data
AToB0Tag	Actual transformation parameter structure; 8-bit or 16-bit data
mediaWhitePointTag	Media XYZ white point
copyrightTag	Profile copyright information
chromaticAdaptationTag	Converts XYZ color from the actual illumination source to PCS illuminant. Required only if the actual illumination source is not D50.

The AToB0Tag and BToA0Tag represent a model that can use a multi-dimensional lookup table. The model is described in detail in the tag descriptions and the general lookup table tag element structures.

For color transformation profiles, the device profile dependent fields are set to zero if irrelevant.

Additional multidimensional tags (AToB1Tag, AToB2Tag., BToA1Tag, BToA2Tag) may also be included. If these are present, their usage shall be as defined in Table 20.

In addition, a gamutTag may be included. The usage of this tag is identical as in output profiles (Section 6.3.3.2).

© 2003 ICC

6.3.4.3 Abstract Profile

This profile represents abstract transforms and does not represent any device model. Color transformations using abstract profiles are performed from PCS to PCS. Abstract profiles cannot be embedded in images.

Table 31 — Abstract profile required tags

Tag Name	General Description
profileDescriptionTag	Structure containing invariant and localizable versions of the profile name for display
AToB0Tag	Actual transformation parameter structure; 8-bit or 16-bit data
mediaWhitePointTag	Media XYZ white point
copyrightTag	Profile copyright information
chromaticAdaptationTag	Converts XYZ color from the actual illumination source to PCS illuminant. Required only if the actual illumination source is not D50.

The AToB0Tag represents a PCS to PCS model that can use a multi-dimensional lookup table. The model is described in detail in the tag descriptions and the general lookup table tag element structures.

6.3.4.4 Named Color Profile

Named color profiles can be thought of as sibling profiles to device profiles. For a given device there would be one or more device profiles to handle process color conversions and one or more named color profiles to handle named colors. There might be multiple named color profiles to account for different consumables or multiple named color vendors. This profile provides a PCS and optional device representation for a list of named colors. Named color profiles are device-specific in that their data is shaped for a particular device.

Table 32 — Named color required tags

Tag Name	General Description
profileDescriptionTag	Structure containing invariant and localizable versions of the profile name for display
namedColor2Tag	PCS and optional device representation for named colors
mediaWhitePointTag	Media XYZ white point
copyrightTag	Profile copyright information
chromaticAdaptationTag	Converts XYZ color from the actual illumination source to PCS illuminant. Required only if the actual illumination source is not D50.

The namedColor2Tag provides a PCS and optional device representation for each named color in a list of named colors. The PCS representation is provided to support general color management functionality. It is very useful for display and emulation of the named colors.

© 2003 ICC

When using a named color profile with the device for which it is intended, the device representation of the color specifies the exact device coordinates for each named color. The PCS representation in conjunction with the device's output profile can provide an approximation of these exact coordinates. The exactness of this approximation is a function of the accuracy of the output profile and the color management system performing the transformations.

The combination of the PCS and device representations provides for flexibility with respect to accuracy and portability.

6.4 Tag descriptions

This clause specifies the individual tags used to create all possible portable profiles in the ICC Profile Format. The appropriate tag typing is indicated with each individual tag description. Note that the signature indicates only the type of data and does not imply anything about the use or purpose for which the data is intended.

Any of the tags in Table 33 can be used as optional tags if they are not used in the required set for a particular profile and are not specifically excluded in a profile definition.

The tag types are described in Section 6.5.

Table 33 — Tag list (Part 1 of 2)

Tag Name	General Description
AToB0Tag	Multidimensional transformation structure
AToB1Tag	Multidimensional transformation structure
AToB2Tag	Multidimensional transformation structure
blueMatrixColumnTag	Relative XYZ values of blue phosphor or colorant
blueTRCTag	The third column in the matrix used in TRC/matrix transforms (This column is combined with the linear blue channel during the matrix multiplication)
BToA0Tag	Multidimensional transformation structure
BToA1Tag	Multidimensional transformation structure
BToA2Tag	Multidimensional transformation structure
calibrationDateTimeTag	Profile calibration date and time
charTargetTag	Characterization target such as IT8/7.2
chromaticAdaptationTag	Converts XYZ color from the actual illumination source to PCS illuminant. Required only if the actual illumination source is not D50.
chromaticityTag	Set of phosphor/colorant chromaticity
colorantOrderTag	Identifies the laydown order of colorants
colorantTableTag	Identifies the colorants used in the profile
copyrightTag	Profile copyright information
deviceMfgDescTag	Displayable description of device manufacturer
deviceModelDescTag	Displayable description of device model
gamutTag	Out of gamut: 8-bit or 16-bit data

© 2003 ICC

Table 33 — Tag list (Part 2 of 2)

grayTRCTag	Gray tone reproduction curve
greenMatrixColumnTag	The second column in the matrix used in TRC/matrix transforms (This column is combined with the linear green channel during the matrix multiplication)
greenTRCTag	Green channel tone reproduction curve
luminanceTag	Absolute luminance for emissive device
measurementTag	Alternative measurement specification information
mediaBlackPointTag	Media XYZ black point
mediaWhitePointTag	Media XYZ white point
namedColor2Tag	PCS and optional device representation for named colors
outputResponseTag	Description of the desired device response
preview0Tag	Preview transformation: 8-bit or 16-bit data
preview1Tag	Preview transformation: 8-bit or 16-bit data
preview2Tag	Preview transformation: 8-bit or 16-bit data
profileDescriptionTag	Structure containing invariant and localizable versions of the profile name for displays
profileSequenceDescTag	An array of descriptions of the profile sequence
redMatrixColumnTag	The first column in the matrix used in TRC/matrix transforms (This column is combined with the linear red channel during the matrix multiplication)
redTRCTag	Red channel tone reproduction curve
technologyTag	Device technology information such as LCD, CRT, Dye Sublimation, etc.
viewingCondDescTag	Viewing condition description
viewingConditionsTag	Viewing condition parameters

6.4.1 AToB0Tag

Tag Type: lut8Type or lut16Type or lutAtoBType
Tag Signature: ‘A2B0’ (41324230h)

This tag defines a color transform from Device to PCS using lookup table tag element structures. The processing mechanisms are described in lut8Type or lut16Type or lutAtoBType.

6.4.2 AToB1Tag

Tag Type: lut8Type or lut16Type or lutAtoBType
Tag Signature: ‘A2B1’ (41324231h)

This tag defines a color transform from Device to PCS using lookup table tag element structures. The processing mechanisms are described in lut8Type or lut16Type or lutAtoBType.

© 2003 ICC

6.4.3 AToB2Tag

Tag Type: lut8Type or lut16Type or lutAtoBType
Tag Signature: 'A2B2' (41324232h)

This tag defines a color transform from Device to PCS using lookup table tag element structures. The processing mechanisms are described in lut8Type or lut16Type or lutAtoBType.

6.4.4 blueMatrixColumnTag

Tag Type: XYZType
Tag Signature: 'bXYZ' (6258595Ah)

The third column in the matrix used in TRC/matrix transforms.

6.4.5 blueTRCTag

Tag Type: curveType or parametricCurveType
Tag Signature: 'bTRC' (62545243h)

Blue channel tone reproduction curve. The first element represents no colorant (white) or phosphors (black) and the last element represents 100 percent colorant (blue) or 100 percent phosphor (blue).

6.4.6 BToA0Tag

Tag Type: lut8Type or lut16Type or lutBtoAType
Tag Signature: 'B2A0' (42324130h)

This tag defines a color transform from PCS to Device using the lookup table tag element structures. The processing mechanisms are described in lut8Type or lut16Type or lutBtoAType.

6.4.7 BToA1Tag

Tag Type: lut8Type or lut16Type or lutBtoAType
Tag Signature: 'B2A1' (42324131h)

This tag defines a color transform from PCS to Device using the lookup table tag element structures. The processing mechanisms are described in lut8Type or lut16Type or lutBtoAType.

6.4.8 BToA2Tag

Tag Type: lut8Type or lut16Type or lutBtoAType
Tag Signature: 'B2A2' (42324132h)

This tag defines a color transform from PCS to Device using the lookup table tag element structures. The processing mechanisms are described in lut8Type or lut16Type or lutBtoAType.

6.4.9 calibrationDateTimeTag

Tag Type: dateTimeType
Tag Signature: 'calt' (63616C74h)

© 2003 ICC

Profile calibration date and time. Initially, this tag matches the contents of the profile header's creation date/time field. This allows applications and utilities to verify if this profile matches a vendor's profile and how recently calibration has been performed.

6.4.10 charTargetTag

Tag Type: textType
Tag Signature: 'targ' (74617267h)

This tag contains the name of the registered characterization data set, or it contains the measurement data for a characterization target. This tag is provided so that distributed utilities can identify the underlying characterization data, create transforms "on the fly" or check the current performance against the original device performance.

The first seven characters of the text shall identify the nature of the characterization data.

If the first seven characters are "ICCHDAT", then the remainder of the text shall be a single space followed by the Reference Name of a characterization data set in the ICC Characterization Data Registry and terminated with a NULL byte (00h). The Reference Name in the text must match exactly (including case) the Reference Name in the registry.

If the first seven characters match one of the identifiers defined in an ANSI or ISO standard, then the tag embeds the exact data file format defined in that standard. Each of these file formats contains an identifying character string as the first seven characters of the format, allowing an external parser to determine which data file format is being used. This provides the facilities to include a wide range of targets using a variety of measurement specifications in a standard manner.

NOTE: It is highly recommended that the profileDescriptionTag also include an identification of the characterization data that was used in the creation of the profile (e.g. "Based on CGATS TR 001").

6.4.11 chromaticAdaptationTag

Tag Type: s15Fixed16ArrayType
Tag Signature: 'chad' (63686164h)

This tag converts an XYZ color, measured at a device's specific illumination conditions, to an XYZ color in the PCS illumination conditions after complete adaptation.

The tag reflects a survey of the currently used methods of conversion, all of which can be formulated as a matrix transformation (see Annex E). Such a 3 by 3 chromatic adaptation matrix is organized as a 9-element array of signed 15.16 numbers (s15Fixed16ArrayType tag). Similarly as in the other occurrences of a 3 by 3 matrix in the ICC tags, the dimension corresponding to the matrix rows varies least rapidly while the one corresponding to the matrix columns varies most rapidly.

$$array = \begin{bmatrix} a0 & a1 & a2 & a3 & a4 & a5 & a6 & a7 & a8 \end{bmatrix} \tag{19}$$

$$\begin{bmatrix} Xpcs \\ Ypcs \\ Zpcs \end{bmatrix} = \begin{bmatrix} a0 & a1 & a2 \\ a3 & a4 & a5 \\ a6 & a7 & a8 \end{bmatrix} \begin{bmatrix} Xsrc \\ Ysrc \\ Zsrc \end{bmatrix} \tag{20}$$

© 2003 ICC

Where *XYZsrc* represents the measured value in the actual device viewing condition and *XYZpcs* represents the chromatically adapted value in the PCS.

The chromatic adaptation matrix is a combination of three separate conversions:

1) Conversion of source CIE XYZ tristimulus values to cone response tristimulus values.

2) Adjustment of the cone response values for an observer's chromatic adaptation.

3) Conversion of the adjusted cone response tristimulus back to CIE XYZ values.

6.4.12 chromaticityTag

Tag Type: chromaticityType
Tag Signature: 'chrm' (6368726Dh)

The data and type of phosphor/colorant chromaticity set.

6.4.13 colorantOrderTag

Tag Type: colorantOrderType
Tag Signature: 'clro' (636C726Fh)

This tag specifies the laydown order of colorants.

6.4.14 colorantTableTag

Tag Type: colorantTableType
Tag Signature: 'clrt' (636C7274h)

This tag identifies the colorants used in the profile by a unique name and an XYZ or L*a*b* value.

This is a required tag for profiles where the color space defined in the header is xCLR, where x is one of the allowed numbers from 2 through Fh, per Table 13. See Section 6.3.3.2, Section 6.3.4.1.

6.4.15 copyrightTag

Tag Type: multiLocalizedUnicodeType
Tag Signature: 'cprt' (63707274h)

This tag contains the text copyright information for the profile.

6.4.16 deviceMfgDescTag

Tag Type: multiLocalizedUnicodeType
Tag Signature: 'dmnd' (646D6E64h)

Structure containing invariant and localizable versions of the device manufacturer for display. The content of this structure is described in 6.5.12.

6.4.17 deviceModelDescTag

Tag Type: multiLocalizedUnicodeType
Tag Signature: 'dmdd' (646D6464h)

© 2003 ICC

Structure containing invariant and localizable versions of the device model for display. The content of this structure is described in 6.5.12.

6.4.18 gamutTag

Tag Type: lut8Type or lut16Type or lutBtoAType
Tag Signature: 'gamt' (67616D74h)

Out of gamut tag. The processing mechanisms are described in lut8Type or lut16Type or lutBtoAType.

This tag takes PCS values as its input and produces a single channel of output. If the output value is 0, the PCS color is in-gamut. If the output is non-zero, the PCS color is out-of-gamut, with the output value "n+1" being at least as far out of gamut as the output value "n".

6.4.19 grayTRCTag

Tag Type: curveType or parametricCurveType
Tag Signature: 'kTRC' (6B545243h)

Gray tone reproduction curve. The tone reproduction curve provides the necessary information to convert between a single device channel and the CIEXYZ encoding of the profile connection space. The first element represents black and the last element represents white.

6.4.20 greenMatrixColumnTag

Tag Type: XYZType
Tag Signature: 'gXYZ' (6758595Ah)

The second column in the matrix used in TRC/matrix transforms.

6.4.21 greenTRCTag

Tag Type: curveType or parametricCurveType
Tag Signature: 'gTRC' (67545243h)

Green channel tone reproduction curve. The first element represents no colorant (white) or phosphors (black) and the last element represents 100 percent colorant (green) or 100 percent phosphor (green).

6.4.22 luminanceTag

Tag Type: XYZType
Tag Signature: 'lumi' (6C756D69h)

Absolute luminance of emissive devices in candelas per square meter as described by the Y channel. The X and Z channels are ignored in all cases.

6.4.23 measurementTag

Tag Type: measurementType
Tag Signature: 'meas' (6D656173h)

Alternative measurement specification such as a D65 illuminant instead of the default D50.

© 2003 ICC

6.4.24 mediaBlackPointTag

Tag Type: XYZType
Tag Signature: 'bkpt' (626B7074h)

This tag specifies the media black point and contains the CIE 1931 XYZ colorimetry of the black point of the actual medium.

NOTE Previous revisions of this specification contained an error indicating that this tag is used to calculate ICC-absolute colorimetry. This is not the case.

6.4.25 mediaWhitePointTag

Tag Type: XYZType
Tag Signature: 'wtpt' (77747074h)

This tag, which is used for generating ICC-absolute colorimetric intent, specifies the XYZ tristimulus values of the media white point. If the media is measured under an illumination source which has a chromaticity other than D50, the measured values must be adjusted to D50 using the chromaticAdaptationTag matrix before recording in the tag. For reflecting and transmitting media, the tag values are specified relative to the perfect diffuser (which is normalized to a Y value of 1,0) for illuminant D50. For displays, the values specified must be those of D50 (i.e. 0,9642, 1,0 0,8249) normalized such that Y = 1,0.

See Annex A for a more complete description of the use of the media white point.

6.4.26 namedColor2Tag

Tag Type: namedColor2Type
Tag Signature: 'ncl2' (6E636C32h)

Named color information providing a PCS and optional device representation for a list of named colors.

6.4.27 outputResponseTag

Tag Type: responseCurveSet16Type
Tag Signature: 'resp' (72657370h)

Structure containing a description of the device response for which the profile is intended. The content of this structure is described in 6.5.16.

NOTE The user's attention is called to the possibility that the use of this tag for device calibration may require use of an invention covered by patent rights. By publication of this specification, no position is taken with respect to the validity of this claim or of any patent rights in connection therewith. The patent holder has, however, filed a statement of willingness to grant a license under these rights on reasonable and nondiscriminatory terms and conditions to applicants desiring to obtain such a license. Details may be obtained from the publisher.

6.4.28 preview0Tag

Tag Type: lut8Type or lut16Type or lutBtoAType
Tag Signature: 'pre0' (70726530h)

Preview transformation from PCS to device space and back to the PCS. The processing mechanisms are described in lut8Type or lut16Type or lutBtoAType.

© 2003 ICC

This tag contains the combination of tag B2A0 and tag A2B1.

6.4.29 preview1Tag

Tag Type: lut8Type or lut16Type or lutBtoAType
Tag Signature: 'pre1' (70726531h)

Preview transformation from the PCS to device space and back to the PCS. The processing mechanisms are described in lut8Type or lut16Type or lutBtoAType.

This tag contains the combination of tag B2A1 and tag A2B1.

6.4.30 preview2Tag

Tag Type: lut8Type or lut16Type or lutBtoAType
Tag Signature: 'pre2' (70726532h)

Preview transformation from PCS to device space and back to the PCS. The processing mechanisms are described in lut8Type or lut16Type or lutBtoAType.

This tag contains the combination of tag B2A2 and tag A2B1.

6.4.31 profileDescriptionTag

Tag Type: multiLocalizedUnicodeType
Tag Signature: 'desc' (64657363h)

Structure containing invariant and localizable versions of the profile description for display. The content of this structure is described in 6.5.12. This invariant description has no fixed relationship to the actual profile disk file name.

6.4.32 profileSequenceDescTag

Tag Type: profileSequenceDescType
Tag Signature: 'pseq' (70736571h)

Structure containing a description of the profile sequence from source to destination, typically used with the DeviceLink profile. The content of this structure is described in 6.5.15.

6.4.33 redMatrixColumnTag

Tag Type: XYZType
Tag Signature: 'rXYZ' (7258595Ah)

The first column in the matrix used in TRC/matrix transforms.

6.4.34 redTRCTag

Tag Type: curveType or parametricCurveType
Tag Signature: 'rTRC' (72545243h)

Red channel tone reproduction curve. The first element represents no colorant (white) or phosphors (black) and the last element represents 100 percent colorant (red) or 100 percent phosphor (red).

© 2003 ICC

6.4.35 technologyTag

Tag Type: signatureType
Tag Signature: 'tech' (74656368h)

Device technology information such as CRT, Dye Sublimation, etc. The encoding is such that:

Table 34 — Technology signatures

Technology	Signature	Hex Encoding
Film Scanner	'fscn'	6673636Eh
Digital Camera	'dcam'	6463616Dh
Reflective Scanner	'rscn'	7273636Eh
Ink Jet Printer	'ijet'	696A6574h
Thermal Wax Printer	'twax'	74776178h
Electrophotographic Printer	'epho'	6570686Fh
Electrostatic Printer	'esta'	65737461h
Dye Sublimation Printer	'dsub'	64737562h
Photographic Paper Printer	'rpho'	7270686Fh
Film Writer	'fprn'	6670726Eh
Video Monitor	'vidm'	7669646Dh
Video Camera	'vidc'	76696463h
Projection Television	'pjtv'	706A7476h
Cathode Ray Tube Display	'CRT '	43525420h
Passive Matrix Display	'PMD '	504D4420h
Active Matrix Display	'AMD '	414D4420h
Photo CD	'KPCD'	4B504344h
PhotoImageSetter	'imgs'	696D6773h
Gravure	'grav'	67726176h
Offset Lithography	'offs'	6F666673h
Silkscreen	'silk'	73696C6Bh
Flexography	'flex'	666C6578h

6.4.36 viewingCondDescTag

Tag Type: multiLocalizedUnicodeType
Tag Signature: 'vued' (76756564h)

Structure containing invariant and localizable versions of the viewing conditions. The content of this structure is described in 6.5.12.

© 2003 ICC

6.4.37 viewingConditionsTag

Tag Type: viewingConditionsType
Tag Signature: ‘view’ (76696577h)

Viewing conditions parameters. The content of this structure is described in 6.5.25.

6.5 Tag type definitions

This clause specifies the type and structure definitions used to create all of the individual tagged elements in the ICC Profile Format. The data type description identifiers are indicated at the right margin of each data or structure definition. An effort was made to make sure one-byte, two-byte and four-byte data lies on one-byte, two-byte and four-byte boundaries respectively. This required occasionally including extra spaces indicated with “reserved for padding” in some tag type definitions. Value 0 is defined to be of “unknown value” for all enumerated data structures.

All tags, including private tags, have as their first four bytes (0..3) a tag signature (a 4-byte sequence) to identify to profile readers what kind of data is contained within a tag. This encourages tag type reuse and allows profile parsers to reuse code when tags use common tag types. The second four bytes (4..7) are reserved for future expansion and must be set to 0 in this version of the specification. Each new tag signature and tag type signature must be registered with the International Color Consortium (see 0.7) in order to prevent signature collisions.

Where not specified otherwise, the least-significant 16 bits of all 32-bit flags in the type descriptions below are reserved for use by the International Color Consortium.

When 7-bit ASCII text representation is specified in types below, each individual character is encoded in 8 bits with the most-significant bit set to zero. The details are presented in 5.3.11.

6.5.1 chromaticityType

The chromaticityType information provides basic chromaticity data and type of phosphors or colorants of a monitor to applications and utilities. The byte stream is given below..

