

International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

H.222.0

Amendment 1
(12/2017)

SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS

Infrastructure of audiovisual services – Transmission
multiplexing and synchronization

Information technology – Generic coding of moving
pictures and associated audio information – Part 1:
Systems

**Amendment 1: Ultra-low latency and 4K and
higher resolution support for transport of JPEG
2000 video**

Recommendation ITU-T H.222.0 (2017) –
Amendment 1



ITU-T H-SERIES RECOMMENDATIONS
AUDIOVISUAL AND MULTIMEDIA SYSTEMS

| | |
|---|--------------------|
| CHARACTERISTICS OF VISUAL TELEPHONE SYSTEMS | H.100–H.199 |
| INFRASTRUCTURE OF AUDIOVISUAL SERVICES | |
| General | H.200–H.219 |
| Transmission multiplexing and synchronization | H.220–H.229 |
| Systems aspects | H.230–H.239 |
| Communication procedures | H.240–H.259 |
| Coding of moving video | H.260–H.279 |
| Related systems aspects | H.280–H.299 |
| Systems and terminal equipment for audiovisual services | H.300–H.349 |
| Directory services architecture for audiovisual and multimedia services | H.350–H.359 |
| Quality of service architecture for audiovisual and multimedia services | H.360–H.369 |
| Telepresence | H.420–H.429 |
| Supplementary services for multimedia | H.450–H.499 |
| MOBILITY AND COLLABORATION PROCEDURES | |
| Overview of Mobility and Collaboration, definitions, protocols and procedures | H.500–H.509 |
| Mobility for H-Series multimedia systems and services | H.510–H.519 |
| Mobile multimedia collaboration applications and services | H.520–H.529 |
| Security for mobile multimedia systems and services | H.530–H.539 |
| Security for mobile multimedia collaboration applications and services | H.540–H.549 |
| VEHICULAR GATEWAYS AND INTELLIGENT TRANSPORTATION SYSTEMS (ITS) | |
| Architecture for vehicular gateways | H.550–H.559 |
| Vehicular gateway interfaces | H.560–H.569 |
| BROADBAND, TRIPLE-PLAY AND ADVANCED MULTIMEDIA SERVICES | |
| Broadband multimedia services over VDSL | H.610–H.619 |
| Advanced multimedia services and applications | H.620–H.629 |
| Ubiquitous sensor network applications and Internet of Things | H.640–H.649 |
| IPTV MULTIMEDIA SERVICES AND APPLICATIONS FOR IPTV | |
| General aspects | H.700–H.719 |
| IPTV terminal devices | H.720–H.729 |
| IPTV middleware | H.730–H.739 |
| IPTV application event handling | H.740–H.749 |
| IPTV metadata | H.750–H.759 |
| IPTV multimedia application frameworks | H.760–H.769 |
| IPTV service discovery up to consumption | H.770–H.779 |
| Digital Signage | H.780–H.789 |
| E-HEALTH MULTIMEDIA SERVICES AND APPLICATIONS | |
| Personal health systems | H.810–H.819 |
| Interoperability compliance testing of personal health systems (HRN, PAN, LAN, TAN and WAN) | H.820–H.859 |
| Multimedia e-health data exchange services | H.860–H.869 |

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

**Information technology – Generic coding of moving pictures and
associated audio information – Part 1: Systems**

Amendment 1

**Ultra-low latency and 4K and higher resolution support for
transport of JPEG 2000 video**

Summary

Amendment 1 to ITU-T H.222.0 (2017) | ISO/IEC 13818-1:2017 fixes interoperability issues in the transport of JPEG 2000 Part 1 (Rec. ITU-T T.800 | ISO/IEC 15444-1) by removing references to Rec. ITU-T T.800 | ISO/IEC 15444-1 Annex M and updating the definition of the elementary stream header to make it self-contained in ISO/IEC 13818-1 Annex S. It further adds support for JPEG 2000 Ultra-Low Latency (ULL) encoding and transport of professional video, audio and data over Internet Protocol networks, by specifying the use of horizontal, independent JPEG 2000 stripes. Finally, it supports higher resolutions (4K or higher) of JPEG 2000 video images by adding a new block mode. This new mode allows implementers to divide a given frame into blocks.

