            A File Format for the Exchange of Images in the Internet

Status of This Memo

Abstract

   This document defines a standard file format for the exchange of
   fax-like black and white images within the Internet. It is a product
   of the Network Fax Working Group of the Internet Engineering Task
   Force (IETF).

   The standard is:

        ** The file format should be TIFF-B with multi-page files
           supported. Images should be encoded as one TIFF strip
           per page.

        ** Images should be compressed using MMR when possible. Images
           may also be MH or MR compressed or uncompressed. If MH or MR
           compression is used, scan lines should be "byte-aligned".

        ** For maximum interoperability, image resolutions should
           either be 600, 400, or 300 dpi; or else be one of the
           standard Group 3 fax resolutions (98 or 196 dpi
           vertically and 204 dpi horizontally).

   Note that this specification is self contained and an implementation
   should be possible without recourse to the TIFF references, and that
   only the specific TIFF documents cited are relevant to this
   specification. Updates to the TIFF documents do not change this
   specification.

   Experimentation with this file format specified here is encouraged.

1. Introduction

   The purpose of this document is to define a standard file format for
   exchange of black and white images using the Internet. Since many
   organizations have already started to accumulate and exchange scanned
   documents it is important to reach agreement about an interchange
   file format in order to promote and facilitate the exchange and
   distribution of such documents. These images may originate from
   scanners, software, or facsimile (fax) machines. They may be
   manipulated by software, communicated, shared, duplicated, displayed,
   printed by laser printers, or faxed.

   This file format provides for the uniform transfer of high quality
   images at a reasonable cost and with reasonable speed whether these
   files are generated by scanners, totally by software (e.g., text-to-
   fax, bitmap-to-fax, OCR, etc), or by fax. Also the intent of this
   document is to remain compatible with future moves to multi-level
   (i.e., gray-scale), higher resolution, or color images. The format
   proposed here is supported by both commercially available hardware
   and commercial and public domain software for most popular platforms
   in current use.

   The file format for images is a totally separate issue from how such
   files are to be communicated. For example, FTP or SMTP could be used
   to move an image file from one host to another, although there are
   complications in the use of SMTP as currently implemented due to file
   size and the need to move binary data. (There is currently a
   proposal for removing these limitations from SMTP and in particular
   extending it to allow binary data. See reference [1].)

   One major potential application of the communications format defined
   here is to allow images to be sent to fax machines using the
   Internet. It is intended that one or more separate companion
   documents will be formulated to address the issues of standardization
   in the areas of protocols for transmitting images through the
   Internet and the issues of addressing fax machines and routing faxes.
   Just as the exchange format is separate from the transmission
   mechanism, it is also separate from how hosts store images.

   This document specifies a common exchange format; it does not require
   a host to store images in the format specified here, only to convert
   between the host's local image storage formats and the exchange
   format defined here for the purpose of exchanging images with other
   hosts across the network.

   This standard specifies the use of TIFF (Tagged Image File Format,
   see below) as a format for exchange of image files. This is not a
   specific image encoding, but a framework for many encoding

techniques, that can be used within the TIFF framework. For example,
within TIFF it is possible to use MMR (the data encoding of CCITT
Group 4 fax, see below), MH or MR (the data encodings of CCITT Group
3 fax), or other encoding methods.

Which encoding technique to use is not specified here. Instead, with
time the encoding schemes used by most document providers will emerge
as the de-facto standard. Therefore, we do not declare any as "the
standard data encoding scheme," just as we do not declare that
English is the standard publication language. (However, we expect
that most document providers will use MMR in the immediate future
because it offers much better compression ratios than MH or MR.)

Similarly, TIFF does not require that an image be communicated at a
specific resolution. Resolution is a parameter in the TIFF
descriptive header. We do suggest that images now be sent using one
of a set of common resolutions in the interests of interoperability,
but the format accommodates other resolutions that may be required by
specialized applications or changing technologies.

Occasionally, image files will have to be converted, such as in the
case where a document that was scanned at 400 dpi is to be printed on
a 300 dpi printer. This conversion could be performed by the
document provider, by the consumer, or by a third party. This
document specifies neither who performs the conversion, nor which
algorithms should be used to accomplish it.

Note that this standard does not attempt to define an exchange format
for all image types that may be transmitted in the Internet. Nothing
in this standard precludes it from being used for other image type
such as gray-scale (e.g., JPEG) or color images but, for the purposes
of standardization, the scope of this document is restricted to
monochromatic bitmapped images.

The developers of this standard recognize that it may have a limited
lifespan as Office Document Architecture (ODA) matures and comes into
use in the Internet; ultimately the class of images covered by this
standard will likely be subsumed by the more general class of images
supported by the ODA standards. However, at present, there does not
appear to be a sufficient installed base of ODA compliant software
and the ODA standards are not fully mature. This standard is
intended to fill the need for a common image transfer format until
ODA is ready. Finally, we believe that it should be possible to
automatically map images encoded in the format specified here into a
future ODA-based image interchange format, thus providing a
reasonable transition path to these future standards.