Table 35 — chromaticityType encoding

Byte Offset	Content	Encoded as...
0..3	’chrm’ (6368726Dh) type signature	
4..7	reserved, must be set to 0	
8..9	Number of Device Channels	uInt16Number
10..11	encoded value of phosphor or colorant type	see below
12..19	CIE xy coordinate values of channel 1	u16Fixed16Number[2]
20..27	CIE xy coordinate values of channel 2	u16Fixed16Number[2]
28..35	CIE xy coordinate values of channel 3	u16Fixed16Number[2]
36..end	CIE xy coordinate values of other channels (if needed)	u16Fixed16Number[...]

When using this type, it is necessary to assign each color space component to device channel. Table 43 in 6.5.7 shows these assignments.

© 2003 ICC

The encoding for the phosphor or colorant type field is such that:.

Table 36 — Phosphor or colorant encoding

Phosphor or Colorant Type	Encoded Value	Channel 1 x,y	Channel 2 x,y	Channel 3 x,y
unknown	0000h	any	any	any
ITU-R BT.709	0001h	(0,640, 0,330)	(0,300, 0,600)	(0,150, 0,060)
SMPTE RP145-1994	0002h	(0,630, 0,340)	(0,310, 0,595)	(0,155, 0,070)
EBU Tech.3213-E	0003h	(0,64 0,33)	(0,29, 0,60)	(0,15, 0,06)
P22	0004h	(0,625, 0,340)	(0,280, 0,605)	(0,155, 0,070)

When the encoded value is 0000h, the actual set of chromaticity values shall be described. Otherwise, the chromaticity values shall match the table values for the given phosphor type.

6.5.2 colorantOrderType

Table 37 — colorantOrderType encoding

Byte Offset	Content	Encoded as...
0..3	‘clro’ (636c726fh) type signature	
4..7	reserved, must be set to 0	
8..11	Count of colorants	uInt32Number
12	One-based number of the colorant to be printed first.	
13..n	The remaining count-1 colorants are described in a manner consistent with the first colorant	

This is an optional tag which specifies the laydown order in which colorants will be printed on an n-colorant device. The laydown order may be the same as the channel generation order listed in the colorantTableTag or the channel order of a color space such as CMYK, in which case this tag is not needed.

When this is not the case (for example, ink-towers sometimes use the order KCMY), this tag may be used to specify the laydown order of the colorants.

The size of the array is the same as the number of colorants. The first position in the array contains the number of the first colorant to be laid down, the second position contains the number of the second colorant to be laid down, and so on, until all colorants are listed. The colorant numbers are one-based.

When this tag is used, the "count of colorants" must be in agreement with the color space signature of Section 6.1.5.

© 2003 ICC

6.5.3 colorantTableType

Table 38 — colorantTableType encoding

Byte Offset	Content	Encoded as...
0..3	'clrt' (636c7274h) type signature	
4..7	reserved, must be set to 0	
8..11	Count of colorants	uInt32Number
12..43	First colorant name (32 byte field, null terminated, unused bytes must be set to zero)	7-bit ASCII
44..49	First colorant coordinates in the PCS colorspace of the profile as described in Section 6.1.6 (the Profile Connection Space Signature in the header). Only 16-bit PCS coordinates are allowed.	uInt16Number
50..n	The remaining colorants, if count > 1, described using the format of bytes 12-49 of the first colorant	

The purpose of this tag is to identify the colorants used in the profile by a unique name and an XYZ or L*a*b* value to give the colorant an unambiguous value. The first colorant listed is the colorant of the first device channel of a lut tag. The second colorant listed is the colorant of the second device channel of a lut tag, and so on.

The XYZ or L*a*b* value may also be used to derive the visual density of the colorant, which trapping algorithms may then use to determine overlay values. The visual density is calculated using the formula:

$$\text{Visual Density} = -\log_{10}(Y) \quad \text{if the PCS is in XYZ} \tag{21}$$

If the PCS is in L*a*b*, convert L* to Y using:

$$Y = L^* / 903{,}3 \quad \text{if } L^* < 8 \tag{22}$$

$$Y = ((L^* + 16) / 116)^3 \quad \text{if } L^* >= 8 \tag{23}$$

The colorant coordinates are provided only for convenience and, for many profile classes, should be populated by processing the individual colorants through the AToB1Tag of the profile if this tag exists, otherwise the user must supply the coordinates if this tag is to be used. An individual colorant has the maximum value in the channel corresponding to that colorant and the minimum value in all other channels.

For example, using a 3CLR 8-bit profile, the individual colorant for the first channel would be (255, 0, 0). Processing this color through the AToB1Tag would produce the first colorant coordinates listed in bytes 44-49.

This tag is required for output profiles and named color profiles where the device colorspace (Section 6.1.5, Color Space Signature) is one of the xCLR color spaces. When this tag is used, the "count of colorants" must be in agreement with the color space signature of Section 6.1.5.

© 2003 ICC

6.5.4 curveType

The curveType contains a 4-byte count value and a one-dimensional table of 2-byte values. The byte stream is given below.

Table 39 — curveType encoding

Byte Offset	Content	Encoded as...
0..3	'curv' (63757276h) type signature	
4..7	reserved, must be set to 0	
8..11	count value specifying number of entries that follow	uInt32Number
12..end	actual curve values starting with the zeroth entry and ending with the entry *count*-1.	uInt16Number[...]

The count value specifies the number of entries in the curve table except as follows:

when *count* is 0, then a linear response (slope equal to 1,0) is assumed,

when *count* is 1, then the data entry is interpreted as a simple gamma value encoded as a u8Fixed8Number. Gamma is interpreted canonically and not as an inverse.

when *count* is 2, then the data entries shall be set so that the correct results are obtained when linear interpolation is used to generate intermediate values.

6.5.5 dataType

The dataType is a simple data structure that contains either 7-bit ASCII or binary data, i.e. textType data or transparent 8-bit bytes. The length of the string is obtained by subtracting 12 from the element size portion of the tag itself. If this type is used for ASCII data, it must be terminated with a 00h byte.

Table 40 — dataType encoding

Byte Offset	Content
0..3	'data' (64617461h) type signature
4..7	reserved, must be set to 0
8..11	data flag, 00000000h represents ASCII data, 00000001h represents binary data, other values are reserved for future use
12..end	a string of (element size - 12) ASCII characters or (element size - 12) bytes

© 2003 ICC

6.5.6 dateTimeType

This dateTimeType is a 12-byte value representation of the time and date. The actual values are encoded as a dateTimeNumber described in 5.3.1.

Table 41 — dateTimeType encoding

Byte Offset	Content	Encoded as...
0..3	'dtim' (6474696Dh) type signature	
4..7	reserved, must be set to 0	
8..19	date and time	dateTimeNumber

6.5.7 lut16Type

This structure converts an input color into an output color using tables with 16-bit precision. This type contains four processing elements: a 3 by 3 matrix (only used when the input color space is XYZ), a set of one dimensional input lookup tables, a multidimensional lookup table, and a set of one dimensional output tables. Data is processed using these elements via the following sequence:

© 2003 ICC

(matrix) ⇒ (1d input tables) ⇒ (multidimensional lookup table) ⇒ (1d output tables).

Table 42 — lut16Type encoding

Byte Offset	Content	Encoded as...
0..3	'mft2' (6D667432h) [multi-function table with 2-byte precision] type signature	
4..7	reserved, must be set to 0	
8	Number of Input Channels	uInt8Number
9	Number of Output Channels	uInt8Number
10	Number of CLUT grid points (identical for each side)	uInt8Number
11	Reserved for padding (required to be 00h)	
12..15	Encoded e00 parameter	s15Fixed16Number
16..19	Encoded e01 parameter	s15Fixed16Number
20..23	Encoded e02 parameter	s15Fixed16Number
24..27	Encoded e10 parameter	s15Fixed16Number
28..31	Encoded e11 parameter	s15Fixed16Number
32..35	Encoded e12 parameter	s15Fixed16Number
36..39	Encoded e20 parameter	s15Fixed16Number
40..43	Encoded e21 parameter	s15Fixed16Number
44..47	Encoded e22 parameter	s15Fixed16Number
48..49	Number of input table entries	uInt16Number
50..51	Number of output table entries	uInt16Number
52..n	Input tables	uInt16Number[...]
n+1..m	CLUT values	uInt16Number[...]
m+1..o	Output tables	uInt16Number[...]

The input, output, and grid tables contained in a lut16Type each embodies a one- or multi-dimensional function which maps an input value in the "domain" of the function to an output value in the "range" of the function.

The domain of each of these tables is defined to consist of all real numbers between 0,0 and 65535,0, inclusive. The first entry is located at 0,0, the last entry at 65535,0, and intermediate entries are uniformly spaced using an increment of 65535,0/(M-1). For the input and output tables, M is the number of entries in the table. For the CLUT, M is the number of grid points along each dimension. Note that since the increment of 65535,0/(M-1) is not necessarily an integer, the domain is specified to be over the real numbers rather than restricting it to the integers only.

The range of a function used to generate the contents of a table is likewise defined to be all real numbers between 0,0 and 65535,0, inclusive. Because the contents of a table are encoded using 16 bits of precision, it is necessary to round each real value to the nearest 16-bit integer.

© 2003 ICC

This means that both the domain and range of the functions represented by the elements of the lut16Type as a whole are all real numbers between 0,0 and 65535,0, inclusive. In many situations it is necessary to convert between these 16-bit values and some other bit precision.

See Annex A for additional guidance on this topic.

The matrix is organized as a 3 by 3 array. The dimension corresponding to the matrix rows varies least rapidly and the dimension corresponding to the matrix columns varies most rapidly and is shown in matrix form below.

$$\begin{bmatrix} e00 & e01 & e02 \\ e10 & e11 & e12 \\ e20 & e21 & e22 \end{bmatrix}$$

When using the matrix of an output profile, and the input data is XYZ, we have

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} e00 & e01 & e02 \\ e10 & e11 & e12 \\ e20 & e21 & e22 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (24)$$

Each input X, Y or Z is an unsigned 1.15 number and each matrix entry is a signed 15.16 number. Therefore, each multiplication in the matrix multiply is 1.15 * s15.16 = s16.31 and the final sum is also s16.31 (48 bits). From this sum we take bits 31..16 as the unsigned integer result for X', Y', or Z'. These are then used as the inputs to the input tables of the multidimensional LUT. This normalization is used since the number of fractional bits in the input data must be maintained by the matrix operation.

The matrix is mandated to be an identity matrix unless the input is in the XYZ color space.

The input tables are arrays of 16-bit unsigned values. Each input table consists of a minimum of two and a maximum of 4096 two-byte integers. Each input table entry is appropriately normalized to the range 0-65535. The inputTable is of size (InputChannels * inputTableEntries * 2) bytes. When stored in this tag, the one-dimensional lookup tables are assumed to be packed one after another in the order described below.

The CLUT is organized as an n-dimensional array with a given number of grid points in each dimension, where n is the number of input channels (input tables) in the transform. The dimension corresponding to the first input channel varies least rapidly and the dimension corresponding to the last input channel varies most rapidly. Each grid point value contains m two-byte integers, where m is the number of output channels. The first sequential two-byte integer of the entry contains the function value for the first output function, the second sequential two-byte integer of the entry contains the function value for the second output function, and so on until all the output functions have been supplied. Each two-byte integer in the CLUT is appropriately normalized to the range of 0 - 65535. The equation for computing the byte size of the CLUT is:

$$CLUTSize = (GridPoints^{InputChannels} * OutputChannels * 2) \text{ bytes} \quad (25)$$

The output tables are arrays of 16-bit unsigned values. Each output table consists of a minimum of two and a maximum of 4096 two-byte integers. Each output table entry is appropriately normalized to the range 0 - 65535. The outputTable is of size (OutputChannels * outputTableEntries * 2) bytes. When stored in this tag, the one-dimensional lookup tables are assumed to be packed one after another in the order described in the following paragraph.

© 2003 ICC

If the number of data points in a one-dimensional table, or in a particular dimension of the CLUT, is two, the data for those points shall be set so that the correct results are obtained when linear interpolation is used to generate intermediate values.

When using this type, it is necessary to assign each color space component to an input and output channel. The following table shows these assignments. The channels are numbered according to the order in which their table occurs. Note that additional color spaces can be added simply by defining the signature, channel assignments, and creating the tables.

Table 43 — lut16Type channel encodings

Color Space	Channel 1	Channel 2	Channel 3	Channel 4
‘XYZ ’	X	Y	Z	
‘Lab ’	L	a	b	
‘Luv ’	L	u	v	
‘YCbr’	Y	Cb	Cr	
‘Yxy ’	Y	x	y	
‘RGB ’	R	G	B	
‘GRAY’	K			
‘HSV ’	H	S	V	
‘HLS ’	H	L	S	
‘CMYK’	C	M	Y	K
‘CMY ’	C	M	Y	
‘2CLR’	Ch. 1	Ch. 2		
‘3CLR’	Ch. 1	Ch. 2	Ch. 3	
‘4CLR’	Ch. 1	Ch. 2	Ch. 3	Ch. 4

The color space used on the PCS side of a lut16Type tag (which may be either the input or output space, or both in the case of an abstract profile) is identified by the Profile Connection Space field in the profile header (see 6.1.6). This field does not distinguish between 8-bit and 16-bit PCS encodings. For the lut16Type tag, the 'Lab' signature is defined to specify a legacy 16-bit CIELAB encoding and the 'XYZ ' signature is defined to specify the 16-bit XYZ encoding. Note that this definition only applies to the encoding used at the Profile Connection Space side of the tag. It does NOT apply when these signatures are used on the "Color Space of Data" field in the profile header (see 6.1.5), except in the case of an abstract profile.

For color values that are in the Lab color space on the PCS side of the tag, this tag uses a legacy 16-bit Lab encoding, not the 16-bit CIELAB PCS encoding that is defined in Annex A.2. This encoding is retained for backwards compatibility with profile version 2.

To convert color values from this tag's legacy 16-bit Lab encoding to the 16-bit CIELAB PCS encoding defined in Annex A.2, multiply all values with 65535/65280 (that is, FFFFh/FF00h). Any color values that are in the value range of legacy 16-bit PCS Lab, but not in the 16-bit CIELAB PCS encoding defined in Annex A.2, shall be clipped on a per-component basis when transforming from legacy 16-bit PCS Lab to the 16-bit CIELAB PCS encoding defined in Annex A.2. To convert color values from the 16-bit CIELAB PCS encoding defined in Annex A.2 to this tag's legacy 16-bit Lab encoding, divide all values with 65535/65280.

© 2003 ICC

The L* values have a different encoding than the a* and b* values. The L* encoding is:

Table 44 — L* encoding

Value (L*)	16 bit
0	0000h
100,0	FF00h
100 + (25500/65280)	FFFFh

Although the 16-bit encoding can represent values slightly greater than 100,0, these are not valid PCS L* values and they should not be used.

The a* and b* encoding is:

Table 45 — a* or b* encoding

Value (a* or b*)	16-bit
-128,0	0000h
0	8000h
127,0	FF00h
127 + (255/256)	FFFFh

Note that the 16-bit encoding can represent values slightly greater than 127,0. Since a* and b* have no defined limits, these are valid PCS values.

6.5.8 lut8Type

This structure converts an input color into an output color using tables of 8-bit precision. This type contains four processing elements: a 3 by 3 matrix (only used when the input color space is XYZ), a set of one dimensional input lookup tables, a multidimensional lookup table, and a set of one dimensional output tables. Data is processed using these elements via the following sequence:

(matrix) ⇒ (1d input tables) ⇒ (multidimensional lookup table) ⇒ (1d output tables).

Table 46 — lut8Type encoding (Part 1 of 2)

Byte Offset	Content	Encoded as...
0..3	‘mft1’ (6D667431h) [multi-function table with 1-byte precision] type signature	
4..7	reserved, must be set to 0	
8	Number of Input Channels	uInt8Number
9	Number of Output Channels	uInt8Number
10	Number of CLUT grid points (identical for each side)	uInt8Number
11	Reserved for padding (fill with 00h)	
12..15	Encoded e00 parameter	s15Fixed16Number
16..19	Encoded e01 parameter	s15Fixed16Number

© 2003 ICC

Table 46 — lut8Type encoding (Part 2 of 2)

20..23	Encoded e02 parameter	s15Fixed16Number
24..27	Encoded e10 parameter	s15Fixed16Number
28..31	Encoded e11 parameter	s15Fixed16Number
32..35	Encoded e12 parameter	s15Fixed16Number
36..39	Encoded e20 parameter	s15Fixed16Number
40..43	Encoded e21 parameter	s15Fixed16Number
44..47	Encoded e22 parameter	s15Fixed16Number
48..m	Input tables	uInt8Number[...]
m+1..n	CLUT values	uInt8Number[...]
n+1..o	Output tables	uInt8Number[...]

The input, output, and grid tables contained in a lut8Type each embodies a one- or multi-dimensional function which maps an input value in the "domain" of the function to an output value in the "range" of the function.

The domain of each of these tables is defined to consist of all real numbers between 0,0 and 255,0, inclusive. The first entry is located at 0,0, the last entry at 255,0, and intermediate entries are uniformly spaced using an increment of 255,0/(M-1). For the input and output tables, M is 256. For the CLUT, M is the number of grid points along each dimension. Note that since the increment of 255,0/(M-1) is not necessarily an integer, the domain is specified to be over the real numbers rather than restricting it to the integers only. The range of a function used to generate the contents of a table is likewise defined to be all real numbers between 0,0 and 255,0, inclusive.

Because the contents of a table are encoded using 8 bits of precision, it is necessary to round each real value to the nearest 8-bit integer. This means that both the domain and range of the functions represented by the elements of the lut8Type as a whole are all real numbers between 0,0 and 255,0, inclusive. In many situations it is necessary to convert between these 8-bit values and some other bit precision.

See Annex A for additional guidance on this topic.

The color space used on the PCS side of a lut8Type tag (which may be either the input or output space, or both in the case of an abstract profile) is identified by the Profile Connection Space field in the profile header (see 6.1.6). This field does not distinguish between 8-bit and 16-bit PCS encodings. For the lut8Type tag, the 'Lab' signature is defined to specify the 8-bit CIELAB encoding. Note that this definition only applies to the encoding used as the Profile Connection Space side of the tag. It does NOT apply when these signatures are used on the "Color Space of Data" field in the profile header (see 6.1.5), except in the case of an abstract profile.

An 8-bit XYZ PCS has not been defined, so the interpretation of a lut8Type in a profile that uses the XYZ PCS is implementation specific. Because of the resulting ambiguity and because an 8-bit linear quantization of XYZ results in poor quality, it is recommended that the lut8Type tag not be used in profiles that employ the XYZ PCS.

The matrix is organized as a 3 by 3 array. The dimension corresponding to the matrix rows varies least rapidly and the dimension corresponding to the matrix columns varies most rapidly and is shown in matrix form below.

© 2003 ICC

$$\begin{bmatrix} e00 & e01 & e02 \\ e10 & e11 & e12 \\ e20 & e21 & e22 \end{bmatrix}$$

When using the matrix of an output profile, and the input data is XYZ, we have

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} e00 & e01 & e02 \\ e10 & e11 & e12 \\ e20 & e21 & e22 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (26)$$

Each input X, Y or Z is an unsigned 1.15 number and each matrix entry is a signed 15.16 number. Therefore, each multiplication in the matrix multiply is 1.15 * s15.16 = s16.31 and the final sum is also s16.31 (48 bits). From this sum we take bits 31..16 as the unsigned integer result for X', Y', or Z'. These are then scaled to the range 0-255 and used as the inputs to the input tables of the multidimensional LUT. This normalization is used since the number of fractional bits in the input data must be maintained by the matrix operation.

The matrix is mandated to be an identity matrix unless the input is in the XYZ color space.

The input tables are arrays of 8-bit unsigned values. Each input table consists of 256 one-byte integers. Each input table entry is appropriately normalized to the range 0-255. The inputTable is of size (InputChannels * 256) bytes. When stored in this tag, the one-dimensional lookup tables are assumed to be packed one after another in the order described below.

The CLUT is organized as an n-dimensional array with a given number of grid points in each dimension, where n is the number of input channels (input tables) in the transform. The dimension corresponding to the first input channel varies least rapidly and the dimension corresponding to the last input channel varies most rapidly. Each grid point value is an m-byte array, where m is the number of output channels. The first sequential byte of the entry contains the function value for the first output function, the second sequential byte of the entry contains the function value for the second output function, and so on until all the output functions have been supplied. Each byte in the CLUT is appropriately normalized to the range 0 - 255. The equation for computing the byte size of the CLUT is:

$$CLUTSize = (GridPoints^{InputChannels} * OutputChannels) \text{ bytes} \quad (27)$$

The output tables are arrays of 8-bit unsigned values. Each output table consists of 256 one-byte integers. Each output table entry is appropriately normalized to the range 0 - 255. The outputTable is of size (OutputChannels * 256) bytes. When stored in this tag, the one-dimensional lookup tables are assumed to be packed one after another in the order described in the following paragraph.

