INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

T.44

Amendment 1

(02/2000)

SERIES T: TERMINALS FOR TELEMATIC SERVICES

Mixed raster content (MRC)

Amendment 1: Accommodation of new Annex B

ITU-T Recommendation T.44 – Amendment 1

(Previously CCITT Recommendation)

ITU-T T-SERIES RECOMMENDATIONS

TERMINALS FOR TELEMATIC SERVICES

For further details, please refer to ITU-T List of Recommendations.

ITU-T RECOMMENDATION T.44

MIXED RASTER CONTENT (MRC)

AMENDMENT 1

Accommodation of new Annex B

Summary

Amendment 1 comprises a new Annex B which defines provisions for sharing resources across pages, stripes and layers and provisions for using colour tags as a means of representing text colour, as well as modifications to the main body, and Annex A of Recommendation T.44, required for introducing Annex B. These provisions play an essential role in the application of JBIG2 (i.e. Recommendation T.88). JBIG2 realizes significant compression gains by sharing symbol dictionaries (i.e. a shared resource) across pages and page components. Additionally, the colours of JBIG2 encoded text may be represented by colour tags to realize even greater compression gains over conventional bitmap representation of the text colours.

In addition, this amendment includes corrections for two technical oversights that were detected as a result of implementation experience.

Source

Amendment 1 to ITU-T Recommendation T.44, was prepared by ITU-T Study Group 8 (1997-2000) and was approved under the WTSC Resolution No. 1 procedure on 10 February 2000.

FOREWORD

ITU (International Telecommunication Union) is the United Nations Specialized Agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of the ITU. The ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Conference (WTSC), which meets every four years, establishes the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

The approval of Recommendations by the Members of the ITU-T is covered by the procedure laid down in WTSC Resolution No. 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

INTELLECTUAL PROPERTY RIGHTS

The ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. The ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, the ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementors are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database.

CONTENTS

MIXED RASTER CONTENT

AMENDMENT 1

Accommodation of new Annex B

(Geneva, 2000)

A) Modifications to the main body, and Annex A of Recommendation T.44, required for introducing Annex B

1) Subclauses 9.1 and A.9.1

Add the following new paragraphs to the end of each subclause:

A distillation of the bit and byte ordering conventions of Annex B/T.81 that are applied throughout this Recommendation follows:

Bits are packed into octets starting at the most significant bit. If a decoder is reading a sequence of bits from a bit-stream, it shall first read the most significant bit of the first octet, the next most significant bit, and so on, then proceed to the next octet.

All multi-octet values shall be interpreted in a most-significant-first manner: the first octet of each value is most significant, and the last octet is the least significant.

2) Subclause 9.2

Add the following new paragraphs:

Marker segments defined in this Recommendation and located between the Start of Page Marker Segment (SOP) and EOP marker shall have the following structure, which is consistent with that of the SOP:

APP13 (X'FFED'), Length of segment, identifier, Optional length (if needed), parameters and/or segment data;

where the Optional length is included when two octets are insufficient to represent the Length of segment and four octets are needed.

In typical use, the 2-octet Length of segment should be sufficient to represent the length of the marker segment header and data, not including the 2-octet APP13 (X'FFED') marker itself; in which case the Optional length will be omitted. In cases where two octets are insufficient, the 2-octet Length of segment will take on a value of zero (0), and the Optional length will be used. If the 2-octet Length of segment is any value less than six (which is the clear minimal size of a marker segment for both the 2-octet length and the 4-octet identifier), then the Optional length is required. Values between one and five are reserved for future use.

All APP13 (X'FFED') markers currently adhere to this rule with the exception of EOH, which doesn't include the data length in the Length of segment, but includes that length separately as the only parameter in the marker segment header.

3) Subclause 9.2.1

Modify Tables 1/T.44 and 2/T.44 as follows:

Table 1/T.44 – Mask (even-numbered layer) coder octet(s)

Octet bit number	Coder used
LSB 0	One dimensional T.4 (MH) coding
1	Two dimensional T.4 (MR) coding
2	T.6 (MMR) coding
3	T.82 (JBIG1) coding applying Recommendation T.85
4	T.88 (JBIG2) coding, Annex B/ T.44 is required~~Reserved~~
5	Reserved
6	Reserved
MSB 7	Extend, add another octet that follows immediately
NOTE – New bi-level coders (i.e. a ~~5th,~~ 6th and 7th coder) would be assigned bit numbers ~~4,~~ 5 and 6 respectively. Bit 7, the extend bit, would be set when adding another octet to accommodate additional coders, such as an 8th which would be assigned to bit number 8.	

Table 2/T.44 – Image (odd numbered layer) coder octet(s)

Octet bit number	Coder used
LSB 0	T.81 (JPEG) coding
1	T.82 (JBIG1) coding applying Recommendation T.43
2	T.45 "Run-length Colour Encoding", Annex B/T.44 is required[1]~~Reserved~~ (Note 1)
3	Reserved
4	Reserved
5	Reserved
6	Reserved
MSB 7	Extend, add another octet that follows immediately
NOTE 1 – Coding scheme(s) referencing this note, shall use the SLC (Start of Layer Coded data) marker segment, defined in the Layer Data Structure clause of Annex A/T.44. This means that Mode 1 shall not be used with the referencing coding scheme.	
NOTE 2 – New multi-level coders (i.e. ~~3rd~~ 4th through 7th coder) would be assigned bit numbers ~~2~~ 3 through 6 respectively. Bit 7, the extend bit, would be set when adding another octet to accommodate additional coders, such as an 8th which would be assigned to bit number 8.	

4) Subclauses 9.2.1 and A.9.2.1

a) *Modify the* "Version" *parameter of both these subclauses to reflect the new T.44 version as follows:*

Version: 1 octet Revision number, X'0~~0~~1' indicting revision "~~0~~1".

b) *Correct the typographical error of the* "Version" *parameter by changing*

indicting

to:

indicating

c) "Mask coders" *parameter:*

i) change the name of the parameter from:

Mask coders

to:

Mask layer coders

ii) Change the last sentence as follows:

The value shall be fixed to zero "0" in the event that there is no mask layer coder (i.e. no coded mask layer data present).

d) Change the second sentence of the "Page width" *parameter as follows:*

For pages with two or more layers, the main mask layer image width defines the page width in units of main mask resolution.

to:

For pages with two or more layers, the main mask layer image width defines the page width ~~in~~ using units of the main mask resolution.