2. Relationship to Fax

   Transmission of facsimile (fax) images over phone lines is becoming
   increasingly widespread. The standard of most fax machines in the
   U.S. is CCITT Group 3 (G3), specified in Recommendations T.4 and
   T.30 [2] and in EIA Standards EIA-465 and EIA-466. G3 faxes are 204
   dots per inch (dpi) horizontally and 98 dpi (196 dpi optionally, in
   fine-detail mode) vertically. Since G3 neither assumes error free
   transmission nor retransmits when errors occur, the encoding scheme
   used is differential only over small segments never exceeding 2 lines
   at standard resolution or 4 lines for fine-detail. (The incremental
   G3 encoding scheme is called two-dimensional and the number of lines
   so encoded is specified by a parameter called k.)

   CCITT Group 4 fax (G4) is defined by the T.400 and T.500 series of
   Recommendations as well as Recommendation T.6 [2]. It provides for
   400 dpi (both vertical and horizontal) and is a fully two-dimensional
   encoding scheme (k is infinite) called MMR (Modified Modified READ,
   where READ stands for: Relative Element Address Designate). G4
   assumes an error free transmission medium (generally an X.25 Public
   Data Network, or PDN). Because of this, G4 is not in widespread use
   in the U.S. today.

   The traditional fax bundles together four independent issues:

        (1) Data presentation and compression;
        (2) Data transmission;
        (3) Image input from paper ("scanning"); and
        (4) Image output to paper ("printing").

   This bundling supports, for example, the high quality CCITT Group 4
   (G4) images (400x400 dpi) but only over X.25 public data networks
   with error correction, and similarly it supports the mid-quality
   CCITT Group 3 (204x98 and 204x196 dpi) but only over phone voice
   circuits (the Switched Telephone Network, or STN) without error
   correction. This bundling does not support the use of any other data
   transmission capabilities (e.g., FTP over LANs and WANs), nor
   asynchrony between the scanning and the printing, nor image storage,
   nor the use of the popular laser printers for output (even though
   they are perfectly capable of doing so).

   In conventional fax, images are never stored. In today's computer
   network environment, a better model is:

        (1) Images are scanned into files or created by software;
        (2) These image files are stored, manipulated, or communicated;
        (3) Images in a file are printed or displayed.

The only feature of the CCITT fax that should be used is the encoding technique (preferably MMR, but with MR or MH allowed) which may be implemented with a variety of fax-oriented chips at low cost due to the popularity of fax.

"Sending a fax" means both encoding (and decoding) the fax images as well as transmitting the data. Since the Internet ALREADY provides several mechanisms for data transmission (in particular, FTP for general file transmission), it is unnecessary to use the data transmission methods specified in the CCITT standard. Within the Internet, each fax image should be stored in a file and these files could be transferred (e.g., using FTP, SMTP, RPC-based methods, etc.).

Fax machines should be considered just as scanners and printers are, as I/O devices between paper and files; but not as a transmission means. Higher quality Group 4 images are thus supported at low cost, while enjoying the freedom to use any computerized file transfer and duplication mechanism, standard laser printers, multiple printing (possibly at multiple remote sites) of the same image without having to rescan it physically, and a variety of software for various processing of these images, such as OCR and various drawing programs. We should be able to interoperate with files created by fax machines, scanners, or software and to be able to print all of them on fax machines or on laser printers.

The CCITT Recommendations assume realtime communications between fax machines and do not therefore specify any kind of fax file format. We propose using TIFF [3] which seems to be emerging as a standard, for encapsulation of encoded images. Because they assume realtime communications, the CCITT fax protocols require negotiations to take place between the sender and receiver. For example, they negotiate whether to use two-dimensional coding (and with what k parameter) and what (if any) padding there is between scan lines.

In our approach, the image in the file is already compressed in a particular manner. If it is to be sent to an ordinary fax machine using a fax board/modem, that board will perform the negotiations with the receiving fax machine. In the cases where the receiver cannot handle the type of compression used in the file, it will be necessary to convert the image to another compression scheme before transmission. (Most fax cards seem to either store images using the default values of the parameters which are negotiated or in a format which can quickly be converted to this. With currently available hardware and software, any necessary format conversion should be easy to accomplish.)

In conventional fax, if the compression used for a particular image

is "negative" (i.e., the compressed form is larger than the
uncompressed form, something that happens quite frequently with
dithered photographic images), the larger compressed form of the
image is still sent. If the images are first scanned into files,
this problem could be recognized and the smaller, uncompressed file
sent instead. (Also, Recommendations T.4 and T.6 [2] allow for an
"uncompressed mode." Thus, lines which have negative compression may
each be sent uncompressed. However, very few G3 fax machines support
this mode.)

3. Image File Format

   Image files should be in the TIFF-B format which is the bi-level
   subclass of TIFF. TIFF and TIFF-B are described in reference [3],
   cited at the end of this document. Images should be compressed using
   MMR (the G4 compression scheme) because it offers superior
   compression ratios. However, images may also be compressed using MH
   or MR (the G3 methods). MMR offers much better compression ratios
   than these (which are used in G3 fax because of the lack of an
   error-free communications path).

   TIFF-F, described in [4], is the proposed subclass of TIFF-B for fax
   images. However, since TIFF-F was intended for use with G3, it
   recommends against certain features we recommend. Specifically, it
   suggests not using MMR or MR compression (we recommend MMR and allow
   MR) and prohibits uncompressed mode (which we allow and suggest for
   some photographic images). Apart from these, the TIFF-F restrictions
   should be followed. (Complete compatibility between the format
   specified here and TIFF-F can only be guaranteed for MH compressed
   images.)