If the number of data points in a one-dimensional table, or in a particular dimension of the CLUT, is two, the data for those points shall be set so that the correct results are obtained when linear interpolation is used to generate intermediate values.

When using this type, it is necessary to assign each color space component to an input and output channel. The following table shows these assignments. The channels are numbered according to the order in

© 2003 ICC

which their table occurs. Note that additional color spaces can be added simply by defining the signature, channel assignments, and creating the tables.

Table 47 — lut8Type channel encodings

Color Space	Channel 1	Channel 2	Channel 3	Channel 4
‘XYZ ’	X	Y	Z	
‘Lab ’	L	a	b	
‘Luv ’	L	u	v	
‘Yxy ’	Y	x	y	
‘YCbr’	Y	Cb	Cr	
‘RGB ’	R	G	B	
‘GRAY’	K			
‘HSV ’	H	S	V	
‘HLS ’	H	L	S	
‘CMYK’	C	M	Y	K
‘CMY ’	C	M	Y	
‘2CLR’	Ch. 1	Ch. 2		
‘3CLR’	Ch. 1	Ch. 2	Ch. 3	
‘4CLR’	Ch. 1	Ch. 2	Ch. 3	Ch. 4

6.5.9 lutAtoBType

This structure converts an input color value to an output color value. The type contains up to five processing elements: a set of one dimensional curves, a 3 by 3 matrix with offset terms, a set of one dimensional curves, a multidimensional lookup table, and a set of one dimensional output curves. Data are processed using these elements via the following sequence:

("A" curves)-> (multidimensional lookup table)->("M" curves)-> (matrix) ->("B" curves).

NOTE The processing elements are not in this order in the tag to allow for simplified reading and writing of profiles.

It is possible to use any or all of these processing elements. At least one processing element must be included. Some processing elements are not allowed in certain circumstances.

The following combinations are allowed for the lutAtoBType:

B
M - Matrix - B
A - CLUT - B
A - CLUT - M - Matrix - B

Other combinations may be achieved by setting processing element values to identity transforms.

© 2003 ICC

When using this type, it is necessary to assign each color space component to an input and output channel. This assignment is specified in Table 43.

Table 48 — lutAtoBType encoding

Byte Offset	Content	Encoded as...
0..3	‘mAB ’ (6D414220h) [multi-function A-to-B table] type signature	
4..7	reserved, must be set to 0	
8	Number of Input Channels	uInt8Number
9	Number of Output Channels	uInt8Number
10..11	Reserved for padding, must be set to 0	
12..15	Offset to first "B" curve*	uInt32Number
16..19	Offset to matrix	uInt32Number
20..23	Offset to first "M" curve*	uInt32Number
24..27	Offset to CLUT	uInt32Number
28..31	Offset to first "A" curve*	uInt32Number
32..n	First data entry	

Each curve and processing element must start on a 4-byte boundary. To achieve this, each item may be followed by up to three 00h pad bytes as needed.

The offset entries (bytes 12-31) point to the various processing elements found in the tag. The offsets indicate the number of bytes from the beginning of the tag to the desired data. If any of the offsets are zero, that is an indication that processing element is not present and the operation is not performed.

Either the XYZ or Lab PCS may be used with this tag type.

6.5.9.1 "A" Curves

There are the same number of "A" curves as there are input channels. The "A" curves may only be used when the CLUT is used. The curves are stored sequentially, with 00h bytes used for padding between them if needed. Each "A" curve is stored as an embedded curveType or a parametricCurveType. The length is as indicated by the convention of the respective curve type. Note that the entire tag type, including the tag type signature and reserved bytes, is included for each curve.

6.5.9.2 CLUT

The CLUT appears as an n-dimensional array, with each dimension having a number of entries corresponding to the number of grid points.

The CLUT values are arrays of 8 bit or 16 unsigned values, normalized to the range of 0-255 or 65535.

The CLUT is organized as an n-dimensional array with a variable number of grid points in each dimension, where n is the number of input channels in the transform. The dimension corresponding to the first channel varies least rapidly and the dimension corresponding to the last input channel varies most rapidly. Each grid point value is an m-integer array, where m is the number of output channels. The first sequential

© 2003 ICC

integer of the entry contains the function value for the first output function, the second sequential integer of the entry contains the function value for the second output function and so on until all of the output functions have been supplied. The equation for computing the byte size of the CLUT is defined below.

nGrid1 * nGrid2 *... * nGridN * number of output channels * sizeof (channel component) (28)

Table 49 — lutAtoBType CLUT encoding

Byte Offset	Content	Encoded as...
0..15	Number of grid points in each dimension. Only the first n entries are used, where n is the number of input channels. Unused entries shall be set to 00h.	uInt8Number[16]
16	Precision of data elements in bytes. Must be either 01h or 02h.	uInt8Number
17..19	Reserved for padding, must be set to 0	
20..n	CLUT data points (arranged as described in the text).	uInt8Number[...] or uInt16Number[...]

If the number of input channels does not equal the number of output channels, the CLUT must be present.

If the number of grid points in a one-dimensional table, or in a particular dimension of the CLUT, is two, the data for those points shall be set so that the correct results are obtained when linear interpolation is used to generate intermediate values.

6.5.9.3 "M" Curves

There are the same number of "M" curves as there are output channels. The curves are stored sequentially, with 00h bytes used for padding between them if needed. Each "M' curve is stored as an embedded curveType or a parametricCurveType. The length is as indicated by the convention of the respective curve type. Note that the entire tag type, including the tag type signature and reserved bytes, is included for each curve. The "M" curves may only be used when the matrix is used.

6.5.9.4 Matrix

The matrix is organized as a 3x4 array. The elements appear in order from e1-e12. The matrix elements are each s15Fixed16Numbers.

array = $[e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9, e_{10}, e_{11}, e_{12}]$

The matrix is used to convert data to a different color space, according to the following equation:

$$\begin{bmatrix} Y1 \\ Y2 \\ Y3 \end{bmatrix} = \begin{bmatrix} e1 & e2 & e3 \\ e4 & e5 & e6 \\ e7 & e8 & e9 \end{bmatrix} \bullet \begin{bmatrix} X1 \\ X2 \\ X3 \end{bmatrix} + \begin{bmatrix} e10 \\ e11 \\ e12 \end{bmatrix} \quad (29)$$

Each input X1, X2, or X3 is an u16Fixed16Number and each matrix entry is a s15Fixed16Number. Therefore, each multiplication in the matrix multiply is 1.15*s15.16=s16.31 and the final sum is also s16.31(48

© 2003 ICC

bits). From this sum we take bits 31-16 as the unsigned integer result for Yl, Y2, or Y3. These are used as the inputs to the "B" curves.

The matrix is allowed only if the number of output channels, or "M" curves, is 3.

6.5.9.5 "B" Curves

There are the same number of "B" curves as there are output channels. The curves are stored sequentially, with 00h bytes used for padding between them if needed. Each "B" curve is stored as an embedded curveType or a parametricCurveType. The length is as indicated by the convention of the respective curve type. Note that the entire tag type, including the tag type signature and reserved bytes, are included for each curve.

6.5.10 lutBtoAType

This structure converts an input color value to an output color value. The type contains up to five processing elements: a set of one dimensional curves, a 3 by 3 matrix with offset terms, a set of one dimensional curves, a multidimensional lookup table, and a set of one dimensional output curves. Data are processed using these elements via the sequence defined below.

("B" curves)->(matrix)->("M" curves)->(multidimensional lookup table)->("A" curves).

It is possible to use any or all of these processing elements. At least one processing element must be included. Some processing elements are not allowed in certain circumstances.

The following combinations are allowed for the lutBtoAType:

B
B - Matrix - M
B - CLUT - A
B - Matrix - M - CLUT - A

Other combinations may be achieved by setting processing element values to identity transforms.

© 2003 ICC

When using this type, it is necessary to assign each color space component to an input and output channel. This assignment is specified in Table 43.

Table 50 — lutBtoAType encoding

Byte Offset	Content	Encoded as...
0..3	‘mBA ’ (6D424120h) [multi-function B-to-A table] type signature	
4..7	reserved, must be set to 0	
8	Number of Input Channels	uInt8Number
9	Number of Output Channels	uInt8Number
10-11	Reserved for padding, must be set to 0	
12..15	Offset to first "B" curve*	uInt32Number
16..19	Offset to matrix	uInt32Number
20..23	Offset to first "M" curve*	uInt32Number
24..27	Offset to CLUT	uInt32Number
28..31	Offset to first "A" curve*	uInt32Number
32..n	First data entry	

Each curve and processing element must start on a 4-byte boundary. To achieve this, each item may be followed by up to three 00h pad bytes as needed.

The offset entries (bytes 12-31) point to the various processing elements found in the tag. The offsets indicate the number of bytes from the beginning of the tag to the desired data. If any of the offsets are zero, that processing element is not present and the operation is treated as an identity transformation.

Either the XYZ or Lab PCS may be used with this tag type.

6.5.10.1 "B" Curves

There are the same number of "B" curves as there are input channels. The curves are stored sequentially, with 00h bytes used for padding between them if needed. Each "B" curve is stored as an embedded curveType tag or a parametricCurveType. The length is as indicated by the convention of the proper curve type. Note that the entire tag type, including the tag type signature and reserved bytes, is included for each curve.

6.5.10.2 Matrix

The matrix is organized as a 3x4 array. The elements of the matrix appear in the type in order from e1-e12. The matrix elements are each s15Fixed16Numbers.

$$\text{array} = [e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8, e_9, e_{10}, e_{11}, e_{12}]$$

The matrix is used to convert data to a different color space, according to the following equation:

© 2003 ICC

$$\begin{bmatrix} Y1 \\ Y2 \\ Y3 \end{bmatrix} = \begin{bmatrix} e1 & e2 & e3 \\ e4 & e5 & e6 \\ e7 & e8 & e9 \end{bmatrix} \bullet \begin{bmatrix} X1 \\ X2 \\ X3 \end{bmatrix} + \begin{bmatrix} e10 \\ e11 \\ e12 \end{bmatrix} \quad (30)$$

Each input X1, X2, or X3 is an u16Fixed16Number and each matrix entry is a s15Fixed16Number. Therefore, each multiplication in the matrix multiply is 1.15*s15.16=s16.31 and the final sum is also s16.31(48 bits). From this sum we take bits 31-16 as the unsigned integer result for YI, Y2, or Y3. These are used as the inputs to the "M" curves.

The matrix is allowed only if the number of input channels, or "B" curves, is 3.

6.5.10.3 "M" Curves

There are the same number of "M" curves as there are input channels. The curves are stored sequentially, with 00h bytes used for padding between them if needed. Each 'M' curve is stored as an embedded curveType or a parametricCurveType. The length is as indicated by the convention of the proper curve type. Note that the entire tag type, including the tag type signature and reserved bytes, are included for each curve. The "M" curves may only be used when the matrix is used.

6.5.10.4 CLUT

The CLUT appears as an n-dimensional array, with each dimension having a number of entries corresponding to the number of grid points.

The CLUT values are arrays of 8 bit or 16 unsigned values, normalized to the range of 0-255 or 65535.

The CLUT is organized as an n-dimensional array with a variable number of grid points in each dimension, where n is the number of input channels in the transform. The dimension corresponding to the first channel varies least rapidly and the dimension corresponding to the last input channel varies most rapidly. Each grid point value is an m-integer array, where m is the number of output channels. The first sequential integer of the entry contains the function value for the first output function, the second sequential integer of the entry contains the function value for the second output function and so on until all of the output functions have been supplied. The equation for computing the byte size of the CLUT is:

nGrid1 * nGrid2 *...* nGridN * number of output channels * sizeof (channel component) (31)

Table 51 — lutBtoAType CLUT encoding

Byte Offset	Content	Encoded as...
0..15	Number of grid points in each dimension. Only the first n entries are used, where n is the number of input channels. Unused entries shall be set to 00h.	uInt8Number[16]
16	Precision of data elements in bytes. Must be either 01h or 02h.	uInt8Number
17..19	Reserved for padding.	
20..n	CLUT data points (arranged as described in the text).	uInt8Number[...] or uInt16Number[...]

© 2003 ICC

If the number of grid points in a one-dimensional table, or in a particular dimension of the CLUT, is two, the data for those points shall be set so that the correct results are obtained when linear interpolation is used to generate intermediate values.

If the number of input channels does not equal the number of output channels, the CLUT must be present.

6.5.10.5 "A" Curves

There are the same number of "A" curves as there are output channels. The "A" curves may only be used when the CLUT is used. The curves are stored sequentially, with 00h bytes used for padding between them if needed. Each "A" curve is stored as an embedded curveType or a parametricCurveType. The length is as indicated by the convention of the proper curve type. Note that the entire tag type, including the tag type signature and reserved bytes, is included for each curve.

6.5.11 measurementType

The measurementType information refers only to the internal profile data and is meant to provide profile makers an alternative to the default measurement specifications.

Table 52 — measurementType structure

Byte Offset	Content	Encoded as...
0..3	'meas' (6D656173h) type signature	
4..7	reserved, must be set to 0	
8..11	encoded value for standard observer	see below
12..23	XYZ tristimulus values for measurement backing	XYZNumber
24..27	encoded value for measurement geometry	see below
28..31	encoded value for measurement flare	see below
32..35	encoded value for standard illuminant	see below

The encoding for the standard observer field is such that:

Table 53 — Standard observer encodings

Standard Observer	Encoded Value
unknown	00000000h
CIE 1931 standard colorimetric observer	00000001h
CIE 1964 standard colorimetric observer	00000002h

© 2003 ICC

The encoding for the measurement geometry field is such that:

Table 54 — Measurement geometry encodings

Geometry	Encoded Value
unknown	00000000h
0/45 or 45/0	00000001h
0/d or d/0	00000002h

The encoding for the measurement flare value is shown below and is equivalent to the basic numeric type u16Fixed16Number in 5.3.4.

Table 55 — Measurement flare encodings

Flare	Encoded Value
0 (0%)	00000000h
1,0 (or 100%)	00010000h

The encoding for the standard illuminant field is such that:

Table 56 — Standard illuminant encodings

Standard Illuminant	Encoded Value
unknown	00000000h
D50	00000001h
D65	00000002h
D93	00000003h
F2	00000004h
D55	00000005h
A	00000006h
Equi-Power (E)	00000007h
F8	00000008h

6.5.12 multiLocalizedUnicodeType

This tag structure contains a set of multilingual Unicode strings associated with a profile. Each string in the set is stored in a separate record with the information about what language and region the string is for.

© 2003 ICC

Table 57 — multiLocalizedUnicodeType

Byte Offset	Content	Encoded as...
0..3	'mluc' (0x6D6C7563) type signature	
4..7	reserved, must be set to 0	
8..11	number of names: the number of name records that follow.	uInt32Number
12..15	name record size: the length in bytes of each name record that follows. Each name record currently consists of the fields first name language code to first name offset.	uInt32Number
16..17	first name language code: language code from ISO-639	uInt16Number
18..19	first name country code: region code from ISO-3166	uInt16Number
20-23	first name length: the length in bytes of the string	uInt32Number
24..27	first name offset: the offset from the start of the tag in bytes	uInt32Number
28..28+12n-1	if more than one name record, store them here	
28+12n...end	Storage area of Unicode characters	

Note that the third field of this tag, the name record size should, for the time being, contain the value 12, which corresponds to the size in bytes of each name record. Any code that needs to access the nth name record should determine the record's offset by multiplying n by the contents of this size field and adding 16. This minor extra effort allows for future expansion of name records, should the need arise, without having to define yet another new tag type.

For the specification of Unicode, see The Unicode Standard published by The Unicode Consortium or visit their website at http://www.unicode.org. For the definition of language code and region codes, see ISO-639 and ISO-3166. The Unicode strings in storage are encoded as 16-bit big-endian, UTF-16BE, and should not be NULL terminated.

If the specific string for the desired region is not stored in the tag, the string with the same language code should be used. If the specific string for the desired language is not stored in the tag, the first string in the tag is used if no other user preference is available.

6.5.13 namedColor2Type

The namedColor2Type is a count value and array of structures that provide color coordinates for 7-bit ASCII color names. For each named color, a PCS and optional device representation of the color are given. Both representations are 16-bit values. The device representation corresponds to the header's "color space of data" field. This representation should be consistent with the "number of device components" field in the namedColor2Type. If this field is 0, device coordinates are not provided. The PCS representation corresponds to the header's PCS field. The PCS representation is always provided. Color names are fixed-length, 32-byte fields including null termination. In order to maintain maximum portability, it is strongly recommended that special characters of the 7-bit ASCII set not be used.

© 2003 ICC

Table 58 — namedColor2Type encoding

Byte Offset	Content	Encoded as...
0..3	'ncl2' (6E636C32h) type signature	
4..7	reserved, must be set to 0	
8..11	vendor specific flag (least-significant 16 bits reserved for ICC use)	
12..15	count of named colors	uInt32Number
16..19	number of device coordinates for each named color	uInt32Number
20..51	prefix for each color name (32-byte field including null termination)	7-bit ASCII
52..83	suffix for each color name (32-byte field including null termination)	7-bit ASCII
84..115	first color root name (32-byte field including null termination)	7-bit ASCII
116..121	first named color's PCS coordinates. The encoding is the same as the encodings for the PCS color spaces as described in Annex A. Only 16-bit L*a*b*, encoded using legacy 16-bit PCS Lab encoding, and XYZ are allowed.	uInt16Number[3]
122..y	first named color's device coordinates. For each coordinate, 0000h represents the minimum value for the device coordinate and FFFFh represents the maximum value for the device coordinate. The number of coordinates is given by the "number of device coordinates" field. If the "number of device coordinates" field is 0, this field is not given.	uInt16Number[...]
y+1..z	if *count* > 1 the remaining *count*-1 colors are described in a manner consistent with the first named color, see byte offsets 84..y.	

For color values that are in the Lab color space on the PCS side of the tag, this tag uses a legacy 16-bit Lab encoding, not the 16-bit CIELAB PCS encoding that is defined in Annex A.2. This encoding is retained for backwards compatibility with profile version 2.

To convert color values from this tag's legacy 16-bit Lab encoding to the 16-bit CIELAB PCS encoding defined in Annex A.2, multiply all values with 65535/65280 (that is, FFFFh/FF00h). Any color values that are in the value range of legacy 16-bit PCS Lab, but not in the 16-bit CIELAB PCS encoding defined in Annex A.2, shall be clipped on a per-component basis when transforming from legacy 16-bit PCS Lab to the 16-bit CIELAB PCS encoding defined in Annex A.2. To convert color values from the 16-bit CIELAB PCS encoding defined in Annex A.2 to this tag's legacy 16-bit Lab encoding, divide all values with 65535/65280.

© 2003 ICC

The L* values have a different encoding than the a* and b* values. The L* encoding is:

Table 59 — L* encoding

Value (L*)	16 bit
0	0000h
100,0	FF00h
100 + (25500/65280)	FFFFh

Although the 16-bit encoding can represent values slightly greater than 100,0, these are not valid PCS L* values and they should not be used.

The a* and b* encoding is:

Table 60 — a* or b* encoding

Value (a* or b*)	16-bit
-128,0	0000h
0	8000h
127,0	FF00h
127 + (255/256)	FFFFh

Note that the 16-bit encoding can represent values slightly greater than 127,0. Since a* and b* have no defined limits, these are valid PCS values.

6.5.14 parametricCurveType

The parametricCurveType describes a one-dimensional curve by specifying one of a predefined set of functions using the parameters. The byte stream is as follows:

Table 61 — parametricCurveType encoding

Byte Offset	Content	Encoded as...
0..3	'para' (70617261h) type signature	
4..7	reserved, must be set to 0	
8..9	function type	uInt16Number (see below)
10..11	reserved, must be set to 0	
12..end	one or more parameters (see below)	s15Fixed16Number [...]

© 2003 ICC

The encoding for the function type field and the parameters are such that:

Table 62 — parametricCurveType function type encoding

Function type	Encoded value	Parameters	Note
$Y = X^{\gamma}$	0000h	γ	
$Y = (aX + b)^{\gamma} \quad (X \geq -b/a)$ $Y = 0 \quad (X < -b/a)$	0001h	$\gamma\ a\ b$	CIE 122-1966
$Y = (aX + b)^{\gamma} + c \ (X \geq -b/a)$ $Y = c \quad (X < -b/a)$	0002h	$\gamma\ a\ b\ c$	IEC 61966-3
$Y = (aX + b)^{\gamma} \quad (X \geq d)$ $Y = cX \quad (X < d)$	0003h	$\gamma\ a\ b\ c\ d$	IEC 61966-2.1 (sRGB)
$Y = (aX + b)^{\gamma} + e \ (X \geq d)$ $Y = (cX + f) \quad (X < d)$	0004h	$\gamma\ a\ b\ c\ d\ e\ f$	

NOTE More functions can be added as necessary.

The order of the parameters in the tag data, Table 61, follows the left-to-right order of the parameters in Table 62.

The domain and range of each function shall be [0,0 1,0]. Any function value outside the range shall be clipped to the range of the function. When unsigned integer data is supplied as input, it shall be converted to the domain be dividing it by a factor of $(2^N) -1$, where N is the number of bits used to represent the input data. When unsigned integer data is required as output, it shall be converted from the range by multiplying it by a factor of $(2^M) -1$, where M is the number of bits used to represent the output data.