5) Subclause 9.2.2

Modify the 2nd, 3rd and 4th paragraphs of this subclause as follows:

Optional marker segments (OMSx) consist of marker and associated parameters. The APP13 (X'FFED') marker initiates identification of the entry. Each optional marker segment is identified by the 3-octet ASCII string plus a hexadecimal count for 'MRCn'. The 'MRCn' identifier is a 4-octet value X'4D',X'52',X'43',X'n', where n may equal~~s~~ X'0A' (10) to a maximum of X'FE' (254). The optional marker segments are located after the Termination Number (TN).

Each optional marker segment (OMSx) has the following structure:

APP13 Marker (X'FFED'), Length of entry, OMSx ident (MRCn), Optional length (if needed), entry parameters and/or~~entry~~ data.

6) Subclause 9.3

Modify the note to the note box of Table 3 to read as follows:

NOTE – Refer to Annex A of this Recommendation for stripes with 4 or more layers. Layers above seven (7) would require an additional octet for representation. Bit 7, the extend bit, would be set when adding another octet to accommodate an additional layer such as Layer 8 which would be represented by bit 8.

7) Subclause A.9.1

Add the following new paragraph to the end of this subclause:

Markers and/or marker segments used in association with the coding methods (i.e. encoder marker and/or marker segments) may be defined outside of this Recommendation (i.e. foreign encoder markers and/or marker segments). Foreign encoder markers and/or marker segments may be located within or outside of the data stream. A foreign encoder marker located outside the data stream shall be of the APPn form (i.e. an X'FF octet followed by an octet not equal to X'00' or X'FF' and optionally preceded by extra X'FF' octet codes). Structure of a foreign encoder marker segment located outside the data stream shall be as follows:

APPn, Length of segment, identifier, parameter and/or data.

8) Subclause A.9.5

Modify this subclause as follows:

Layers are coded using ITU-T coding methods indicated in the Start of Page marker segment. A Start of Layer Coded Data (SLC) marker segment precedes the coded layer data. Parameters of the SLC include layer number, coder, resolution, coded image width and height, layer base colour and layer offset. One or more marker segments that contain encoding related parameters might follow the SLC. New encoder related marker segments may be defined as determined by encoding needs. They may be defined within or outside of this Recommendation. Those defined outside of this Recommendation are frequently referenced as foreign encoder marker segments. End of Header (EOH) marker segment terminates the SLC. The EOH contains the coded data length (octet count) of the layer. Encoder marker segments shall be located between the SLC and EOH. The resolutions of all layers are restricted to ITU-T recommended integral factors of the main mask resolution. For example, if the main mask resolution is 400 pels/25.4 mm, the other layers may each be either 100, 200 or 400 pels/25.4 mm.

9) Subclause A.9.5.2

a) Modify the 2nd, 3rd and 4th paragraphs as follows:

Encoder related marker segments consist of marker and associated parameters/data. The APP13 (X'FFED') marker initiates identification of each entry. Each encoder related marker segment is identified by the 3-octet ASCII string plus an hexadecimal count for 'MRCn'. The 'MRCn' identifier is a 4-octet value X'4D',X'52',X'43',X'n', where n may equal~~s~~ X'0C' (12) to a maximum of X'FE' (254).

Each encoder related marker segment has the following structure:

APP13 Marker (X'FFED'), Length of entry, encoder marker segment ident (MRCn), Optional length (if needed), parameters and/or data.

b) Replace the last paragraph as follows:

~~"Definition of encoder marker segments is for further study"~~Specific encoder marker segments are defined elsewhere within this Recommendation or outside of this Recommendation. A foreign encoder marker segment shall also have the structure:

APPn, Length of segment, identifier, parameter/data; where the APPn marker consist of an X'FF' octet followed by an octet not equal to X'00' or X'FF' and optionally preceded by extra X'FF' octet codes.

B) New Annex B

Add the following new Annex B

Annex B

MRC Mode 4 – Shared resources and colour tags

Introduction and background

Improved compression, in terms of both reduced size and less error for lossy methods, is achieved when the compression method closely models and matches the data to be compressed. This is giving rise to a new wave of compression methods that have explicit models of some types of data. MPEG4 includes support for describing objects moving over a static background, something that is seen in video images of the world. ITU-T Rec. T.88 ISO/IEC 14492 (JBIG2) represents scanned bi-level data by segmenting it into text, halftone, and other regions, and then using separate specialized compression methods to store these regions. The text regions are compressed by extracting symbols (individual text characters), and forming symbol dictionaries. The same symbol shapes (each one representing a text character from a certain font in a certain size) are used across multiple text regions and pages, to improve compression. Halftone regions are similarly represented using dictionaries of halftone patterns.

Using these dictionaries, JBIG2 can achieve a large increase in compression relative to other bi-level image compression methods: 3-5 times greater compression than T.82 (JBIG1) or T.6 (MMR) is typical, and factors of over 20 times better compression than MMR have been observed.

Of course, in order to achieve these large compression factors, each piece of data must be used to the maximum possible extent. This means that a single symbol dictionary must be used by multiple pages wherever possible, which necessarily has consequences for any system using JBIG2, as most systems have considered pages to be completely independent entities.

An imaging model using JBIG2, such as MRC, should incorporate provisions for the use of shared data. This entails: having some way to define a shared resource, to be used by multiple coded entities (pages, stripes or layers); referring to that shared resource at the point where it is to be used; and later instructing the decoder that the resource is no longer required and may be flushed from memory. The shared data marker segment (SDM), introduced in this annex, is intended to provide this functionality.

It should be noted that SDM is not restricted to JBIG2; its structure is flexible and could be used for other coding methods. For example, a set of JPEG Huffman tables could be stored in a shared resource and then used by multiple JPEG-coded layers, reducing file size. Similarly, a palette table could be defined once, then used by multiple T.43-coded layers.