      [NOTE: Aldus Corp., the TIFF Developer, considers fax
      applications to be outside the scope of mainstream TIFF
      since it is not a part of general publishing which is
      what TIFF was originally designed for. They specify the
      LZW [5] compression scheme rather than MMR. We, however,
      are concerned with the transmission and storage of images
      rather than publishing. Therefore, we are more concerned
      with compression ratios and compatibility with CCITT fax
      than Aldus is.]

   TIFF itself allows for gray-scale and color images. Image files
   should be restricted to TIFF-B for now because most of the currently
   available hardware is bi-level (1 bit per pixel). In the future,
   when gray-scale or color scanners, printers, and fax becomes
   available, the file format suggested here can already accommodate it.
   (For example, though JPEG is not currently a TIFF defined compression
   type, work is currently underway for including it as such.)

        [NOTE: In this document, we will use the term "reader"
        or "TIFF reader" to refer to the process or device
        which reads and parses a TIFF file.]

3.A. TIFF File Format

   Figure 1 below (reproduced here from Figure 1 of reference [3])
   depicts the structure of a TIFF file.

   TIFF files start with a file header which specifies the byte order
   used in the file (i.e., Big or Little Endian), the TIFF version
   number, and points to the first "Image File Directory" (IFD). If the
   first two bytes are hex 4D4D, the byte order is from most to least
   significant for both 16 and 32 bit integers (Big Endian). If the
   first two bytes are hex 4949, the byte order is from least to most
   significant (Little Endian). In both formats, character strings are
   stored into sequential bytes and are null terminated.

   The next two bytes (called the TIFF Version) must be 42 (hex 002A).
   This does not refer to the current TIFF revision number. The
   following 4 bytes contain the offset (in bytes from the beginning of
   the file) to the first IFD.

   An IFD contains a 2 byte count of the number of entries in the IFD, a
   sequence of 12 byte directory entries, and a 4 byte pointer to the
   next IFD. One of these fields (StripOffsets) points to (parts of) an
   image in the file. There may be more than one image in the file
   (e.g., a "multi-page" TIFF file) and therefore more then one IFD.
   IFD field entries may appear in any order.

   Each directory entry is 12 bytes and consists of a tag, its type, a
   length, and an offset to its value. If the value can fit into 4
   bytes (i.e., if the type is BYTE, SHORT, or LONG), the actual value
   rather than an offset is given. If the value is less than 4 bytes
   (i.e., if the type is BYTE or SHORT), it is left-justified within the
   4 byte value offset. More details about directory entries and the
   possible tags will be given in Section 3.C.

   All pointers (called offsets in the TIFF reference [3]) are the
   number of bytes from the beginning of the file and are 4 bytes long.
   The first byte of the file has an offset of 0. In the case of only
   one image per file, there should therefore be only one IFD. The last
   IFD's pointer to the next IFD is set to hex 00000000 (32 bits).