6.5.15 profileSequenceDescType

This type is an array of structures, each of which contains information from the header fields and tags from the original profiles which were combined to create the final profile. The order of the structures is the order

© 2003 ICC

in which the profiles were combined and includes a structure for the final profile. This provides a description of the profile sequence from source to destination, typically used with the DeviceLink profile.

Table 63 — profileSequenceDescType structure

Byte Offset	Content
0..3	'pseq' (70736571h) type signature
4..7	reserved, must be set to 0
8..11	count value specifying number of description structures in the array
12..end	*count* profile description structures

Each profile description structure has the format:

Table 64 — Profile Description structure

Byte Offset	Content
0..3	Device manufacturer signature (from corresponding profile's header)
4..7	Device model signature (from corresponding profile's header)
8..15	Device attributes (from corresponding profile's header)
16..19	Device technology information such as CRT, Dye Sublimation, etc. (corresponding profile's technology signature)
20..m	displayable description of device manufacturer (corresponding profile's deviceMfgDescTag)
m+1..n	displayable description of device model (corresponding profile's deviceModelDescTag)

If the deviceMfgDescTag and/or deviceModelDescTag is not present in a component profile, then a "placeholder" tag should be inserted. This tag should have a 0 in the number of names field in the multiLocalizedUnicodeType structure with no name record or strings.

Also note that the entire tag, including the tag type, should be stored.

If the technologyTag is not present, bytes 16..19 should be 00000000h.

6.5.16 responseCurveSet16Type

ICC profiles for display and output devices will produce the desired color only while the device has a particular relationship between normalized device codes and physical colorant amount (the reference response). If the response of the device changes (the current response), the profile will no longer produce the correct result. In many cases it is impractical to produce a new profile for the current response, but the change can be compensated for by modifying the single channel device codes.

The purpose of this tag type is to provide a mechanism to relate physical colorant amounts with the normalized device codes produced by lut8Type or lut16Type tags so that corrections can be made for variation in the device without having to produce a new profile. The mechanism can be used by applications to allow users with relatively inexpensive and readily available instrumentation to apply corrections to individual output color channels in order to achieve consistent results.

© 2003 ICC

Two pieces of information are necessary for this compensation: the reference response and the current response. This tag type provides a mechanism that allows applications that create profiles to specify the reference response. The way in which applications determine and make use of the current response is not specified at this time.

The measurements are of the standard variety used in the photographic, graphic arts, and television industries for process control. The measurements are intended to represent colorant amounts and so different measurement techniques are appropriate for different device types.

It is the job of the profile creator to provide reference response data in as many measurement units as practical and appropriate so that applications may select the same units that are measured by the user's instrument. Since it is not possible in general to translate between measurement units, and since most instruments only measure in one unit, providing a wide range of measurement units is vital. The profile originator must decide which measurement units are appropriate for the device.

Here are some examples of suitable measurement units: For process colors, density should be reported. Red-filter density should be reported for the cyan channel, green-filter for the magenta channel, blue-filter for the yellow channel, and visual for the black channel. For other colorants, such as Spot colors or Hi-Fi colors, it is the responsibility of the profile creator to select the appropriate units of measure for the system being profiled. Several different density standards are used around the world, so it is important that profile creators report in as many different density units as possible. Examples of suitable density measurements are: Status T, Status E, Status I and DIN.

This structure relates normalized device codes that would result from a lut16Type tag with density measurements of the resulting colorant amount. normalized device codes resulting from a lut8Type tag should first be multiplied by 257 (101h).

For those fields that have been structured in arrays of channel data, the channels are ordered as specified for the appropriate color space in Table 43.

Table 65 — responseCurveSet16Type structure

Byte Offset	Content	Encoded as...
0..3	'rcs2' (72637332h) [response curve set with 2-byte precision] type signature	
4..7	reserved, must be set to 0	
8..9	number of channels	uInt16Number
10..11	count of measurement types	uInt16Number
12..m	*count* relative offsets from byte 0 of this structure. Each will point to the response data for the measurement unit.	uInt32Number[...]
m+1..n	*count* response curve structures	see below

© 2003 ICC

Each response curve structure has the format:

Table 66 — Curve structure

Byte Offset	Content	Encoded as...
0..3	measurement unit signature	see below
4..m	number of measurements for each channel: 4 = count of chan. 1 measurements, 8 = count of chan. 2 measurements, m-3 = count of final channel measurements	uInt32Number[...]
m+1..n	*number-of-channels* measurements of patch with the maximum colorant value	XYZNumber[...]
n+1..p	*number-of-channels* response arrays. Each array contains *number-of-measurements* response16Numbers appropriate to the channel	response16Number[...]

NOTE The XYZ values are CIE XYZ tristimulus values as described in 5.3.10. The response arrays must be ordered with normalized device code elements increasing.

The measurement unit is encoded as follows:

Table 67 — Curve measurement encodings

Measurement Unit	Signature	Hex Encoding
Status A: ANSI PH2.18 densitometer response. This is the accepted standard for reflection densitometers for measuring photographic color prints.	‘StaA’	53746141h
Status E: A densitometer response which is the accepted standard in Europe for color reflection densitometers.	‘StaE’	53746145h
Status I: A densitometer response commonly referred to as narrow band or interference-type response.	‘StaI’	53746149h
Status T: Wide band color reflection densitometer response which is the accepted standard in the United States for color reflection densitometers.	‘StaT’	53746154h
Status M: Standard densitometer response for measuring negatives.	‘StaM’	5374614Dh
DIN: Measurement according to DIN standard with no polarizing filter.	‘DN ’	444E2020h
DIN with Polarizing Filter	‘DN P’	444E2050h
Narrow band DIN	‘DNN ’	444E4E20h
Narrow band DIN with Polarizing filter	‘DNNP’	444E4E50h

© 2003 ICC

6.5.17 s15Fixed16ArrayType

This type represents an array of generic 4-byte/32-bit fixed point quantity. The number of values is determined from the size of the tag.

Table 68 — s16Fixed16ArrayType encoding

Byte Offset	Content
0..3	‘sf32’ (73663332h) type signature
4..7	reserved, must be set to 0
8..end	an array of s15Fixed16Number values

6.5.18 signatureType

The signatureType contains a four-byte sequence used for signatures. Typically this type is used for tags that need to be registered and can be displayed on many development systems as a sequence of four characters. Sequences of less than four characters are padded at the end with spaces, 20h.

Table 69 — signatureType encoding

Byte Offset	Content
0..3	‘sig ’ (73696720h) type signature
4..7	reserved, must be set to 0
8..11	four-byte signature

6.5.19 textType

The textType is a simple text structure that contains a 7-bit ASCII text string. The length of the string is obtained by subtracting 8 from the element size portion of the tag itself. This string must be terminated with a 00h byte.

Table 70 — textType encoding

Byte Offset	Content
0..3	‘text’ (74657874h) type signature
4..7	reserved, must be set to 0
8..end	a string of (element size - 8) 7-bit ASCII characters

© 2003 ICC

6.5.20 u16Fixed16ArrayType

This type represents an array of generic 4-byte/32-bit quantity. The number of values is determined from the size of the tag.

Table 71 — u16Fixed16ArrayType encoding

Byte Offset	Content
0..3	‘uf32’ (75663332h) type signature
4..7	reserved, must be set to 0
8..end	an array of u16Fixed16Number values

6.5.21 uInt16ArrayType

This type represents an array of generic 2-byte/16-bit quantity. The number of values is determined from the size of the tag.

Table 72 — uInt16ArrayType encoding

Byte Offset	Content
0..3	‘ui16’ (75693136h) type signature
4..7	reserved, must be set to 0
8..end	an array of unsigned 16-bit integers

6.5.22 uInt32ArrayType

This type represents an array of generic 4-byte/32-bit quantity. The number of values is determined from the size of the tag.

Table 73 — uInt32ArrayType encoding

Byte Offset	Content
0..3	‘ui32’ (75693332h) type signature
4..7	reserved, must be set to 0
8..end	an array of unsigned 32-bit integers

© 2003 ICC

6.5.23 uInt64ArrayType

This type represents an array of generic 8-byte/64-bit quantity. The number of values is determined from the size of the tag.

Table 74 — uInt64ArrayType encoding

Byte Offset	Content
0..3	‘ui64’ (75693634h) type signature
4..7	reserved, must be set to 0
8..end	an array of unsigned 64-bit integers

6.5.24 uInt8ArrayType

This type represents an array of generic 1-byte/8-bit quantity. The number of values is determined from the size of the tag.

Table 75 — uInt8ArrayType encoding

Byte Offset	Content
0..3	‘ui08’ (75693038h) type signature
4..7	reserved, must be set to 0
8..end	an array of unsigned 8-bit integers

6.5.25 viewingConditionsType

This type represents a set of viewing condition parameters including: CIE ’absolute’ illuminant white point tristimulus values and CIE ’absolute’ surround tristimulus values.

Table 76 — viewingConditionsType encoding

Byte Offset	Content	Encoded as...
0..3	‘view’ (76696577h) type signature	
4..7	reserved, must be set to 0	
8..19	CIE ’absolute’ XYZ values for illuminant (in which Y is in cd/m^2)	XYZNumber
20..31	CIE ’absolute’ XYZ values for surround (in which Y is in cd/m^2)	XYZNumber
32..35	illuminant type	as described in measurementType

The viewing condition described in this tag is the actual viewing condition assumed for the media for which the profile is defined, specified in CIE absolute units. Note that the luminanceTag must be the same as the Y value given in this tag.

© 2003 ICC

6.5.26 XYZType

The XYZType contains an array of three encoded values for the XYZ tristimulus values. The number of sets of values is determined from the size of the tag. The byte stream is given below. Tristimulus values must be non-negative. The signed encoding allows for implementation optimizations by minimizing the number of fixed formats.

.

Table 77 — XYZType encoding

Byte Offset	Content	Encoded as...
0..3	'XYZ ' (58595A20h) type signature	
4..7	reserved, must be set to 0	
8..end	an array of XYZ numbers	XYZNumber

© 2003 ICC

Annex A
Color spaces

The International Color Profile Format supports a variety of both device-dependent and device-independent color spaces divided into three basic families: 1) CIEXYZ based, 2) RGB based, and 3) CMY based.

The CIE color spaces are defined in CIE publication 15.2 on Colorimetry. A subset of the CIEXYZ based spaces are also defined as connection spaces. The device dependent spaces below are only representative and other device dependent color spaces may be used without needing to update the profile format specification or the software that uses it.

Table 78 — CIE color spaces

Base Space	Description	Derivative Space
CIEXYZ	base CIE device-independent color space	CIELAB
GRAY	monochrome device-dependent color space	
RGB	base additive device-dependent color space	HLS, HSV
CMY	base subtractive device-dependent color space	CMYK

A.1 Profile Connection Spaces

A key component of these profiles is a well-defined profile connection space. This space is the interface which provides an unambiguous connection between the input and output profiles as illustrated in the diagram below. The profile connection space is based on the CIE 1931 standard colorimetric observer. This experimentally derived standard observer provides a very good representation of the human visual system color matching capabilities. Unlike device dependent color spaces, if two colors have the same CIE colorimetry they will match if viewed under the same conditions. Because the imagery is typically produced for a wide variety of viewing environments, it is necessary to go beyond simple application of the CIE system.

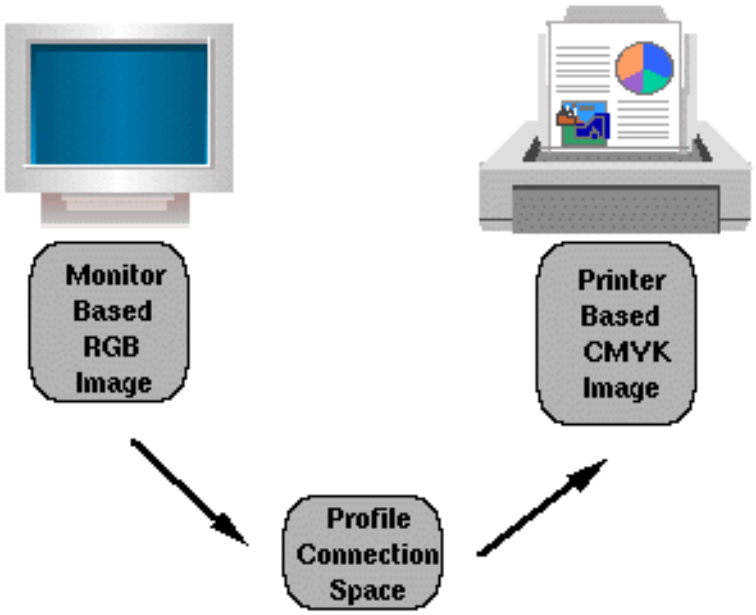

Figure 3 — Profile connection space illustration

© 2003 ICC

The profile connection space is defined as the CIE colorimetry which, in the case of the perceptual rendering intent, will produce the desired color appearance if rendered on a reference imaging media and viewed in a reference viewing environment. This reference corresponds to an ideal reflection print viewed in an ISO P2 standard viewing booth.

The default measurement parameters for the profile connection space and all other color spaces defined in this specification are based on the ISO 13655 standard, “Graphic technology - Spectral measurement and colorimetric computation for graphic arts images.” Essentially this defines a standard illuminant of D50, the 1931 CIE standard colorimetric observer, and $0^o/45^o$ or $45^o/0^o$ reflectance measurement geometry. The reference viewing condition shall be ISO 3664 viewing condition P2 using the recommended 20% surround reflectance. This is a graphics arts and photography print comparison environment using a D50 illuminant at an illumination level of 500 lux.

One of the first steps in profile building involves determining the colorimetry of a set of colors from some imaging media or display. If the imaging media or viewing environment differ from the reference, it will be necessary to adapt the measured colorimetry to that appropriate for the profile connection space. These adaptations account for such differences as white point chromaticity and luminance relative to an ideal reflector, maximum density, viewing surround, viewing illuminant, and flare. Currently, it is the responsibility of the profile builder to do this adaptation.

However, the possibility of allowing a variable illuminant in the PCS is under active consideration by the International Color Consortium. For this reason, a PCS illuminant field is in the profile header, but must be set to the CIE Illuminant D50 [X=0,9642 Y=1,0000 Z=0,8249]. Note that the PCS illuminant field should not be confused with the viewing conditions tag defined in clause 6.4.36:“viewingCondDescTag” and clause 6.4.37:“viewingConditionsTag”.

The PCS is based on media-relative colorimetry. This is in comparison to ICC-absolute colorimetry. In ICC-absolute colorimetry colors are represented with respect to the illuminant, for example D50, and a perfect diffuser for reflecting and transmitting media. In media-relative colorimetry, colors are represented with respect to a combination of the illuminant and the media’s white, e.g. unprinted paper. The translation from media-relative colorimetry XYZ data, XYZ_r to ICC-absolute colorimetric data, XYZ_a, is given by

$$X_a = \left(\frac{X_{mw}}{X_i}\right) \cdot X_r \qquad \text{A.1.0.0.0-0}$$

$$Y_a = \left(\frac{Y_{mw}}{Y_i}\right) \cdot Y_r \qquad \text{A.1.0.0.0-1}$$

$$Z_a = \left(\frac{Z_{mw}}{Z_i}\right) \cdot Z_r \qquad \text{A.1.0.0.0-2}$$

where XYZ_{mw} represents the chromatically adapted media white point as found in the mediaWhitePointTag and XYZ_i represents the PCS illuminant white (which is D50).

The actual media and actual viewing conditions will typically differ from the reference conditions. The profile specification defines tags which provide information about the actual white point and black point of a given media or display. These tags may be used by a CMM to provide functionality beyond that of the default. For example, an advanced CMM could use the tags to adjust colorimetry based on the Dmin of a specific media. A tag is also provided to describe the viewing environment. This information is useful in choosing a profile appropriate for the intended viewing method.

© 2003 ICC

A.2 PCS Encodings

There are many ways of encoding CIE colorimetry. This specification provides two methods in order to satisfy conflicting requirements for accuracy and storage space. These encodings, a CIELAB encoding and a 16-bit/component CIEXYZ encoding, are described below. The CIEXYZ space represents a linear transformation of the average color matching data, obtained by mixing red, green and blue lights to match all spectral colors, derived experimentally in the 1920s. The CIELAB space represents a transformation of the CIEXYZ space into one that is nearly perceptually uniform. This uniformity allows color errors to be equally weighted throughout its domain. While supporting multiple CIE encodings increases the complexity of color management, it provides immense flexibility in addressing different user requirements such as color accuracy and memory footprint.

The relationship between PCS CIEXYZ and PCS CIELAB is given by the usual set of equations as defined in ISO 13655:1996 but where the media white point (rather than the illuminant) is used as the relevant white point. Thus:

X/Xn is replaced by Xr/Xi (or Xa/Xmw) A.2.0.0.0-1

Y/Yn is replaced by Yr/Yi (or Ya/Ymw) A.2.0.0.0-2

Z/Zn is replaced by Zr/Zi (or Za/Zmw) A.2.0.0.0-3

where X/Xn, Y/Yn, and Z/Zn are defined in ISO 13655:1996, Annex B, section B.1.

NOTE The equations are as follows:

$L^* = 116[f(Y/Y_n)] - 16$ A.2.0.0.0-4

$a^* = 500[f(X/X_n) - f(Y/Y_n)]$ A.2.0.0.0-5

$b^* = 200[f(Y/Y_n) - f(Z/Z_n)]$ A.2.0.0.0-6

for: $X/X_n > 0{,}008856$, $f(X/X_n) = (X/Xn)^{1/3}$ A.2.0.0.0-7

$Y/Y_n > 0{,}008856$, $f(Y/Y_n) = (Y/Yn)^{1/3}$ A.2.0.0.0-8

$Z/Z_n > 0{,}008856$, $f(Z/Z_n) = (Z/Zn)^{1/3}$ A.2.0.0.0-9

for: $X/X_n <= 0{,}008856$, $f(X/X_n) = 7{,}7870(X/X_n)+16/116$ A.2.0.0.0-10

$Y/Y_n <= 0{,}008856$, $f(Y/Y_n) = 7{,}7870(Y/Y_n)+16/116$ A.2.0.0.0-11

$Z/Z_n <= 0{,}008856$, $f(Z/Z_n) = 7{,}7870(Z/Z_n)+16/116$ A.2.0.0.0-12

where $7{,}7870 = [29 / (6 * \mathrm{sqrt}(3))]^2$

It is important to understand that the PCS encodings do not represent a quantization of the connection space. The purpose of the encodings is to allow points within the space to be specified. Since the processing models benefit from interpolation between table entries, the interpolated AToB results should be used as the inputs to the BToA transforms. The AToB results should not be rounded to the nearest encoding value.

© 2003 ICC

For the CIEXYZ encoding, each component (X, Y, and Z) is encoded as a fixed unsigned 16-bit quantity which has 15 fractional bits (u1.15).

An example of this encoding is:

Table 79 — CIEXYZ encoding

0	0000h
1,0	8000h
1 + (32767/32768)	FFFFh

The largest valid XYZ values are those of the PCS illuminant (specified in the profile header).[1)] This encoding was chosen to allow for PCS illuminants that have an X or Z greater than 1,0.

NOTE: CIE specifies that for reflecting and transmitting media Y should be normalized such that it has the value 100 for the perfect reflector or transmitter. In this specification, for reasons of coding efficiency, Y is specified such that it has the value 1 for the perfect reflector or transmitter.

For the CIELAB PCS encodings, the L* values have a different encoding than the a* and b* values. The L* encoding is:

Table 80 — CIELAB L* encoding

Value (L*)	**8-bit**	**16-bit**
0	00h	0000h
100,0	FFh	FFFFh

The a* and b* encoding is:

Table 81 — CIELAB a* or b* encoding

Value (a* or b*)	**8-bit**	**16-bit**
-128,0	00h	0000h
0	80h	8080h
127,0	FFh	FFFFh

Note that this is not "two's complement" encoding, but a linear scaling after an offset of 128. This encoding was chosen to prevent discontinuities in CLUTs when going from negative to positive values.

You can convert between the 8-bit and 16-bit encodings by multiplying or dividing by 257.

Because of the way that PCS encodings map to input tables, the BToAn tags must be able to handle invalid PCS values. However, the results of sending invalid values to these tags is up to the creator of the profile.

[1)] For a D50 illuminant, the largest valid XYZ values are [0,9642 1,0 0,8249], or [7B6Bh, 8000h, 6996h] in encoded form. Note that the PCS illuminant values are stored in s15.16 format, so you must translate them to u1.15 format to find the encoded PCS limits.

© 2003 ICC

An important point to be made is that the PCS is not necessarily intended for the storage of images. A separate series of “interchange color spaces” may be defined in a future version of this specification for this purpose. The design choices made for these spaces (colorimetric encoding, reference media, viewing conditions, etc.) might be different than that of the PCS.