Another opportunity that is afforded by JBIG2 is improved compression of the foreground layer for documents containing coloured text. In most cases, if a document contains text, each individual text character is a single, flat, colour (e.g. black or red), and the number of such colours is limited. The foreground layer in this case looks like a number of coloured 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 and/or halftone regions, then decoding it essentially yields a sequence of (XPositon, YPosition, Symbol ID) triples. 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 colour of that individual character (sometimes called the symbols "colour tag"), allows storage of the foreground layer in a very small amount of space: using run-length coding on those colours. The total space taken by the foreground layer can be as small as a few tens of bytes.

For example, if the mask layer contained two characters, an "R" in red and a "B" in blue, then the mask layer would decompress to:

(100, 0, "R")

(120, 0, "B")

and the foreground layer would decompress to:

(#7AD29C) [corresponding to CIELAB (48.0, 65.5, 48.0) using default gamut range]

(#3A9B1D) [corresponding to CIELAB (23.1, 20.4, –52.1) using default gamut range]

or some other suitable representation of the colours, such as indexes into a palette. Matching up the "R" symbol with the colour #7AD29C and drawing the symbol's shape in red gives the correct result. This is a single drawing operation, and is extremely efficient.

Storing the foreground layer in this manner, using colour tags, allows very compact representation, and efficient decoding. However, since the mask layer is transmitted before the foreground layer, the decoder needs to be warned that the upcoming foreground layer is simply a list of colours (one per JBIG2 symbol in the mask layer), not a complete image. For this reason, we need to put a flag in the mask layer that warns the decoder "foreground is compressed using colour tags". The decoder can then defer drawing the mask until the foreground layer has also been decoded.

B.1 Scope

This annex defines Mode 4 to Recommendation T.44, extending the MRC model to accommodate shared data and provisions for colour tags. The provisions of Mode 4 shall use the Mode 3 structure for implementation. Applications implementing Mode 4 shall support Modes 1, 2 and 3.

B.2 References

The references of the main body of this Recommendation apply, plus the following.

– ITU-T Recommendation T.45 (2000), *Run-length colour encoding.*

– ITU-T Recommendation T.88 (2000) | ISO/IEC 14492:2000, *Information technology – Lossy/lossless coding of bi-level images*. (Commonly referred to as JBIG2 standard.)

– ITU-T Recommendation T.89[1], *Application profiles for Recommendation T.88 – Lossy/lossless coding of bi-level images (JBIG2) for facsimile apparatus*.

B.3 Definitions

The definitions in Annex A of this Recommendation apply, plus the following additional definitions:

B.3.1 Create shared data marker segment (SDMc), encoded as APP13 (X'FFED'), Length of segment, SDM ident (MRC3), Optional length (if needed), parameters, shared data.

B.3.2 Colour-interpreter encoder marker segment (CLIe), encoded as APP13 (X'FFED'), Length of segment, CLIe ident (MRC13), parameters.

B.3.3 Disposition shared data marker segment (SDMd), encoded as APP13 (X'FFED'), Length of segment, SDMd ident (MRC4), parameters.

B.3.4 Generic region: a region that codes pixels individually or in runs — a non-text or non-halftone region.

B.3.5 Halftone region: a region containing halftone patterns that is coded by drawing a set of patterns into a bitmap, placing the patterns according to a halftone grid.

B.3.6 Meta-data: coding data external to the coded data stream that is required in the interpretation of the data stream and may be shared between pages and other document entities.

B.3.7 JBIG2 encoder marker segment (JB2e), encoded as APP13 (X'FFED'), Length of segment, JB2e ident (MRC12), parameters.

B.3.8 Joint Bi-level Image Experts Group (JBIG), and also shorthand for the encoding methods, JBIG1 and JBIG2 described in Recommendations T.82 and T.88 respectively, which were defined by this group.

[1] Presently at the stage of draft.

B.3.9 Refinement region: a region that codes pixels by modifying a reference bitmap to produce an output bitmap.

B.3.10 Text region: a region containing text characters that is coded by drawing a set of symbol instances into a bitmap.

B.4 Shared data

JBIG2 compresses text regions by extracting symbols (individual text characters), and forming symbol dictionaries. The same symbol shapes (each one representing a text character from a certain font in a certain size) are used across multiple text regions and pages, to improve compression. Halftone regions are similarly represented using dictionaries of halftone patterns. The symbol dictionaries that are used over multiple regions and pages are referenced as shared data or shared resources. Using a single symbol dictionary over multiple pages, wherever possible, maximizes compression. The practice of using data over multiple pages contrasts with that of most systems where pages are considered to be completely independent entities.

MRC makes provision for shared resources by introducing three new functions:

1) The "create" function is used to establish a set of shared data. For future access, an ID (identification number) is assigned to the shared data when it is created/defined.

   There are four flags assigned to the "create" function to indicate the scope of application of the shared data:

   a) The "global" flag is used to indicate that the shared data is available for application over the entire document, across multiple pages.

   b) The "page" flag is used to indicate that the shared data is available for application over the rest of the current page, across multiple stripes.

   c) The "stripe" flag is used to indicate that the shared data is available for application over the rest of the current stripe, across multiple layers.

   d) The "layer" flag is used to indicate that the shared data is available for application over the current layer.

The scope flags are intended to reduce the need for the "forget" function, see item 2 below. A scope of "global" implies that the shared data are to be retained until the end of the document data stream, or until a "forget" or "use/forget" function indicates that they may be discarded, whichever comes first. A scope of "page" implies that the shared data can be discarded once the next EOP is encountered, or a "forget" or "use/forget" function combination indicates that they may be discarded, whichever comes first. A scope of "stripe" implies that the shared data can be discarded once the next SOSt is encountered, or a "forget" or "use/forget" function indicates that they may be discarded, whichever comes first. A scope of "layer" implies that the shared data can be discarded once the next SLC is encountered, or a "forget" or "use/forget" function combination indicates that they may be discarded, whichever comes first.

2) The "forget" function is used to instruct the decoder that the identified share data is no longer required and may be flushed from memory. The "forget" function may be applied to one or more sets of shared data at a time by referencing one or more shared data ID.

3) The "use" function is used to instruct the decoder to implement the identified share data in its decode operation. The "use" function may be applied to one or more sets of shared data at a time by referencing one or more shared data ID.

