

INTERNET-DRAFT
Fax Working Group
March 1st, 2001
[draft-ietf-fax-tiff-fx-extension1-01.txt](#)

L. McIntyre
Xerox Corporation
D. Abercrombie
Xerox Corporation
W. Rucklidge
Intelligent Markets
R. Buckley
Xerox Corporation
March 2001

TIFF-FX Extension Set 1

Status of this Memo:

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at
<<http://www.ietf.org/ietf/1id-abstracts.txt>>

The list of Internet-Draft Shadow Directories can be accessed at
<<http://www.ietf.org/shadow.html>>.

A version of this draft document is intended for submission to the RFC editor as a Proposed Standard for the Internet Community. Discussion and suggestions for improvement are requested.

Copyright Notice

Copyright (C) The Internet Society 2001. All Rights Reserved.

Abstract

This document is an Internet draft for extensions to TIFF-FX [RFC XXXX], extension set 1.

[[[RFC XXXX is a placeholder for the pending Draft Standard revision to [RFC 2301](#), currently documented in [draft-ietf-fax-tiff-fx-09.txt](#).]]]

This draft describes extensions to RFC XXXX to enable new features or conditions to TIFF-FX. The features are described by a set of guidelines for all TIFF-FX extensions, and a set of 5 extension types which enable: increased set of resolutions and image widths, expanding Profile M from 3 layers to N layers, the use of shared data as a general mechanism for sharing data across images and pages, a binary profile for JBIG2 coding, and an extension to Profile M for JBIG2 and "colour tag" coding. These extensions do not required modification of existing TIFF-FX implementations.

The IETF has been notified of intellectual property rights claimed in regard to some or all of the specification contained in this document. For more information consult the online list of claimed property rights in [<http://www.ietf.org/ipr.html>](http://www.ietf.org/ipr.html).

The revisions to draft 00 are summarized in the list attached as Annex A to this document.

Table of Contents

E1.1 INTRODUCTION	4
E1.1.1 Scope.....	4
E1.1.2. Overview of this draft	5
E1.2. TIFF Fields Required and Recommendations for All TIFF-FX Extensions.....	6
E1.2.1. Required Fields.....	6
E1.2.1.1. GlobalParametersIFD Field.....	6
E1.2.1.2. TIFF-FXExtensions Field.....	6
E1.2.2. Recommended Fields.....	8
E1.2.2.1. FaxProfile Field.....	8
E1.2.2.2. MultiProfiles Field.....	8
E1.3. TIFF-FX Extension 1: Resolution and ImageWidth Extensions.....	10
E1.4. TIFF-FX Extension 2: N-Layer Profile M Extension.....	14
E1.4.1. Introduction.....	14
E1.4.2. Section 8.1 . Overview.....	15
E1.4.3. Section 8.1.1 . MRC 3-layer model.....	15
E1.4.4. Section 8.1.2 . A TIFF Representation for the MRC 3-layer model.....	16
E1.4.5. Section 8.2.1 . Baseline Fields.....	18
E1.4.6. Section 8.2.2 . Extension Fields.....	19
E1.4.7. Section 8.2.3 . New Fields.....	19
E1.4.8. Section 8.4 . Rules and Requirements for Images.....	21
E1.4.9. Section 8.5 . Profile M - MRC Fax Profile Summary.....	22
E1.5. TIFF-FX Extension 3: Shared Data Extension.....	22
E1.5.1. Overview.....	23
E1.5.2. Required TIFF Fields.....	23
E1.5.2.1. Baseline Fields.....	23
E1.5.2.2. Extension Fields.....	23
E1.5.2.3. New Fields.....	23
E1.5.3. Recommended Fields.....	23
E1.5.4. Shared Data.....	23
E1.5.4.1. SharedDataList.....	26
E1.5.4.2. Representation of Shared Data in TIFF.....	27
E1.6. TIFF-FX Extension 4: Profile T - Lossy and Lossless JBIG2 Black-and-White Fax profile.....	28
E1.6.1. Overview.....	28
E1.6.2. Required TIFF Fields.....	30
E1.6.2.1. Baseline Fields.....	31
E1.6.2.2. Extension Fields.....	33
E1.6.2.3. New Fields.....	33
E1.6.3. Recommended TIFF Fields.....	33
E1.6.4. JBIG2 Shared Data.....	36
E1.6.4.1. The JBIG2 Shared Data.....	36
E1.6.4.2. JBIG2 SharedDataMemory	38
E1.6.5. The JBIG2 Image Strip.....	39
E1.6.5.1. Image Strip Format.....	41

E1.6.7. Rules and Requirements for Images-----	45
E1.6.8. Profile T - Lossy and Lossless JBIG2 Black-and-White Fax profile Summary-----	46
E1.7. TIFF-FX Extension 5: JBIG2 Color Extension of Profile M-----	48
E1.7.1. Overview-----	48
E1.7.2. Required TIFF Fields-----	49
E1.7.2.1. Baseline Fields-----	49
E1.7.2.2. Extension Fields-----	49
E1.7.2.3. New Fields-----	49
E1.7.3. Recommended TIFF Fields-----	49
E1.7.4. JBIG2 Shared Data-----	50
E1.7.5. The Image Strip-----	50
E1.7.6. Representation of JBIG2 data in TIFF-----	50
E1.7.7. Colour Tag (Color Symbol Compression)-----	50
E1.7.7.1. Why use Colour Tag Compression-----	50
E1.7.7.2. Colour Tag Terms of Use in TIFF-----	50
E1.7.8. Rules and Requirements for Images-----	52
E1.7.9. JBIG2 Extension of Profile M Summary-----	53
E1.8. Security Considerations-----	56
E1.9. References-----	56
E1.10. Authors' Addresses-----	57
Full Copyright Statement-----	57
Annex A. List of edits to draft-ietf-fax-tiff-fx-Extension1-00 -----	59

E1.1. Introduction

This document describes extensions to RFC XXXX [RFCXXXX], titled "File Format for Internet Fax", also known as TIFF-FX, to augment the features and capabilities for the data content and structure generated by the current suite of ITU-T Recommendations for Group 3 facsimile. As an RFC XXXX extension, this specification should be read in conjunction with RFC XXXX.

[[RFC XXXX is a placeholder for the pending Draft Standard revision to [RFC 2301](#), currently documented in [draft-ietf-fax-tiff-fx-09.txt](#).]]

These Recommendations and the TIFF fields described here support the following new facsimile profile:

Profile T: TIFF-FX Extension 4: Black-and-White JBIG2 Extension - a JBIG2 profile for binary only T.88|ISO/IEC 14492 [[T.88](#)] coded data, built upon Profile M and the Shared Data extension (TIFF-FX Extension 5).

and create new extensions for the following new features:

- TIFF-FX Extension 1: Resolution and ImageWidth Extensions - extends the resolutions and image widths available for all Profiles, with the exception of Profile S
- TIFF-FX Extension 2: N-Layer Profile M Extension - extends the 3-layer model to N layers
- TIFF-FX Extension 3: Shared Data Extension - enables data to be shared among different images and pages; an enabler for additional compression gains when using JBIG2 encoding
- TIFF-FX Extension 5: JBIG2 Color Extension of Profile M - the binary and color extension to Profile M which enables JBIG2 coding using T.88 (JBIG2, binary) and T.45 [[T.45](#)] "Run-length Colour Encoding", required for colour tag extensions to JBIG2.

This extension specification of TIFF-FX for facsimile is known as TIFF-FX Extension Set 1.

E1.1.1 Scope

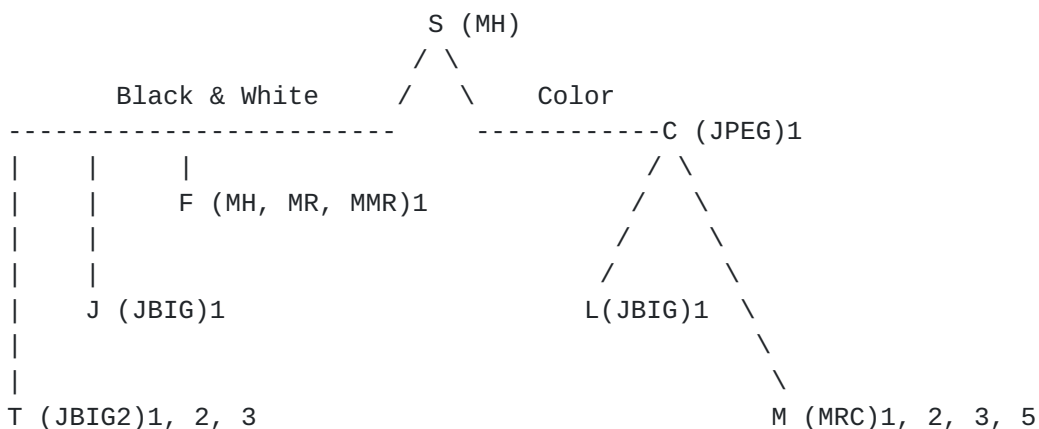
This document defines extensions to RFC XXXX, titled "File Format for Internet Fax", known as TIFF-FX. These extensions add new functionality to the profiles of TIFF-FX, with the exception of Profile S, which is highly constrained. Most of these extensions can be independently used; although some extensions may rely on others.

Unless otherwise noted, the primary reference is [RFC XXXX] "File Format for Internet Fax" (TIFF-FX), which references the following as it's primary references: the current TIFF specification [TIFF], selected TIFF Technical Notes [TTN1, TTN2] and [T.30].

E1.1.2 Overview of this draft

This Section gives an overview of TIFF-FX Extension Set 1.0. Section E1.2 describes the requirements and recommendations associated with all TIFF-FX extensions defined within this document. Sections E1.3 through E1.7 describe the five new extensions. Section E1.3 defines the new resolutions and associated image widths. E1.4 defines the N-Layer extension to the 3-Layer MRC model. Sections E1.5 through E1.7 are all associated with making provisions for JBIG2 encoding. The ShardData structure of Section E1.5 accommodates a single representation of reusable data, such as JBIG2 symbols, that appear or are used multiple times within a file. The sharing or reusing of image components within and across pages is the fundamental difference between JBIG2 and other encodings. The new JBIG2 black-and-white profile, Profile T, is defined in Section E1.6, while the JBIG2 color extension to Profile M is defined in Section E1.7.

The following tree diagram shows the relationship among profiles, between profiles and coding methods, and between profiles and extensions. Profiles are represented by their single character designation (e.g. S, F and C), coding methods are represented by their multi-character acronym (e.g. MH, MR and JBIG2), and extensions are represented by their numeric designation (e.g. 1, 2 and 5).



This document is an extension to RFC XXXX and portions are specified by defining specific modifications to sections of RFC XXXX, as such it should be read in conjunction with RFC XXXX. Throughout this document, section number references without an "E1" prefix should be

interpreted as [RFC XXXX] section references.

McIntyre et al.

Expires September 2001

[Page 5]

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [\[REQ\]](#).

E1.2. TIFF Fields Required or Recommended for All TIFF-FX Extensions

E1.2.1. Required Fields

E1.2.1.1. GlobalParametersIFD Field

Status of the GlobalParametersIFD (GPIFD) is being changed from Recommended to Required, for all extensions. This change is based on the fact that more than one required extension, such as SharedData and TIFF-FXExtensions, have global file implications (i.e. apply across multiple pages or Primary IFDs), warranting their location within the GPIFD. Accommodation of these required fields within the GPIFD then requires change in status of the GPIFD to Required.

E1.2.1.1.1 Instructions

Remove the GlobalParametersIFD field from [Section 2.2.4](#) "New TIFF fields recommended for fax profiles" and append the following revised definition to Sections [4.2.3](#), [5.2.3](#), [6.2.3](#), [7.2.3](#) and [8.2.3](#) "New Fields":

GlobalParametersIFD(400) IFD or LONG

The GlobalParametersIFD, defined in [Section 2.2.4](#), is an IFD containing global parameters. It SHALL be present for all TIFF-FX extensions. This reflects a modification to the [Section 2.2.4](#) definition where GlobalParametersIFD is defined as a Recommended field. The RFC XXXX GlobalParametersIFD definition is further modified in that it is permitted to contain fields that are NOT permitted in any other IFD. The new SharedData field is one such field that is not permitted in any other IFD, see the Shared Data Extension below.

It is recommended that a TIFF writer place this field in the first IFD, where a TIFF reader would find it quickly. If a conflict exists between fields in the GlobalParametersIFD and in the image IFDs, then the data in the image IFD SHALL prevail.

E1.2.1.2. TIFF-FXExtensions Field

The TIFF-FXExtensions field is introduced to be an identification mechanism for all TIFF-FX extensions and SHALL be present when an extension is used. The extensions are identified by bit value assignment, accommodating use of more than one extensions at the same time. The TIFF-FXExtensions field SHALL be placed within the

GlobalParametersIFD.

McIntyre et al.

Expires September 2001

[Page 6]

E1.2.1.2.1 Instructions

Add the TIFF-FXExtensions field to Sections [4.2.3](#), [5.2.3](#), [6.2.3](#), 7.2.3 and 8.2.3 "New Fields", as defined below:

TIFF-FXExtensions(34687)

LONG

Count = 1

This field identifies the RFC XXXX extension(s) that apply. The field accommodates the need to continually enhance the functionality of TIFF-FX. This field SHALL only appear in the GlobalParametersIFD, it is NOT permitted in any other IFD. It is required and SHALL be present with use of any TIFF-FX extension.

A value of 1 in a bit location indicates the corresponding TIFF-FX Extension is used. More than one bit set to 1 means more than one TIFF-FX extension is used.

Current TIFF-FX Extensions:

Bit number	Extension number	Meaning
0	1	Resolution and ImageWidth Extensions If bit 0 is 1, then any of the X and YResolutions from 300 x 600 to 1200 x 1200 pixels per resolution unit, and the corresponding ImageWidths may be used.
1	2	N-Layer Profile M Extension If bit 1 is 1, then the provision for more than three MRC layers is used.
2	3	Shared Data Extension If bit 2 is 1, then Shared Data provisions is used.
3	4	Profile T - Black-and-White JBIG2 Extension If bit 3 is 1, then Black-and-White JBIG2 coding is used (i.e. Compression = 34715, TIFF-FXExtensions bit 2 is frequently set and bi-level constrains to the MRC model are applied).


```

+-----+-----+-----+
| 4 | 5 | JBIG2 Extension of Profile M |
|   |   | If bit 4 is 1, JBIG2 is used for Mask layer coding |
|   |   | and colour tags may be used for foreground layers |
|   |   | (i.e. Compression = 34715 and TIFF-FXExtensions |
|   |   | bit 2 is frequently set). |
+-----+-----+-----+
Default = 0, no extensions being used.