   The entries in an IFD must be sorted in ascending order by Tag.

```
                    Header
             +--------+--------+
         0   |        |        | Byte Order               Directory Entry
             +--------+--------|                     +--------+--------+
         2   |        |        | Version(42)      X  |        |        | Tag
             +--------+--------|                     +--------+--------|
         4   |        |        | Offset of      X+2  |        |        | Type
             +-    - A -    -+   0th IFD              +--------+--------|
         6   |        |        |                X+4  |        |        |
             +--------+--------+                     +-    -        -+  Length
                      |                              |        |        |
                      |                              +--------+--------+
                      |                         X+8  |        |        | Value
                      V                              +-    - Y -    -+    or
                                                     |        |        | Value
                    IFD                              +--------+--------+ Offset
             +--------+--------+
         A   |    - B -        | Entry Count                    |
             +--------+--------|                                |
             |        |        |                                |
       A+2   |        |        | Entry 0                        V
             |        |        |
             +--------+--------+                     +--------+--------+
             |        |        |                     |        |        |
      A+14   |        |        | Entry 1          Y  |        |        | Value
             |        |        |                     |        |        |
             +--------+--------+                     +--------+--------|
             |        |        |
      A+26   |        |        | Entry 2
             |        |        |
             +--------+--------+
             |        |        |       .
             |        |        |       .
             |        |        |       .
             +--------+--------+
             |        |        |
                                   Entry B-1
             |        |        |
             +--------+--------+
             |        |        | Offset of
  A+2+B*12   |    - C -    +     Next IFD
             |        |        |
             +--------+--------+
                      |
                      V
               (next IFD)
```
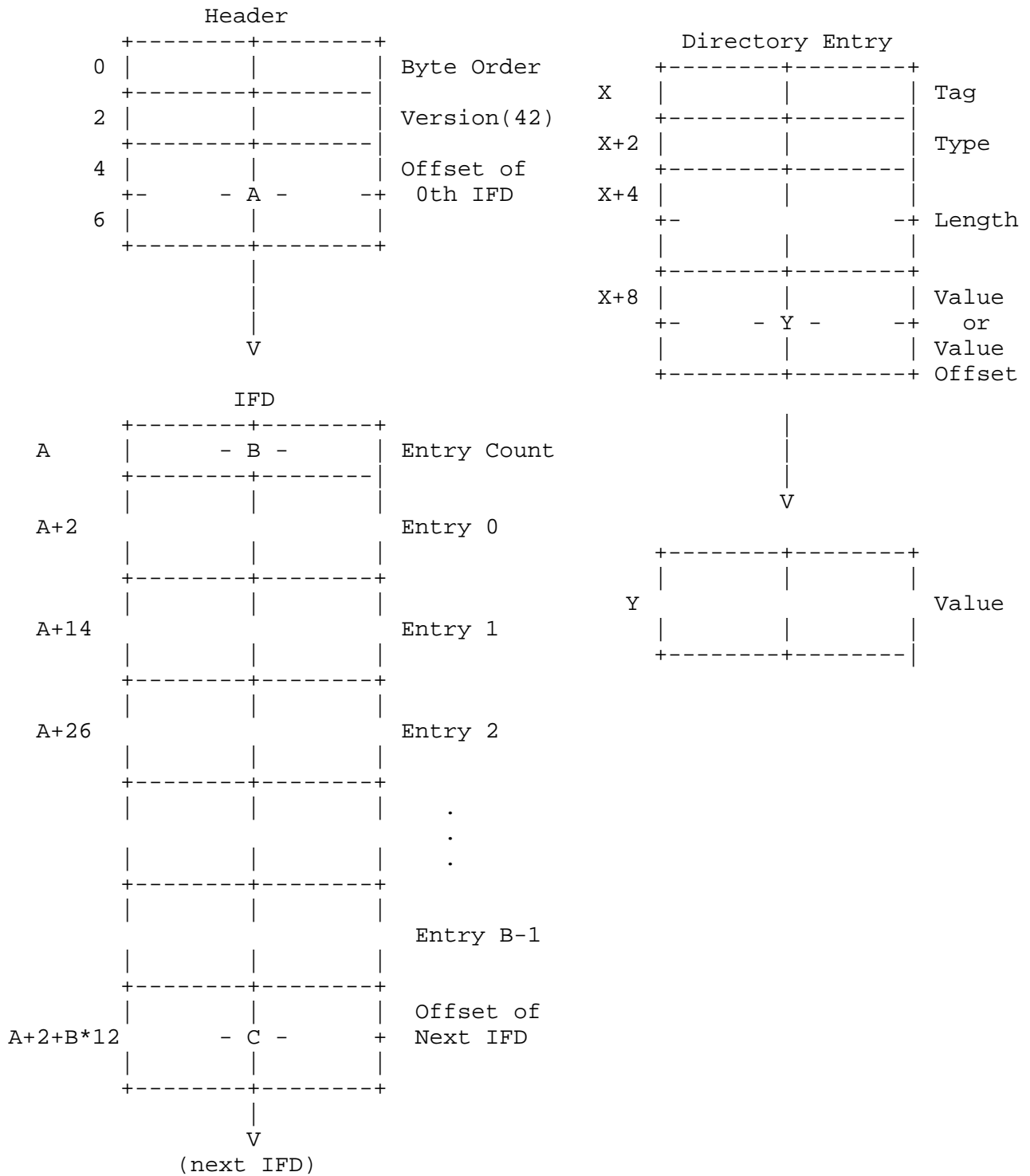
                Figure 1: The Structure of a TIFF File

3.B. Image Format and Encoding Issues

   Images in TIFF files are organized as horizontal strips for fast
   access to individual rows. One can specify how many rows there are
   in each strip and all of the strips are the same size (except
   possibly the last one). Each strip must begin on a byte boundary but
   successive rows are not so required. For two-dimensional G3
   compression (MR), each strip must begin with an "absolute" one-
   dimensional line. For MMR (G4) compression, each strip must be
   encoded as if it were a separate image.

   For a variety of reasons, each page must be a single strip (e.g., not
   broken up into multiple strips).

   One problem with multiple strips per page is that images which come
   from G4 fax machines as well as most scanned images will be generated
   as a single strip per page. These would have to be decoded and re-
   encoded as multiple strips (remember that for MMR compression, each
   strip must be start with a one-dimensionally encoded line).

   Another problem with multiple strips per page arises in MR
   compression. Here, there MAY be at most k-1 two-dimensionally
   encoded lines following a one-dimensionally encoded line, but this is
   not required. It is possible to have one-dimensional lines more
   frequently than every k lines. However, since each strip (except
   possibly the last one) is required to be the same size, it may be
   necessary to re-encode the image to insure that each strip starts
   with a one-dimensional line. This is not a problem if each page is a
   single strip.

        [NOTE: The TIFF document [3] suggests using strips which
        are about 8K bytes long. However, TIFF-F [4] recommends
        that each page be a single strip regardless of its size.
        The format specified in this document follows the TIFF-F
        recommendation.]

   Also, as TIFF-F recommends, all G3 encoded images (MH and MR) should
   be "byte-aligned." This means that extra zero bits (fill bits) are
   added before each EOL (end-of-line) so that every line starts on a
   byte boundary.

   In addition, as in the TIFF-F specification, the RTC (Return to
   Control signal which consists of 6 continuous EOL's) of G3 shall not
   be included at the end of G3 encoded documents. RTC is to be
   considered part of the G3 transmission protocol and not part of the
   encoding. Most, if not all, G3 fax modems attach RTC to outgoing
   images and remove it from incoming ones.