The "use" and "forget" functions may be used independently or in combination with each other. When used in combination, the decoder is instructed to use the identified shared data resources for that layer and then flush from memory.

B.5 Colour tags

In most cases, if a document contains text, each individual text character is a single, flat, colour (e.g. black or red), and the number of such colours is limited. The foreground layer in this case looks like a number of coloured blobs, one for each character, each one having the shape of the corresponding character.

Colour tags can be taken advantage of for documents containing coloured text based on its: improved compression of foreground image layers (odd numbered layers ≥ 3), high speed encode and decode, ease of transcode to PDLs (printer description languages). If the corresponding mask layer is compressed using JBIG2, then decoding it essentially yields a sequence of (X, Y, Symbol ID) triples. Each triple indicates that the symbol (from some dictionary) specified by "Symbol ID" should be drawn at the location "(X, Y)". Simply augmenting this triple with a fourth component, the colour of that individual character (sometimes called the symbols "colour tag"), allows storage of the foreground layer in a very small amount of space. The foreground is represented by a T.45 run-length encoded list of colours, one per JBIG2 symbol in the mask layer. The colours may be represented by discrete colours (i.e. in CIELAB space), indexed colours, such as in palette tables or 1 bit/component RGB/CMY(K).

A JBIG2-coded mask layer, within a stripe, might contain any combination of generic, halftone, refinement or text regions. Colour tags may only be used in association with JBIG2 mask layers, within a stripe, that contain only text regions (as colour tags "attach" to text). The foreground associated with a mask layer containing generic, refinement or halftone regions (or text and generic or text and halftone or text and refinement regions) shall be coded in the traditional MRC manner (i.e. using a multi-level coder). In other words, colour tags may only be used with foregrounds that are associated with mask layers within a stripe that contains only text regions.

B.5.1 Mask generation (rendering) when JBIG2 encoded

Recommendation T.88 defines a collection of encoding parameters and components, which are mixed and matched to generate various application profiles. Generation of the mask layer data stream will require knowledge of the specific profile used during JBIG2 encoding. Additionally, if colour tags are used to encode the foreground associated with a JBIG2 encoded mask it will be necessary to know this information prior to decoding the mask.

Subclause B.6.3 makes provision for a T88Options function, used to notify the decoder of: the JBIG2 profile used, whether colour tags are used, any other parameters and/or data required to decode the data stream. The T88Options function uses a series of flag bits to identify each option.

The "tags to follow" options flag bit is used to alert JBIG2 decoders to defer drawing the mask layer until the foreground layer has also been decoded. Given that mask layers are transmitted before the corresponding foreground layers, the decoder needs to be warned that the upcoming foreground layer is simply a list of colours (one per JBIG2 symbol in the mask layer), not a complete image. For this reason, the "tags to follow" flag bit is needed in the mask layer to warn the decoder that the foreground is compressed using colour tags. The decoder can then defer drawing the mask until the foreground layer has also been decoded.

B.5.2 Foreground generation (rendering) when mask is JBIG2 encoded

If the mask layer is coded with JBIG2 and the foreground layer is coded with T.45, then the foreground image is the image obtained by:

- decoding the text regions in the mask layer into a list of (X, Y, Symbol ID) triples; ordered as found in the JBIG2 data;
- decoding the T.45 data into a list of corresponding colour values (CVAL); this list shall have the same number of elements as the list of triples;

- matching each (X, Y, Symbol ID) triple with the corresponding colour value, giving a list of (X, Y, Symbol ID, CVAL) quads;
- drawing these quads, in order from first to last, into the foreground image.

Thus, if two symbol instances overlap, the colour from the later symbol instance overwrites the colour from the first one.

B.6 Data format

B.6.1 Overview

The second paragraph of A.9.1/T.44 is augmented to add shared data marker segment (SDMx) and a series of encoder marker segments (EMSe). The paragraph now reads as follows:

The MRC page structure for this application has the following elements: parameters, markers, and entropy-coded data segments. Parameters and markers are often organized into marker segments. Parameters are integers of length ½, 1, 2 or more octets. Markers are assigned two or more octet codes, an X'FF octet followed by an octet not equal to X'00' or X'FF' and optionally preceded by extra X'FF' octet codes. This application accommodates marker segments to denote the start of page (SOP), additional optional marker segments (OMSx), the start of a stripe (SOSt), the shared data marker segment (SDMx), the start of layer coded data (SLC), encoder related marker segments such as JBIG2 encoder marker segment (JB2e) and colour-interpreter encoder marker segment (CLIe), and end of header marker segment (EOH). The SDMx, JB2e and CLIe marker segments are defined in Annex B/T.44. The MRC Magic Number (i.e. JPEG SOI) is used immediately preceding the application marker as part of the SOP marker segment. Prior to the first SOSt, the JPEG EOI is used as a termination number. The end of a page (EOP) is defined as X'FFD9FFD9'. These markers are inserted by the encoder, and understood by the decoder in addition to all markers used for the coding methods, such as SOS (start of scan) of Recommendation T.81.

NOTE – Entire JBIG2 encoded data streams (i.e. including JBIG2 stripes and headers) are inserted directly following the EOH marker segment.

B.6.2 Start of page marker segment

Start of page marker segment is defined per Mode 3 of Annex A/T.44 with revision to the "Mode:" description to read as follows:

Mode:	1 octet	X'04', indicating Mode 4. Each mode identifies a different level of performance. Mode 4 identifies mandatory writer and reader provision for SDMx (shared data marker segment) along with optional writer and mandatory reader provision for colour tags. The SDMx and colour tag provisions shall be used in conjunction with the SLC (start of layer coded marker segment) supported N-layer mode of T.44 as defined by Mode 3 of Annex A/T.44. Applications supporting Mode 4 shall support the capabilities defined in Mode 3.

B.6.3 JBIG2 encoder marker segment (JB2e), MRC12 entry

This entry specifies parameters and/or data that are required in decoding a JBIG2 encoded data stream. It is used to specify the JBIG2 profile and whether colour tags or any future JBIG2 options are used in the data stream. The JBIG2 facsimile profiles are defined in Recommendation T.89. Structure of JB2e entry is as follows:

APP13, length, JB2e ident., parameters/data.