```

E1.2.2. Recommended Fields

E1.2.2.1. FaxProfile Field

The FaxProfile field is revised for two reasons: 1) acknowledge the new MultiProfiles field, specified below, and make provisions for association of the two fields. 2) update the current profile list to include the new Profile T, specified later in this document.

E1.2.2.1.1 Instructions

Replace the existing FaxProfile field in [Section 2.2.4](#) "New TIFF fields recommended for fax profiles" with the following:

FaxProfile(402) = 0 - 7, 255. BYTE

The profile that applies to this file; a profile is a subset of the full set of permitted fields and field values of TIFF-FX and it's extensions.

This field is used to indicate the profile used within the file when only one profile is used. The MultiProfiles field is used when a TIFF-FX extension or more than one profile is used within the file. A FaxProfile value of 255 (X'FF') indicates that the MultiProfiles field is used.

The currently defined FaxProfile values associated with a profile are:

- 0: does not conform to a profile defined for TIFF for facsimile
- 1: minimal black & white lossless, Profile S
- 2: extended black & white lossless, Profile F
- 3: lossless JBIG black & white, Profile J
- 4: lossy color and grayscale, Profile C
- 5: lossless color and grayscale, Profile L
- 6: Mixed Raster Content, Profile M
- 7: lossy and lossless JBIG2 black & white, Profile T
- 255: more than one profile and/or an extension is used. See MultiProfiles field, which indicates the profiles and/or profile(s) plus extension(s) used in the file

E1.2.2.2. MultiProfiles Field

This field is used to indicate when extension(s) or two or

more different profiles are used within a single file. RFC XXXX makes no statement with regard to the appropriateness of using encodings of two or more different profiles within a file. There is no question as to the value of such a provision. This is illustrated by an example where the first ten black-and-white text pages of a fifteen page document are processed using Profile F MMR (G4) encoding while the last five color pages are processed using Profile C JPEG encoding. The existing FaxProfile field is used within the file to signal one encoding profile. In response to requests received from implementors, this extension introduces the MultiProfiles field, to be used when an extended profile or more than one profile is used within a file.

Files containing extension(s) or more than one profile SHOULD contain the MultiProfiles field, identifying the profiles or extension(s) present. When present, the MultiProfiles field SHALL reside within the GlobalParameterIFD. The number of profiles present within a file is ambiguous without the MultiProfiles or FaxProfile field. It is highly likely that readers of files without the MultiProfiles or FaxProfile will assume that only one profile is present, the profile that is consistent with the Compression type and other relevant values that are present within the first IFD. Such readers may experience difficulty if different Compression types and/or other relevant parameters are encountered in subsequent IFDs within the file.

E1.2.2.2.1 Instructions

- a) Append the new MultiProfiles field, specified below, to [Section 2.2.4](#) "New TIFF fields recommended for fax profiles" following the ModeNumber field:

MultiProfiles(34689) LONG
Count = N

The extended profile (i.e. profile plus extension) or more than one profiles that apply to this file; a profile is a subset of the full set of permitted fields and field values of TIFF for facsimile. This field is used when an extended profile or more than one profile are used within the file. This field SHALL only be present when the FaxProfile field is absent or has a value of 255 (X'FF'). A value of 1 in more than one bit location indicates the corresponding profiles or profile(s) plus extension(s) that are used. The currently defined bits are:

MultiProfiles[0]:

- Bit 0: minimal black & white lossless, Profile S
- Bit 1: extended black & white lossless, Profile F
- Bit 2: lossless JBIG black & white, Profile J

Bit 3: lossy color and grayscale, Profile C
Bit 4: lossless color and grayscale, Profile L
Bit 5: Mixed Raster Content, Profile M
Bit 6: lossy and lossless JBIG2 black & white, Profile T
Bit 7: Extension 1, resolution and imagewidth extensions
Bit 8: Extension 2, N-Layer Profile M extension
Bit 9: Extension 3, shared data extension
Bit 10: Extension 5, JBIG2 extension of Profile M
Bits 11-31: reserved for future use.

WARNING: Files containing extensions or more than one profile "SHOULD" contain the MultiProfiles field, identifying the profiles present. See the description from the second paragraph of Section E1.2.2.2.

NOTE: count (N) > 1 should be used to accommodate future expansion beyond 32 bits.

- b) Append the new 4.5.7 "Multiple Profiles within a file" section, specified below, to [Section 4.5](#) "Implementation Warnings" following [Section 4.5.6](#):

4.5.7 Multiple Profiles within a file

Files containing more than one profile "SHOULD" contain the MultiProfiles field, identifying the profiles present. The number of profiles present within a file is ambiguous without the MultiProfiles or FaxProfile field. It is highly likely that readers of files without the MultiProfiles or FaxProfile fields will assume that only one profile is present, the profile that is consistent with the Compression type and other relevant values that are present within the first IFD. Such readers may experience difficulty if different Compression types and/or other relevant parameters are encountered in subsequent IFDs within the file.

E1.3. TIFF-FX Extension 1: Resolution and ImageWidth Extensions

The ITU-T has approved a new set of X and YResolutions, along with corresponding image widths (i.e. page widths). This extension makes provisions for using these extended resolutions.

The TIFF-FXExtensions field, below, SHALL be present when this extension is used.

TIFF-FXExtensions(34687) bit 0 is 1
See Section E1.2.1.2.

LONG

E1.3.1 Instructions

- a) Replace the 2nd sentence of [Section 2.2.2](#) "Additional TIFF fields required for all fax profiles" XResolution field with the following:

The ITU-T Recommendations for facsimile specify a small number of horizontal resolutions: 100, 200, 300, 400, 600, 1200 pixels per inch, and 80, 160 pixels per centimeter (or 204, 408 pixels per inch).

- b) Replace the 2nd sentence of [Section 2.2.2](#) "Additional TIFF fields required for all fax profiles" YResolution field with the following:

The ITU-T Recommendations for facsimile specify a small number of vertical resolutions: 100, 200, 300, 400, 600, 800, 1200 pixels per inch, and 38.5, 77, 154 pixels per centimeter (or 98, 196, 391 pixels per inch).

- c) Append the following five rows to the resolution table that follows the YResolution field in [Section 2.2.2](#) "Additional TIFF fields required for all fax profiles":

+-----+	+-----+	+-----+	+-----+
300	600	600	1200
+-----+	+-----+	+-----+	+-----+
400	800	1200	1200
+-----+	+-----+	+-----+	+-----+
600	1200	1200	1200
+-----+	+-----+	+-----+	+-----+
1200	1200	1200	1200
+-----+	+-----+	+-----+	+-----+

- d) Replace the ImageWidth field in [Section 4.2.1](#). "Baseline fields" with the following:

ImageWidth(256) SHORT or LONG

RequiredByTIFFBaseline

This profile supports the following fixed page widths: 1728, 2592, 3456, 5184, 10368 (corresponding to North American Letter and Legal, ISO A4 paper sizes), 2048, 3072, 4096, 6144, 12288 (corresponding to ISO B4 paper size), and 2432, 3648, 4864, 7296, 14592 (corresponding to ISO A3 paper size).

No default; must be specified

NOTE: Historical TIFF-F did not include support for the following widths related to higher resolutions: 2592, 3072, 3648, 3456, 4096, 4864, 5184, 6144, 7296, 10368, 12288 and 14592. Historical TIFF-F

documents also included the following values related to A5 and A6

McIntyre et al.

Expires September 2001

[Page 11]

widths: 816 and 1216. Per the most recent version of [T.4], A5 and A6 documents are no longer supported in Group 3 facsimile, so the related width values are now obsolete. See [section 4.5.2](#) for more information on inch/metric equivalencies and other implementation details.

e) Replace the XResolution field in [Section 4.2.1](#). "Baseline fields" with the following:

XResolution(282) = 200, 204, 300, 400, 408, 600, 1200 RATIONAL
RequiredByTIFFBaseline
The horizontal resolution of the image is expressed in pixels per resolution unit. In pixels/inch, the allowed values are: 200, 204, 300, 400, 408, 600 and 1200. See [Section 2.2.2](#) for inch-metric equivalency.
No default, must be specified

NOTE: The values of 200 and 408, 600 and 1200 have been added to the historical TIFF-F values, for consistency with [T.30]. Some existing TIFF-F implementations may also support values of 80 pixels/cm, which is equivalent to 204 pixels per inch. See [section 4.5.2](#) for information on implementation details.

f) Replace the YResolution field in [Section 4.2.1](#). "Baseline fields" with the following:

YResolution(283) = 98, 100, 196, 200, 300, 391, 400, RATIONAL
600, 800 and 1200
RequiredByTIFFBaseline
The vertical resolution of the image is expressed in pixels per resolution unit. In pixels/inch, the allowed values are: 98, 100, 196, 200, 300, 391, 400, 600, 800 and 1200 pixels/inch.
See [Section 2.2.2](#) for inch-metric equivalency.
No default, must be specified

NOTE: The values of 100, 200, 391, 600, 800 and 1200 have been added to the historical TIFF-F values, for consistency with [T.30]. Some existing TIFF-F implementations may also support values of 77 and 38.5 (cm), which are equivalent to 196 and 98 pixels per inch respectively. See [Section 4.5.2](#) for more information on implementation details.

NOTE: Not all combinations of XResolution, YResolution and ImageWidth are legal. The following table gives the legal combinations and corresponding paper size [T.30].

g) Replace the Resolution and ImageWidth table that follows YResolution in [Section 4.2.1](#). "Baseline fields" with the following:

XResolution x YResolution		ImageWidth		
200x100, 204x98				
200x200, 204x196		1728	2048	2432
204x391				
300 x 300, 300 x 600		2592	3072	3648
408 x 391, 400 x 400		3456	4096	4864
400x800				
600 x 600, 600 x 1200		5184	6144	7296
1200 x 1200		10368	12288	14592
Letter, A4		B4	A3	
Legal				
		Paper Size		

h) Replace the ImageWidth field in [Section 6.2.1](#). "Baseline Fields" with the following:

ImageWidth(256). SHORT or LONG

This profile supports the following fixed page widths: 864, 1024, 1216, 1728, 2048, 2432, 2592, 3072, 3456, 3648, 4096, 4864, 5184, 6144, 7296, 10368, 12288, 14592.

i) Replace the XResolution and YResolution fields in [Section 6.2.1](#). "Baseline Fields" with the following:

XResolution(282) = 100, 200, 300, 400, 600, 1200. RATIONAL

YResolution(283) = 100, 200, 300, 400, 600, 1200. RATIONAL

The resolution of the image is expressed in pixels per resolution unit. In pixels per inch, allowed XResolution values are: 100, 200, 300, 400, 600 and 1200. The base color fax profile requires the pixels to be square, hence YResolution SHALL equal XResolution. Base resolution is 200 pixels per inch and SHALL be supported by all implementations of this profile.

NOTE: The functional equivalence of inch-based and metric-based resolutions is maintained, per Annex E.6.5 in [T.4]. See table in Sec. 2.2.2.

NOTE: Not all combinations of XResolution, YResolution and ImageWidth are legal. The following table gives the legal combinations for inch-based resolutions and the corresponding paper sizes [T.30].

j) Replace the Resolution and ImageWidth table that follows YResolution

McIntyre et al.

Expires September 2001

[Page 13]

in [Section 6.2.1](#). "Baseline fields" with the following:

XResolution x YResolution		ImageWidth		
100 x 100		864	1024	1216
200 x 200		1728	2048	2432
300 x 300		2592	3072	3648
400 x 400		3456	4096	4864
600 x 600		5184	6144	7296
1200 x 1200		10368	12288	14592
		Letter,A4	B4	A3
		Legal		
		Paper Size		

k) Replace the XResolution and YResolution fields in [Section 7.2.1](#). "Baseline Fields" with the following:

XResolution(282) = 100, 200, 300, 400, 600, 1200. RATIONAL

YResolution(283) = 100, 200, 300, 400, 600, 1200. RATIONAL

The resolution of the image is expressed in pixels per resolution unit. In pixels per inch, allowed XResolution values are: 100, 200, 300, 400, 600 and 1200. The lossless color fax profile requires the pixels to be square, hence YResolution SHALL equal XResolution. Base resolution is 200 pixels per inch.

E1.4. TIFF-FX Extension 2: N-Layer Profile M Extension

E1.4.1 Introduction

This extension tracks [T.44] by defining the N-layer extension to the 3-layer Mixed Raster Content profile or Profile M of TIFF for facsimile. It also extends the provision for using a single multi-stripped IFD to represent a multi-stripped layer, with no coding parameter changes between stripes, to layers other than the Primary Mask. By applying the appropriate black-and-white, bi-level, constraints from Extension 4 (Section E1.6), the N-layer model and rules of this extension may also be applied to Profile T.

Often times, the contents of a page can be broken up into more

components than a 3-layer representation would allow. For instance, a complex magazine article may be represented by:

McIntyre et al.

Expires September 2001

[Page 14]

- a low resolution background image, for a low variance picture. Let's say this is 100 dpi, and is color.
- a high resolution mask layer, for the large amount of text and lines. Let's say this is 400 dpi (and binary).
- a low resolution foreground image, for the text and line color. Let's say this is 100 dpi, and is color.
- several medium resolution images, for the pictures that are scattered throughout the page. Let's say they are 200 dpi, and are color. Of these, several may be odd shapes (non-rectangular), which would require separate masks to select their regions.

Expanding the 3-layer MRC model to N-layers allows for us to represent a richer variety of image types and page structure.

This N-layer and multi-stripped single IFD Profile M Extension is specified by defining specific modifications to the text of Profile M. As a result of the specification method, this extension should be read in conjunction with Profile M.