A.3 External and internal conversions

CMMs or other applications that use ICC tags to perform color transformations typically need to perform two types of data processing in addition to table interpolation. First, because the color values being processed (such as image pixels) may not match the native precision of an ICC tag (such as a lut16Type or lut8Type), it may be necessary to alter the precision of the input to (or results from) these transforms. Second, because there is more than one PCS encoding, it may be necessary to convert the output from a first transform before applying it to the input of a second transform. These two types of additional processing may be thought of as primarily affecting the **external** and **internal** interfaces of ICC processing, respectively.

In the first (external) case, the appropriate conversion method is to multiply each color value by $(2^M-1)/(2^N-1)$, where N is the starting number of bits and M is the required number of bits. This converts a number with values from 0 to (2^N-1) to a number with values from 0 to (2^M-1). For example, to prepare an 8-bit image value for input to a lut16Type tag the scale factor is $(2^{16}-1)/(2^8-1)$ = 65535,0/255,0 = 257,0. Note that the colors represented by the scaled numbers (be they device coordinates or some other color space) are not altered by the change in precision. For example, if a particular image value represents an L* of 31,0, then the scaled value also represents an L* of 31,0. Additionally, if an integer value is required from the scaling operation, it should be obtained via rounding rather than truncation.

In the second (internal) case, the appropriate conversion uses the CIE equations to convert between CIEXYZ and CIELAB.

Any colors in PCS XYZ encoding range that are outside of the PCS LAB encoding range shall be clipped on a per-component basis to the outside limits of the range of PCS LAB when transforming from XYZ into LAB. Conversely, any colors that occur in PCS LAB encoding range that are outside of the encoding range of PCS XYZ shall be clipped on a per-component basis to the PCS XYZ range when transforming from LAB into XYZ

A.4 Rendering Intents

In general, actual device gamuts will not be large enough to reproduce the desired color appearances communicated by the PCS values. Four rendering intents (gamut mapping styles) are provided to address this problem. Each one represents a different compromise. The colorimetric rendering intents operate on measurement-based colorimetric values as chromatically adapted to the PCS illuminant D50. This adaptation will usually be indicated in the chromaticAdaptationTag. The other rendering intents operate on colorimetric values which are corrected in an as-needed fashion to account for any differences between devices, media, and viewing conditions.

A.4.1 Colorimetric Intents

The colorimetric intents preserve the relationships between in-gamut colors at the expense of out-of-gamut colors. Mapping of out-of-gamut colors is not specified but should be consistent with the intended use of the transform.

© 2003 ICC

A.4.1.1 MediaWhitePoint Tag

The mediaWhitePointTag contains CIE 1931 XYZ colorimetry of the white point of the actual medium. If the viewer completely adapts to the white point of the medium (as is often the case with monitors) this tag should be set to Xi, Yi, Zi (the PCS white point). If chromatic adaptation is being applied to the PCS values, the adaptation should be applied to the mediaWhitePointTag values as well.

A.4.1.2 Media-Relative Colorimetric Intent

This intent re-scales the in-gamut, chromatically adapted tristimulus values such that the white point of the actual medium is mapped to the PCS white point (for either input or output). It is useful for colors that have already been mapped to a medium with a smaller gamut than the reference medium (and therefore need no further compression).

A.4.1.3 ICC-Absolute Colorimetric Intent

For this intent, the chromatically adapted tristimulus values of the in-gamut colors are unchanged. It is useful for spot colors and when simulating one medium on another (proofing). Note that this definition of ICC-absolute colorimetry is actually called “relative colorimetry” in CIE terminology, since the data has been normalized relative to the perfect diffuser viewed under the same illumination source as the sample.

A.4.1.4 Applying the ICC-Absolute Colorimetric Intent

Profiles do not contain a separate transform for the ICC-absolute colorimetric intent. When this intent is needed, it is generated using the media-relative colorimetric intent and scaling the PCS values by the ratio of the destination profile mediaWhitePointTag to the source profile mediaWhitePointTag (as described in this annex). This applies to profiles using the TRC/matrix model, which is defined to use the media-relative colorimetric intent.

A.4.2 Perceptual Intent

The exact gamut mapping of the perceptual intent is vendor specific and involves compromises such as trading off preservation of contrast in order to preserve detail throughout the tonal range. It is useful for general reproduction of images, particularly pictorial or photographic-type images.

A.4.3 Saturation Intent

The exact gamut mapping of the saturation intent is vendor specific and involves compromises such as trading off preservation of hue in order to preserve the vividness of pure colors. It is useful for images which contain objects such as charts or diagrams.

© 2003 ICC

Annex B
Embedding Profiles

This annex details the requirements and options for embedding device profiles within PICT, EPS, TIFF, JFIF, and GIF image files. All profiles except Abstract and DeviceLink profiles can be embedded. The complete profile must be embedded with all tags intact and unchanged.

B.1 Embedding ICC profiles in PICT files

Apple has defined a QuickDraw picture comment type for embedded ICC profiles. The picture comment value of 224 is followed by a 4-byte selector that describes the type of data in the comment. Using a selector allows the flexibility to embed more CMM related information in the future. The following selectors are currently defined:

Table 82 — PICT selectors

Selector	Description	
0	Beginning of an ICC profile.	Profile data to follow.
1	Continuation of ICC profile data.	Profile data to follow.
2	End of ICC profile data.	No profile data follows.

Because the dataSize parameter of the PicComment procedure is a signed 16-bit value, the maximum amount of profile data that can be embedded in a single picture comment is 32763 bytes (32767 - 4 bytes for the selector). You can embed a larger profile by using multiple picture comments of selector type 1. The profile data must be embedded in consecutive order, and the last piece of profile data must be followed by a picture comment of selector type 2.

All embedded ICC profiles, including those that fit within a single picture comment, must be followed by the end-of-profile picture comment (selector 2), as shown in the following examples.

EXAMPLE 1: Embedding a 20K profile.

```
PicComment kind = 224, dataSize = 20K + 4, selector = 0, profile data = 20K

PicComment kind = 224, dataSize = 4, selector = 2
```

EXAMPLE 2: Embedding a 50K profile.

```
PicComment kind = 224, dataSize = 32K, selector = 0, profile data = 32K - 4 PicComment
kind = 224, dataSize = 18K + 8, selector = 1, profile data = 18K + 4

PicComment kind = 224, dataSize = 4, selector = 2
```

In ColorSync 1.0, picture comment types `CMBeginProfile` (220) and `CMEndProfile` (221) are used to begin and end a picture comment. The `CMBeginProfile` comment is not supported for ICC profiles; however, the `CMEndProfile` comment can be used to end the current profile and begin using the System Profile for both ColorSync 1.0 and 2.0.

The `CMEnableMatching` (222) and `CMDisableMatching` (223) picture comments are used to begin and end color matching in both ColorSync 1.0 and 2.0

See “Advanced Color Imaging on the Mac OS”, Apple Computer 1995, for more information about picture comments.

© 2003 ICC

B.2 Embedding ICC profiles in EPS files

There are two places within EPS files that embedding International Color Consortium (ICC) profiles are appropriate. 1) Associated with a screen preview. 2) Associated with the page description. Embedding ICC profiles within a screen preview is necessary so that applications using this screen preview to display a representation of the EPS page description can do so with accurate colors. Embedding ICC profiles within a page description is necessary so that sophisticated applications, such as OPI server software, can perform color conversions along with image replacement. For general information concerning PostScript's Document Structuring Conventions (DSC), the EPS file format, or specific PostScript operators, see the PostScript Language Reference Manual, second edition.

1) There are a variety of different methods of storing a screen preview within an EPS file depending on the intended environment. For cross platform applications with embedded ICC profiles, TIFF screen previews are recommended. The TIFF format has been extended to support the embedding of ICC profiles. ICC profiles can also be embedded in a platform specific manner. For example on the Macintosh, Apple has defined a method for embedding ICC profiles in PICT files (see B.1).

Note that a given page description may use multiple distinct color spaces. In such cases, color conversions must be performed to a single color space to associate with the screen preview.

2) ICC profiles can also be embedded in the page description portion of an EPS file using the `%%BeginICCProfile:` / `%%EndICCProfile` comments. This convention is defined as follows.

```
%%BeginICCProfile:  <profileid> <numberof> [<type> [<bytesorlines>]]
<profileid> ::= <text> (Profile ID)
<numberof> ::= <int> (Lines or physical bytes)
<type> ::= Hex | ASCII (Type of data)
<bytesorlines> ::= Bytes | Lines  (Read in bytes or lines)
%%EndICCProfile (no keywords)
```

These comments are designed to provide information about embedded ICC profiles. If the type argument is missing, ASCII data is assumed. ASCII refers to an ASCII base-85 representation of the data. If the `bytesorlines` argument is missing, `<numberof>` shall be considered to indicate bytes of data. If `<numberof>` = -1, the number of bytes of data are unknown. In this case, to skip over the profile one must read data until the encountering the `%%EndICCProfile` comment.

`<profileID>` provides the profile's ID in order to synchronize it with PostScript's setcolorspace and findcolorrendering operators and associated operands (see below). Note that `<numberof>` indicates the bytes of physical data, which vary from the bytes of virtual data in some cases. With hex, each byte of virtual data is represented by two ASCII characters (two bytes of physical data). Although the PostScript interpreter ignores white space and percent signs in hex and ASCII data, these count toward the byte count.

Each line of profile data shall begin with a single percent sign followed by a space (%). This makes the entire profile section a PostScript language comment so the file can be sent directly to a printer without modification. The space avoids confusion with the open extension mechanism associated with DSC comments.

ICC profiles can be embedded within EPS files to allow sophisticated applications, such as OPI server software, to extract the profiles, and to perform color processing based on these profiles. In such situations it is desirable to locate the page description's color space and rendering intent, since this color space and rendering intent may need to be modified based on any color processing. The `%%BeginSetColorSpace:`

© 2003 ICC

/ %%EndSetColorSpace and %%BeginRenderingIntent: / %%EndRenderingIntent comments are used to delimit the color space and rendering intent respectively.

```
%%BeginSetColorSpace: <profileid>
<profileid> ::= <text> (ICC Profile ID)
%%EndSetColorSpace (no keywords)
```

<profileid> provides the ICC profile's ID corresponding to this color space. The ICC profile with this profile must have occurred in the PostScript job using the %%BeginICCProfile: / %%EndICCProfile comment convention prior to this particular %%BeginSetColorSpace: comment.

An example usage is shown here for CIE 1931 (XYZ)-space with D65 white point that refers to the ICC profile with <profileid> = XYZProfile.

```
%%BeginSetColorSpace: XYZProfile
[/CIEBasedABC <<
/WhitePoint [0.9505 1 1.0890]
/RangeABC [0 0.9505 0 1 0 1.0890]
/RangeLMN [0 0.9505 0 1 0 1.0890]
>>] setcolorspace
%%EndSetColorSpace
```

Note that the setcolorspace command is included within the comments. The PostScript enclosed in these comments shall not perform any other operations other than setting the color space and shall have no side effects.

```
%%BeginRenderingIntent: <profileid>
<profileid> ::= <text>            (ICC Profile ID)
%%EndRenderingIntent (no keywords)
```

<profileid> provides the ICC profile's ID corresponding to this rendering intent. The ICC profile with this profile must have occurred in the PostScript job using the %%BeginICCProfile: / %%EndICCProfile comment convention prior to invocation of this particular %%BeginRenderingIntent: comment.

An example usage is shown here for the Perceptual rendering intent that refers to the ICC profile with <profileid> = RGBProfile.

```
%%BeginRenderingIntent: RGBProfile
/Perceptual findcolorrendering pop
/ColorRendering findresource setcolorrendering
%%EndRenderingIntent
```

Note that the setcolorrendering command is included within the comments. The PostScript enclosed in these comments shall not perform any other operations other than setting the rendering intent and shall have no side effects.

© 2003 ICC

B.3 Embedding ICC profiles in TIFF files

The discussion below assumes some familiarity with TIFF internal structure. It is beyond the scope of this document to detail the TIFF format, and readers are referred to the “TIFF™ Revision 6.0” specification, which is available from Adobe Systems Incorporated.

The International Color Consortium has been assigned a private TIFF tag for purposes of embedding ICC device profiles within TIFF image files. This is not a required TIFF tag, and Baseline TIFF readers are not currently required to read it. It is, however, strongly recommended that this tag be honored.

An ICC device profile is embedded, in its entirety, as a single TIFF field or Image File Directory (IFD) entry in the IFD containing the corresponding image data. An IFD should contain no more than one embedded profile. A TIFF file may contain more than one image, and so, more than one IFD. Each IFD may have its own embedded profile. Note, however, that Baseline TIFF readers are not required to read any IFDs beyond the first one.

The structure of the ICC Profile IFD Entry is as follows.

Table 83 — ICC profile IFD entry structure

Byte Offset	Content
0..1	The TIFFTag that identifies the field = 34675(8773.H)
2..3	The field Type = 7 = UNDEFINED (treated as 8-bit bytes).
4..7	The Count of values = the size of the embedded ICC profile in bytes.
8..11	The Value Offset = the file offset, in bytes, to the beginning of the ICC profile.

Like all IFD entry values, the embedded profile must begin on a 2-byte boundary, so the Value Offset will always be an even number.

A TIFF reader should have no knowledge of the internal structure of an embedded ICC profile and should extract the profile intact.

B.4 Embedding ICC profiles in JPEG files

The JPEG standard (ISO/IEC 10918-1) supports application specific data segments. These segments may be used for tagging images with ICC profiles. The APP2 marker is used to introduce the tag. Given that there are only 15 supported APP markers, there is a chance of many applications using the same marker. ICC tags are thus identified by beginning the data with a special null terminated byte sequence, "ICC_PROFILE".

The length field of a JPEG marker is only two bytes long; the length of the length field is included in the total. Hence, the values 0 and 1 are not legal lengths. This would limit maximum data length to 65533. The identification sequence would lower this even further. As it is quite possible for an ICC profile to be longer than this, a mechanism must exist to break the profile into chunks and place each chunk in a separate marker. A mechanism to identify each chunk in sequence order would thus be useful.

The identifier sequence is followed by one byte indicating the sequence number of the chunk (counting starts at 1) and one byte indicating the total number of chunks. All chunks in the sequence must indicate the same total number of chunks. The one-byte chunk count limits the size of embeddable profiles to 16707345 bytes.

© 2003 ICC

B.5 Embedding ICC profiles in GIF files

The GIF89a image file format supports Application Extension blocks, which are used for "application specific" information. These blocks may be used for tagging images with ICC profiles.

The Application Identifier for an embedded profile shall be the following 8 bytes: "ICCRGBG1". The Authentication Code shall be "012". The entire profile shall be embedded as application data, using the conventional technique of breaking the data into chunks of at most 255 bytes of data.

© 2003 ICC

Annex C
Relationship between ICC Profiles and PostScript CSAs and CRDs

C.1 Introduction

When ICC profiles are used to generate PostScript color space arrays (CSAs) or color rendering dictionaries (CRDs) it is useful to be able to identify the profile used to define the CSA or CRD. This can be achieved by adding the following keys to the CSA or CRD. This mechanism does not rely on comments, and enables a parser to obtain the original profile from outside the PostScript file.

C.2 Profile identification keys for a PostScript CSA

The following keys are recommended by Adobe Systems for inclusion in PostScript (and EPS) color space arrays.

/CreationDate (string)

Identifies the date and time at which the color space array was created or most recently modified.

The value of this entry should be coordinated with the calibrationDateTimeTag attribute of any associated ICC profile, and its syntax should conform to the international standard ASN.1, defined in the document ISO/IEC 8824.

/RenderingIntent (name or string)

Identifies the rendering intent that this color space array is designed to achieve. Must be one of: AbsoluteColorimetric, RelativeColorimetric, Saturation or Perceptual.

/Description (string)

7-bit ASCII description string from the ICC profile 'desc' tag.

/Copyright (string)

7-bit ASCII copyright string from the ICC profile 'cprt' tag.

Note: In version 4 profiles, the copyright and description strings are multi-lingual. Only the U.S. English string from the ICC profile is present in the CSA/CRD. If the ICC profile does not contain a U.S. English string, one should be computed from the first multi-lingual string.

/ColorSpace (string)

Color model of the profile data from the ICC profile header. Must be the 4-character ASCII string representing the ColorSpace signature (see section 6.1.5).

/ProfileID (hexadecimal string)

This is the Profile ID of the ICC profile. This shall be encoded as hexadecimal data, enclosed in < and >.

For version 4 profiles, Profile ID is generally present in the profile header. For those ICC profiles not containing Profile ID, a Profile ID should be computed using the method described in section 6.1.13.

© 2003 ICC

Example CSA from Photoshop:

```
[ /CIEBasedABC
  <<
    /CreationDate    (19990603000000)
    /RenderingIntent (Perceptual)
    /Description     (not Adobe RGB (1998))
    /ColorSpace      (RGB )
    /Copyright       (Copyright 1999 Adobe Systems Incorporated)
    /ProfileID       <33BC7F1C156FA0D72F8F717AE5886BD4>
    /DecodeLMN       [{2.1992 exp}bind {2.1992 exp}bind {2.1992 exp}bind]
    /MatrixLMN       [0.3805 0.7083 0.9959
                      0.1282 0.0593 0.7144
                      0.4554 0.2324 0.0145]
    /WhitePoint      [0.9642 1.0000 0.8249]
  >>
]
```

C.3 Profile identification keys for a PostScript CRD

The following keys are recommended by the PostScript Language Reference Manual for inclusion in PostScript colorrendering dictionaries:

/CreationDate (string)

Identifies the date and time at which the color rendering dictionary was created or most recently modified.

The value of this entry should be coordinated with the calibrationDateTimeTag attribute of any associated ICC profile, and its syntax should conform to the international standard ASN.1, defined in the document ISO/IEC 8824-1.

/RenderingIntent (name or string)

Identifies the rendering intent that this color rendering dictionary is designed to achieve. Must be one of: AbsoluteColorimetric, RelativeColorimetric, Saturation or Perceptual.

The use of the following additional keys is also recommended in cases where it is important to establish a clear relationship between the CRD and the ICC profile from which it was derived.

/Description (string)

7-bit ASCII description string from the ICC profile 'desc' tag.

/Copyright (string)

7-bit ASCII copyright string from the ICC profile 'cprt' tag.

Note: In ICC4 Profiles, the copyright and description strings are multi-lingual. Only the U.S. English string from the ICC Profile is present in the CSA/CRD. If the ICC Profile does not contain a U.S. English string, one should be computed from the first multi-lingual string.

© 2003 ICC

/ColorSpace (string)

Color Model of the profile data from the ICC profile header. Must be the 4-character ASCII string representing the ColorSpace signature (see section 6.1.5).

/ProfileID (string)

ASCII string representation of the hex-encoded Profile ID of the ICC Profile.

For version 4 profiles, Profile ID is generally present in the profile header. For those ICC profiles not containing Profile ID, a Profile ID should be computed using the method described in section 6.1.13.

© 2003 ICC

Annex D
Profile Connection Space

The information necessary to adequately define the Profile Connection Space (PCS) is contained in annex D.1. While complete, this information is difficult to interpret without additional explanation and background material. This supporting information, along with examples and suggestions, is contained in annex D.2. These two types of material have been separated to facilitate future movement of the ICC Profile Specification into a formal standard in which normative information must be separated from informative information. Annex D.1 represents the normative information necessary to define the PCS. Annex D.2 would form the informative part of an ISO or IEC standard.

D.1 Requirements

D.1.1 The PCS Definition

The Profile Connection Space is the reference color space in which colors shall be encoded in order to provide an interface for connecting source and destination transforms. The PCS values shall constitute an encoding of a CIE colorimetric specification.

In perceptual transforms the PCS values shall represent hypothetical measurements of a color reproduction on a reference medium. By extension, for the perceptual intent, the PCS represents the appearance of that reproduction as viewed in the reference viewing environment by a human observer adapted to that environment.

In transforms for the media-relative colorimetric intent the PCS values shall represent media-relative measurements of the captured original (for input profiles), or media-relative color reproductions produced by the output device (for output profiles).

In transforms for the ICC-absolute colorimetric intent the PCS values shall represent measurements of the captured original relative to a hypothetical perfectly reflecting or transmitting diffuser (for input profiles), or color reproductions produced by the output device relative to a hypothetical perfectly reflecting or transmitting diffuser (for output profiles).

In transforms for the media-relative and ICC-absolute colorimetric intents, the PCS values may represent a color rendering of the actual original captured for input profiles. Likewise for output profiles, the PCS values may be color rendered by the output device to the actual medium. However, wherever ICC profiles are used, the PCS values resulting from such transforms shall be interpreted as the colorimetry of the original and reproduction, regardless of whether such colorimetry is the actual colorimetry.

D.1.2 PCS Colorimetric Specification

The colorimetric values used by the PCS shall be as described in Annex A. The colorimetry shall be assumed not to contain any flare or other defect caused by inadequacies in the optical system of the instrument and illumination used to make the measurements, but is assumed to include the surface reflection component normally associated with the prescribed measurement geometry.

D.1.3 PCS Encoding

In transforms for the media-relative colorimetric, perceptual, and saturation rendering intents (intents other than ICC-absolute colorimetric), the white point of the actual medium, and the white point of the reference medium are represented in PCS XYZ and PCS Lab formats as follows.