The JB2e encoder marker segment is defined as follows:

APP13 marker:	2 octets	X'FFED'
Length of Segment:	2 octets	Length of segment in octets, MSB to LSB, as an integer value not including APP13.
JB2e Ident:	4 octets	'MRC12', represented as a 3-octet ASCII string plus an hexadecimal count (i.e. X'4D',X'52',X'43',X'0C'). This X'0C'-terminated string "MRC" uniquely identifies this marker segment as the JBIG2 encoder marker segment.
T88Options:	1 or more octet(s)	with bit setting indicating applied option(s) flag, as shown in Table B.1/T.44. More than one bit may be set to indicate a combination of T88Options.

Table B.1/T.44 – T88Options octet(s)

Octet bit number	T88Options definition
LSB 0	JBIG2 Fax Profile 1 per T.89
1	JBIG2 Fax Profile 2 per T.89
2	JBIG2 Fax Profile 3 per T.89
3	Reserved for JBIG2 Fax Profile to be defined in T.89
4	Reserved for JBIG2 Fax Profile to be defined in T.89
5	Reserved for JBIG2 Fax Profile to be defined in T.89
6	Tags to follow – used to alert the JBIG2 decoder to defer drawing the mask layer until the foreground layer has also been decoded. This is applied when the foreground colours are represented with colour tags. (Note 1.)
MSB 7	Extend, add another octet that follows immediately
NOTE 1 – If this bit is set, then JBIG2 Fax Profile 2 must be used (i.e. bit #1 must also be set). NOTE 2 – Bit 7, the extend bit, would be set when adding another octet to accommodate additional options, such as an 8th which would be assigned to bit number 8. NOTE 3 – Options for additional JBIG2 fax profiles will be added to bits 3 through 5 as they are defined in Rec. T.89.	

B.6.4 Shared data marker (SDMx) segment

This marker segment makes provision for the use of shared data, shared resources. It provides a means to: create/define a shared resource that is available to be used by multiple coded entities (i.e. pages, stripes or layers); refer to a shared resource at the point where it is to be used; instruct the decoder that the shared resource is no longer required and may be flushed from memory. In applications of JBIG2 encoding, the SDMx is typically used in mask layers (even numbered layers). The SDMx may be located prior to SOSt marker segments (i.e. between the TN and the first SOSt or between end of coded data stream for one stripe and the SOSt for the next stripe), between SOSt and SLC, between SLC and EOH, between end of coded data for one layer and SLC for the next layer, or between end of stripe data and EOP. More than one SDMx may appear at the same location within the page structure.

Structure of each SDMx entry is as follows:

APP13 (X'FFED'), length, SDMx ident, Optional length (if needed), parameters/data.

SDMx represents specific shared data marker segments, where "x" is replaced by a specific character used to represent each shared data marker segment.

B.6.4.1 Create shared data marker (SDMc) segment

The SDMc is used in creating/defining a shared resource that is available to be used by multiple coded entities (i.e. pages, stripes or layers). Only one set of shared data may be created per SDMc.

The Create Shared Data Marker Segment (SDMc) is defined as follows:

APP13 marker:	2 octets	X'FFED'
Length of Segment:	2 octets	length of segment including length of shared data stream in octets, MSB to LSB, as an integer value not including APP13. Note that if two octets are insufficient, that the Optional length will be used, and this Length of Segment value will be zero.
SDMc Ident:	4 octets	'MRC3', represented as a 3-octet ASCII string plus an hexadecimal count (i.e. X'4D',X'52',X'43',X'03'). This X'03'-terminated string "MRC" uniquely identifies this marker segment as the Create Shared Data Marker.
Optional length (if needed)	4 octets	length of segment when two octets are not sufficient. If used, Length of segment should have a value of zero.
ID:	4 octets	with value providing unique identification of the shared data being created. IDs are unique, and as such their values shall not be reassigned within the document.
Scope:	1 octet	with value indicating breadth of application, as shown in Table B.2/T.44, of the shared data being created/defined.
Shared data stream	(Length of segment – 11 or 15 octets)	data stream being created by this marker segment and referenced by the above ID.

Table B.2/T.44 – Share data scope octet

Octet value	Share data scope
0	Global – shared data is available for application across the entire document (i.e. over more than one page).
1	Page – shared data is available for application across an entire page (i.e. over more than one stripe).
2	Stripe – shared data is available for application across an entire stripe (i.e. over more than one layer).
3	Layer – shared data is available for application across an entire layer within a stripe (i.e. over one layer of a stripe).
4-255	Reserved

B.6.4.2 Disposition Shared Data Marker (SDMd) segment

The SDMd is used to: refer to a shared resource at the point where it is to be used; and/or instruct the decoder that the shared resource is no longer required and may be flushed from memory.

The Disposition Shared Data Marker Segment (SDMd) is defined as follows:

APP13 marker:	2 octets	X'FFED'
Length of Segment:	2 octets	Length of segment in octets, MSB to LSB, as an integer value not including APP13 and the data when present. Note that two octets should be sufficient for the length, although the use of Optional length may still be used, if necessary.
SDMd Ident:	4 octets	'MRC4', represented as a 3-octet ASCII string plus an hexadecimal count (i.e. X'4D',X'52',X'43',X'04'). This X'04'-terminated string "MRC" uniquely identifies this marker segment as the Disposition Shared Data Marker.
Disposition:	1 octet	with value indicating disposition, as shown in Table B.3/T.44, of the shared data referenced by the associated ID(s). The corresponding bit shall be set to "1" for each disposition type being applied. The "forget" and "use" dispositions may be used independently or in combination with each other. They may also apply to one or more "ID(s)" at a time. The "forget" disposition signals deletion of the shared data(s) referenced by the ID(s) below. The "Use" disposition signals application of the shared data(s) referenced by the ID(s) below. Combining the "forget" and "use" dispositions signals application and deletion at the end of the layer data stream of the shared data(s) referenced by the ID(s) below.
Count:	2 octets	with value indicating number of share data ID(s) being addressed by the "forget" and/or "use" disposition commands.
ID:	4 × Count	with value(s) identifying the shared data being addressed by the disposition commands.