E1.4.2. [Section 8.1](#). Overview

The objective is to change 3-layer to N-layer references.

2nd paragraph - replace the 3rd and 4th sentences with the following:

By itself, MRC does not define any new coding methods or resolutions. Instead it defines an N-layer (where N is a positive integer) image model for structuring and combining the scanned image data. The MRC N-layer model has been applied here using the TIFF format to yield a data structure which differs from [T.44] though it applies the same coding methods, uses the same compressed image data streams and is consistent with the TIFF principle of a single IFD per image.

E1.4.3. [Section 8.1.1](#). MRC 3-layer model

The objective is to change 3-layer to N-layer references, specify the presence of multiple foreground and mask layers, distinguish role of Primary Mask relative to secondary Mask layers and their interactions:

- a) Title and 1st paragraph - replace the title and paragraph with the following:

[8.1.1](#). MRC N-layer model

The N layers of the MRC model are Background (layer 1), Mask (even numbered layers such as 2, 4, 6, ..., N-1, where N is odd) and Foreground (odd numbered layers such as 3, 5, 7, ..., N, where N is

odd) pairs. The Mask and Foreground layers occur in Mask and

McIntyre et al.

Expires September 2001

[Page 15]

Foreground pairs (i.e. layers 2 & 3, 4 & 5, 6 & 7, ..., N-1 & N pairs, where N is odd). The odd numbered layers (Background and Foreground) are typically encoded with multi-level coders while the even numbered layers (Mask) are encoded with bi-level coders. Each layer may appear only once on a page and the images of each layer are coded independently of those of the other layers.

The final image is obtained by using the Mask layers to determine which of the Foreground layer pixels are rendered over the Background layer or composite of layers below the Mask. When the Mask layer pixel value is 1, the corresponding pixel from the Foreground layer is selected; when it is 0, the corresponding pixel from the Background layer or composite of layers below the Mask is selected.

Details are given in the Introduction of this section, and in Annex A of [T.44].

b) 4th paragraph - replace with the following:

Not all pages, and not all parts of a page, require 3 or more layers. If a page consists of only one layer, then that layer is the primary image whether it is a Background, Mask, or Foreground layer. If there is more than one layer, then the Primary Mask (layer 2) SHALL be one of the layers, in which case it is the primary image. In all cases, the primary image SHALL be page size.

c) 5th paragraph, 1st sentence - replace with the following:

MRC [T.44] allows a page to be transmitted as a series of stripes with each stripe consisting of 1, 2, 3 or N layers.

d) 6th paragraph - replace with the following:

Furthermore, color fax also requires the spatial resolutions of Background, Foreground and secondary Mask images to be legal fax values that are also integer factors of the Primary Mask image resolution. For example, if the Primary Mask Layer resolution is 400 pixels per inch, then allowed resolutions for the Foreground, Background and secondary Mask (layers 4, 6, ..., N-1, where N is odd) layers are 100, 200 or 400 pixels per inch; if the Primary Mask is at 300 pixels per inch, then allowed values are 100 and 300. The Foreground, Background and secondary Mask layer image resolutions can be set independent of each other.

E1.4.4. [Section 8.1.2](#). A TIFF Representation for the MRC 3-layer model

The objectives are to change 3-layer to N-layer references, specify multiple foreground and mask layers, define their IFD and SubIFD relationships and structural layout:

- a) Title and 1st paragraph - replace the title and paragraph with the following:

8.1.2. A TIFF Representation for the MRC N-layer model

In the TIFF representation of the N-layer MRC model, each page is represented by a single IFD, called the Primary IFD. The nextIFD offset associated with a Primary IFD SHALL point to the Primary IFD of the next page. If the page consists of a single layer, then the Primary IFD represents that layer. If more than one layer is present, the Primary IFD represents the Primary Mask layer and the other layers are represented by a set of child IFDs that are referenced through the SubIFD extension field [TTN1] of the Primary IFD. To distinguish MRC-specific SubIFDs from other SubIFDs, the NewSubFileType field SHALL have Bit 4 ON, indicating an MRC-related IFD. A new ImageLayer field is also introduced that consists of two values that identify the layer (Background [layer 1], Mask [layer 2, 4, 6, ..., N, where N is even], or Foreground [layer 3, 5, 7, ..., N, where N is odd]) and the order within the layer (first, second, ...image of the layer); see [Section 8.2.3](#).

- b) 5th paragraph, last sentence - replace the sentence with the following:

In all cases, if the Primary Mask layer exists, it SHALL be represented by a single IFD and a single set of coding parameters.

- c) 6th paragraph, - replace with the following:

The use of SubIFDs to store child IFDs is described in [TTN1]. When a Mask is the primary image, the secondary Mask, Background and Foreground layer images are represented with child IFDs that are referenced by the SubIFDs field in the Primary IFD. There are many possible ways to represent the secondary Mask, Background and Foreground layer images: (1) the SubIFD field of the Primary IFD is an array of pointers to all child image IFDs, one entry per child image; (2) the SubIFD field is a single pointer to a linked list of all child image IFDs; (3) the SubIFD field is an array of N-1 pointers, where the first pointer is to a linked list of all Background layer (layer 1) image IFDs, the second pointer is to a linked list of all the first Foreground layer (layer 3) image IFDs, the third pointer is to a linked list of all the second Mask layer (layer 4) IFDs, the N-1 pointer is to a linked list of all the Nth layer IFDs. A Profile M writer SHOULD structure the Background, Foreground and secondary Mask layer images using (3), as shown in the example below. Furthermore, the child IFDs representing the Background, Foreground and secondary Mask layer images SHOULD be

ordered in the file in the same order as they occur on the page.
However, a Profile M reader must scan all available child IFDs to

McIntyre et al.

Expires September 2001

[Page 17]

locate and identify IFDs associated with MRC layers.

- d) IFD - SubIFD tree diagram that follows the 6th paragraph - replace the tree diagram with the following:

(nextIFD)

```

PRIMARY IFD PAGE 0 -----> PRIMARY IFD PAGE 1--> ...
  ImageLayer = [2,1]
  NewSubFileType = 18
  SubIFD[0] ----- SubIFD[1] ---- ... --- SubIFD[N-2]
    |               |               |
    V               V               V
  Child IFD       Child IFD       Child IFD
    ImageLayer = [1,1]   ImageLayer = [3,1]   ImageLayer = [N,1]
    NewSubFileType = 16   NewSubFileType = 16   NewSubFileType =16
    |                   |                   |
    |(nextIFD)          |(nextIFD)          |(nextIFD)
    V                   V                   V
  Child IFD       Child IFD       Child IFD
    ImageLayer = [1,2]   ImageLayer = [3,2]   ImageLayer = [N,2]
    NewSubFileType = 16   NewSubFileType = 16   NewSubFileType =16
    |                   |                   |
    |(nextIFD)          |(nextIFD)          |(nextIFD)
    V                   V                   V
  Child IFD       Child IFD       Child IFD
    ImageLayer = [1,3]   ImageLayer = [3,3]   ImageLayer = [N,3]
    NewSubFileType = 16   NewSubFileType = 16   NewSubFileType =16
    |                   |                   |
    |(nextIFD)          |(nextIFD)          |(nextIFD)
    V                   V                   V
    0                   0                   0

```

E1.4.5. [Section 8.2.1](#). Baseline Fields

The objectives are to reflect presence of the multiple foreground and mask layers and the distinct Primary Mask layer in some fields, and to specify the impact of having no image data in the multiple mask and foreground layer pairs.

- a) ImageWidth - replace with the following:

ImageWidth(256). SHORT or LONG
 Primary images define the page widths, which SHALL be limited to the same set of widths as defined in Profile C, the base color profile; see [Section 6.2.1](#). In Profile M, the width of an image (i.e. Foreground, Background, or Mask layer) in the coded data stream may be less than the page width, unless the Background, Foreground or Mask is the primary image (i.e. the Primary IFD), in which case the

width of the coded data stream is the page width. The ImageWidth field SHALL always store the actual width of the coded data.

b) Compression - replace the first sentence with the following:

For Mask layers, see Sections [4.2.1](#) and [5.2.1](#)

c) PhotometricInterpretation - replace with the following:

PhotometricInterpretation(262) = 0, 2, 5, 10. SHORT

For Mask layers, 0. For Foreground and Background layers, see Sections [6.2.1](#) and [7.2.1](#).

d) StripByteCounts - replace the field with the following:

StripByteCounts(279) SHORT or LONG

In Profile M, it is permissible for the StripByteCounts value for a given strip, other than one corresponding to a mask, to have a zero entry. This means there is no encoded image data corresponding to that strip. Instead, for the Foreground or Background layers the current default image color should be used for the strip. The standard default image colors are black for the Foreground layers and white for the Background layer. The ImageBaseColor field can be used to specify other default image colors, see 8.2.3.

e) YResolution - replace the last sentence with the following:

The resolution of secondary Mask, Background and Foreground layers SHALL each be an integer factor of the Primary image, which is the Primary Mask layer, when it is present; see [Section 8.4](#).

E1.4.6. [Section 8.2.2](#). Extension Fields

The objective is to reflect presence of the multiple mask layers.

a) T6Options - replace the field with the following:

T6Options(293) = 0. SHORT

For Mask layers, see [Section 4.2.2](#).

E1.4.7. [Section 8.2.3](#). New Fields

The objectives are to reflect presence of the multiple mask and foreground layers, their impact on the ImageLayer field and resulting extension of the ImageLayer[0] field values. The newly required TIFF-FXExtension field and minimal bit setting are specified.

a) T82Options - replace the field with the following:

T82Options(435) LONG

For Mask layers, see [Section 5.2.3](#).

b) ImageLayer - replace the field, descriptive paragraph and

McIntyre et al.

Expires September 2001

[Page 19]

ImageLayer[0]field with the following:

ImageLayer(34732).

LONG

Count = 2

Image layers are defined such that layer 1 is the Background layer. Layers above layer 1 are arranged in mask (i.e. even numbered layers) and foreground (i.e. odd numbered layers) pairs. The mask selects pixels from the associated foreground that will be rendered on top of the Background or composite of layers below the mask. For example, layer 2 (i.e. the Primary Mask or first mask layer) selects pixels from layer 3 (i.e. the first foreground layer) that will be rendered on top of the Background. Layer 4 (i.e. the second mask layer) selects pixels from layer 5 (i.e. the second foreground layer) that will be rendered on top of the composite of layers 1 through 3 below. The ImageLayer field contains two values, describing the layer to which the image belongs and the order in which it is imaged.

ImageLayer[0] = 1, 2, 3, ..., N.

- 1: Image is a Background image, i.e., the image that will appear whenever the Primary Mask contains a value of 0. Background images typically contain low-resolution, continuous-tone imagery.
 - 2: Image is the Primary Mask layer. In MRC, if more than one layer is present then the Primary Mask layer (layer 2) is present, it SHALL be the Primary IFD and be full page in extent.
 - 3: Image is the first Foreground image, i.e. the image that will be rendered on top of the lower layer (layer 1) whenever the Primary Mask contains a value of 1. The Foreground image generally defines the color of text or lines, but may also contain high-resolution imagery.
 - 4: Image is the second Mask layer (layer 4).
 - 5: Image is the second Foreground image (layer 5), i.e. the image that will be rendered on top of the composite of layers 1 through 3 below whenever the second Mask contains a value of 1.
 - N-1: If N is odd, then layer N-1 is the (N-1)/2-th Mask layer.
 - N: If N is odd, then layer N is the (N-1)/2-th Foreground image, i.e. the image that will be rendered on top of the composite of layers 1 through N-2 below whenever the (N-1)/2 Mask contains a value of 1.
- If N is even, then layer N is the N/2-th Mask layer, which will be rendered black (i.e. default color of the unspecified foreground layer pair) on top of the composite layers of 1 through N-1, whenever the mask image contains a value of 1.

c) TIFF-FX extensions - append the following field at the end of the section:

TIFF-FXExtensions(34687) bit 1 is 1

LONG

See Section E1.2.1.2.

McIntyre et al.

Expires September 2001

[Page 20]

E1.4.8. [Section 8.4](#). Rules and Requirements for Images

The objectives are: i) to reflect presence and impact of the multiple mask and Foreground layers and the Primary IFD role, ii) to accommodate multi-stripped single IFD representations. These objectives are realized through changes in [Section 8.4](#) rules:

- a) replace the introductory sentence, along with rules 1, 3, 7 and 8; b)
- b) split rule 3 into two parts and renumber accordingly;
- c) append a new rule 10 as reflected below.

Profile M defines a fundamental set of rules for images in the N-layer representation. These rules, with the appropriate black-and-white (bi-level) constraints, MAY also be applied to Profile T.

1. If more than one layer exists, then a binary Mask layer SHALL be present and the first mask layer SHALL be the primary image. Mask layers SHALL support the binary data representations defined in [Section 3](#) and MAY support the binary data representations defined in other Sections, including E1.6. PhotometricInterpretation SHALL be set to "0". If only one layer exists, then the image corresponding to that layer is the Primary Image.
3. The Background and Foreground images SHALL support the color representations defined in [Section 6](#) and MAY support other color representations.
4. Images other than the Primary Image (i.e. Primary IFD) MAY optionally cover only a portion of the stripe or page.
8. In MRC Internet Fax, each layer is transmitted as a sequence of TIFF strips. If the page consists of a single layer, then all page stripes SHALL be stored in the single Primary IFD. In this case, coding parameters SHALL NOT change between strips. If the page consists of more than one layer, then all stripes of the Primary Mask layer SHALL be stored in the single Primary IFD as a series of corresponding strips. Without constraint on coding parameter changes, all stripes of the Foreground/Background/secondary Mask layers SHALL be stored in separate single-stripped IFDs. These IFDs are referenced by the Primary IFD's SubIFD field, containing an ImageLayer field with ImageLayer[0] identifying Background (layer 1), Foreground layers (layer 3, 5, 7, ..., N, where N is odd) or mask layers (layer 2, 4, 6, ..., N-1, where N is odd), and ImageLayer[1] identifying order in which images within a single layer are to be imaged. The TIFF XPosition and Yposition fields are used to indicate the placement of these images with respect to the primary image.