For MMR (G4) encoded files, readers should be able to read images
with only one EOFB (End Of Facsimile Block) at the end of the page
and should not assume that Facsimile Blocks are of any particular
size. (It has been reported that some MMR readers assume that all
Facsimile Blocks are the maximum size.)

Systems may optionally choose to store the entire image uncompressed
if the compression increases the size of the image file. Also,
uncompressed mode (specified in Group3Options or Group4Options, see
below) allows portions of the image to be uncompressed.

The multi-page capability of TIFF is supported and should be used for
multi-page documents. TIFF files which have multiple pages have an
IFD for each page of the document each of which describes and points
to a single page image. (Note: though the current TIFF specification
does not specifically prohibit having a single IFD point to an image
which is actually multiple pages, with one strip for each page, most
if not all TIFF readers would probably not be able to read such a
file. Therefore, this should not be done.)

   [A NOTE ON TIFF AND MULTI-PAGE DOCUMENTS:

      Since most publications (e.g., reports, books, and
      magazine articles) are composed of more than a single
      page, multi-page TIFF files should be used where
      appropriate. However, many current TIFF implementations
      now only handle single-page files.

      It is hoped that in the future, more TIFF implementations
      will handle multi-page files correctly. In the meantime,
      it would be useful to develop a utility program which
      could join several single-page TIFF files into a single
      multi-page file and also separate a multi-page TIFF file
      into several single page files.

      For example, the utility could take a single TIFF file
      with N pages, called doc.tif, and create the files
      doc.000, doc.001, doc.002, ..., doc.N. doc.000 would be
      an ASCII listing of the files created. This naming
      scheme is compatible with that used by the image systems
      we have seen which only handle single page files.

      In going the other way, the N+1 single page files could
      be combined into a single multi-page TIFF file. In this
      case, if the file doc.000 exists but contains information
      contrary to what is found in looking for the files
      doc.001, doc.002, ..., the program would notify the user.]

3.C. TIFF Fields

   TIFF is tag or field based. The various fields and their format are
   listed in [3]. There are Basic Fields (common to all TIFF files),
   Informational Fields (which provide useful information to a user),
   Facsimile Fields (used here), and Private Fields.

   Each directory entry contains:

       The Tag for the field (2 bytes)

       The field Type (2 bytes)

       The field Length (4 bytes)
           (This is in terms of the data type, not in bytes. For
            example, a single 16-bit word or SHORT has a Length
            of 1, not 2)

       The Value Offset (4 bytes)
           (Pointer to the actual value, which must begin on a
            word boundary. Therefore, this offset will always
            be an even number. If the Value fits into 4 bytes, the
            Value Offset contains the Value instead. If the Value
            takes less than 4 bytes, it is left justified)

   The allowed types and their codes are:

        1 = BYTE   8-bit unsigned integer (1 byte)

        2 = ASCII   8-bit ASCII terminated with a null (variable
                        length)

        3 = SHORT   16-bit unsigned integer (2 bytes)

        4 = LONG   32-bit unsigned integer (4 bytes)

        5 = RATIONAL   Two LONGs (64 bits) representing the
                        numerator and denominator of a fraction.
                        In this document, RATIONAL's will be written
                        as numerator/denominator. (8 bytes)

   For ASCII, the Length specifies the number of characters and includes
   the null. It does not, however, include padding if such is
   necessary.

   (Note that ASCII strings of length 3 or less may be stored in the
   Value Offset field instead of being pointed to.)

The following fields should be used in a TIFF image file. Only the
Basic Fields are mandatory; the others are optional (except that for
MH and MR encoded files, the Group3Options Facsimile Field is
mandatory). The optional fields have default values which are given
in the TIFF specification. (Note that the TIFF reference [3]
recommends not relying on the default values.)

Some fields contain one or more flag bits all stored as one value.
In these cases, the bit labeled 0 is the least significant bit (i.e.,
Little Endian order). Where there is more than one suggested value
for a tag, the possible values are separated by |.

Note that some fields (such as ImageLength or ImageWidth) can be of
more than one type.

It would be useful to develop a TIFF viewer and editor which would
allow one to read, add, and edit the fields in a TIFF file. Such an
editor would display fields in sorted order and force the inclusion
of all mandatory fields. Also, resolution and position should always
be displayed or specified together with their units.

3.C.1. Basic Fields (Mandatory)