Table B.3/T.44 – Share data disposition octet(s)

Octet bit number	Share data disposition
LSB 0	Use – apply shared data to the following layer
1	Forget – discard shared data: immediately, in the case that the "use" bit is not set; or after the layer, in the case that the "use" bit is set
2	Reserved
3	Reserved
4	Reserved
5	Reserved
6	Reserved
MSB 7	Reserved
NOTE – New disposition commands (i.e. 3rd through 8th command) would be assigned bit numbers 2 through 7 respectively.	

B.6.5 Interpretation and representation of run-length colour encoded data

Colours values (CVAL) from T.45 "Run-length Colour Encoding" encoded layers shall be interpreted using parameters of the Start of Layer Coded Data (SLC) marker segment, defined in A.9.5.1/T.44, the Colour-interpreter Encoder Marker Segment (CLIe) and any foreign encoder marker segments (i.e. encoder marker segments defined outside of this Recommendation) appearing between the SLC and EOH pair. The CLIe and foreign encoder marker segments shall and may, respectively, be included to provide complete colour interpretation. CLIe is specified within this annex while foreign encoder marker segments, used in the specification of gamut range, illuminant data and palette data, are defined outside of this annex. As per Annex A/T.44, the CLIe and any foreign encoder marker segments shall be located between the SLC and the EOH (End of Header) pair. The EOH is immediately followed by the run-length encoded layer data.

There are a number of SLC parameters that are not required when T.45 is used for colour value encoding. The value of the following unnecessary SLC parameters shall be set to "0" (zero) when T.45 is used:

- resolution
- width
- height
- layer base colour
- offset

The layer number and coder are the only SLC parameters that contain valid information when T.45 is used.

B.6.5.1 Colour-interpreter Encoder Marker Segment (CLIe)

The CLIe identifies colour-encoding rules and is required to interpret T.45 encoded colour values (CVAL). The CLIe may be used for the interpretation of colour values that have been encoded with other encoders. This marker segment is mandatory for all run-length colour encoded layers. Structure of the CLIe is as follows:

APP13, length, CLIe ident. ('MRC13'), colour interpreter.

The marker segment is defined as follows:

APP13 marker:	2 octets	X'FFED'
Length of Segment:	2 octets	total entry field octet count, MSB to LSB, including the octet count itself. It excludes the APP13 marker.
CLIe ident:	4 octets	'MRC13', represented as a 3-octet ASCII string plus an hexadecimal count (i.e. X'4D',X'52',X'43',X'0D'). This X'0D' terminated string "MRC" uniquely identifies this marker segment as the Colour-interpreter Encoder Marker Segment, CLIe.
ColorInterpreter:	1 octet	with value indicating colour interpreter as shown in Table B.4/T.44. The interpreter specifies the colour space, bit depth (i.e. number of bits/component) and possibly other colour parameters such as gamut range, illuminant and white point.

Table B.4/T.44 – Colour interpreter octet(s)

Octet value	Coder used
0	Continuous-tone colour using CIELAB 8 bits/component per Rec. T.42
1	Continuous-tone colour using CIELAB 12 bits/component per Rec. T.42
2-15	Reserved
16	3-bit colour (1 bit/colour) using RGB primaries per 6.2.1/T.43 (Note 1)
17	3-bit colour (1 bit/colour) using CMY primaries per 6.2.1/T.43 (Note 1)
18	4-bit colour (1 bit/colour) using CMYK primaries per 6.2.1/T.43 (Note 1)
19-31	Reserved
32	Palettized colour image using the palette, colour space and bit depth defined in the G3FAX3 marker segment per 6.2.2 and 7.2.2.4/T.43, included within the same SLC (Note 2)
33-255	Reserved
NOTES 1 – 1 bit/component image data is treated as a special case of palette image representation in which colours values are specified by a bit plane stack of named colours, per 6.2.1/T.43, rather than an exact colour value. Palette tables are omitted in these cases since there is no actual palette data. NOTE 2 – G3FAX3 marker segment, as specified in B.6.5.4, is required for colour value interpretation.	

B.6.5.2 Coded image gamut range data

The G3FAX1marker segment, defined in E.6.6/T.4, shall be used in the interpretation of T.45 encoded colour values (CVAL) when it is present between the SLC and EOH. Presence of the G3FAX1 is optional in the interpretation of T.45 encoded colour values (CVAL), however, if present, it shall be used. The G3FAX1 may be used in the interpretation of colour values that have been encoded with other encoders. The G3FAX1 is similar to the OMSg (layer base colour gamut range Optional Marker Segment) defined in 9.2.2.1/T.44, with the difference being that G3FAX1 is applied to the encoded data while OMSg is applied to the layer base colour. The G3FAX1 marker segment, as define in Annex E/T.4, is reproduced in Appendix I of this annex for the readers' convenience and information.

B.6.5.3 Coded image illuminant data

The G3FAX2 marker segment, defined in Recommendation E.6.7/T.4, shall be used in the interpretation of T.45 encoded colour values (CVAL) when it is present between the SLC and EOH. Presence of the G3FAX2 is optional in the interpretation of T.45 encoded colour values (CVAL), however, if present it shall be used. The G3FAX2 may be used for the interpretation of colour values that have been encoded with other encoders. The G3FAX2 is similar to the OMSi (layer base colour illuminant Optional Marker Segment) defined in 9.2.2.2/ T.44, with the difference being that G3FAX2 is applied to the encoded data while OMSi is applied to the layer base colour. The G3FAX2 marker segment, as defined in Annex E/ T.4, is reproduced in Appendix I.2 of this annex for the readers' convenience and information.

B.6.5.4 Coded image palette data

The G3FAX3 marker segment, defined in 7.2.2.4/ T.43, shall be used in the interpretation of T.45 encoded colour values (CVAL) when the CVALs are defined in terms of palette indices. The G3FAX3 marker segment shall be present between the SLC and EOH when T.45 encoded CVALs are defined in terms of palette indices. The G3FAX3 marker segment may be used in the interpretation of colour values that have been encoded with other encoders. The G3FAX3 marker segment, as defined in Recommendation T.43, is reproduced in Appendix I.3 of this annex for the readers' convenience and information. A sample G3FAX3 code stream is also presented in Appendix I.3 for information.