© 2003 ICC

Table 84 — White point encodings

Component	Value	8-bit Encoding	16-bit Encoding		Component	Value	Encoding
L*	100	255	65535		**X**	0,9642	31595
a*	0	128	32896		**Y**	1,0000	32768
b*	0	128	32896		**Z**	0,8249	27030

In transforms for the media-relative colorimetric intent the perfect absorber (a theoretical medium that reflects absolutely no light) is represented in PCS XYZ and PCS Lab formats as follows. Other reflectance values are mapped linearly to PCS XYZ.

Table 85 — Perfect absorber encoding

Component	Value	8-bit Encoding	16-bit Encoding		Component	Value	Encoding
L*	0	0	0		**X**	0,0	0
a*	0	128	32896		**Y**	0,0	0
b*	0	128	32896		**Z**	0,0	0

In transforms for the perceptual and saturation intents the black point of the reference medium is represented in PCS XYZ and PCS Lab formats as follows. This is here called the PCS perceptual black point.

Table 86 — Black point encoding of reference media

Component	Value	8-bit Encoding	16-bit Encoding		Component	Value	Encoding
L*	3,1373	8	2056		**X**	0,00336	110
a*	0	128	32896		**Y**	0,0034731	114
b*	0	128	32896		**Z**	0,00287	94

NOTE: Due to limited numerical precision, Y encoded as 114 does not exactly match L* encoded as 8.

NOTE: An adjustment must be applied to older perceptual transforms that use a PCS value of zero to represent the black point of the reference medium. The zero point of such a transform needs to be mapped to the above PCS perceptual black point. The white point remains unchanged. All other values are mapped linearly in XYZ. The following equations can be used for the adjustment of such a transform to the above PCS encoding.

$$Xp = Xt * (1 - Xb / Xi) + Xb \qquad \text{D.1.3.0.0-0}$$

$$Yp = Yt * (1 - Yb / Yi) + Yb \qquad \text{D.1.3.0.0-1}$$

$$Zp = Zt \ *(1 - Zb / Zi) + Zb \qquad \text{D.1.3.0.0-2}$$

© 2003 ICC

Where:

Xt, Yt, Zt = original PCS XYZ value in the transform
Xb, Yb, Zb = CIE XYZ values for the PCS perceptual black point (X = 0,00336, Y = 0,0034731, Z = 0,00287)
Xi, Yi, Zi = CIE XYZ values of the PCS white point (X = 0,9642, Y = 1,0000, Z = 0,8249)
Xp, Yp, Zp = the adjusted PCS XYZ value

D.1.4 The Reference Viewing Environment

The reference viewing condition applies only to the perceptual transform. If the actual viewing environment differs from the reference viewing environment perceptual transforms must compensate for the differences in viewing environments.

The reference viewing environment, as specified in A.1, shall be taken as the aim viewing environment underpinning the use of the PCS. It shall be based on standard viewing condition P2, as specified for graphic arts and photography in ISO 3664. It is characterized by an "average" surround, which means that the illumination of the image shall be assumed to be similar to the illumination of the rest of the environment. The surfaces immediately surrounding the image shall be assumed to be a uniform matte gray with a reflectance of 20%. ISO 3664 specifies that the illumination in the plane of viewing shall be that of CIE standard illuminant D50 within specific tolerances. The chromaticity of this illumination (x_{10} = 0,3478, y_{10} = 0,3595 in the CIE x_{10}, y_{10} chromaticity diagram) shall define the chromatic adaptation state associated with the PCS. ISO 3664 describes the appropriate illumination level for practical appraisal of prints as 500 lux and ISO 3664 describes the appropriate illumination level for practical appraisal of prints as 500 lux, which is specified to be typical of actual home and office viewing environments. This was deemed to be most appropriate for the reference viewing environment, which shall also be assumed to have a level of stray light such that the veiling glare observed by a viewer, in addition to that measured using the prescribed measurement geometry, (the viewing flare[1)]) is a factor of 0,0075 (3/4%) times the illumination level as reflected by the reference white, with the same chromaticity as the illumination (4,25 cd/m^2).

D.1.5 The Reference Medium

The reference medium applies only to the perceptual transform.

The reference medium shall be a hypothetical print on a substrate having a neutral reflectance of 89%. The darkest printable color on this medium shall have a neutral reflectance of 0,30911%, which is 0,34731% of the substrate reflectance. These are the white point and black point of the reference medium. The reference medium therefore has a linear dynamic range of 287,9 :1 and a density range of 2,4593.

D.2 Explanation and Background Material (informative)

The following material is provided to aid in the understanding and interpretation of the PCS requirements.

[1)] E. J. Giorgianni and T. E. Madden, *Digital Color Management*, Addison-Wesley, Reading, Massachusetts, 1998, p. 561.

© 2003 ICC

D.2.1 Introduction

The concept of a Profile Connection Space is a vital element in the ICC architecture. It allows the profile transforms for input, display, and output devices to be decoupled so that they can be produced independently. A well-defined PCS provides the common interface for the individual device profiles. It is the virtual destination for input transforms and the virtual source for output transforms. If the input and output transforms are based on the same PCS definition, even though they are created independently, they can be paired arbitrarily at run time by the color-management engine and will yield consistent and predictable results when applied to color values.

The key to effective use of the profile specification is an unambiguous definition of the PCS. However, there is probably no definition that will yield optimal results for all possible color-management scenarios involving all possible input media, all possible output media, and all possible market preferences. Where trade-offs are necessary, the preference has been to serve the needs of applications in graphic arts and desktop publishing. For this reason the PCS definition is biased somewhat toward scenarios that result in output to reflection-print media such as offset lithography, off-press proofing systems, computer-driven printers of various kinds, and photographic paper. Even with this bias, the PCS will provide good results in other applications such as video production, slide production, and presentation graphics.

D.2.2 Encoding of PCS Measurements

In Annex A to the specification, the PCS is defined to be based on colorimetry relative to the media white point. These two factors are accommodated by the encoding part of the PCS definition.

Part of the PCS encoding, for the perceptual intent, normalizes the reference medium's white point to the PCS white point. This procedure corresponds to using media-relative colorimetry when the reference medium's white point is the media white point (media-relative colorimetric intent).

For the perceptual intent, the normalized CIE XYZ values of the reference medium black point are used as the color rendering target black point values. This provides a specific reference in the PCS for the black point and dynamic range of the target ideal reflection print.

The choice of a reference medium with a realistic black point for the perceptual intent provides a well-defined aim when tonal remapping is required. Inputs with a dynamic range greater than a reflection print (for example, a slide film image, or the colorimetry of high-range scenes) can have their highlights and shadows smoothly compressed to the range of the print in such a way that these regions can be expanded again without undue loss of detail on output to wide-range media. Note that while this does not impose a limit on the precision of the PCS values, it does require that appropriate precision be maintained in both the image data and the calculations using that data.

In transforms for the colorimetric intents, the range of valid (but not necessarily physically realizable) PCS XYZ values is unrelated to the reference media black point. Instead they reflect instrument readings without tonal remapping. In theory, the dynamic range of the PCS for colorimetric transforms is infinite.

All transforms in an output profile should be able to process all values in the PCS, regardless of whether the values are outside of the destination device gamut.

NOTE The PCS encoding defined here is different to that in version 3 of the ICC specification which defined the PCS as being the encoded colorimetry of an ideal reflection print on a spectrally non-selective substrate with 100% reflectance. This ideal print had an infinite dynamic range, since black could have 0% reflectance.

© 2003 ICC

D.2.3 Color Measurements

In order to establish the relationship between the colorimetry encoded in the PCS (and oriented toward the reference medium and reference environment) and the measured colorimetry of an actual medium, intended for an actual viewing environment, it is useful to describe the measurement conditions more precisely.

In general, the actual viewing illumination source may have a spectral power distribution different from D50. In such cases, the actual illumination source should be used in the color measurements, or, equivalently, the actual illumination spectrum should be used in calculating tristimulus values from the measured spectral reflectances or transmittances of the medium. If the chromaticity of the illumination source is different from that of D50, corrections for chromatic adaptation may be needed and must be incorporated into the colorimetric transforms (see D.2.4 and D.2.6.1) and the perceptual transform (see section D.2.4 and D.2.7.7) by the profile builder. For example, an Alexandrite stone appears to be a purple color when viewed under tungsten illumination. The same stone appears to be sea-green when viewed under daylight. If an image of such a stone is captured under tungsten illumination, its PCS colorimetry (as produced by the input profile) should correspond to a purple color.

For media intended for the graphic arts, it is best that the color measurements conform to ISO 13655, Graphic technology - Spectral measurement and colorimetric computation for graphic arts images. Here, the illumination source spectral power distribution for the calculation of colorimetry is specified to be that of D50. No corrections for chromatic adaptation are required in this case, since the chromaticity of the illumination source is that of D50. Other corrections, as discussed below, may still be applicable. Note that the fluorescent D50 simulators found in typical professional viewing booths, although their chromaticity may agree closely with standard D50, may have rather different spectral distributions (different from each other and different from the CIE definition) so that the measured, or calculated, tristimulus values can vary noticeably. Often, a better description of the observed color can be obtained by basing the colorimetry on the actual, rather than the theoretical, illumination source[1)]. The CIE color rendering and metamerism index criteria specified in ISO 3664 can be used to determine if an actual source is sufficiently close to D50 to minimize spectrally caused visual effects. In critical applications, filtered tungsten D50 simulators may be the best choice to minimize these effects.

As specified in D.1.2, the measurements are assumed not to be contaminated with flare due to the use of low quality instruments or poor measurement technique. This does not imply that it is necessary to remove any surface reflections that are a typical component of $0^o/45^o$ measurements of reflection materials. It is important to note that the difference in flare between the specifications for measurement and viewing is neither a contradiction nor does it add complexity. It is simply a statement of current practice. The measurement conditions have been chosen so as to not require any corrections to high quality measurements of the type typically collected for color management purposes. Similarly, the 3/4% flare of the reference viewing environment was chosen since this is representative of the amount of stray light contributed by high quality, but realistic environments in actual use.

Because the PCS is more a specification of how to reproduce a desired appearance than it is a specification of the appearance itself, it is not necessary (or desirable) to add the 3/4% flare to the measurements before encoding a color in the PCS. Instead, the 3/4% viewing flare is specified to allow compensation for any potential difference between the actual viewing environment and the reference environment.

---

1) D. Walker, "The Effects of Illuminant Spectra on Desktop Color Reproduction', in *Device Independent Color Imaging,* R. Motta and H. Berberian, ed., *Proc. SPIE,* 1909**,** 1993, pp. 236-246.

© 2003 ICC

D.2.4 Chromatic Adaptation

When a person is looking at a real-world scene, the color stimulus presented to the retina by any visible surface in the scene depends on the spectral composition of the light with which the surface is illuminated. This stimulus is what colorimetry attempts to measure, by stipulating how a mixture of three specified stimuli would match it, for a standard observer. If the illuminant is changed the stimulus will also change. Colorimetry measures the change in stimulus and predicts a different color. However, because of adaptation the appearance of the color does not change significantly - despite the change in stimulus incident on the eye - and this seems to indicate a serious limitation in colorimetry which was not created to measure appearance - but only whether two colors match. But this is not the case - because a change also occurs in the white point stimulus it can be used in defining metrics of appearance, of varying complexity, which can predict the change in appearance. To understand this we need some understanding of the way the visual system adapts so flexibly to the color and intensity of the incident light.

The mechanism can be modeled as follows: Through some means, the system infers the color and strength of the presumed illumination source. (In a normal scene, this inference may be based on specular highlights, or the apparent colors of known objects, or some kind of scene average, etc.; for reproductions, the inference may be made from the image itself or, as when viewing reflection prints, objects in the real world surrounding the image.) The system then uses this information to adjust the "gain" applied to the "cone responses" to the color stimuli (the actual process is not well understood, and is most likely more complicated). The result of this adaptation is that the signals received by the brain are much less dependent on the brightness and chromaticity of the illumination source, so that objects can be more easily recognized, regardless of whether the light source is bright or dim, yellowish or bluish, etc. The adaptive mechanism does not compensate perfectly for the change of illuminant, however, so objects do appear somewhat different under different illumination. Note that this mechanism is operative in bright environments; adaptation to the dark is a separate phenomenon.

There are several available models which may be used to represent this process, among them XYZ scaling, the von Kries transformation, the Bradford transformation, and CMCCAT97. The choice of which model to use depends upon the device and the environment in which it is used. Often this choice depends upon the difference in chromaticity between the illumination sources. If the difference is small, a simple model may be suitable, but if the difference is large, a more complicated model may be needed. In some cases it may be necessary to consider the effects of the use of differing methods in the source and destination profiles.

This aspect of the PCS definition provides some flexibility to the color management system as a whole. For example, it is possible to transform data from a medium intended for tungsten illumination to a medium intended for cool-white-fluorescent: the input profile handles the adaptation from tungsten to D50, and the output profile handles the adaptation from D50 to cool-white.

D.2.5 Aesthetic Considerations and the Media White Point

Aside from the adaptive effects mentioned above, there is frequently a strong aesthetic preference for maintaining highlight detail in all renderings of an image. One way to guarantee this result for typical reflection media is to modify the colorimetry of the reproduction so as to factor out the colorimetry of the substrate. This approach is called "media-relative colorimetry," i.e., colorimetry relative to the substrate. (In contrast, "ICC-absolute colorimetry" is called "relative colorimetry" in the CIE terminology, since the data are normalized relative to the perfect diffuser viewed under the same illumination source as the sample. See CIE Publications 15.2-1986, Colorimetry (second edition).) According to this media-relative method the PCS color [100, 0, 0] in CIELAB is associated with the blank substrate, regardless of its actual colorimetry, and all other colors are modified accordingly.

© 2003 ICC

However, there are applications in which the goal is to reproduce the actual colors of an image (within the limitations of color gamut and dynamic range), even if highlight detail must be sacrificed. For instance, the goal may be to simulate one medium on another, for proofing purposes. In these cases, the "ICC-absolute", or "CIE relative", colorimetry is required. The ICC specification provides a mechanism for converting "media-relative" into "ICC-absolute" colorimetry. The profile mediaWhitePointTag defines the colorimetry of the actual substrate, corrected for differences in the viewing conditions, in CIE 1931 XYZ coordinates. (See section D.2.6.2.) Annex A describes the mechanism for using these coordinates in the required conversion.

If the white point mapping discussed above is present in both the input and the output transforms, the white point of the input medium will be mapped, by way of the PCS illumination source, to the white point of the output medium (media-relative colorimetric intent). The ICC-absolute colorimetric rendering intent will also be accurately enabled through the use of the mediaWhitePointTags.

For the perceptual intent, just as it is necessary to correct for viewing environment differences, we need to convert the colorimetry of the actual medium to that desired for the reference medium. This can include mapping the white point of the actual medium to the white point of the reference medium. The white point of the reference medium is then mapped to the PCS white point (see section D.1.3 and step 5 of section D.2.7.7).

In other cases, the goal may be to introduce color shifts which provide a unique aesthetic effect[1)]. In these cases the white point of the actual medium may be mapped to a color other than the white point of the reference medium. This is another means by which unique value may be added to profiles while maintaining data interoperability.

D.2.6 Discussion of Relative Colorimetric Intent

D.2.6.1 Relative and Absolute Intents

For ICC-absolute colorimetric transformations in the context of ICC profiles the media XYZ tristimulus values are reproduced relative to the illumination source or perfect diffuser. The reproduction provided by the ICC-absolute colorimetric intent is said to be illuminant-relative, and L* = I00 for the perfect diffuser. The PCS tristimulus values for an ICC-absolute colorimetric transform of course are also illuminant-relative, that is, PCS L* = 100 for the perfect diffuser. The profile format does not define an explicit transform for ICC-absolute colorimetric intent. For a given profile, the PCS-side XYZ tristimulus values for the ICC-absolute colorimetric intent are obtained from the media-relative colorimetric transform, see below.

For media-relative colorimetric transforms in the context of ICC profiles the media XYZ tristimulus values are reproduced relative to the media white point. Given the media-relative reproduction provided by the media-relative colorimetric intent, L* = 100 for media white. The PCS tristimulus values for a media-relative colorimetric transform are also media-relative, that is, PCS L* = 100 for media white.

The PCS-side XYZ tristimulus values of a colorimetric transform ($XYZ_{\text{mediawhite-relative under D50}}$ for media-relative colorimetric transforms, $XYZ_{\text{under D50}}$ for ICC-absolute colorimetric transforms) are calculated from CIE XYZ tristimulus values ($XYZ_{\text{illuminant-relative}}$) of the device under the actual illumination source ($XYZ_{\text{illuminant}}$)

1) E. J. Giorgianni and T. E. Madden, op. cit., p. 425.

© 2003 ICC

When the actual illumination source differs from the PCS reference illumination source, CIE D50, chromatic adaptation as described in section D.2.4 is required for transforming tristimulus values between the two illumination sources. The PCS-side tristimulus values are obtained as follows:

$$XYZ_{\text{under D50}} = \text{ChromaticAdaptationMatrix}\,(XYZ_{D50}, XYZ_{\text{illuminant}}) * XYZ_{\text{illuminant-relative}} \qquad \text{D.2.6.1.0-1}$$

$$XYZ_{\text{under D50 mediawhite}} = \text{ChromaticAdaptationMatrix}\,(XYZ_{D50}, XYZ_{\text{illuminant}}) * XYZ_{\text{illuminant-relative mediawhite}} \qquad \text{D.2.6.1.0-2}$$

$$\begin{aligned} X_{\text{mediawhite-relative}} &= X_{D50} / X_{\text{under D50 mediawhite}} * X_{\text{under D50}} \\ Y_{\text{mediawhite-relative}} &= Y_{D50} / Y_{\text{under D50 mediawhite}} * Y_{\text{under D50}} \\ Z_{\text{mediawhite-relative}} &= Z_{D50} / Z_{\text{under D50 mediawhite}} * Z_{\text{under D50}} \end{aligned} \qquad \text{D.2.6.1.0-3}$$

Where:

> ChromaticAdaptationMatrix (illuminant 2, illuminant 1) is a 3 by 3 matrix that adjusts XYZ tristimulus values from illuminant 1 to illuminant 2 using chromatic adaptation, see section D.2.4

$XYZ_{\text{under D50 mediawhite}}$ must be stored in the mediaWhitePointTag, whether the illuminant is D50 or not.

ChromaticAdaptationMatrix should be stored in the chromaticAdaptationTag, 'chad' (63686164h), when the actual illumination source is not CIE D50.

Note that the scaling between media-relative and ICC-absolute colorimetric values is done under the PCS illumination source. Also note that the observer is assumed to be adapted to the perfect diffusers not to the media white.

If the actual illumination source is CIE D50, that is, the same as the PCS illumination source, the above equations are simplified to:

$$\begin{aligned} X_{\text{under D50}} &= X_{\text{illuminant-relative}} \\ Y_{\text{under D50}} &= Y_{\text{illuminant-relative}} \\ Z_{\text{under D50}} &= Z_{\text{illuminant-relative}} \end{aligned} \qquad \text{D.2.6.1.0-4}$$

$$\begin{aligned} X_{\text{under D50 mediawhite}} &= X_{\text{illuminant-relative mediawhite}} \\ Y_{\text{under D50 mediawhite}} &= Y_{\text{illuminant-relative mediawhite}} \\ Z_{\text{under D50 mediawhite}} &= Z_{\text{illuminant-relative mediawhite}} \end{aligned} \qquad \text{D.2.6.1.0-5}$$

$$\begin{aligned} X_{\text{mediawhite-relative}} &= X_{D50} / X_{\text{under D50 mediawhite}} * X_{\text{under D50}} \\ Y_{\text{mediawhite-relative}} &= Y_{D50} / Y_{\text{under D50 mediawhite}} * Y_{\text{under D50}} \\ Z_{\text{mediawhite-relative}} &= Z_{D50} / Z_{\text{under D50 mediawhite}} * Z_{\text{under D50}} \end{aligned} \qquad \text{D.2.6.1.0-6}$$

The ICC profile format does not include an explicit transform for ICC-absolute colorimetric intent. On creating a profile only the $XYZ_{\text{mediawhite-relative under D50}}$ values are stored in a profile, not $XYZ_{\text{under D50}}$. When using a profile, after obtaining the media-relative colorimetric transform of the profile, the PCS-side XYZ tristimulus values for the ICC-absolute colorimetric intent are calculated from the media-relative colorimetric transform through a simple scaling operation:

$$\begin{aligned} X_{\text{under D50}} &= X_{\text{under D50 mediawhite}} / X_{D50} * X_{\text{mediawhite-relative}} \\ Y_{\text{under D50}} &= Y_{\text{under D50 mediawhite}} / Y_{D50} * Y_{\text{mediawhite-relative}} \\ Z_{\text{under D50}} &= Z_{\text{under D50 mediawhite}} / Z_{D50} * Z_{\text{mediawhite-relative}} \end{aligned} \qquad \text{D.2.6.1.0-7}$$

© 2003 ICC

The following symbols are used above. The prefix XYZ identify tristimulus values in the form of a 3 rows by 1 column vector..