9. When the Primary Mask image is present, the resolution of
Background, Foreground and secondary Mask images SHALL each be an

integer factor of the Primary Mask image. For example, if the Primary Mask image is 400pixels/inch, then the Background, Foreground or secondary Mask images may be at 400 pixels/inch (400/1), 200 pixels/inch (400/2) or 100 pixels/inch (400/4).

10. When there are no coding parameter changes between stripes, a single multi-stripped IFD (i.e. each strip corresponding to a stripe) MAY be used to represent a multi-stripped layer that is not a Primary Mask layer.

E1.4.9. [Section 8.5](#). Profile M - MRC Fax Profile Summary

The objective is to identify the secondary Mask layers as being among the data blocks pointed to by the SubIFD offsets, reflect the required status of the GlobalParametersIFD, append the required TIFF-FXExtensions field to the table and callout the need to activate the N-Layer Profile M Extension bit.

- a) Revise the following [Section 8.5](#) table entries to read as shown:

+-----+-----+		+-----+
SubIFDs	<IFD>: byte offset to FG, BG,	
	or secondary Mask IFDs	
+-----+-----+		+-----+
+-----+-----+		+-----+
GlobalParameters	IFD: global parameters IFD	
IFD**		
+-----+-----+		+-----+

- b) Append the following to [Section 8.5](#) table:

+-----+-----+		+-----+
TIFF-FXExtensions	n: extension(s) identification number,	
**	bit 1 for N-Layer Profile M Extension	
	SHALL be among the set bits	
+-----+-----+		+-----+

E1.5. TIFF-FX Extension 3: Shared Data Extension

This section defines the Shared Data extension. Shared Data accommodate a single representation of reusable resources, such as image data or encoding tables, that appear or are used multiple times within a file. Most importantly, it provides a mechanism for access to the redundant resources by multiple components (i.e. sharing) within the encoding process. The sharing of resources is a new encoding provision defined in ITU-T Rec. T.44 Annex B [T.44Amd1].

E1.5.1. Overview

Many pages and/or documents have repeating components such as shapes or symbols, words, images and meta-data (i.e. Huffman Tables). This is especially true for images of text, where the repeating shapes are the characters themselves. The SharedData field is defined below to leverage the redundancy of these repeating components by storing each component once, and then referring to each stored component as many times as possible. Reference to a resource must be made with an explicit and unique reference to the Shared Data, from within the data stream that uses it (the referencing data stream).

E1.5.2. Required TIFF Fields

E1.5.2.1. Baseline Fields

None

E1.5.2.2. Extension Fields

None

E1.5.2.3. New Fields

GlobalParametersIFD(400)	IFD or LONG
See Section E1.2.1	

TIFF-FXExtensions(34687) bit 2 SHALL be 1	LONG
See Section E1.2.1	

SharedData(34689)	LONG
See Section E1.5.4.	

E1.5.3. Recommended TIFF Fields

None

E1.5.4 Shared Data

The following new field is defined:

SharedData(34689)	LONG
Count = 1	

The SharedData field contains the file offset (E), in bytes, from the beginning of the file of the first Shared Data Table. Each Shared Data Table contains a count of the number of Shared Data Entries that physically exist in the table (whether populated with an entry or not) , the Shared Data Entries themselves, and the file location of the next Shared Data Table (if any).

There can be as many Shared Data Tables as necessary to describe the number of shared data items, but there SHALL be only one SharedData field in a file, and it SHALL be in the GlobalParametersIFD (recommended to be in the first IFD in the file). The SharedData field is required when sharing data and SHALL be present. A SharedData value of zero "0" means that there are no Shared Data Tables present and that no data are currently being shared.

Each Shared Data Table consists of a 2-byte count of the number of physical entries, a sequence of 16-byte shared data entries, and a 4-byte offset to the next Shared Data Table. The last shared data table has a "next table file offset" of 0.

NOTE: In all the tables shown below, all multi-byte quantities are Stored in the endianness convention established by the TIFF file ("II" or "MM").

Shared Data Table

Bytes	Byte offsets	Type	Description
2	E - E+1	SHORT	Items in table - (i) The number of shared data entries that are physically in this table, whether populated (non-zero byte entries) or not.
16	E+2 - E+17	-	Item 1 - First shared data entry.
16	E+18 - E+33	-	Item 2 - Second shared data entry.
. Etc.	-	Item i - i-th item entry. (i determined by "items in table" value above.)
4	N - N+3	LONG	Next table file offset: 4-byte file offset, in bytes from beginning of file, of the next Shared Data Table. Where i = shared data entries in this table (see "items in table", above), and each shared data item entry is 16 bytes, $N = E+2+16*i$.

Shared Data Entries

Bytes	Byte offsets	Type	Description
4	Y - Y+3	LONG	Shared Data bytes - Size of the shared data data block, in bytes.
4	Y+4 - Y+7	LONG	Shared Data file offset - File offset of this shared data data block, in bytes, from beginning of file.
2	Y+8 - Y+9	SHORT	SharedDataType - The type of data being represented as a shared data. The table below contains the type of shared data currently defined.
2	Y+10 - Y+11	SHORT	Last page used - The page (primary IFD) ordinal (1, 2, ...) of the last page that references this shared data. A value of 0 indicates that the last page to use the shared data is not known. The page ordinal SHALL be based on the order within the primary IFD chain.
4	Y+12 - Y+15	LONG	SharedDataMemory - the number of bytes required by the referencing data stream to hold this shared data. If the shared data is encoded then the memory required SHOULD be consistent with the decoded form of the shared data. A value of 0 indicates that the SharedDataMemory is not known, or this field is not applicable. When defining a shared data and SharedDataMemory is applicable, a formula for computing the SharedDataMemory SHALL be given within the definition of the referencing data stream. An example of such a formula may be found in Section E1.6.4.2 for JBIG2.

Current SharedDataTypes:

SharedDataType Value	Description
0	undefined
1	JBIG2 Shared Data (i.e. JBIG2 symbol dictionaries, pattern dictionaries, Huffman tables, etc)

Shared Data ID:

The reference to a shared data entry SHALL be by a unique ID, which SHALL be the offset of the entry in the list described by the Shared Data Tables; the first shared data entry SHALL have ID 0. Note that the "list" of shared data entries is what the series of Shared Data Tables SHALL contain, when collected together. The order of the shared data tables SHALL determine the order of the shared data IDs. For example, if the first three tables (A, B and C) have 3, 1 and 2 shared data entry respectively; then table A will refer to shared data with IDs 0, 1, and 2; table B will refer to the shared data with ID 3; and table C will refer to shared data with IDs 4 and 5. Note that if copying shared data from one file to another, the shared data ID will likely need revision to be brought in alignment with the destination table; additionally, any copied data that refers to the copied shared data will require a change in the reference to the new ID.

E1.5.4.1 SharedDataList

The data stream that requires inclusion of one or more Shared Data (i.e. the reference data stream) SHALL include a list of IDs in a SharedDataList. For instance, an image strip that requires a shared data in order to be a complete compressed data stream, will include a SharedDataList. It is up to the interpretation of the reference data type (i.e. Compression type) as to how the shared data will be used.

A SharedDataList, located at file offset P, in bytes, from the beginning of the file, SHALL contain a 2-byte count of the number of IDs (i) in the list, followed by i 4-byte IDs.

SharedDataList Structure

Bytes	Byte offsets	Type	Description
2	P - P+1	SHORT	The number of shared data IDs (i)

	4		P+2 - P+5		LONG		First shared data ID	
+	-	-	-	+	-	-	-	+

McIntyre et al.	Expires	September	2001	[Page 26]
-----------------	---------	-----------	------	-----------

+-----+	+-----+	+-----+	+-----+	+-----+
4	...	LONG	...	
+-----+	+-----+	+-----+	+-----+	+-----+
4	$P+2+(i-1)*4 -$	LONG	$i\text{-th shared data ID}$	
	$P+5+(i-1)*4$			
+-----+	+-----+	+-----+	+-----+	+-----+

E1.5.4.2 Representation of Shared Data in TIFF

The representation of Shared Data in a TIFF file is shown below:

+-----+	+-----+	+-----+	+-----+	+-----+
		"II" or "MM"		
		+-----+		
	TIFF Header	42		
		+-----+		
		FirstIFD offset (IFD0)		:---
	+-----+	+-----+		
		...		
	:---	<-----		:---
		...		
	v +-----+	+-----+		
		...		
		+-----+		
	:--> IFD0	GlobalParametersIFD offset		:---
		+-----+		
		...		
	+-----+	+-----+		v
		...		
	:---	<-----		:---
		...		
	v +-----+	+-----+		
		...		
		+-----+		
	:--> GlobalParametersIFD	SharedData offset (1st Shared Data Table offset)		:---
		+-----+		
		...		
	TIFF file +-----+	+-----+		v
		...		
	:---	<-----		:---
		...		
	+-----+	+-----+		


```

+-----+-----+
| | | | Number of items in table |
+-----+-----+
| v | | Byte count |
| | | +-----+
| | | | File offset |
| | | +-----+
| | | Shared Data 0 | SharedDataType|
| | | +-----+
|--> Shared Data Table 0 | | Last page used|
| | | +-----+
| | | | SharedData |
| | | | Memory |
+-----+-----+
| | | Shared Data 1 | ... |
+-----+-----+
| | | ... | ... |
+-----+-----+
| | | Next Shared Data Table offset |
+-----+-----+
| | ... |
+-----+-----+

```

E1.6. TIFF-FX Extension 4: Profile T - Lossy and Lossless JBIG2 Black-and-White Fax profile

This section defines the lossy and lossless JBIG2 black-and-white profile or Profile T of TIFF for facsimile. JBIG2, ITU-T Rec. T.88 / ISO-IEC 14492 [T.88], is frequently referenced as symbol or token-based compression because it makes use of repeating shapes. The Profile T designation is representative of the term token-based compression. It must be noted that there are modes of JBIG2 that do not make use of repeating shapes, such as generic region coding. All profile T readers and writers SHALL also be able to read and write Profile S, since Profile S is the minimal binary TIFF-FX profile.

E1.6.1. Overview

This section describes a black-and-white profile that uses JBIG2 compression. The ITU-T has approved the lossy and lossless modes of JBIG2 for Group 3 facsimile. JBIG2 compression is typically used in accordance with the application rules given in ITU-T Rec. T.89 [T.89].

This profile is essentially a JBIG2 encoded black-and-white constrained profile of Profile M plus the Shared Data

extension. The MRC 3-layer model and TIFF representation, defined in

[Section 8.1](#) and constrained to a single image layer (i.e. only one layer with image data), SHALL be used. The N-layer extension (Extension2) MAY be used to accommodate more demanding binary image requirements. Odd-numbered layers (i.e. Background and Foreground) are optional since they SHALL NOT contain image data, and are typically omitted. Rational for Profile T using the Profile M structure is traceable to T.44 Annex B [T.44Amd1], in which MRC is currently the only ITU-T encoding structure that has provisions to accommodate meta data, such as JBIG2 dictionaries.

The behavior and structure of this profile is that of a plain single-layer image, similar to that of Profiles S, F and J, with some added structure to accommodate JBIG2's meta-data. Within this context, bit 4 of the NewSubFileType field SHALL be set, indicating the MRC imaging model with multiple layers. The Compression type SHALL be set to 34715 (JBIG2 Compression). When JBIG2 resources, such as symbol or pattern dictionaries, are used across more than image (i.e. stripe, layer or page) the SharedData field SHALL be used to store them. These requirements are consistent with ITU-T's definitions in Rec. T.4 Annex H [T.4Amd1] and Rec. T.44 Annex B [T.44 Amd1].

JBIG2 compression utilizes the fact that many images have repeating shapes or symbols. This is especially true for images of text, where the repeating shapes are the characters themselves. Symbol or token-based compression makes use of these repeating shapes, by storing a set of images for the symbols once, and then referring to the symbol images as many times as possible.

In a multi-page document, the same shape can occur on multiple pages. In fact, this is quite common: any shape occurring on the first page, is likely to show up on other pages as well. JBIG2 compression is improved by collecting all the repeating shapes from multiple pages, and allowing this collection to be referred to by more than one page. This collection of symbol images is referred to as a symbol dictionary. A document can contain more than one symbol dictionary. For example, one symbol dictionary might contain all the symbols that occurred on more than one page; each page then might have its own symbol dictionary, containing symbols that occurred only on that page.

There are two typical ways of compressing symbols on a page using (or generating) a symbol dictionary. One is by finding an exact match (lossless), and the other is by allowing a small amount of discrepancy between the symbol candidate, and the matching symbol in the symbol dictionary (lossy). The mode used can be distinguished by the value of the "Lossless" bit in the T88Options[0] field, which is defined below.

The representation of a symbol-compressed image consists of a shared

data list, which identifies which "JBIG2 Shared Data" are used, and a "JBIG2-coded position block". A JBIG2 Shared Data is used to represent a variety of JBIG2 shared entities (i.e. JBIG2 resources) such as symbol dictionaries, pattern dictionaries, which are used in halftone coding, and Huffman tables. The symbol dictionaries are typically contained within one or more JBIG2 Shared Data, represented within the Shared Data provisions of the Shared Data Extension. The JBIG2-coded position block consists of a series of symbol X and Y-Position coordinates, plus the IDs of the symbols located by the coordinates. The symbol bitmaps are stored in the referenced symbol dictionaries.

Each TIFF strip in an IFD whose Compression is equal to 34715 (the TIFF Compression value defined for JBIG2 below) SHALL be formatted as described in Section E1.6.5 "The JBIG2 Image Strip".

E1.6.1.1 Why Use JBIG2

The symbol-based compression model is incorporated in the JBIG2 standard. This standard boosts compression of text-like documents by retaining similar shapes across multiple images. In the case of text, the shapes are the character symbols, and for multi-page documents, the set of unique character symbols may be fairly small, especially when the same font is used on each page.

A typical image of binary text, at 300 dpi might take about 80 kilobytes to store using T.6 compression; a similar JBIG2 file might only require around 20 kilobytes to store. The compression gain can be up to a factor of two for multi-page files. Sharing dictionaries between multiple images (i.e. stripes layers or pages) makes this high compression possible, but requires some way to refer to the dictionaries used by more than one page.