B.6.6 Data format summary

B.6.6.1 High level data format summary

<table>
<tr><td>SOP</td><td colspan="5">X'FFD8 X'FFED', Length, MRC0, Version, Mode, ...</td></tr>
<tr><td>TN</td><td colspan="5">X'FFD9'</td></tr>
<tr><td>OMSg</td><td colspan="5">X'FFED', Length, MRC10, Gamut data</td></tr>
<tr><td>OMSi</td><td colspan="5">X'FFED', Length, MRC11, Illuminant data</td></tr>
<tr><td rowspan="17">Page data</td><td rowspan="14">Stripe 1</td><td>SOSt</td><td colspan="3">X'FFED', Length, MRC1, Type, stripe height</td></tr>
<tr><td rowspan="13">Stripe data</td><td rowspan="6">Layer 2 (L2)</td><td>SLC</td><td>X'FFED', Length, MRC2, Layer number, coder, res., width, height, layer base colour, offset</td></tr>
<tr><td>SDMc</td><td>X'FFE3', Length, MRC3, Optional length, IDs, scope, data</td></tr>
<tr><td>SDMd</td><td>X'FFED', Length, MRC4, Disposition, count, IDs</td></tr>
<tr><td>JB2e</td><td>X'FFED', Length, MRC12, T88Options</td></tr>
<tr><td>EOH</td><td>X'FFED', Length, MRC255, coded data length</td></tr>
<tr><td colspan="2">Coded Data</td></tr>
<tr><td colspan="3">L1</td></tr>
<tr><td rowspan="4">L3</td><td colspan="2">SLC</td></tr>
<tr><td>CLIe</td><td>X'FFED', Length, MRC13, ColorInterpreter</td></tr>
<tr><td colspan="2">EOH</td></tr>
<tr><td colspan="2">Coded Data</td></tr>
<tr><td colspan="3">–
–
–</td></tr>
<tr><td colspan="3">LN</td></tr>
<tr><td colspan="5">–
–
–</td></tr>
<tr><td rowspan="2">Stripe N</td><td colspan="4">SOSt</td></tr>
<tr><td colspan="4">Stripe data</td></tr>
<tr><td>EOP</td><td colspan="5">X'FFD9FFD9'</td></tr>
</table>

B.6.6.2 Detail data format summary

MRC Magic Number

SOP marker segment

APP13 marker

Length of Segment

MRC0 SOP identifier

Version

Mode

Mask coder

Image layer coder

Mask resolution

Page width

TN

Layer base colour Gamut optional (OMSg) marker segment

APP13

Length of Segment

MRC10 OMSg identifier

    gamut range data

Layer base colour Illuminant optional (OMSi) marker segment

APP13

Length of Segment

MRC11 OMSi identifier

    illuminant data

Optional marker segments

APP13

Length of Segment

MRCn (n = 14 to 254) identifier

        ...

Shared data marker (SDMc) segment

        ...

Shared data marker (SDMc) segment

        ...

Shared data marker (SDMd) segment

        ...

Page data

Stripe 1

- *SOSt marker segment*
- APP13 marker
- Length of Segment
- MRC1 SOSt identifier
  - Type of stripe
  - Stripe height
- *Stripe data*
- *Main Mask layer (Layer 2)*
  - *SLC marker segment*
  - APP13 marker
  - Length of segment
  - MRC2 SLC identifier
    - layer number
    - coder
    - Resolution
    - Layer width
    - Layer height
    - Layer base colour
    - Offset

SDMc marker segment

APP13 marker

Length of segment

MRC3 SDMc identifier

Optional length (if needed)

ID

Scope

Shared data --------

SDMd marker segment

APP13 marker

Length of segment

MRC4 SDMd identifier

Disposition

Count

ID

JBIG2 encoder marker (JB2e) segment

APP13 marker

Length of segment

MRC12 JB2e identifier

T88Options

End of header (EOH) marker segment

APP13 marker

Length of segment

MRC255 EOH identifier

coded data length

Layer coded data --------

Background layer

SLC marker segment

:

:

Encoder marker segments

...

...

EOH marker segment

Layer coded data --------

Foreground layer

SLC marker segment

:

:

CLIe marker segments

:

G3FAX1 marker segments

:

G3FAX2 marker segments

:

EOH marker segment

Layer coded data -------

Layer 4

SLC marker segment

:

:

SDMd marker segment

...

JBIG2 encoder marker (JB2e) segment

EOH marker segment

Layer coded data --------

Layer 5

SLC marker segment

:

:

CLIe marker segments

G3FAX3 marker segments

:

EOH marker segment

Layer coded data --------

:

:

Layer N

SLC marker segment

:

:

SDMc marker segments

...

EMSe marker segments

EOH marker segment

Layer coded data -------

SDMd marker segments

...

Stripe 2

SOSt marker segment

APP13 marker

Stripe data

Main Mask layer (Layer 2)

Layer coded data --------

Background layer

Layer coded data --------

Foreground layer

Layer coded data -------

Layer 4

Layer coded data --------

Layer 5

Layer coded data --------

:

:

Layer N

Layer coded data -------

Stripe 3

Stripe n

SDMd marker segment

EOP (X'FFD9', X'FFD9')

Appendix I.1

G3FAX1 marker segment

The G3FAX1 marker segment, referenced in B.6.5.2 and defined in E.6.6/T.4, is reproduced here for the readers' convenience and information.

E.6.6/T.4 "FAX option identifier: G3FAX1 for gamut range"

X'FFE1' (APP1), length, G3FAX option identifier, gamut range data

The above terms are defined as follows:

Length: (Two octets) – Total APP1 field octet count including the octet count itself, but excluding the APP1 marker.

FAX identifier: (Six octets) – X'47', X'33', X'46', X'41', X'58', X'01' This X'01' – terminated string "G3FAX" uniquely identifies this APP1 marker as containing FAX information about optional gamut range data. (The FAX option identifiers are referred to as G3FAX1 - G3FAX255, meaning the octet-terminated string "G3FAX", X'nn').