Table 87 — Relative and absolute rendering intent equation symbols

ChromaticAdaptationMatrix	a 3 by 3 matrix for chromatic adaptation between two illumination sources
X, Y, Z, XYZ_{D50}	relative CIE XYZ tristimulus values for the PCS illumination source, CIE D50. X=0,9642, Y= 1, Z = 0,8249
X, Y, Z, $XYZ_{illuminant}$	relative CIE XYZ tristimulus values for the perfect reflecting diffuser under the actual illumination source. Y = 1
X, Y, Z, $XYZ_{illuminant\text{-}relative}$	relative CIE XYZ tristimulus values for a color patch on the media under the actual illumination source, flare-free. Y = 1 for the perfect reflecting diffuser
X, Y, Z, $XYZ_{illuminant\text{-}relative\ mediawhite}$	relative CIE XYZ tristimulus values for the media white point under the media white actual illumination source, flare-free. Y = 1 for the perfect reflecting diffuser
X, Y, Z, $XYZ_{under\ D50\ mediawhite}$	relative CIE XYZ tristimulus values for the media white point under the PCS illumination source
X, Y, Z, $XYZ_{mediawhite\text{-}relative}$	PCS-side XYZ tristimulus values of a media-relative colorimetric transform
X, Y, Z, $XYZ_{under\ D50}$	PCS-side XYZ tristimulus values for ICC-absolute colorimetry

D.2.6.2 Procedural Summary

The various colorimetric adjustments discussed above can be organized into a computational procedure for calculating PCS coordinates for device-profile transforms. The procedure presented here is applicable to reflection media input and output profiles; monitor transforms are typically computed in a simplified manner, although it is certainly possible to treat monitors in the same way as other input and output devices in order to achieve more accurate image display.

The procedure is given in the device-to-PCS direction for the media-relative colorimetric rendering intent (AToB1Tag) transform.

1. Obtain CIE 1931 XYZ tristimulus values (XYZ measured) for a set of color patches on the device or media to be profiled. More information about measurement procedures is provided in section D.2.3. There should be at least one measurement of the "media white" and one measurement of the illumination source or perfect reflecting diffuser.

2. Remove flare from the measured XYZ values as needed to match the PCS measurement conditions, creating flare-free XYZ values (XYZ flare-free)

3. If necessary, scale the flare-free measurement values so they are relative to the actual illumination source by dividing all values by the measured Y value of the perfect diffuser. After scaling Y = 1 for the perfect diffuser.

X illuminant-relative = X flare-free / Y flare-free for perfect diffuser
Y illuminant-relative = Y flare-free / Y flare-free for perfect diffuser
Z illuminant-relative = Z flare-free / Y flare-free for perfect diffuser D.2.6.2.0-1

© 2003 ICC

4. If the chromaticity of the illumination source is different from that of D50, convert the illuminant-relative XYZ values from the illumination source white point chromaticity to the PCS white point chromaticity using an appropriate chromatic adaptation transform and equation D.2.6.1.0-1. This may be done by applying one of the transformations mentioned in D.2.4. The transform used should be specified in the chromaticAdaptationTag.

$$XYZ_{\text{under D50}} = \text{ChromaticAdaptationMatrix} * \text{XYZ illuminant-relative} \qquad \text{D.2.6.2.0-2}$$

5. Record the converted media white point in the mediaWhitePointTag. Optionally, record the converted black point in the mediaBlackPointTag.

6. Convert colorimetry from D50 illuminant-relative to mediawhite-relative values, by scaling each value by the ratio of the PCS D50 illumination source over the converted media white point, using equation D.2.6.1.0-3. After scaling, the XYZ values for the media white point measurement will be equal to the XYZ values of the PCS D50 illumination source.

$$\begin{aligned} &X \text{ mediawhite-relative} = X_{\text{under D50}} * X \text{ D50 illuminant} / X_{\text{under D50}} \text{ mediawhite} \\ &Y \text{ mediawhite-relative} = Y_{\text{under D50}} * Y \text{ D50 illuminant} / Y_{\text{under D50}} \text{ mediawhite} \\ &Z \text{ mediawhite-relative} = Z_{\text{under D50}} * Z \text{ D50 illuminant} / Z_{\text{under D50}} \text{ mediawhite} \end{aligned} \qquad \text{D.2.6.2.0-3}$$

7. Optionally, convert the adjusted PCS XYZ coordinates to PCS L*a*b* as described in Annex A.

8. Encode the PCS XYZ coordinates or the PCS L*a*b* coordinates digitally in 8-bit or 16-bit representations, as defined in Annex A.

These values can now be used to populate the AToB1Tag.

D.2.6.3 Example

This example shows how the standard data for SWOP, as published in CGATS TR001, could be used when building a device to PCS transform for the media-relative colorimetric intent. The TR001 data can be used as the measurement data needed for step one in D.2.6.2. The actual viewing environment is a graphic arts viewing booth with an illuminance level of 2000 lux. The example shows how white and black would be converted into PCS values for a transform implementing the media-relative colorimetric rendering intent of a profile.

1. The white (no colorant, Patch 26 of IT8.7/3) and black (100% of all colorants, Patch 24 of IT8.7/3) patches have zero flare CIE XYZ values of

Table 88 — Zero flare CIE XYZ values

	white	black
X	0,7067	0,0097
Y	0,7346	0,0101
Z	0,5703	0,0080

2. These measurements do not need to be corrected for flare. The white and black values are unchanged.

3. These values are already relative to the illumination source, so they do not need to be scaled. The white and black values are unchanged.

© 2003 ICC

4. This illumination source is D50, so no chromatic adaptation is needed. The white and black values are unchanged.

5. Record the white and black values in the media white and black point tags.

6. The CIE XYZ values are mapped to PCS by multiplying them by the ratio of the PCS white point to the actual media white point under D50 illumination source:

Table 89 — CIE XYZ to PCS multipliers

	ratio	white	black
X	0.9642 / 0.7067	0,9642	0,0134
Y	1 / 0.7364	1,0000	0,0138
Z	0.8249 / 0.5703	0,8249	0,0116

7. Convert PCS XYZ to PCS L*a*b*:

Table 90 — PCS XYZ to PCS L*a*b* conversion

	white	black
L*	100	11,8
a*	0	0,28
b*	0	-0,3

8. Convert PCS XYZ and PCS L*a*b* to PCS encodings:

Table 91 — PCS XYZ and PCS L*a*b* to PCS conversion

16-bit	white	black	16-bit	white	black	8-bit	white	black
X	31595	439	**L***	65535	7733	**L***	255	30
Y	32768	452	**a***	32896	32968	**a***	128	128
Z	27030	380	**b***	32896	32819	**b***	128	128

Note that the 8-bit L*a*b* encoding of black is imprecise because of the limited precision afforded by 8-bit data.

D.2.7 A Discussion of Perceptual Rendering Intent

D.2.7.1 Colorimetry and Appearance

One possible definition for the PCS is that it specifies the colorimetry of an image reproduction. Colorimetry, as established by the CIE, is a system of measurement and quantification of visual color stimuli. As such, it is independent of any particular device, medium, or process. This makes it a suitable candidate for

© 2003 ICC

a common interface. With this choice, the output reproduction of an image would present the same color stimuli to an observer as the input, even if it employs a different process of color reproduction. This seems to guarantee the same colors on all media, which would make it the right definition for the PCS for the purposes of color management.

Unfortunately, this simple definition is inadequate for appearance matching as requested by the perceptual intent. The appearance of a color depends not only on the color stimulus presented to the retina, but also on the state of visual adaptation of the observer. In certain cases, different media require different visual color stimuli because they will be viewed in different environments. For example, differences in surround condition or illumination source chromaticity will cause the observer to experience different visual adaptation effects. In order to preserve the same color appearance in these environments, the colorimetry must be corrected to compensate for the adaptation of the human visual system and for physical differences in the viewing environments. By extension, these effects occur in images where the immediate surround of any color in an image consists of other colors in the image. If the relation of any of the colors in the image is changed, for example because of gamut limitations, the color stimuli required to reproduce the image may change, even though the viewing environment for the whole image does not change. **Color appearance is still an active research topic.** Color appearance models for single stimuli function well, and are applicable to images when gamut limitations do not come into play. The science in support of generalized image appearance modeling is less well developed.

There are also aesthetic reasons why it may be necessary or desirable to alter the colorimetry for specific media. For instance, hard-copy media - even those intended for the same viewing environment - differ considerably in their dynamic range and color gamut. A well-crafted rendering of an image on a specific medium will take advantage of the capabilities of that medium without creating objectionable artifacts imposed by its limitations. For instance, the tone reproduction of the image should attempt to provide sufficient contrast in the midtones without producing blocked-up shadows or washed-out highlights. The detailed shape of the tone curve will depend on the brightest and darkest tones (the maximum and minimum reflectances) attainable in the medium. Clearly, there is considerable art involved in shaping the tone-reproduction and color-reproduction characteristics of different media and much of this art is based on subjective, aesthetic judgments. As a result, the substrate and the colorants used in a medium will be exploited to impart a particular personality to the reproduction that is characteristic of the medium. In reproducing an image on various types of media, it may be desirable to adjust the colorimetry to accommodate the differing characteristics of those media. In any case, it is necessary to accommodate the gamut differences. Such considerations go beyond the simplistic matching of color stimuli or even of color appearance.

These adjustments need to be incorporated in the color transforms of the device profiles. Since the PCS is the common interface of these profiles, it has to be defined in a way that facilitates these adjustments. Thus, although the definition of the PCS may be based on the principles of colorimetry, it must also take into account various issues that lie outside the realm of colorimetry and that involve adaptive corrections, pragmatic considerations, and aesthetic judgments.

D.2.7.2 Purpose and Intent of the PCS

These considerations led to a fundamental statement that the PCS for perceptual rendering intent represents desired appearance. The term "desired" implies that the PCS is oriented towards colors to be produced on an output medium. Obviously, "desired" is open to various interpretations, but in order to enable the decoupling of input and output transforms, it must be interpreted in a way that, to the extent possible, transcends the capabilities and limitations of the specific color-reproduction processes, devices, and media for which profiles are to be provided.

For instance, an input profile for a slide scanner should attempt to yield "desired" colors, represented in the PCS, that are independent of the gamut and aesthetics of any specific output medium. This independence, which decouples the PCS colors from the device colors, allows the input profile to be used in con-

© 2003 ICC

junction with any output profile. These desired colors will be based on the colors of the input slide but are not necessarily identical to those colors or limited to the gamut of the slide medium. They are the colors that would be desired on output if the characteristics of the potential output media could be transcended.

Similarly, the output profile for a color printer must reproduce the desired colors within the capabilities and limitations of the output medium and device. This reproduction may involve some adjustment of the colors, but it transcends the characteristics of any specific input medium and permits the use of the output profile in conjunction with a variety of different input profiles.

With this PCS definition, it is the responsibility of the profile transforms to handle any required corrections or modifications to the colorimetry of a reproduction. Input profiles are responsible for modifying the colorimetry of the input media to account for adaptation, flare, and gamut limitations. They also must provide the artistic intent implicit in the word "desired", which allows latitude for variation. For instance, the "desired" colors may be a close facsimile of the original, an aesthetic re-rendering of the original, or a simulation of a specific reproduction medium different from both the input and output media.

Output profiles for media that are viewed in environments different from the reference are responsible for modifying the colorimetry to account for the differences in the observer's state of adaptation as well as any substantial differences in viewing flare present in these environments. This is needed in order to preserve color appearance. Profiles must also incorporate adjustments to the dynamic range and color gamut of the image in order to accommodate the limitations of the actual medium.

D.2.7.3 Reference Medium and Reference Viewing Environment

While a profile is needed which represents desired appearance and transcends the actual device, it is difficult to know how to generate such a profile. It is helpful here to conceptualize a "reference medium" which is a hypothetical medium on which the colors are being rendered (see section D.1.5). It has a large gamut and dynamic range which approximate the limits of current reflection-print technology. It is described using "realworld" specifications so that even though the medium is not real, it can be treated as if it were real.

It is also necessary to define a "reference viewing environment" which is the environment in which the reference medium is being viewed (see section D.1.4). This environment is used to determine the observer's adaptation state and establishes the connection between color stimulus and color appearance.

The concept of a reference medium viewed in the reference viewing environment helps the profile designer to understand how to produce "desired appearance" in the PCS. At the same time, it preserves the goal of decoupling the characteristics of actual media through a virtual intermediate reproduction description. Where the real viewing environment differs from that of the reference environment chromatic adaptation may be an important component in the set of adaptation transforms that are applied to obtain conformance with the reference viewing environment.

NOTE For the perceptual intent, the colorimetry represented in the PCS is that of the image as optimally color rendered to the perceptual intent reference medium and viewing conditions. If the illumination source used to view the actual image has a chromaticity different from that of D50, this color rendering will typically include some type of chromatic adaptation. However, the color rendering used to produce the reference medium image colorimetry will also consider other factors, such as dynamic range and gamut mapping, adaptation for other differences between the reference and actual viewing conditions, and preferential color adjustments. For this reason, it may not make sense to invert the chromatic adaptation as specified in the chromaticAdaptationTag, because the result will be the reference medium colorimetry transformed to be relative to the actual illumination source, which may not produce the colorimetry of the actual image. There is no guarantee the colorimetry produced by the inverse of the chromaticAdaptationTag will be optimal for the reference medium under the actual illumination source, since the color rendering to the reference medium could include optimizations based on the D50 reference white.

© 2003 ICC

D.2.7.4 Aesthetic Considerations and the Media White Point

As discussed in section D.2.5, for the perceptual intent the white point of the actual medium can be mapped to the white point of the reference medium. On the other hand, based on aesthetic considerations, the white point of the actual medium can be mapped to a color other than the white point of the reference medium.

In either case, the white point of the reference medium will correspond, after scaling, to the PCS white point (see section D.1.3 and step 5 of section D.2.7.7). This is another means by which unique value may be added to profiles while maintaining data interoperability.

D.2.7.5 Brightness Adaptation and Tone-scale Correction

One of the most fundamental corrections that must be applied to the measured colorimetry has to do with issues of tone reproduction and overall brightness level. These issues involve adaptive effects, as well as aesthetic and pragmatic considerations.

When viewing a reflection print under normal viewing conditions (i.e., where the print and the area surrounding the print are similarly illuminated), the observer becomes adapted to things perceived as white in the environment. A reflection print is perceived as an object in this environment. Now, the brightest areas in the image are those in which the paper (or other substrate) is blank (no colorant). Since the reflectance of any actual paper is limited (typically 85% to 90%), the medium viewed in this environment cannot realistically create the appearance of specular highlights or other very bright objects that may have existed in the original scene, which can be several times brighter than 100% diffuse white, let alone the paper substrate. Thus, the highlights must be considerably compressed in the reproduction.

On the other hand, slides or movies projected in a darkened room do not suffer from the same limitation. In the absence of dominant external references, the observer's state of adaptation is controlled by the bright image on the screen. Thus, these media are designed to reproduce diffuse white at a lower luminance than the maximum attainable, which leaves some headroom for the reproduction of specular highlights and other very bright tones. To the adapted observer, these tones actually have the appearance of being brighter than 100% diffuse white; they sparkle and shine with a more realistic intensity than is possible for a print viewed under normal conditions. Thus, their representation in the PCS would require an apparent luminance greater than that of the white reference ($Y > 1$, or $L^* > 100$). The same illusion is possible with back-lit transparencies and video, as long as the viewing environment is sufficiently dim that the observer is adapted primarily to the image, rather than the surround.

Of course, there are limits to the apparent brightness that can be simulated by these media, but they are far higher than those of reflection prints in a normal surround - perhaps 200%, as compared with 90%, relative to diffuse white. The practical consequence of this difference is that the tonal compression of highlights is much less severe in the case of movies, slides, and video, than in the case of typical prints on paper.

All real media have a limit at the dark end of the tone scale, so that tonal compression is required in the shadows as well. Furthermore, the level of flare in the intended viewing environment has a strong effect on the apparent tone scale, particularly in the shadows and three-quarter tones; media designed for viewing conditions with different levels of flare tend to incorporate different amounts of flare compensation in their tone reproduction.

PCS colorimetry must also be corrected to account for the change in color appearance caused by differences in the absolute luminance level. For example, the 500 lux illuminance of the reference viewing environment is specified to be typical of actual home and office viewing environments. Corrections will typically be needed to correct for the darker, less colorful appearance of reproductions when they are

© 2003 ICC

viewed at lower levels of illumination, or the lighter, more colorful appearance when they are viewed at higher levels of illumination.

In photographic systems, the tone-reproduction characteristics are implemented in the construction of the sensitized layers and the chemistry of the emulsions and developers, or in the case of digital photography, in the image processing. In video, they are implemented in the electronics of the camera and receiver. Thus, a color management system usually deals with an image originating from a medium or device that has already imposed its own tone characteristic on the luminances captured from a scene, so that the highlights and shadows are already compressed. However, it is often necessary to reproduce the image on a different medium, for which the original compression may be less than ideal. In such cases, for best results, the tone scale of the image should be adjusted for the output medium.

D.2.7.6 The Reference Medium and Tonal Compression

The PCS and its reference medium provide a convenient interface for the tone-scale adjustments just discussed. Input transforms apply adjustments to map the tone scale of the original medium onto that of the reference; output transforms incorporate adjustments to map the tone scale of the reference medium onto that of the output medium.

These adjustments can take on many different forms, depending on the aesthetic effect to be achieved. In some cases, the appearance of the original may be accurately preserved; in others, it may be preferable to make deliberate alterations in the appearance, in order to optimize the rendering for the output medium or to simulate a third medium. This range of possibilities is implicit in the phrase "desired color appearance" in the PCS definition for the perceptual intent.

Output to media with a dynamic range different from that of the reference medium may be handled by tone-shaping techniques which compress or expand the tone scale to the range the device can handle. Furthermore, in output profiles, the different "rendering intents" can incorporate different adjustments. Some perceptual transforms, for example, can be designed to preserve the tone scale of the reference medium, clipping abruptly at the minimum reflectance if necessary, while other perceptual transforms may apply a more subtle reshaping of the highlight and shadow tones.

Input from media with a dynamic range different from the reference medium also may have tone-shaping techniques applied, along with luminance scaling to maintain brightness balance. These adjustments should be invertible (in the sense that they match the precision of the data and the computation) for high-quality output to the same devices. For instance, images with an extended highlight range (such as those from scanned photographic transparencies) must be remapped for the reference medium, so that the highlights will be compressed to the range of the PCS.

The details of these techniques may vary with the intended market, the specified "rendering intent", and aesthetic choices made by the profile builder. If the intent is to preserve the appearance of the original, adjustments to the tone scale can be limited to those compensating for differences between the actual viewing conditions and those of the reference environment. These include the effects of brightness adaptation, surround adaptation, and viewing flare. In other cases, there is plenty of latitude for profile vendors to differentiate their products with respect to aesthetic choices, while still basing their profile transforms on the common definition of the PCS. Thus, proprietary art can be fostered and encouraged in a context of interoperability.

D.2.7.7 Procedural Summary

The various colorimetric adjustments discussed above can be organized into a computational procedure for calculating PCS coordinates for device-profile transforms. The procedure presented here is applicable to reflection media input and output profiles; monitor transforms are typically computed in the simplified

© 2003 ICC

manner described below, although it is certainly possible to treat monitors in the same way as other input and output devices in order to achieve more accurate image display.

The procedure is given in the device-to-PCS direction for the perceptual rendering intent (AToB0Tag) transform. This procedure is intended as a conceptual guide, recognizing that artistic preferences may result in significant variations on this procedure.

1. Obtain CIE 1931 XYZ tristimulus values for a set of color patches on the device or media to be profiled. More information about measurement procedures is provided in section D.2.3. There should be at least one measurement of the "media white." Additionally, it is necessary to obtain the colorimetry of the adaptive white point.[1)] Apply steps 2 and 3 from section D.2.6.2.

2. If the chromaticity of the adaptive white point is different from that of D50, convert colorimetry from the device adaptive white point chromaticity to the PCS white point chromaticity using an appropriate chromatic adaptation transform. This may be done by applying one of the transformations mentioned in D.2.4.

3. Other corrections must be applied to the data to account for any differences in viewing conditions between the actual environment and the reference environment. These include, but are not limited to, tonal adjustments for differences in viewing flare, general brightness adaptation, and surround effects.[2)]

4. Convert the corrected colorimetry to "desired" colors for the reference medium. The medium white and black points are mapped to the reference medium white and black points. In general, there is considerable freedom in this step, depending on aesthetic considerations. Optimal renderings will frequently require adjustments to the tone scale and color reproduction, especially when there is a significant difference in dynamic range or color gamut between the actual medium and the reference medium.

5. Scale the reference-medium CIE XYZ coordinates to PCS values, so that the reference medium white point maps to the PCS white point. This scaling is conceptually equivalent to transforming the "desired reference medium image" to media-relative PCS using a media-relative colorimetric intent mapping and equations D.2.6.1.0-4 through D.2.6.1.0-6 in section D.2.6.1.

6. Optionally, convert the PCS XYZ coordinates to PCS L*a*b* as described in Annex A.

7. Encode the PCS XYZ coordinates or the PCS L*a*b* coordinates digitally in 8-bit or 16-bit representations, as defined in Annex A.