E1.6.2. Required TIFF Fields

This section describes the TIFF fields required, in addition to those in Sections [8.2](#) and E1.5.2, for Profile T. Additionally it describes those not required in [Section 8.2](#) and constrained differently from those in [Section 8.2](#).

Note that these fields are defined under the premise that all odd-numbered layers are omitted, in conformance to the Profile T black-and-white constraint. However, there are application conditions that could benefit from inclusion of these odd-numbered layers and setting the associated StripByteCounts to 0. This may be true when an application wishes to represent a reverse video image (i.e. a white image on a black background). Under these conditions the list of required fields and/or values will be modified per the

StripByteCounts field below.

McIntyre et al.

Expires September 2001

[Page 30]

E1.6.2.1. Baseline Fields

ImageWidth(256). SHORT or LONG

Same page widths as Profile F, the extended black-and-white profile, and the ImageWidth extensions available to Profile F; see [Section 4.2.1](#) and E1.3.

The change in ImageWidth reference, to those of Profile F rather than Profile C, is consistent with the black-and-white constraint of this profile.

Note that for layers other than layer 2, the ImageWidth could be a fragment of the page width, as when the XPosition is greater than 0.

BitsPerSample(258) = 1. SHORT

Constraint is consistent with the black-and-white constraint.

SamplesPerPixel(277) = 1. SHORT

Constraint is consistent with the black-and-white constraint.

Compression(259) = 34715. SHORT

34715 = JBIG2 coding, ITU-T Rec. T.88 / ISO-IEC 14492 (Lossy/Lossless coding of Bi-level Images, frequently referenced as symbol-based compression), T88Options field may be present when using JBIG2.

The format of the data pointed to by the StripByteOffsets when Compression= 34715 is described later.

FillOrder(266) = 1, 2. SHORT

See [Section 8.2.1](#)

PhotometricInterpretation(262) = 0. SHORT

The '0' constraint is consistent with the black-and-white constraint of Section E1.6.1 and the MRC mask layer PhotometricInterpretation constraint defined in [Section 8.2.1](#).

ResolutionUnit(296) = 2. SHORT

See [Section 8.2.1](#).

StripByteCounts(279) SHORT or LONG

In the event that SubIFDs consists of odd-numbered layers, then the value of the StripByteCounts for all odd-numbered values of the ImageLayer[0] field SHALL be fixed to zero "0". In Profile M, it is permissible for the StripByteCounts value for a given odd-numbered layer strip to have a zero entry. This means there is no encoded image data corresponding to that strip. Instead, the current default image color should be used for the strip. The standard default image colors are white for the Background layer and black for the Foreground layer(s).

It would be efficient to omit all odd numbered layers, which are optional for Profile T. However, it may be useful to identify portions of a background and/or foreground image where the default color should be reversed; namely where a background image portion is all black, or where reverse video appears.

To define a child IFD specifying an odd-numbered layer (i.e. foreground or background) but containing no encoded image data, create an IFD with the following settings:

ImageLayer[0]:	specified odd numbered layer
ImageLayer[1]:	specified order
RowsPerStrip:	strip height
ImageLength:	image height
ImageWidth:	specified value, often the Primary IFD width
BitsPerSample:	8
PhotometricInterpretation:	10
SamplesPerPixel:	3
Compression:	1 (none)
XPosition:	specified value, frequently 0
YPosition:	specified value, frequently to top of strip
XResolution:	that of the Primary IFD
YResolution:	that of the Primary IFD
StripByteCounts:	single 0 value
StripOffsets:	single 0 entry
NewSubFileType:	bit 4 set to 1 (MRC)
ImageBaseColor:	default is white and black for background and foreground layers respectively, the reverse may be specified, no other colors SHALL be specified

XResolution(282) RATIONAL

Same XResolution as Profile F, the extended black-and-white profile, and the XResolution extensions available to Profile F; see [Section 4.2.1](#) and E1.3.

The change in XResolution reference, to those of Profile F rather than Profile M, is consistent with the black-and-white constraint of this profile.

YResolution(283) RATIONAL

Same YResolution as Profile F, the extended black-and-white profile, and the YResolution extensions available to Profile F; see [Section 4.2.1](#) and E1.3.

The change in YResolution reference, to those of Profile F rather than Profile M, is consistent with the black-and-white constraint of

this profile.

McIntyre et al.

Expires September 2001

[Page 32]

Note that unlike Profile M, Profile F and Profile T may both have non-square resolutions (i.e. different X and YResolution values).

E1.6.2.2. Extension Fields

These fields SHALL NOT be present:

- ChromaSubSampling(530)
- ChromaPositioning(531)
- Indexed(346)
- T4Options(292)
- T6Options(293)

These fields are as described in [Section 8.2.2](#):

- SubIFDs(330)
- XPosition(286)
- YPosition(287)

E1.6.2.3. New Fields

These [Section 8.2.3](#) fields SHALL NOT be present:

- Decode(433)
- T82Options(435)

These fields, described in [Section 8.2.3](#) SHALL be present:

- StripRowCounts(559)
- ImageLayer(34732)

The ImageBaseColor field has been moved from required to recommended.

The fields described in E1.5.2.3 SHALL be present.

- GlobalParametersIFD(400)
- TIFF-FXExtensions(34687)
 - Bit 3 of the TIFF-FXExtensions field SHALL be set to 1.

E1.6.3. Recommended TIFF Fields

See Sections [8.3](#), with exception of GlobalParametersIFD, which has being changed to required, and append these fields:

ImageBaseColor(434). The field MAY be omitted, in which case the default color of black and white SHALL be applied to foreground and background layers respectively. When present, the default Background or Foreground colors are typically typically revised to black or white respectively, see StripByteCounts above.

MultiProfiles(34688). See Section E1.2.2.2

SharedData(34689). This Section E1.5.4 defined field SHALL be present when data is being shared between pages.

T88Options(34690)

LONG

Count = 1 or 2

The T88Options field contains one or two values, describing options applied to the encoded data stream and any application profile to which the encoded data stream may conform. It SHALL only be present when the IFD's Compression field is equal to 34715 (JBIG2). The content provides an aid to TIFF readers, in that they describe JBIG2 options that may or may not be handled by a specific JBIG2 decoder. None of the values in the field are required for correct decoding, and the field may be ignored. In the event that this field is omitted, a reader SHALL assume that the data stream is encoded per the ITU-T T.89 Base profile (i.e. profile identification number 0x00000101), see T88Options[1] below.

T88Options[0] = 1, 2, ..., 7.

This value represents options that are being applied to the encoded data stream.

NOTE: In all the tables shown below, all multi-byte quantities are stored in the endianness convention established by the TIFF file ("II" or "MM").

The following options are defined:

Bit number	Meaning
0	HuffmanCodingNotPresent If bit 0 is 1, then the JBIG2-compressed data in this IFD SHALL NOT use Huffman or MMR (T.6) coding.
1	ArithmeticCodingPresent If bit 1 is 1, then the JBIG2-compressed data in this IFD MAY contain segments that contain arithmetic (MQ) coding.
2	Lossless If bit 2 is 1, then the JBIG2-compressed data in this IFD SHALL be a lossless representation of the original image.
3	SingleStriped If bit 3 is 1, then each TIFF strip SHALL contain a JBIG2 Position Block that has only one JBIG2 stripe (not the same as a TIFF strip). Note: There is a limit of 32767 lines per JBIG2 stripe in the event that multi-stripe mode is used.

4	TextStripesMixed	If bit 4 is 1, then some JBIG2-compressed stripes that have text region segment(s) MAY also have region segments with other data types (e.g. generic or halftone region segment).
5	ColourTagsFollow	If bit 5 is 1, then this IFD SHALL be in a mask layer whose corresponding foreground layer SHALL be coded with JBIG2 (Compression =34715) and the JBIG2-data SHALL be augmented with ITU-T Rec. T.45 (Run-length colour encoding) [T.45] coded colour tags.
6	ColourTagsPresent	If bit 6 is 1, then this IFD SHALL be in a foreground layer, which SHALL be coded with JBIG2 (Compression =34715) and the JBIG2-data SHALL be augmented with T.45-coded colour tags.
7	HalftoneRegionPresent	If bit 7 is 1, then the JBIG2-compressed data in this IFD SHALL contain halftone region segment(s).

Note: Bits 5 and 6 SHALL be 0 within Profile T, which is constrained to black-and-white images.

Also, bits 8-31 are reserved for future use, and SHALL be 0.

T88Options[1] = 0x000000101, 0x000000102, 0x000000103.

This value represents the JBIG2 profile identification number. This value may be omitted.

Default = 0x000000101.

The profile identification number of the T88Options identifies a subset of the full set of permitted parameters and parameter values that the JBIG2 coded data stream is in compliance with. None of the values of the T88Options field is required for correct decoding of a JBIG2 coded data stream and may be ignored. It allows a decoder to find out quickly which of the set of JBIG2 parameters and parameter values may be needed to decode a given data stream.

The four-byte profile identification numbers 0x00000000 through 0xFFFFFFFF are administered by ISO/IEC JTC1 SC29. ISO/IEC JTC1 SC29 has reserved profile identification numbers 0x000000100 through 0x00000FFF for ITU-T disposition. Interpretation of profiles 0x00000100 through 0x00000FFF is documented in ITU-T Rec. T.89, while interpretation of profiles 0x00000000 through 0x000000FF is documented in ISO/IEC 14492 | ITU-T T.88.

Current profiles:

JBIG2 Profile ID	Description
0x000000101	ITU-T T.89 Base (minimal Fax Application Profile) - MMR and Huffman coding, minimum of two stripes per page, stripes containing text region segment(s) SHALL NOT contain other region segments
0x000000102	ITU-T T.89 Upper MMR - MMR and Huffman coding, minimum of two stripes per page, accommodates halftone and colour tags
0x000000103	ITU-T T.89 Lower Arithmetic - Arithmetic coding, minimum of two stripes per page, stripes containing text region segment(s) SHALL NOT contain other region segments

NOTE: In this table, the term "stripe" means JBIG2 stripe (i.e. a stripe within a JBIG2 Position Block), not a TIFF strip, nor a TIFF-FX/MRC page stripe.

E1.6.4 JBIG2 Shared Data

For JBIG2 Shared Data, the SharedDataType value in the Shared Data Entry SHALL have a value of 1.

E1.6.4.1 The JBIG2 Shared Data

The JBIG2 Shared Data stored in a TIFF file contains three pieces:

1. A JBIG2 Shared Data version
2. A JBIG2-coded shared data block
3. Extensions to the JBIG2 Shared Data (these extensions contain data that are outside the scope of T.88-JBIG2)

The JBIG2-coded shared data block can have a series of JBIG2 segments. The following segment types may occur:

- Symbol dictionary segment
- Pattern dictionary segment
- JBIG2 Extension segment (future extensions to be defined within the T.88 JBIG2 standard)
- Supported profiles segment
- Table segment

Each segment in a JBIG2-coded shared data block SHALL be associated with no page (i.e., SHALL have a page association field value of 0,

as described in T.88 - JBIG2 [Section 7.2.6](#)).

McIntyre et al.

Expires September 2001

[Page 36]

To put it into perspective, when sharing data is required, a file contains one SharedData field, which is located in the GlobalParametersIFD. The SharedData field contains the file offset of the first Shared Data Table. Each Shared Data Table SHOULD include one or more Shared Data Entry (i.e. JBIG2 Shared Data). A populated entry SHALL describe the location and size of the JBIG2 Shared Data themselves. There can be as many Shared Data Tables as necessary to describe the number of JBIG2 Shared Data items.

The JBIG2 image strip (i.e. the reference data stream), described in the next section, that requires inclusion of one or more JBIG2 Shared Data SHALL include a list of JBIG2 Shared Data IDs in a SharedDataList.

E1.6.4.1.1 JBIG2 Shared Data Format

The JBIG2 Shared Data format consists of a version number, which identifies the version of the format that follows, followed by the size of the JBIG2 data block, then the data block itself, followed by any extensions. The start of the JBIG2 Shared Data is located at file offset T, in bytes, from the beginning of the file.

The JBIG2 Shared Data has the following format:

Bytes	Byte offsets	Type	Description
1	T	BYTE	The JBIG2 Shared Data version number, which is currently 0.
4	T+1 - T+4	LONG	Size of JBIG2-coded shared data block (not including extensions) = Y.
Y	T+5 - T+4+Y	BYTE	The JBIG2-coded shared data block.
2	SHORT	Extension type for first extension (these extensions contain data that are outside the scope of T.88-JBIG2)
4	LONG	Size of first extension=Z1
Z1	BYTE	Data for first extension
2	SHORT	Extension type for second extension
4	LONG	Size of second extension=Z2
Z2	BYTE	Data for second extension


```

+-----+-----+-----+-----+
| ....|      . . . . | ....| Further extensions |
+-----+-----+-----+-----+

```

E1.6.4.2 JBIG2 SharedDataMemory

For JBIG2 dictionary Shared Data the SharedDataMemory requirement is represented by the total amount of decoded symbol dictionary information a decoder will accommodate in memory at one time to decode a file. This SharedDataMemory requirement is referenced as the decoder memory. If decoder memory is specified (non-zero value), then it follows the formula described in [T.89]. Namely, the decoder memory, composed of two components (a fixed and per-symbol component), is a measure of how much memory is required to hold a decoded dictionary. The fixed component does not depend on the number of symbols, while the per-symbol component does depend on the number of symbols. The decoder memory algorithm does have dependence on whether Huffman (see note 1) or Arithmetic (see note 2) coding is used and whether the dictionaries contain symbols or halftone patterns (see note 3).

decoder memory = fixed component + per-symbol component

fixed component = $2^{\{\text{direct coding template size}\}}$
 $+ 2^{\{\text{refinement coding template size}\}} + 8K$

per-symbol component = $\text{SUM } (32 + (\text{round32}(W_i) (H_i) / 8) \text{ over } i,$
 $i = 1 \text{ to } N$

where:

i = i th symbol in the dictionary

N = the number of symbols in the dictionary

32 = a fixed per-symbol overhead required to represent:

- width of symbol
- height of symbol
- symbol ID Huffman code
- length of symbol ID Huffman code
- pointer to memory where symbol bitmap resides

round32 = is a rounding up to the next multiple of 32 bits (e.g.,
 33 rounds to 64, 128 rounds to 128)

W_i = width of the i th symbol

H_i = height of the i th symbol

This means that for each symbol there are 32 bytes overhead, plus H_i rows of bitmap data, each of which is $\text{round32}(W_i)/8$ bytes.