Gamut range data: (Twelve octets) – The data field contains six two-octet signed integers. For example X'0064' represents 100. The calculation from a real value L* to an eight bit value, L, is made as follows:

$$L = (255/Q) \times L^* + P$$

where the first integer of the first pair, P, contains the offset of the zero point in L* in the eight most significant bits. The second integer of the first pair, Q, contains the span of the gamut range in L*. Rounding to the nearest integer is performed. The second pair contains offset and range values for a*. The third pair contains offset and range values for b*. If the image is gray-scale (L* only), the field still contains six integers, but the last four are ignored.

NOTE – This representation is in accord with Recommendation T.42. When the twelve bits/pel/component option is used, the range and offset are represented as above in eight bits. These represent the eight most significant bits of the zero-padded twelve-bit number in the offset, and the eight-bit integer range data as above. Appropriately higher precision calculation should be used.

For example, the gamut range L* = [0, 100], a* = [–85, 85], and b* = [–75, 125] would be selected by the code:

X'FFE1',X'0014',X'47',X'33',X'46',X'41',X'58',X'01',X'0000',X'0064',X'0080',X'00AA',X'0060',X'00C8'.

Appendix I.2

G3FAX2 marker segment

The G3FAX2 marker segment, referenced in B.6.5.3 and defined in E.6.7/T.4, is reproduced here for the readers' convenience and information.

E.6.7/T.4 "FAX option identifier: G3FAX2 for illuminant data"

X'FFE1' (APP1), length, G3FAX option identifier, illuminant data. This option is for further study with the exception of the default case; the specification of the default illuminant, CIE Illuminant D50, may be added for information.

Length: (Two octets) – Total APP1 field octet count including the octet count itself, but excluding the APP1 marker.

FAX identifier: (Six octets) – X'47', X'33', X'46', X'41', X'58', X'02'. This X'02'-terminated string "G3FAX" uniquely identifies this APP1 marker as containing optional illuminant data.

Illuminant data: (Four octets) – The data consist of a four octet code identifying the illuminant. In the case of a standard illuminant, the four octets are one of the following:

CIE Illuminant D50: X'00', X'44', X'35', X'30'

CIE Illuminant D65: X'00', X'44', X'36', X'35'

CIE Illuminant D75: X'00', X'44', X'37' X'35'

CIE Illuminant SA: X'00', X'00', X'53', X'41'

CIE Illuminant SC: X'00', X'00', X'53', X'43'

CIE Illuminant F2: X'00', X'00', X'46', X'32'

CIE Illuminant F7: X'00', X'00', X'46', X'37'

CIE Illuminant F11: X'00', X'46', X'31', X'31'

CIE Illuminant D50: X'00', X'44', X'35', X'30'

In the case of a colour temperature alone, the four octets consist of the string 'CT', followed by the temperature of the source in degrees Kelvin represented by an unsigned two-octet integer. For example, a 7500 °K illuminant is indicated by the code:

X'FFE1', X'000C', X'47', X'33', X'46', X'41', X'58', X'02', X'43', X'54', X'1D4C'.

Appendix I.3

G3FAX3 marker segment

I.3.1 Marker segment definition

The G3FAX3 marker segment, referenced in B.6.5.4 and defined in 7.2.2.4.1/T.43, is reproduced here for the readers' convenience and information.

7.2.2.4.1/T.43 "G3FAX3/G4FAX3 entry for colour palette table"

Colour palette table is specified using the Entry Marker X'FFE3' as follows:

X'FFE3' (Entry Marker), length (4 octets), FAX identifier 3, table ID, $t_{entries}$, colour table data.

Length: (4 octets) – Total G3FAX3/G4FAX3 entry field octet count including the octet count itself, but excluding the Entry Marker.

FAX identifier 3: (6 octets) – X'47', X'3m', X'46', X'41', X'58', X'03' (m = 3 or 4) This identifier specifies G3FAX3/G4FAX3 entry.

Table ID: (2 octets) – This specifies the type of colour palette table.

0: table specified in CIELAB space (8 bits/comp. precision)

4: table specified in CIELAB space (12 bits/comp. precision).

$T_{entries}$: (4 octets) – It specifies the number of the colour palette table entries. This value should have the following relations:

N: Number of bits specified in G3FAX0/G4FAX0.

mb: octets/component in the table:

1: 8 bits precision

2: 12 bits precision

$2^{**}(N - 1) < t_{entries} \leq 2^{**}N$

length = 16 + (3 $*$ $t_{entries}$ $*$ mb).

Colour table data: ((3 $*$ $t_{entries}$ $*$ mb) octets) – This data consists of $t_{entries}$ colour palette table entries. Each table entry which consists of 3 components, is in sequential order from index = 0 to index = $t_{entries}$ – 1. Each component consists of one or two octets value. Its length is specified by the table ID. Each component value is represented by CIELAB space defined in Recommendation T.42.

I.3.2 Code string example

The table below is an example of the code string for the following colour palette table. It assumes that the table is specified in CIELAB space (8 bit/comp. precision), $t_{entries} = 236$.

Colour palette table example for 236 entries and 8 bit accuracy:

Index	Component values (8 bits)		
	L*	a*	b*
0	255	128	96
1	0	128	96
2	128	128	96
–	–	–	–
–	–	–	–
–	–	–	–
235	220	128	220

Code String Example:

X'FFE3' Entry Marker	X'000002D4' length	X'47', X'33', X'46', X'41', X'58', X'03' FAX-identifier " G3FAX '3' "	X'0000' table ID = 0	X'000000EC' $t_{entries} = 236$

X'FF', X'80', X'60' index = 0 (255,128,96)	X'00', X'80', X'60' index = 1 (0,128,96)	X'80', X'80', X'60' index = 2 (128,128,96)	... ...	X'DC', X'80', X'DC' index = 235 (220,128,220)

ITU-T RECOMMENDATIONS SERIES

Series A	Organization of the work of the ITU-T
Series B	Means of expression: definitions, symbols, classification
Series C	General telecommunication statistics
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	TMN and network maintenance: international transmission systems, telephone circuits, telegraphy, facsimile and leased circuits
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	**Terminals for telematic services**
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks and open system communications
Series Y	Global information infrastructure and Internet protocol aspects
Series Z	Languages and general software aspects for telecommunication systems