D.2.8 Monitor Display

Some special considerations apply to monitor profiles. Since a CRT monitor is a self-luminous display, the interpretation of tone is somewhat ambiguous: Should full-drive monitor white be regarded as 100% diffuse white? In terms of color appearance, the answer to that question depends on the state of the observer's adaptation, which is influenced by the viewing environment. For example, in a brightly-lit office environment, the observer may adapt to the ambient illumination. In a dim environment, the observer may adapt to the monitor screen itself. In general, it is very difficult to predict the observer's actual state of adaptation.

However, for desktop applications the document editor or graphic artist typically has an expectation that monitor white will be associated with the blank paper (or other substrate) of the output medium, regardless of his or her actual state of adaptation. Thus, for practical reasons, it is important that the monitor profiles

---

1) E. J. Giorgianni and T. E. Madden, op. cit., p. 356.

2) E. J. Giorgianni and T. E. Madden, op. cit., p. 474.

© 2003 ICC

be designed to display paper white at full-drive monitor white (R = G = B = 255 on a typical 24-bit display). Similarly, there is an expectation that R = G = B = 0 corresponds to "black" and will be reproduced by the minimum reflectance of the output medium. These user expectations are based on common practice and convenience and lie outside of strict colorimetry and color-appearance considerations.

Furthermore, the monitor profile transforms that are common on many systems are based on oversimplified mathematical models. Often they take the form of a linear transformation from XYZ to RGB (a 3 x 3 matrix) followed by a simple power law in each channel for gamma correction. Such transforms often fail to model the behavior of the monitor accurately in the shadows, since they ignore the biases that commonly occur in the CRT and support electronics. These biases are variable from unit to unit and are also dependent on the user-selectable settings of contrast and brightness. Fortunately, any departures from colorimetric accuracy that result from these simple models are relatively minor and are partially masked by face-plate reflections, often 3 to 5 percent, so that they are generally tolerated.

These simple monitor profiles will satisfy typical user expectations if monitor white is mapped to the XYZ values of the PCS white point and monitor black is mapped to the PCS black point. This means that monitor white maps to reference medium white and monitor black maps to reference medium black. When processed through a typical monitor profile transform, therefore, reference medium white will be displayed at monitor white and reference medium black will be displayed at monitor black. This provides a practical mapping between the monitor and the reference medium while permitting the use of simple monitor transforms to satisfy common user expectations.

D.2.9 Bibliography

1. CIE Publication 15.2-1986, Colorimetry (second edition).

3. ISO 13655:1996, Graphic technology - Spectral measurement and colorimetric computation for graphic arts images.

4. R. W. G. Hunt, The Reproduction of Colour, Fourth Edition, Fountain Press, 1987.

5. E. J. Giorgianni and T. E. Madden, Digital Color Management, Addison-Wesley, Reading, Massachusetts, 1998.

6. Mark D. Fairchild, Color Appearance Models, Addison-Wesley, Reading, Massachusetts, 1998.

© 2003 ICC

Annex E
Chromatic Adaptation Tag

This section describes the derivation and use of the Chromatic Adaptation Tag in more detail. The first part recommends a chromatic adaptation transform (CAT) for general use. The second part provides a mathematical description of this recommended CAT. The last part provides basic guidelines and instructions for possible use of the Chromatic Adaptation Tag.

The Chromatic Adaptation Tag is required when the actual illumination source has a chromaticity different from that of CIE illuminant D50. Profiling applications can omit this tag or store an identity matrix in this tag when the actual illumination source has a chromaticity identical to D50.

E.1 Calculating the Chromatic Adaptation Matrix

The ICC profile format specification allows the use of different linear (matrix-based) CATs. This flexibility allows profile creators to select the most appropriate CAT for their applications. Criteria for selection include visual performance, the gamut of the image as transformed to the D50 PCS, and other considerations. However, the use of different CATs will produce different results, which may be undesirable. Therefore, it is recommended that the linear Bradford CAT (which is the same as the linearized CIECAM97s transformation) be used when there is no reason to use a different CAT. The linear Bradford CAT has been widely implemented in the digital imaging industry, with demonstrated excellent visual performance. If a profile creator decides to use a CAT other than linear Bradford, they should do so only to address specific known issues, recognizing that the resulting profile will most likely produce different results than profiles from other sources.

E.2 Linearized Bradford/CIECAM97s Transformation

When full adaptation is assumed and a negligible non-linearity in the blue channel is omitted, the Bradford transformation is identical to the CIECAM97s transformation. Under the above assumption both become the same variant of a cone-space transform. Similarly as in von Kries method, the cone response values can be found through the matrix equation:

$$\begin{bmatrix} \rho \\ \gamma \\ \beta \end{bmatrix} = \begin{bmatrix} 0,8951 & 0,2664 & -(0,1614) \\ -(0,7502) & 1,7135 & 0,0367 \\ 0,0389 & -(0,0685) & 1,0296 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = M_{BFD} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \qquad \text{E.2.0.0.0-1}$$

The calculation of corresponding (visually equivalent) CIE XYZ values between two white points is the same as in von Kries transformation and similarly, the following chromatic adaptation matrix can be derived:

$$M_{adapt} = M_{BFD}^{-1} \begin{bmatrix} \rho_{pcs}/\rho_{src} & 0 & 0 \\ 0 & \Upsilon_{pcs}/\Upsilon_{src} & 0 \\ 0 & 0 & \beta_{pcs}/\beta_{src} \end{bmatrix} M_{BFD} \qquad \text{E.2.0.0.0-2}$$

Where:

© 2003 ICC

$$\begin{bmatrix} \rho_{src} \\ \gamma_{src} \\ \beta_{src} \end{bmatrix} = M_{BFD} \begin{bmatrix} X_{WPsrc} \\ Y_{WPsrc} \\ Z_{WPsrc} \end{bmatrix} \qquad \text{E.2.0.0.0-3}$$

$$\begin{bmatrix} \rho_{pcs} \\ \gamma_{pcs} \\ \beta_{pcs} \end{bmatrix} = M_{BFD} \begin{bmatrix} X_{WPpcs} \\ Y_{WPpcs} \\ Z_{WPpcs} \end{bmatrix} \qquad \text{E.2.0.0.0-4}$$

XY_{WPpcs} represents the illuminant of the reference viewing condition.

E.2.1 Applying the Chromatic Adaptation Matrix

The application of the chromaticAdaptationTag is under active study by the ICC. The chromaticAdaptationTag may not apply to the perceptual intent (see D.2.7.3). The user may look at the set of profiles to determine what adjustments can be made. There are several possibilities:

1. No profile has the chromaticAdaptationTag. No action can be taken.

2. All profiles have the chromaticAdaptationTag. If the same method is used, no action should be taken. If different methods are used, the user may choose to undo them first before using a consistent method of their choice.

3. Only one profile has chromaticAdaptationTag. Processing is implementation dependent.

Here is a step by step example of how to do the adjustments if the color transformation is created from two RGB Display profiles containing the chromaticAdaptationTag.

Step 1. Determine if the two methods are the same. If the two matrices are identical, the chromatic adaptation methods are the same. If the matrices are different, the methods could still be the same while the actual viewing illuminants are different. One easy way to test this is: if M1 and M2 represent the chromatic adaptation matrices from profile 1 and 2 respectively, it can be proven that chromatic adaptation algorithms are the same if the following matrix equation holds true: M1 * M2 == M2 * M1. We can stop here if two algorithms are the same.

Step 2. Determine the actual device viewing illuminant for profile 1. This can be achieved by applying the inverse chromatic adaptation matrix to the PCS D50 XYZ value.

Step 3. Invert the red, green, and blue values stored in the colorant tags to the actual device illuminant values. This is accomplished by applying the inverse of the chromatic adaptation matrix for each colorant.

Step 4. Calculate the new chromatic adaptation matrix. Follow the examples of E.1. Use your favorite cone response matrix and generate a new matrix.

Step 5. Generate new D50 relative colorant values for red, green, and blue by applying the matrix calculated in step 4 to colorant values in the device illuminant derived in step 3.

Step 6. Repeat steps 2 to 5 for profile 2.

© 2003 ICC

For profiles with LUT tags, the adjustments can be made after the values are converted into the PCS by adding an extra processing step of undoing and redoing the chromatic adaptation.

© 2003 ICC

Annex F
Summary of spec changes

This annex lists the major changes made to the specification for the past few spec revisions. Minor editorial and cosmetic changes are not listed.

These are the changes made from Revision ICC.1:2001-12

1. Addition to allow the use of multidimensional tables for N-component LUT-based display profiles (Section 6.3.2.3).

   Per: Online ballot #200201
2. Change to the copyright notice, per NPES lawyer recommendation.

   Per: Online ballot #200203
3. Change to recommended CAT (Annex E, E.1, E.2, E.2.1)

   Per: Online ballot #200204
4. Clarification of Section 6.1.11, field type definition.

   Per: Online ballot #200204
5. Rationalizing requirement statement in Section 6.5.3 with references to this section in Section 6.3.3.2, Section 6.3.4.1, and Section 6.4.14.

   Per: Editor judgement
6. Clarification of wording for clauses 5.3.10, 6.4.25, 6.5.25, and Annex A regarding Y range and viewingConditionsTag.

   Per: Online ballot #200111
7. Addition of informative note recommending content for tag originally included in ballot but not added into text of spec for clause 6.4.10:“charTargetTag”.

   Per: Online ballot #200019
8. Fixed Table 62 equations for consistency.

   Per: Editor judgement
9. Added informative note to clause 6.5.10:“lutBtoAType” clarifying use of Matrix element, consistent with online ballot 200210.

   Per: Editor judgement
10. Removal of unused tags: crdInfoTag, deviceSettingsTag, ps2CRD0Tag, ps2CRD1Tag, ps2CRD2Tag, ps2CRD3Tag, ps2CSATag, ps2RenderingIntentTag, screeningDescTag, screeningTag, ucrbgTag, and associated types as applicable. Many changes in Annex C as there are now no longer any PostScript Level 2 tags.

    Per: Online ballot #200209
11. Addition of informative text explaining the preview tags preview0Tag, preview1Tag, and preview2Tag.

    Per: Online ballot #200206
12. Changes to allow use of Matrix and M-curves of an mAB-type A2B0 tag in a DeviceLink profile.

    Per: Online ballot #200210
13. Clarification of clause 6.1.13:“Profile ID”.

    Per: Editor judgement
14. Clarification of why Y max is 1 rather than 100 per CIE in Annex A.2.

    Per: Editor judgement
15. Clarification of relationship between Chromatic Adaptation and Perceptual Rendering Intent.

    Per: Online ballot #200303

© 2003 ICC

16. Addition of Annex C material clarifying relationship between ICC profiles and PostScript CSAs and CRDs.
    Per: Editor judgement
17. Updated Adobe PostScript Language Reference to latest Third Edition.
    Per: Editor judgement
18. Clarification of Profile ID section 6.1.13.
    Per: Editor judgement
19. Removal of obsolete BG, GCR, UCR references in section 5.2.
    Per: Editor judgement
20. Replacement of ProfileVersion to 4.1.0 in section 6.1.3.
    Per: Editor judgement
21. Replacement of output profile: TRC/Matrix with "colorimetric" because of grey printer profiles in Table 20.
    Per: Editor judgement
22. Removal of obsolete definition of ViewingCondDescTag in Table 33.
    Per: Editor judgement
23. Removal of obsolete text in Annex C.
    Per: Editor judgement
24. Removal of "7-bit ASCII" normative statement from copyrightTag description in Table 21 through Table 33.
    Per: Editor judgement
25. Change of Zr/Zn to Zr/Zi in Annex A.2.
    Per: Editor judgement

These are the changes made from Revision ICC.1:2001-04

1. Addition of the colorantOrderTag (Section 6.4.13) , colorantTableTag (Section 6.4.14), colorantOrderType (Section 6.5.2), and colorantTableType (Section 6.5.3).
   Per: Online ballot #200004
2. Replace Annex D, Annex D, with new annex.
   Per: Online ballot #200018
3. Add new lut types, lutAtoBType (Section 6.5.9) and lutBtoAType (Section 6.5.10).
   Per: Online ballot #200005
4. Clarification of colorant tags including changing the name from "Colorant" to "MatrixColumn". Affects 6.3.1.2, 6.3.2.2, 6.3.2.3, 6.4.4, 6.4.20, 6.4.33, and 6.3.
   Per: Online ballot #199907
5. Clarify the PCS CIELAB equations (see A.1).
   Per Online ballot #200011
6. Add new multiLocalizedUnicodeType and apply it to description tags. See 6.5.12, 6.4.15, 6.4.16, 6.4.17, 6.4.31, 6.4.43, 6.4.36, and 6.5.15.
   Per: Online ballot #200006
7. Add new Profile ID field to header.
   Per: Online ballot # 200008
8. Change the definition of the gray profile. See 6.4.19, 6.3.1.1, 6.3.2.1, and 6.3.3.1.
   Per: Online ballot #2000014

© 2003 ICC

9. Addition to Annex A on how to handle PCS encoding bounds.
   Per: Online ballot #2000012
10. Add wording to allow handling of data/time values to be consistent. Clause 5.3.1.
    Per: Online ballot # 200009
11. Addition of parametricCurveType (Section 6.5.14).
    Per: Online ballot #199909
12. Addition of new parametric function (type 4) to parametricCurveType (Section 6.5.14).
    Per: Online ballot #200108
13. Make chromaticAdaptationTag a required tag for all profile types except the DeviceLink and named profile types. This tag was added but not required in the last specification update.
    Per: Online ballot # 2000010
14. Modify annex equation numbering to reflect annex number.
    Per: Editor judgement
15. Clarification of illumination level definition for PCS viewing conditions (A.1)
    Per. Online ballot #200101
16. Extends use of the charTargetTag to include ability to use ICC Characterization Data Registry info. (Section 6.4.10)
    Per. Online ballot #200102
17. Define new PCS Lab value range. (Section 6.5.7, Section 6.5.13, Annex A, and Annex D)
    Per. Online ballot # 200103
18. Include fallback strategy for the use of transform tags. (Section 0.8)
    Per. Online ballot # 200105
19. Clarification of interpolation for degenerate LUTs. (Section 6.5.4, Section 6.5.9, Section 6.5.10, Section 6.5.7, and Section 6.5.8)
    Per. Online ballot #200107
20. Removal of member list from specification. Web site can maintain better accuracy.
    Per. Spec Editing WG
21. Convert numbers to ISO-compliant formats (decimal points become commas and commas become spaces)
    Per. ISO number representation definitions
22. Add appendix-specific numbers to equations in appendices.
    Per. ISO requirements
23. Delete namedColorTag, namedColorType, and textDescriptionType from specification.
    Per: ICC vote
24. Clarification of wording for Section 5.3.10, Section 6.4.25, Section 6.5.26, and Annex A.
    Per. Online ballot #200111
25. Correct LAB conversion equations in A.2.
    Per. Spec Editing WG

These are the changes made from Revision ICC.1:1998-09

1. The descriptions of the various rendering intents have been clarified. (See 4.11 and all of A.4.)
   Per: Online ballot #200015
2. Two new attribute bits have been added to the profile header. (See Table 17.)
   Per: Online ballot #199805
3. The interpretation of multidimensional tags is now defined in more situations. (See Table 20 and all of Section 6.3)
   Per: Online ballot #200016

© 2003 ICC

4. Multidimensional tags are now allowed in monochrome device profiles. (See 6.3.1.1, 6.3.2.1, and 6.3.3.1.)
   Per: Online ballot #200016
5. A new optional tag, chromaticAdaptationTag, has been added. (See 6.4.11 and Annex E.)
   Per: Online ballot #200010
6. A new optional tag, chromaticityTag, has been added. (See 6.4.12 and 6.5.1.)
   Per: Online ballot #199908
7. The description of gamutTag has been reworded to make clear that its input is PCS values. (See 6.4.18.)
   Per: Spec Editing WG
8. The descriptions of lut8Type and lut16Type have been expanded to explain how tables map to one another. (See 6.5.7, 6.5.8, and A.3.)
   Per: Online ballot #199806
9. The offset of the name prefix in namedColorType has been corrected to read "16..t".
   Per: Spec Editing WG
10. A rule was added to prevent the modification of certain text inside textDescriptionType strings.
11. The illumination level of the reference viewing condition is defined as 500 lux (previously it was undefined). (See A.1)
    Per: Online Ballot #200101
12. The term "relative colorimetry" has been changed to "media-relative colorimetry" and the term "absolute colorimetry" has been changed to "ICC-absolute colorimetry" to avoid confusion with CIE terminology. (See A.4.1.2 and A.4.1.3.)
    Per: Online ballot #200015
13. The C header file example has been removed from the specification. It can be found online at the ICC Web site.
    Per: Spec Editing WG
14. The formatting has been altered to more closely match the format of ISO and IEC standards. (NOTE This specification is not an International Standard, and it does not meet all of the ISO/IEC drafting rules.)
    Per: Spec Editing WG

These are the changes made from Version 3.4 (August 1997):

1. The Normative References have been brought up to date and expanded to include all cited International Standards. (See clause 2.)
   Per: Spec Editing WG
2. The CMM Type and Primary Platform signatures are now allowed to be set to zero. (See 6.1.2 and 6.1.7.)
   Per: Resolution voted 1998-03-15
3. The profile version number in the header has been changed to 2.2.0. (See 6.1.3.)
   Per: Resolution voted 1998-07-24
4. The terms “low *n* bits” and “first *n* bits” have been changed to “least-significant *n* bits” to avoid confusion. (See 6.1.8, 6.1.10, 6.1.11, 6.2.2, and 6.5.)
   Per: Spec Editing WG
5. A tag can now only appear once in a profile. (See 6.2.)
   Per: Resolution voted 1998-03-15
6. The table describing the interpretation of context-dependent tags has been expanded to explicitly list the contexts where the interpretation is undefined. (See Table 20.)
   Per: Resolution voted 1998-07-24

© 2003 ICC

7. The rules concerning which classes of profiles can and cannot be embedded in images have been made consistent. (See 6.3.4.2, 6.3.4.3 and B.)
   Per: Resolution voted 1998-03-15
8. A new optional tag, deviceSettingsTag, has been added. (See 6.4.19 and 6.5.8)
   Per: Online ballot #199706
9. A new optional tag, outputResponseTag, has been added. (See 6.4.27 and 6.5.16.) The new basic numeric type response16Number has been added to support the responseCurveSet16Type. (See 5.3.2.)
   Per: Online ballot #199801
10. The possibility for alignment problems in crdInfoType has been pointed out. (See 6.5.4.)
    Per: Spec Editing WG
11. All curveType tags now must follow the same interpretation rules for zero-entry and one-entry tables. (See 6.5.4.)
    Per: Resolution voted 1998-03-15
12. The method for embedding profiles in GIF images has been added. (See B.5.)
    Per: Online ballot #199704
13. The C header file has been updated to reflect the addition of new tags.
    Per: Spec Editing WG

These are the changes made from Version 3.3 (November 1996):

1. The Definitions clause has been updated. (See clause 4.)
   Per: Spec Editing WG
2. A new clause has been added for symbols and abbreviations used in the Specification. Abbreviations that were previously under Definitions have been moved to the new clause. (See 5.2.)
   Per: Spec Editing WG
3. The dataType and textType descriptions now spell out how the data size is calculated. (See 6.5.5 and 6.5.19.)
   Per: Spec Editing WG
4. The method for embedding profiles in JFIF images has been added. (See B.4.)
   Per: Online ballot #199701
5. The C header file has been updated. The conditional compilation has been altered to provide a default definition of the data types in all circumstances. The typedef for `icCrdInfoType` has been altered, and comments have been added to explain its use.
   Per: Spec Editing WG

These are the changes made from Version 3.2 (November 1995):

1. The requirements for Input Profiles have changed. Instead of having categories for RGB and CMYK input devices, the requirements have been changed to cover “three-component matrix-based” profiles and “N-component LUT-based” profiles. (See 6.3.1.2 and 6.3.1.3.)
   Per: Online ballot with results reported on 1996-07-15
2. The list of color space signatures has expanded to include generic color spaces (2colorData to 15colorData). (See Table 13.)
   Per: Resolution voted 1996-07-15
3. The profile version number in the header has been changed to 2.1.0. (See 6.1.3.)
   Per: Included in "NCLR" (generic color space) resolution
4. A new optional tag, crdInfoTag, has been added. (See 6.4.16, 6.5.4, and C.1.)
   Per: Resolution voted 1996-03-28

© 2003 ICC

5. The signature of namedColor2Type was incorrectly listed as ‘ncol’ in Version 3.2 of the spec. The correct signature is ‘ncl2’. (See Table 58.)
   Per: Spec Editing WG
6. The description of the PCS encodings has been rewritten to clarify some issues. The actual encodings have not changed. (See A.1.)
   Per: Resolution voted 1996-07-15
7. The possibility for alignment problems in textDescriptionType and ucrbgType has been pointed out. (See 6.5.24.)
   Per: Resolution voted 1996-07-15
8. The examples for embedding profiles in EPS files have been corrected to show the required colon (:) after `%%BeginSetColorSpace` and `%%BeginRenderingIntent`. (See B.2.)
   Per: Spec Editing WG

© 2003 ICC