Note:

- 1) For Huffman coding there are no templates, so the fixed component is about 8K bytes. The fixed component can in fact be zero if

custom Huffman tables are not used.

- 2) For Arithmetic coding the per-symbol component is the same. The amount of memory needed to store the decoded dictionary bitmaps (that's the $(\text{round32}(W_i) * H_i) / 8$ component) is unchanged. Differences occur in the 32 bytes per-symbol overhead component. The width, height and pointer fractions of the overhead still apply, however the Huffman code parts do not apply. There are, however, context tables for symbol ID probability modeling that take the place of the Huffman code parts. Bottom line, 32 bytes is also a reasonable per-symbol overhead for Arithmetic coding. The template options documented in [T.89] range from a 10 pixels direct bitmap template with no refinement bitmap coding to a 16 pixels direct bitmap template with a 13 pixels refinement bitmap template. Given this range of templates, the fixed component will range from 9K to 80K bytes.
- 3) The same expression holds for pattern dictionaries of halftone image regions, since pattern dictionaries are similar to symbol dictionaries but contain halftone patterns. In this context, the references to symbol above should be taken to include patterns stored in pattern dictionaries. The pattern dictionaries, however, tend to be small relative to symbol dictionaries, since the pattern count is frequently low. This isThis only a few K of memory. It is the space required by a decoder to hold the halftone bit-planes that is of significance and determines the SharedDataMemory. This memory requirement is document in [T.89] to be typically 110% of the resolution dependent page buffer size (i.e. 1.0 Mbytes at 300 dpi and 2.0 Mbytes at 400 dpi).

E1.6.5 The JBIG2 Image Strip

The JBIG2 image stored in a TIFF file will have components that require a TIFF reader to parse through its strip content. This is due to the fact that JBIG2 is represented by two types of data:

1. shared components: namely the symbol or pattern bitmaps that are associated with multiple images, which are compressed into dictionaries and stored in one or more JBIG2 Resources.
2. image specific components: data that is specific to one image only, that with the aid of the shared components, will allow the image to be decoded.

To couple these two component sets together, each JBIG2 Image Strip within an IFD has a corresponding list of shared data IDs in the SharedDataList section of each image strip. The concatenation of the shared data and the JBIG2 position block will comprise a decodable JBIG2 stream for the image described by the IFD for that strip.

A position block is the JBIG2 encoding of the binary image code

stream. Included in the position block are the encoded positions and IDs of the symbols or patterns found in the dictionaries. The

dictionaries may lie within the position block, typically if they are used for only one image (stripe, layer or page), and/or outside of it (i.e. as JBIG2 Shared Data), typically if they are used for more than one image. Within the position block, the following segment types may occur:

1. Page information segment
 - Exactly one page information segment SHALL be present, and it SHALL be the first segment in the JBIG2-coded position block
2. End of page segment
 - Exactly one end of page segment SHALL be present, and it SHALL be the last segment in the JBIG2-coded position block
3. End of stripe segment
4. Symbol dictionary segment
5. Pattern dictionary segment
6. Generic region segment
7. Generic refinement region segment
8. Text region segment (i.e. position and ID for each symbol instance)
9. Halftone region segment (i.e. position and ID for each pattern instance)
10. Extension segment (future extensions to be defined within the T.88 JBIG2 standard, not to be confused with the extension component of the TIFF JBIG2 Image Strip below)
11. Supported profiles segment
12. Table segment.

Each segment in a JBIG2-coded position block SHALL be associated with the page defined by the page information segment.

The TIFF JBIG2 Image Strip SHALL consist of four pieces:

1. A JBIG2 Image Strip version
2. A SharedDataList: list of JBIG2 Shared Data IDs
3. The JBIG2-coded position block
4. Extensions to the image strip (these extensions contain data that are outside the scope of JBIG2, such as colour tags, which are defined within the JBIG2 Extension of Profile M. Colour tags are not permitted within this profile, Profile T).

If the JBIG2 Image Strip contains extensions, each extension is preceded by a two-byte type giving its extension type, and a 4-byte count of its length in bytes. Other data that could possibly be stored in an extension includes ASCII text, hyperlinks, and so on. The current image strip extensions and the data that may be stored in them are defined in Section E1.6.5.1. If a TIFF reader is not able to interpret an extension (if it does not recognize the extension type), then it SHOULD ignore that extension, but may skip over it to find further extensions that it can interpret.

To decode a JBIG2-coded image strip, follow these steps:

1. Retrieve the list of shared data IDs from the SharedDataList.

2. Search the collection of JBIG2 Shared Data for all the JBIG2 Shared Data, such as dictionaries, whose IDs are given in the SharedDataList.
From each one, extract its JBIG2-coded shared data block.
3. Concatenate those JBIG2-coded shared data blocks, in the order in which their IDs are given in the SharedDataList, followed by the JBIG2-coded position block.
4. This concatenated data stream can then be decoded as a JBIG2 data stream. This SHALL be a valid JBIG2 data stream containing a single page: no segment numbers may be duplicated, and so on.

As an optimization, you won't actually concatenate and decode the JBIG2-coded shared data block(s) for every position block that uses them, as that is quite inefficient. Instead, the JBIG2-coded shared data block(s) can be decoded and kept in an intermediate format, designed so that the effect of logical concatenation can be simulated.

E1.6.5.1 Image Strip Format

The image strip has the following format, which includes a SharedDataList, delimited by "====":

+-----+-----+-----+-----+-----+-----+-----+					
Bytes Byte offsets		Type	Description		
+-----+-----+-----+-----+-----+-----+-----+					
1	P	BYTE	Strip format version number, which is		
			currently 0.		
+=====+=====+=====+=====+=====+=====+=====+					
2	P+1 - P+2	SHORT	Number of shared data IDs = X		
			(i.e. JBIG2 Shared Data)		
+-----+-----+-----+-----+-----+-----+-----+					
X*4	P+3 - P+2	LONG	The sequence of shared data IDs of the		
	+(X*4)		JBIG2 Shared Data required by this		
			JBIG2-coded position block. The JBIG2		
			Shared Data (e.g. dictionaries) will be		
			prepended in the order specified by		
			this sequence of IDs, to construct the		
			array of symbols that will be used to		
			decode the JBIG2-coded position block.		
+=====+=====+=====+=====+=====+=====+=====+					
4	P+3+(X*4) -	LONG	Size of JBIG2-coded position block not		
	P+6+(X*4)		including extensions = Y		
+-----+-----+-----+-----+-----+-----+-----+					
Y	BYTE	The JBIG2-coded Position Block.		
+-----+-----+-----+-----+-----+-----+-----+					
2	SHORT	Extension type for first extension		
			(these extensions contain data that are		
			outside the scope of T.88-JBIG2, such		

| | | as colour tags) |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

+	-----+	+	-----+	+	-----+	+
	4			LONG	Size of first extension=Z1
+	-----+	+	-----+	+	-----+	+
	Z1			BYTE	Data for first extension
+	-----+	+	-----+	+	-----+	+
	2			SHORT	Extension type for second extension
+	-----+	+	-----+	+	-----+	+
	4			LONG	Size of second extension=Z2
+	-----+	+	-----+	+	-----+	+
	Z2			BYTE	Data for second extension
+	-----+	+	-----+	+	-----+	+
	Further extensions
+	-----+	+	-----+	+	-----+	+

Current JBIG2 Image Strip Extension Types:

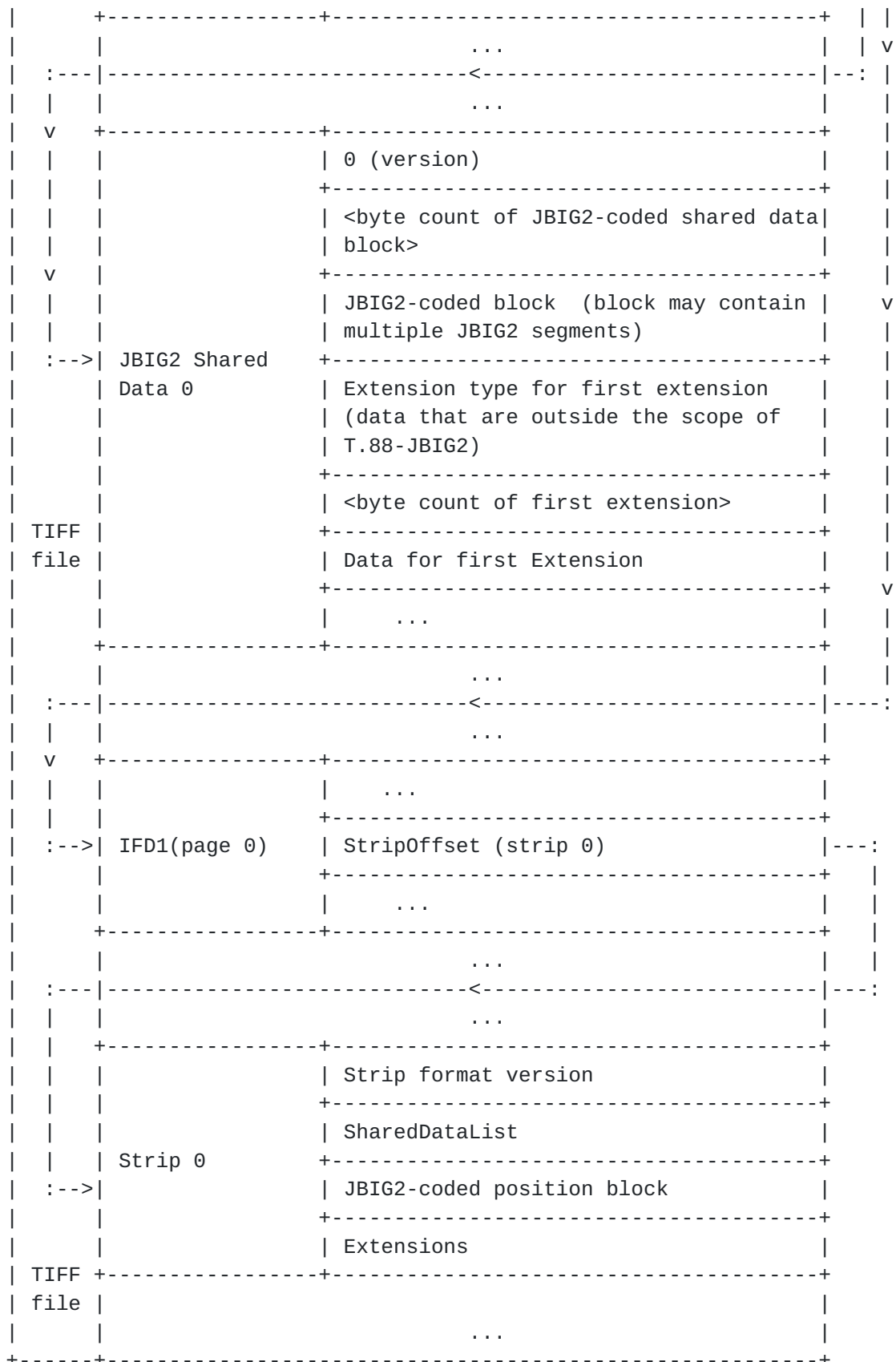
+	-----+	+	-----+	+	-----+	+
	Extension Type		Meaning			
+	-----+	+	-----+	+	-----+	+
	0		Undefined			
+	-----+	+	-----+	+	-----+	+
	1		Colour tags			
			This extension contains colour tag data, T.45			
			encoded.			
			This value SHALL NOT be used when using Profile T			
+	-----+	+	-----+	+	-----+	+

E1.6.6 Representation of JBIG2 data in TIFF

The embedding of JBIG2 data in TIFF then takes the following form:

+	-----+	+	-----+	+	-----+	+
					"II" or "MM"	
					-----+	
	TIFF		TIFF Header		42	
	file				-----+	
					FirstIFD offset (IFD0)	--:
					-----+	
					...	
	:---		-----+		<-----+	--:
					...	
					-----+	

			...		
v			GlobalParametersIFD	--:	
			StripOffsets/StripByteCounts		
-->	IFD0		...		
					v
			...		
			Next IFD offset (IFD 1)	--:-:	
			...		
---			<	--:	
			...		
v					v
			...		
-->	GlobalParameters IFD		SharedData offset (1st Shared Data Table offset)	--:	
TIFF file			...		v
			...		v
---			<	--:	
			...		
			Number of items in table		
			Byte count (of JBIG2 Shared Data 0)		
v					v
			File offset (to JBIG2 Shared Data 0)	--:	
			Shared Data 0	SharedDataType (1)	
					v
-->	Shared Data Table 0		Last page used		
			SharedDataMemory		
					v
TIFF file			Shared Data 1	...	
					v
			
			Next Shared Data Table offset		



E1.6.7 Rules and Requirements for Images

Profile T defines a fundamental set of imaging rules.

1. Typically, only ONE layer with image data SHALL exist and it SHALL be the binary Mask layer, which SHALL be the Primary IFD. JBIG2 Compression(34715) SHALL be used in this layer and the PhotometricInterpretation SHALL be 0.
2. A Foreground and/or Background layer without image data (i.e. IFD(s) with StripByteCounts = 0) MAY be present. If present, then only black and white colors SHALL be used. The Foreground and Background defaults for ImageBaseColor SHALL be black and white respectively.
3. The Primary IFD defines and extends to the entire page boundary; all attached SubIFD images cannot extend beyond the Primary image. Resolution differences may cause some pixels to "hang over" the page boundary, but no new pixels should exist completely beyond the page extent.
4. Images other than the Primary Image (i.e. Primary IFD) MAY optionally cover only a portion of the stripe or page.
5. Each Primary IFD and each MRC-specific SubIFD SHALL have an ImageLayer field to specify which layer the IFD belongs to, and the imaging order of that IFD within the layer.
6. Each Primary IFD SHALL have a NewSubFileType field value set to 18, indicating a single page of a multi-page document (bit 1) and MRC (bit 4).
7. Each MRC-specific child IFD SHALL have a NewSubFileType field value set to 16, indicating MRC (bit 4).
8. In MRC Internet Fax, each layer is transmitted as a sequence of TIFF strips. If the page consists of a single layer, then all page stripes SHALL be stored in the single Primary IFD. In this case, coding parameters cannot change between strips. If the page consists of more than one layer, then all stripes of the Primary Mask layer SHALL be stored in the single Primary IFD as a series of corresponding strips. Should Foreground/Background layers be present, the StripByteCounts SHALL be set to "0". Without constraint on coding parameter changes, each stripe of the Foreground/Background layers SHALL be stored in one IFD containing a single strip, referenced by the Primary IFD's SubIFD field, containing an ImageLayer field with ImageLayer[0] identifying Background (layer 1) or Foreground layer (layer 3), and ImageLayer[1] identifying order in which images within a single

layer are to be rendered. The TIFF XPosition and Yposition fields

are used to indicate the placement of these images with respect to the primary image.