   Basic Fields are those which are fundamental to the pixel
   architecture or visual characteristics of an image. The following
   Basic Fields should be included in a TIFF image file:

```
      FIELD NAME
    (TAG in hex, TYPE)      VALUE            DESCRIPTION
    ------------------      -----            -----------

      BitsPerSample           1             Number of bits
      (0102, SHORT)                         per pixel (bi-level for
                                            now, but may allow
                                            more later)

      Compression             4             Type of Compression
      (0103, SHORT)      (could also be      1 = Uncompressed
                          1 or 3)            3 = G3 (MH or MR)
                                             4 = G4 (MMR)
                                           Use 4 if possible

      ImageLength        <image's length>   Length of the Image
      (0101, SHORT                             in scan lines
        or LONG)

      ImageWidth         <image's width>    Width of the Image
      (0100, SHORT                            in pixels
```

```
        or LONG)

    NewSubFileType        0 usually          Flag bits indicating
    (00FE, LONG)           bit 0: 1 if        the kind of image.
                           reduced           (see the TIFF
                           resolution of      reference [3])
                           another image
                           bit 1: 1 if
                           single page of a
                           multi-page image
                           bit 2: 1 if
                           image defines a
                           transparency
                           mask

    Photometric-          0 for positive
      Interpretation       image (0 imaged
      (0106, SHORT)        as white, 1 as
                           black)
                          1 means reverse
                           black and white

    RowsPerStrip         <Number of Rows>    Number of Rows in
      (0116, SHORT                           Each Strip.  Each
       or LONG)                              page should be a
                                             single strip.

    SamplesPerPixel            1             (since are Bi-level
      (0115, SHORT)                           images)

    StripByteCounts      count1, count2...   Number of Bytes in
      (0117, SHORTs                           each strip of the
       or LONGs)                              images.  (The Value
                                              is an offset which
                                              points to a series
                                              of counts, each of
                                              which is the same
                                              Type, LONG or SHORT.
                                              The Length is the
                                              same as the number
                                              of strips.)

    StripOffsets         off1, off2,...      Pointers to the strips
      (0111, SHORTs                           of the image (remember,
       or LONGs)                              one strip per page).
                                              (The Value is an offset
                                               which points to a
                                               series of offsets,
```

                                        each of which points
                                        to the actual image
                                        data for the strip.)

        ResolutionUnit   2 | 3   Units of Resolution
         (0128, SHORT)   See Below, 3.C.6   2: Inches
                                                    3: Centimeters

        XResolution   See Below, 3.C.6   Resolution in the X
         (011A, RATIONAL)   direction in pixels
                                                   per ResolutionUnit
                                                   (we suggest 400 dots
                                                   per inch when possible)

        YResolution   See Below, 3.C.6   Resolution in the Y
         (011B, RATIONAL)   direction in pixels
                                                   per ResolutionUnit
                                                   (we suggest 400 dots
                                                   per inch when possible)

   3.C.2. Informational Fields (Optional)

      The following Informational Fields are optional. They provide
      useful information to a user. All Field values are ASCII strings.

        NAME (TAG in hex)   DESCRIPTION
        -----------------   -----------

          Artist (013B)   Person Who Created the Image

          DateTime (0132)   Date and Time of Image Creation

          HostComputer (013C)   Name of Computer Image was Created On

          ImageDescription   A Short Text Description
            (010E)

          Make (010F)   Manufacturer of Hardware (Scanner) Used

          Model (0110)   Model Number of Hardware (Scanner) Used

          Software (0131)   Software Package that Created the Image

   3.C.3. Facsimile Fields (Optional, Mandatory for G3 Compression)

      In addition to the above, the Facsimile Fields below should be
      used. The TIFF document recommends that they not be used for
      interchange between applications, but they are now in wide enough

use for just that. These fields are optional and default to 0
(all bits off).

FIELD NAME (TAG in hex, TYPE)	VALUE	DESCRIPTION
Group3Options (0124, LONG)	bit 0: 1 for 2-dimensional coding (i.e., MR with k > 1) bit 1: 1 if uncompressed mode MAY be used, 0 if uncompressed mode IS NOT used.	Flag bits indicating Options for G3
	bit 2: 1 if fill bits have been added	(As allowed by the G3 protocol, fill bits may be added between each line of data and the EOL. Since fill bits are used to "byte-align" G3 image files, bit 2 should be set to 1 for these images.)
Group4Options (0125, LONG)	bit 0: unused bit 1: 1 if uncompressed mode MAY be used, if this bit is 0 it means that uncompressed mode IS NOT used.	Flag bits indicating Options for G4

3.C.4. Storage and Retrieval Fields (Optional)

   The following fields are optional and may be useful for document
   storage and retrieval.

```
        FIELD NAME
     (TAG in hex, TYPE)                 DESCRIPTION
     ------------------                 -----------

     DocumentName                  Name of the Document
        (010D, ASCII)

     PageName                      Name of the Page
        (011D, ASCII)

     PageNumber                    Page Number in a Multi-Page Document
        (0129, SHORTs)             Two SHORT Values are specified, the
                                   first is the page number and the
                                   second is the total number of pages
                                   in the document.  The first page
                                   is page 0.  (NOTE:  This does not
                                   necessarily correspond to page
                                   numbers which may be printed
                                   in the image.)

     XPosition                     X Offset of the Left Side of
        (011E, RATIONAL)           the Image, in ResolutionUnits

     YPosition                     Y Offset of the Top of
        (011F, RATIONAL)           the Image, in ResolutionUnits
```

   3.C.5. TIFF-F Fields (NOT Recommended)

   TIFF-F defines the following new fields for G3 (MH) encoded
   images. Since these fields are not defined in TIFF-B itself,
   their use is not recommended. However, since TIFF-F files may
   include these tags for image data which came from a G3 fax
   machine, readers should be prepared for them.