9. If Background and/or Foreground images are present, their resolution SHALL be that of the Primary Mask image.

10. As define in Rule 10 of [Section 1.4.8](#).

E1.6.8. Profile T - Lossy and Lossless JBIG2 Black-and-White Fax profile Summary

Recommended fields are shown with an asterisk *

Required fields or values are shown with a double asterisk **. If the double asterisk is on the field name, then all the listed values are required of implementations; if the double asterisks are in the Values column, then only the values suffixed with a double asterisk are required of implementations.

Baseline Fields	Values
BitsPerSample	1**: binary mask
Compression	1: None (ImageBaseColor IFD only) 34715**: JBIG2
DateTime*	{ASCII}: date/time in the 24-hour format "YYYY:MM:DD HH:MM:SS"
FillOrder**	1: Most significant bit first 2: Least significant bit first
ImageDescription*	{ASCII}: A string describing the contents of the image.
ImageWidth	1728**, 2048, 2432, 2592, 3072, 3456, 3648, 4096, 4864
ImageLength**	n: total number of scanlines in image
NewSubFileType**	16, 18: Bit 1 indicates single page of a multi-page document on Primary IFD Bit 4 indicates MRC model
Orientation	1**-8, Default 1
PhotometricInter	0**: WhiteIsZero (Mask Layer)

| pretation |
+-----+-----+

McIntyre et al.

Expires September 2001

[Page 46]

+-----+-----+-----+		
ResolutionUnit**	2: inch	
+-----+-----+-----+		
RowsPerStrip	n: number of scanlines per strip	
+-----+-----+-----+		
SamplesPerPixel	1**, 3 (ImageBaseColor IFD only)	
+-----+-----+-----+		
Software*	{ASCII}: name & release number of	
	creator software	
+-----+-----+-----+		
StripByteCounts**	<n>: number of bytes in each strip,	
	fixed to "0" for FG/BG (when present)	
+-----+-----+-----+		
StripOffsets**	<n>: offset from beginning of file to	
	each TIFF strip	
+-----+-----+-----+		
XResolution	200**, 204**, 300, 400, 408 (written in	
	pixels/inch)	
+-----+-----+-----+		
YResolution	98**, 196**, 100**, 200**,	
	300, 391, 400 (written in pixels/inch)	
+-----+-----+-----+		
Extension Fields		
+-----+-----+-----+		
DocumentName*	{ASCII}: name of scanned document	
+-----+-----+-----+		
PageNumber**	n, m: page number followed by total page	
	count	
+-----+-----+-----+		
SubIFDs	<IFD>: byte offset to FG/BG IFDs	
+-----+-----+-----+		
XPosition	horizontal offset in primary IFD	
	resolution units	
+-----+-----+-----+		
YPosition	vertical offset in primary IFD	
	resolution units	
+-----+-----+-----+		
New Fields		
+-----+-----+-----+		
StripRowCounts	<n>: number of scanlines in each strip	
+-----+-----+-----+		
ImageLayer	n, m: layer number, imaging sequence	
	(e.g., strip number)	
+-----+-----+-----+		
GlobalParameters	IFD: global parameters IFD	
IFD**		
+-----+-----+-----+		
ProfileType*	n: type of data stored in TIFF file	

+-----+-----+

McIntyre et al.

Expires September 2001

[Page 47]

FaxProfile*	n: ITU-T compatible fax profile	
CodingMethods*	n: compression algorithms used in file	
VersionYear*	byte sequence: year of ITU-T standard	
ModeNumber*	n: mode (i.e. functional level) of T.44	
	standard	
TIFF-FXExtensions	n: extension(s) identification number,	
**	bit 3 for Profile T SHALL be among the	
	set bits	
MultiProfiles*	n: profiles or profile(s) plus	
	extension(s) applied within the file	
T88Options*	n, m: option numbers, profile number	
SharedData	<n>: offset from beginning of file to	
	the first Shared Data	

E1.7. TIFF-FX Extension 5: JBIG2 Color Extension of Profile M

This section defines extensions to Profile M that augment the pool of coders and encoding mechanisms available for use in the mask and foreground layers when encoding documents with color. The JBIG2, ITU-T Rec. T.88 / ISO-IEC 14492, bi-level coder is made available for use in both mask and foreground layer(s). It must be noted that simple JBIG2 symbol-based compression is limited to matching symbols in a binary image, but ITU-T Rec. T.44 Annex B [T.44Amd1] expands this to include single-color images, such as colored text. The T.44 defined mechanism, referenced as "colour tag" encoding, is only available for use in foreground layers that are associated with JBIG2 coded mask layers. The more conventional multi-level color coders, such as JPEG, are also available for use in the encoding of foreground layer colors when JBIG2 is used in the mask layer(s).

Revise [Section 8](#), 2nd sentence to read as follows:

Implementers of this profile are required to implement Profiles S and C, and may optionally implement other profiles.

E1.7.1. Overview

Use of JBIG2 in Profile M effectively amounts to application of

Profile T (i.e. Section E.2) to the mask layer(s), without imposition

McIntyre et al.

Expires September 2001

[Page 48]

of the Profile T black-and-white constraints that prohibit the presence of background and/or foreground image data (i.e. if present StripByteCounts = 0). This translates to augmenting the MRC model and TIFF representation, defined in Profile M, with the Shared Data extension (Extension 1), and using JBIG2 coding (i.e. Compression = 34715) in the mask layer. The N-layer model and TIFF representation, defined in Extension 2 (Section E1.4) MAY also be used if the corresponding TIFF-FXExtensions bit is set.

Use of JBIG2 and "colour tagging" to encode the colors of the foreground layer(s) translates to attaching discrete color values to the JBIG2 coded symbols (shapes) that are represented in the associated mask layer.

E1.7.2. Required TIFF Fields

This section describes the TIFF fields required, in addition to those in Sections [8.2](#), E1.4 and E1.5.2.3.

E1.7.2.1. Baseline Fields

Augment the [Section 8.2.1](#) compression field with JBIG2 (i.e. value = 34715) as below:

Compression(259) = 1, 3, 4, 7, 9, 10, 34715. SHORT

For Mask layer, see Sections [4.2.1](#), [5.2.1](#) and E1.6.2.1.

For Foreground and Background layers, see Sections [6.2.1](#), [7.2.1](#) and E1.6.2.1. Compression=1 is not used by previous profiles. An IFD used only to specify the default image color for a layer SHALL not have any encoded image data associated with it, i.e., the StripByteCounts field SHALL contain a 0. Since no image data exists in the IFD, the Compression field SHALL be set to 1 indicating no compression. A Compression field value of 1 is not allowed for any other IFDs.

E1.7.2.2. Extension Fields

See [Section 8.2.2](#).

E1.7.2.3. New Fields

Augment [Section 8.2.3](#) with these two fields from E1.5.2.3.

GlobalParametersIFD(400)

TIFF-FXExtensions(34687)

Bit 4 of the TIFF-FXExtensions field SHALL be set to 1.

E1.7.3. Recommended TIFF Fields

See Section E1.6.3.

The note that appears at the end of the T88Options[0] table, prohibiting activation of bits 5 or 6 (i.e. ColourTagsFollow or

ColourTagsPresent respectively) for Profile T SHALL be disregarded.

McIntyre et al.

Expires September 2001

[Page 49]

If the IFD's T88Options[0] field has the ColourTagsFollow or ColourTagsPresent bits set, then the following segment types SHALL NOT occur in the JBIG2-coded position block:

- Pattern dictionary
- Halftone region
- Generic region
- Generic refinement region

E1.7.4 JBIG2 Shared Data

See Section E1.6.4.

E1.7.5 The Image Strip

See Section E1.6.5.

E1.7.6 Representation of JBIG2 data in TIFF

See Section E1.6.6.

E1.7.7 Colour Tag (Color Symbol) Compression

E1.7.7.1 Why Use Colour Tag Compression

Another opportunity that is afforded by JBIG2 is improved compression of the foreground layer for documents containing colored text. In most cases, if a document contains text, each individual text character is a single, flat, color (e.g., black or red), and the number of such colors is limited. The foreground layer in this case looks like a number of colored blobs, one for each character, each one having the shape of the corresponding character. This foreground layer can be compressed using a new method that takes advantage of the JBIG2 structure. If the mask layer is compressed using JBIG2 symbols, then decoding it essentially yields a sequence of (XPosition, YPosition, Symbol ID) triples that represent each symbol instance. Each triple indicates that the symbol (from some dictionary) specified by "Symbol ID" SHOULD be drawn at the location "(X, Y)". Simply augmenting a text region triple with a fourth component, the color of that individual character (the symbols "colour tag"), allows storage of the foreground layer in a very small amount of space, using run-length coding on those colors. The total space taken by the foreground layer can be small in comparison to an encoded contone image.

E1.7.7.2 Colour Tag Terms of Use in TIFF

Colour tags are one of the JBIG2 image strip extensions referenced in Section E1.6.5 (The JBIG2 Image Strip). Their Representation within the image strip is expressed within Section E1.6.5.1 (Image Strip Format). Stepping back and considering the MRC (Mixed Raster Content) mask (bi-level data) and foreground (color data) layer pairs, we

arrive at the following terms of use to be applied when colour tagging is used in foreground representation:

1. the (JBIG2-compressed) bi-level data (position block) SHALL be followed immediately (in the file) by the colour tags. The colour tags SHALL appear as an extension in the JBIG2 image strip, with the image strip extension type = 1 (colour tags, as defined in Section E1.6.5.1). The colour tags SHALL be compressed using ITU-T Rec. T.45.
2. the mask IFD points to the start of the bi-level data, and the associated byte count SHALL include ONLY the bi-level data (the position block, but not the colour tags). The IFD's Compression field SHALL be set to 34715 (JBIG2). If present, the T88Options[0] field SHALL have bit 5 set to 1 (ColourTagsFollow).
3. the foreground IFD points to the bi-level data, and the associated byte count SHALL include BOTH the bi-level data and the colour tag extension. The IFD's Compression field SHALL be set to 34715 (JBIG2). The IFD's PhotometricInterpretation SHALL indicate the color space used to interpret the colors found in the colour tag data. If present, the T88Options[0] field SHALL have bit 6 set to 1 (ColourTagsPresent).
4. in the event that two symbol instances overlap, the reconstructed image SHALL be the one obtained by drawing each JBIG2 symbol with the appropriate color, where the drawing SHALL be done in the order that the JBIG2 symbols appear in the encoded JBIG2 image strip.

Thus, the foreground IFD completely describes an image: it points to enough data to reconstruct a colored image that contains the right color at each pixel selected by the mask. It is reasonable that a decoder will recognize that both a mask IFD and a foreground IFD point to the same JBIG2 data, and decode the JBIG2 data only once (not once for the mask, and again for the foreground). The T88Options[0] bits "ColourTagsFollow" and "ColourTagsPresent" are designed to make the decoder's job easier: if it sees ColourTagsFollow in the T88Options[0] field of a mask IFD, it knows it should defer decoding it until it decodes the corresponding foreground IFD.

Similarly, if it sees ColourTagsPresent in a foreground IFD, then it can optimize the drawing/decoding operations.

Note: This representation has two pointers to the same part of the TIFF-FX file, which is a violation of a TIFF guideline, and could conceivably cause problems in some unsuspecting TIFF editors. However, these unsuspecting TIFF editors would probably not be able to decode the JBIG2 data anyway. The shared pointers occur only in a restricted case.

The nature of the two pointers is illustrated below:

McIntyre et al.

Expires September 2001

[Page 51]

4. Images other than the Primary Image (i.e. Primary IFD) MAY optionally cover only a portion of the stripe or page.

c). Append a Rule 10 to read as defined in Section E1.4.8.

E1.7.9. JBIG2 Extension of Profile M Summary

Recommended fields are shown with an asterisk *

Required fields or values are shown with a double asterisk **. If the double asterisk is on the field name, then all the listed values are required of implementations; if the double asterisks are in the Values column, then only the values suffixed with a double asterisk are required of implementations.