   These three fields deal with corrupted image data which is due to
   the fact that G3 devices may not perform error correction on bad
   data.

```
        FIELD NAME
     (TAG in hex, TYPE)                 DESCRIPTION
     ------------------                 -----------

     BadFaxLines                   Number of Bad fax scan lines
        (0146, SHORT or LONG)      encountered during fax reception
                                   (but not necessarily in the file)

     CleanFaxData                  0 means no bad lines received
        (0147, SHORT)              1 means bad lines were regenerated
```

                               by the receiving device
                          2 means bad lines were detected
                              but not regenerated

        ConsecutiveBadFaxLines   The maximum number of consecutive
          (0148, SHORT or LONG)   bad fax lines (but not necessarily
                                      in the file)

   3.C.6. More on Representing Resolutions

      The tags XResolution and YResolution are both RATIONALs, i.e., the
      ratio of two LONGS. G3 fax resolutions are actually specified in
      dots (or lines) per mm while G4 is in dots per inch (actually,
      dots per 25.4 mm).

      For example, G3 horizontal resolution is defined to be 1728 dots
      per 215 mm which comes out to 80.4 dots per cm or about 203 dots
      per inch. It is frequently referred to as just 200 dpi. To avoid
      any possibility of problems due to round off error, this should be
      represented by having XResolution = 17280/215 and ResolutionUnit =
      3 (cm). However when reading, 204/1 or even 200/1 with
      ResolutionUnit = 2 (inches) should be recognized as representing
      the same resolution.

      For G4, on the other hand, the resolution 400 dots/inch should be
      represented by an XResolution of 400/1 and ResolutionUnit = 2.

      The following table shows various ways of representing the
      standard resolutions in order of preference:

	ResolutionUnit	XResolution	YResolution
G3 normal	3	17280/215	3850/100
	3	80/1	3850/100
	3	17280/215	385/10
	3	80/1	385/10
	2	2042/10	9779/100
	2	204/1	98/1
	2	200/1	100/1
G3 fine	3	17280/215	77/1
	3	80/1	77/1
	2	2042/10	19558/100
	2	204/1	196/1
	2	200/1	200/1

```
     G4 200 dpi       2                200/1              200/1

     G4 300 dpi       2                300/1              300/1

     Other 300 dpi    2                300/1              300/1

     G4 400 dpi       2                400/1              400/1

     600 dpi          2                600/1              600/1
```

   It is suggested that Image readers be able to handle all of the
   above representations.

4. A Sample TIFF Image File

   Below is a sample of what might be in a TIFF file for an MMR (G4)
   encoded single image which is about 100K bytes compressed at 400 dpi.
   A generic outline is given first, followed by a more detailed hex
   listing.

4.A. Sample File

   Comments are to the right and are preceded by a semicolon. Note that
   tags must be sorted in order of the tag codes.

   0:, IFDADDR:, and STRIP0: are addresses within the file and denote
   the number of bytes from the beginning of the file.

   Header:

```
  0:  Byte Order=      hex 4D4D         ;first bytes of the file, from
                                        ;most significant bit to least
                                        ;significant (big endian)
      Version=         42 (hex 002A)    ;Must be 42
      First IFD=       IFDADDR          ;Address of first (and only) IFD
```

   Image File Directory (the only one in this example):

   IFDADDR:

```
      IFD Entry Count=      24          ;(NOT A TAG) Count of
                                        ; Number of IFD Entries


      NewSubFileType=        0
      ImageWidth=         3400          ;8.5 inches at 400 dpi
      ImageLength=        4400          ;11 inches at 400 dpi
      BitsPerSample=         1          ;Bi-Level
```

```
        Compression=            4           ;MMR
        Photometric-
           Interpretation=      0
        DocumentName=           "LAMap1"
        ImageDescription=   "A map of Los Angeles"
        Make=                   "Fujitsu"
        Model=                  "M3093E"
        StripOffsets=           <STRIP0>    ;There is only one strip in
                                            ;this example.  However, note
                                            ;that strips can be in any
                                            ;order.  (Offsets are from the
                                            ;beginning of the TIFF file.)

        SamplesPerPixel=        1           ;Bi-Level
        RowsPerStrip=           4400        ;Entire image in 1 strip
        StripByteCounts=        <COUNT0>    ;Byte count of entire
                                            ;compressed image

        XResolution=            400/1
        YResolution=            400/1
        XPosition=              0/1         ;position of left side of image
        YPosition=              0/1         ;position of top of image
        Group4Options=      hex 00000002    ;bit 1 on means uncompressed
                                            ;mode MAY be used

        ResolutionUnit=         2           ;Inches
        Software=               "Xionics"
        DateTime=               "1990:10:05 15:00:00"
        Artist=                 "Joe Pro"
        HostComputer=           "Tardis.Isi.Edu"

        Next IFD Pointer=   hex 00000000 ;(NOT A TAG) Indicates no
                                            ; more IFDs in this file

     Image Data:

     <STRIP0>:       <actual compressed image data>

     [end of TIFF file]
```

   In this example there is only one strip. Note that if there were
   more than one, the TIFF specification does not require them to be in
   any particular order. Strips may be given in any order and TIFF
   readers must use the StripOffsets to locate them.

   Also, the TIFF document recommends not relying on the default values
   of the tags.

4.B. Detailed Hex Listing

   All offsets and values are represented by hex except for ASCII
   strings which are double quoted. Remember that Value Offsets must
   always be an even number since the value it points to must always be
   on a 16-bit word boundary.

   Entries in the Name column are for reference and are not actually a
   part of the TIFF file.

```
  Offset          Name                  Value
  ----            ------------------    ------------------------------------
Header (first byte is Offset 0):
  0000            Byte Order            4D4D
  0002            Version               002A
  0004            1st. IFD pointer      00000010

IFD (IFDADDR from above is 0010 here):
  0010            Entry Count           0018
  0012            NewSubFileType        00FE   0004   00000001   00000000
  001E            ImageWidth            0100   0004   00000001   00000D48
  002A            ImageLength           0101   0004   00000001   00001130
  0036            BitsPerSample         0102   0003   00000001   00010000
  0042            Compression           0103   0003   00000001   00040000
  004E            Photometric Interp.   0106   0003   00000001   00000000
  005A            DocumentName          010D   0002   00000007   00000136
  0066            ImageDescription      010E   0002   00000015   0000013E
  0072            Make                  010F   0002   00000008   00000154
  007E            Model                 0110   0002   00000007   0000015C
  008A            StripOffsets          0111   0004   00000001   000001A8
  0096            SamplesPerPixel       0115   0003   00000001   00010000
  00A2            RowsPerStrip          0116   0004   00000001   00001130
  00AE            StripByteCounts       0117   0004   00000001   <COUNT0>
  00BA            XResolution           011A   0005   00000001   00000164
  00C6            YResolution           011B   0005   00000001   00000164
  00D2            XPosition             011E   0005   00000001   0000016C
  00DE            YPosition             011F   0005   00000001   0000016C
  00EA            Group4Options         0125   0004   00000001   00000002
  00F6            ResolutionUnit        0128   0003   00000001   00020000
  0102            Software              0131   0002   00000008   00000174
  010E            DateTime              0132   0002   00000014   0000017C
  011A            Artist                013B   0002   00000008   00000190
  0126            HostComputer          013C   0002   0000000F   00000198
  0132            Next IFD Pointer      00000000

 Fields Offsets Point to:
  0136            DocumentName          "LAMap1"
  013E            ImageDescription      "A map of Los Angeles"
```

```
   0154          Make                "Fujitsu"
   015C          Model               "M3093E"
   0164          X,Y Resolution      00000190 00000001
   016C          X,Y Position        00000000 00000001
   0174          Software            "Xionics"
   017C          DateTime            "1990:10:05 15:00:00"
   0190          Artist              "Joe Pro"
   0198          HostComputer        "Tardis.Isi.Edu"

 Image Data (<STRIP0> from above is here 01A8)
   01A8          Compressed Data for single strip, of length <COUNT0> bytes

    [end of TIFF file]
```

NOTE: Since in this example there is only a single strip, there is only
       one count for StripByteCounts and one offset for StripOffsets.
       Thus, each of these only takes 4 bytes and will fit in the
       Value Offset instead of being pointed to.

5. Conclusions

   Bitmapped images transferred within the Internet should be in the
   following format:

        1. The file format should be TIFF-B with multi-page files
           supported. Images should be encoded as one TIFF strip
           per page.

        2. Images should be compressed using MMR when possible. Images
           may also be MH or MR compressed or uncompressed. If MH or MR
           compression is used, scan lines should be "byte-aligned".

        3. For maximum interoperability, image resolutions should
           either be 600, 400, or 300 dpi; or else be one of the
           standard Group 3 fax resolutions (98 or 196 dpi
           vertically and 204 dpi horizontally).

   Note that this specification is self contained and an implementation
   should be possible without recourse to the TIFF references, and that
   only the specific TIFF documents cited are relevant to this
   specification. Updates to the TIFF documents do not change this
   specification.

   Existing commercial off-the-shelf products are available which can
   handle images in the above format. ISI would be delighted to help
   those interested in assembling a system.

6. Acknowledgments

   Many contributions to this work were made by members of the IETF
   Network Fax Working Group especially by its chairman, Mark Needleman
   and by Clifford Lynch of the University of California Office of the
   President, Library Automation. Also, Kiyo Inaba of Ricoh Co. Ltd.
   made a number of helpful suggestions.

7. References

   [1] Borenstein, N., and N. Freed, "Mechanisms for Specifying and
       Describing the Format of Internet Message Bodies", RFC in
       preparation.

   [2] International Telegraph and Telephone Consultative Committee
       (CCITT), Red Book, October, 1984.

   [3] Aldus Corp., Microsoft Corp., "Tag Image File Format
       Specification", Revision 5.0, Final, 1988.

   [4] Cygnet Corporation, "The Spirit of TIFF Class F, 1990", available
       from Cygnet Technologies, 2560 9th., Suite 220, Berkeley, CA
       94710, FAX: (415) 540-5835.

   [5] Welch, T., "A Technique for High Performance Data Compression",
       IEEE Computer, Vol. 17, No. 6, Page 8, June 1984.

8. Security Considerations

   While security issues are not directly addressed by this document, it
   is important to note that the file format described in this document
   is intended for the communications of files between systems and
   across networks. Thus the same precautions and cares should be
   applied to these files as would be to any files received from remote
   and possibly unknown systems.

9. Authors' Addresses

   Alan Katz
   USC Information Sciences Institute
   4676 Admiralty Way #1100
   Marina Del Rey, CA 90292-6695

   Phone: 310-822-1511
   Fax: 310-823-6714
   EMail: Katz@ISI.Edu

   Danny Cohen
   USC Information Sciences Institute
   4676 Admiralty Way #1100
   Marina Del Rey, CA 90292-6695

   Phone: 310-822-1511
   Fax: 310-823-6714
   EMail: Cohen@ISI.Edu