Baseline Fields	Values
BitsPerSample	1**: binary mask 2-8**: bits per color sample for FG/BG 9-16: optional 12 bits/sample
Compression	1: None (ImageBaseColor IFD only) 3**: Modified Huffman and Modified Read (mask layer) 4: Modified Modified Read 7**: JPEG (FG/BG layers) 9: JBIG, per T.85 10: JBIG, per T.43 34715**: JBIG2, per T.88 (Note that T.45 Run-length Colour Encoding is also required for FG layers)
DateTime*	{ASCII}: date/time in the 24-hour format "YYYY:MM:DD HH:MM:SS"
FillOrder**	1: Most significant bit first 2: Least significant bit first
ImageDescription*	{ASCII}: A string describing the contents of the image.
ImageWidth	864, 1024, 1216, 1728**, 2048, 2432, 2592, 3072, 3456, 3648, 4096, 4864
ImageLength**	n: total number of scanlines in image

+-----+-----+-----+		
NewSubFileType**	16, 18:	
	Bit 1 indicates single page of a multi-	
	page document on Primary IFD	
	Bit 4 indicates MRC model	
+-----+-----+-----+		
Orientation	1**-8, Default 1	
+-----+-----+-----+		
PhotometricInterpretation	0**: WhiteIsZero (Mask layer)	
	2: RGB	
	5: CMYK	
	10**: ITULAB (FG/BG layers)	
+-----+-----+-----+		
ResolutionUnit**	2: inch	
+-----+-----+-----+		
RowsPerStrip	n: number or scanlines per strip	
+-----+-----+-----+		
SamplesPerPixel	1**: L* (lightness)	
	3: RGB, LAB, CMY	
	4: CMYK	
+-----+-----+-----+		
Software*	{ASCII}: name & release number of	
	creator software	
+-----+-----+-----+		
StripByteCounts**	<n>: number or bytes in each strip	
+-----+-----+-----+		
StripOffsets**	<n>: offset from beginning of file to	
	each TIFF strip	
+-----+-----+-----+		
XResolution	100, 200**, 300, 400 (written in	
	pixels/inch)	
+-----+-----+-----+		
YResolution	equal to XResolution (pixels SHALL be	
	square)	
+-----+-----+-----+		
Extension Fields		
+-----+-----+-----+		
T4Options	0**: required if Compression is Modified	
	Huffman, EOLs not byte aligned	
	1: required if Compression 2D Modified	
	Read, EOLs are not byte aligned	
	4**: required if Compression Modified	
	Huffman, EOLs byte aligned	
	5: required if Compression 2D Modified	
	Read, EOLs are byte aligned	
+-----+-----+-----+		
T6Options	0: required if Compression is 2D	
	Modified Modified Read	

+-----+-----+

McIntyre et al.

Expires September 2001

[Page 54]

DocumentName*	{ASCII}: name of scanned document	
PageNumber**	n, m: page number followed by total page	
	count	
ChromaSubSampling	(1,1), (2, 2)**	
	(1, 1): equal numbers of lightness and	
	chroma samples horizontally & vertically	
	(2, 2): twice as many lightness samples	
	as chroma horizontally and vertically	
ChromaPositioning	1: centered	
Indexed	0: not a palette-color image	
	1: palette-color image	
SubIFDs	<IFD>: byte offset to FG/BG IFDs	
XPosition	horizontal offset in primary IFD	
	resolution units	
YPosition	vertical offset in primary IFD	
	resolution units	
New Fields		
Decode	minL, maxL, mina, maxa, minb, maxb:	
	minimum and maximum values for L*a*b*	
ImageBaseColor	a, b, c: background color in ITULAB	
StripRowCounts	<n>: number of scanlines in each strip	
ImageLayer	n, m: layer number, imaging sequence	
	(e.g., strip number)	
T82Options	0: T.85 profile of T.82 coding	
GlobalParameters	IFD: global parameters IFD	
IFD**		
ProfileType*	n: type of data stored in TIFF file	
FaxProfile*	n: ITU-T compatible fax profile	
CodingMethods*	n: compression algorithms used in file	

VersionYear*	byte sequence: year of ITU-T standard	
ModeNumber*	n: mode of T.44 standard	
TIFF-FXExtensions	n: extension(s) identification number,	
**	bit 4 for Profile M SHALL be among the	
	set bits	
MultiProfiles*	n: profiles or profile(s) plus	
	extension(s) applied within the file	
T88Options*	n, m: option numbers, profile number	
	- if colour tag is used then bit 1 of n	
	SHALL NOT be set	
	- if bit 5 is set then IFD is in mask	
	layer with colour tag augmented JBIG2	
	coded corresponding foreground layer	
	- if bit 6 is set then IFD is in	
	foreground layer with colour tag	
	augmented JBIG2 coding	
SharedData	<n>: offset from beginning of file to	
	the first Shared Data	

E1.8. Security Considerations

This document describes a file format for Internet fax, which is a series of profiles of TIFF for facsimile. As such, it does not create any security issues not already identified in [TIFF-REG], in its use of fields as defined in [TIFF]. There is also new TIFF fields defined within this specification, but they are of a purely descriptive nature, so that no new security risks are incurred.

Further, the encoding specified in this document does not in any way preclude the use of any Internet security protocol to encrypt, authenticate, or non-repudiate TIFF-encoded facsimile messages.

E1.9. References

The following references are appended to the list in [Section 11](#) of RFC XXXX.

[[[RFC XXX is a placeholder for the pending Draft Standard revision to [RFC 2301](#), currently documented in [draft-ietf-fax-tiff-fx-09.txt](#).]]]

[RFC XXXX] RFC XXXX, Draft Standard "File Format for Internet Fax", known as TIFF-FX (pending issue of [draft-ietf-fax-tiff-fx-09.txt](#))

[T.4 Amd1] Amendment 1 to ITU-T Recommendation T.4, Standardization of group 3 facsimile apparatus for document transmission, March 2000
[T.44Amd1] Amendment 1 to ITU-T Recommendation T.44, Mixed Raster Content (MRC), March 2000.

[T.45] ITU-T Recommendation T.45, Run-length Colour Encoding, March 2000.

[T.88] ITU-T Recommendation T.88|ISO/IEC 14492:2000, Information technology - Lossy/Lossless coding of Bi-level Images. (Commonly referred to as JBIG2 standard.)

[T.89] ITU-T Draft Recommendation T.89, Application Profiles for Recommendation T.88 - Lossy/Lossless Coding of Bi-level Images (JBIG2) for Facsimile, November 2000.

[REQ] [RFC 2119](#), "Key words for use in RFCs to Indicate Requirement Levels", S. Bradner, Harvard University, March 1997.

E1.10 Authors' Addresses

Dave Abercrombie
Xerox Corporation
Mailstop PAHV-121
3400 Hillview Ave
Palo Alto, CA 94304, USA
Voice: +1-650-813-6811
Fax: +1-650-813-6860
Email: aber@pahv.xerox.com

Robert Buckley
Xerox Corporation
Mailstop 0128-30E
800 Phillips Road
Webster, NY 14580, USA
Voice: +1-716-422-1282
Fax: +1-716-265-8871
Email: rbuckley@crt.xerox.com

Lloyd McIntyre
Xerox Corporation
Mailstop PAHV-121
3400 Hillview Ave
Palo Alto, CA 94304, USA
Voice: +1-650-813-6762
Fax: +1-650-845-2340
Email: lmcintyre@pahv.xerox.com

William Rucklidge
Intelligent Markets
410 Jessie St., Suite 602
San Francisco, CA 94103, USA
Voice: +1-415-369-5013
Email: wjr@imarkets.com

Full Copyright Statement

Copyright (C) The Internet Society (2001). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Annex A. List of edits to [draft-ietf-fax-tiff-fx-Extension1-00](#)

Note: "e" below the item number indicates that the change is editorial, while "r" indicates a requirement change. Less significant editorial changes are not identified in this list.

+-----+-----+-----+-----+-----+-----+			
No. Section		Edit March 01, 2000	
+-----+-----+-----+-----+-----+-----+			
1.	Abstract	Added comment clarifying RFC XXXX reference as	
e	E1.1	being current draft-ietf-fax-tiff-fx-09.txt and	
	E1.9	Abstract, 2nd sentence, changed "increased	
		resolutions" to "increased set of resolutions"	
+-----+-----+-----+-----+-----+-----+			
2.	E1.1.2	Expanded description of document layout, added a	
e	E1.9	tree diagram, clarified that some specifications	
		are defined via edits to sections of RFC XXXX,	
		and added the appropriate [REQ] reference	
+-----+-----+-----+-----+-----+-----+			
3.	All	E1, E2, E3, E4 and E5 extension abbreviations	
e		are unnecessary and have been removed	
+-----+-----+-----+-----+-----+-----+			
4.	E1.2.2.1	Clarified that FaxProfile value of 255 signals	
e		use of MultiProfiles field rather than	
		designates the MultiProfiles field	
+-----+-----+-----+-----+-----+-----+			
5.	E1.4.1	3rd paragraph, clarified value of expanding	
e		3-layer MRC model to N-layers	
+-----+-----+-----+-----+-----+-----+			
6.	E1.4.1	added provisions for multi-stripped single IFD	
r	E1.4.8	representation of multi-striped layers other	
	E1.6.7	than Primary Mask layers	
	E1.7.8		
+-----+-----+-----+-----+-----+-----+			
7.	E1.4.3	1st paragraph, last sentence, clarify	
e	a	it is the images of each layer that are coded	
		independently	
+-----+-----+-----+-----+-----+-----+			
8.	E1.4.4	last sentence, changed "layer 2, 4, 6, ..., N-1,"	
e	a	where N is odd" to "layer 2, 4, 6, ..., N, where	
		N is even", clarifying layer reference	
+-----+-----+-----+-----+-----+-----+			
9.	E1.4.4	instructions should indicate replacement of the	
e	c	entire 6th paragraph since there are references	
		to the background and foreground throughout	
+-----+-----+-----+-----+-----+-----+			
10.	E1.4.4	Inserted "secondary Mask" into the 2nd and 3rd	
e	c)	sentence, clarifying that secondary Masks are	
		also included in the SubIFD IFD representations	

+---+-----+-----+

McIntyre et al.

Expires September 2001

[Page 59]

+-----+-----+-----+-----+-----+-----+			
11.	E1.4.4	Deleted edit item "e)" since it is already	
e	e)	addressed in RFC XXXX.	
+-----+-----+-----+-----+-----+-----+			
12.	E1.4.7	Clarified black rendering when ImageLayer[0]=N	
e	b)	and if N is even.	
+-----+-----+-----+-----+-----+-----+			
13.	E1.4.8	item 8, clarified correlation of strips to TIFF	
e	E1.6.7	structure and stripes to page structure	
+-----+-----+-----+-----+-----+-----+			
14.	E1.5	Increased potential for broader usage of Shared	
r	all	Data by removing restriction permitting use only	
		within the Profile M MRC structure plus	
		references to Shared Data being a Profile M	
		extension	
+-----+-----+-----+-----+-----+-----+			
15.	E1.5.2	Reduced the set of Required and Recommended	
e	E1.5.3	fields to only those associated with the Shared	
		Data structure representation. Fields Required	
		or Recommended for the data being stored within	
		the structure will be defined during definition	
		of the particular data type that uses it, as in	
		E1.6 for JBIG2 Shared Data	
+-----+-----+-----+-----+-----+-----+			
16.	E1.5.4	Shared Data Entries table, Byte offsets	
r		Y+10 - Y+11, changed recommendation to a	
		requirement that page ordinal is based on IFD	
		order, this minimizes compatibility concerns	
+-----+-----+-----+-----+-----+-----+			
17.	E1.5.4	Shared Data ID, 4th sentence, clarification	
e		extension	
+-----+-----+-----+-----+-----+-----+			
18.	E1.5.4.1	1st sentence, changed may to SHALL since the	
r		SharedDataList must be present in the stream	
+-----+-----+-----+-----+-----+-----+			
19.	E1.6.1	clarified Profile T image layer constraints	
e			
+-----+-----+-----+-----+-----+-----+			
20.	E1.6.1	Moved rational for Profile T using the Profile M	
e	E1.6.1.1	structure from E1.6.1.1 to 2nd paragraph of	
		E1.6.1 and added clarification	
+-----+-----+-----+-----+-----+-----+			
21.	E1.6.1	clarified requirement to use SharedData only	
e	E1.6.2.3	when resources are being shared between pages	
	E1.6.8		
	E1.7.9		
+-----+-----+-----+-----+-----+-----+			

22.	E1.6.2.3	2nd paragraph, changed "SHOULD" to "MAY" since	
r		there are reasons for inclusion	
23.	E1.6.3	T88Options table, appended note clarifying that	
e		bits 8 - 31 are reserved and set to zero.	
24.	E1.6.3	For consistency, changed "or" to "and" in note	
e		at the end of the T88Options table	
25.	E1.6.3	Added MultiProfiles and SharedData fields	
e			
26.	E1.6.4.1	2nd last paragraph, simplified Shared Data Table	
e		description to reduce redundancy and clarified	
		Shared Data Table requirements	
27.	E1.6.5	3rd paragraph, clarified reason for dictionary	
e		location within or outside the position block	
		and content of text and halftone region segments	
28.	E1.6.5	item 10, clarified presence of JBIG2 defined	
e		extensions located within the position block and	
		TIFF defined extensions located outside the	
		position block but within the TIFF image strip	
29.	E1.6.5.1	1st sentence and byte 2 of table, changed	
e		ResourceList and "resource IDs" to	
		SharedDataList and "shared data IDs"	
30.	E1.6.6	IFD0, changed location of "Next IFD offset	
e		(IFD 1)" to the bottom of the 3 entries	
31.	E1.6.8	Inserted MultiProfiles field into the summary	
e	E1.7.9	tables following TIFF-FXExtensions field	
32.	E1.7	Appended qualification that is more generic,	
e		inclusive of new profiles	
33.	E1.7.1	1st paragraph, 2nd sentence, deleted "3-layer"	
e		as the restriction is unnecessary	
34.	E1.7.2.3	Clarified that GlobalParametersIFD and	
e		TIFF-FXExtensions are the fields being appended	
		to those of 8.2.3	
35.	E1.7.7.1	5th sentence, clarified that the triplets	
e		represent symbol instance	

[Page 61]

+-----+			+-----+		
36.	E1.7.7.2	TIFF diagram, changed "Symbol & positions" to			
e		"Position block", which is more encompassing			
+-----+			+-----+		
37.	E1.7.7.2	terms of use items 2, 3 and 4, changed SHOULD to			
r		SHALL, eliminating any ambiguity			
+-----+			+-----+		
38.	Title	To reduce confusion with individual extensions,			
e		changed to "TIFF-FX Extension Set 1"			
+-----+			+-----+		
39.	all	Changed tag values and compression type for			
r		TIFF-FXExtensions, MultiProfiles, SharedData,			
		T88Options and JBIG2 from public to private			
		values, avoiding potential Adobe constraints			
+-----+			+-----+		
40.	E1.2.1.2	Current TIFF-FX Extensions table, Bit numbers 3			
e	E1.6.2.3	and 4, and TIFF-FXExtensions, changed "bit 2 is			
	E1.7.2.3	set" or bit 2 SHALL be set to "bit 2 is			
	E1.6.8	frequently set", consistent with dictionaries			
	E1.7.9	being in locations other than Shared Data			
+-----+			+-----+		