History

| Edition | Recommendation | Approval | Study Group | Unique ID* |
|---------|---------------------------------------|------------|-------------|---|
| 1.0 | ITU-T H.222.0 | 1995-07-10 | 15 | 11.1002/1000/1071 |
| 1.1 | ITU-T H.222.0 (1995) Amd. 1 | 1996-11-11 | 16 | 11.1002/1000/3834 |
| 1.2 | ITU-T H.222.0 (1995) Amd. 2 | 1996-11-11 | 16 | 11.1002/1000/4096 |
| 1.3 | ITU-T H.222.0 (1995) Technical Cor. 1 | 1998-02-06 | 16 | 11.1002/1000/4532 |
| 1.4 | ITU-T H.222.0 (1995) Amd. 3 | 1998-02-06 | 16 | 11.1002/1000/4228 |
| 1.5 | ITU-T H.222.0 (1995) Amd. 4 | 1998-02-06 | 16 | 11.1002/1000/4229 |
| 1.6 | ITU-T H.222.0 (1995) Amd. 5 | 1999-05-27 | 16 | 11.1002/1000/4498 |
| 1.7 | ITU-T H.222.0 (1995) Amd. 6 | 1999-05-27 | 16 | 11.1002/1000/4671 |
| 2.0 | ITU-T H.222.0 | 2000-02-17 | 16 | 11.1002/1000/5142 |
| 2.1 | ITU-T H.222.0 (2000) Technical Cor. 1 | 2001-03-01 | 16 | 11.1002/1000/5419 |
| 2.2 | ITU-T H.222.0 (2000) Technical Cor. 2 | 2002-03-29 | 16 | 11.1002/1000/5675 |
| 2.3 | ITU-T H.222.0 (2000) Amd. 1 | 2002-12-14 | 16 | 11.1002/1000/6190 |
| 2.4 | ITU-T H.222.0 (2000) Amd. 1/Cor. 1 | 2003-06-29 | 16 | 11.1002/1000/6449 |
| 2.5 | ITU-T H.222.0 (2000) Amd. 2 | 2003-06-29 | 16 | 11.1002/1000/6363 |
| 2.6 | ITU-T H.222.0 (2000) Amd. 3 | 2004-03-15 | 16 | 11.1002/1000/7208 |
| 2.7 | ITU-T H.222.0 (2000) Technical Cor. 3 | 2005-01-08 | 16 | 11.1002/1000/7435 |
| 2.8 | ITU-T H.222.0 (2000) Amd. 4 | 2005-01-08 | 16 | 11.1002/1000/7436 |
| 2.9 | ITU-T H.222.0 (2000) Amd. 5 | 2005-01-08 | 16 | 11.1002/1000/7437 |
| 2.10 | ITU-T H.222.0 (2000) Technical Cor. 4 | 2005-09-13 | 16 | 11.1002/1000/8560 |
| 3.0 | ITU-T H.222.0 | 2006-05-29 | 16 | 11.1002/1000/8802 |
| 3.1 | ITU-T H.222.0 (2006) Amd. 1 | 2007-01-13 | 16 | 11.1002/1000/9024 |
| 3.2 | ITU-T H.222.0 (2006) Amd. 2 | 2007-08-29 | 16 | 11.1002/1000/9214 |
| 3.3 | ITU-T H.222.0 (2006) Cor. 1 | 2008-06-13 | 16 | 11.1002/1000/9471 |

* To access the Recommendation, type the URL <http://handle.itu.int/> in the address field of your web browser, followed by the Recommendation's unique ID. For example, <http://handle.itu.int/11.1002/1000/11830-en>.

| | | | | |
|------|------------------------------------|------------|----|------------------------------------|
| 3.4 | ITU-T H.222.0 (2006) Cor. 2 | 2009-03-16 | 16 | 11.1002/1000/9692 |
| 3.5 | ITU-T H.222.0 (2006) Amd. 3 | 2009-03-16 | 16 | 11.1002/1000/9691 |
| 3.6 | ITU-T H.222.0 (2006) Cor. 3 | 2009-12-14 | 16 | 11.1002/1000/10621 |
| 3.7 | ITU-T H.222.0 (2006) Cor. 4 | 2009-12-14 | 16 | 11.1002/1000/10622 |
| 3.8 | ITU-T H.222.0 (2006) Amd. 4 | 2009-12-14 | 16 | 11.1002/1000/10623 |
| 3.9 | ITU-T H.222.0 (2006) Amd. 5 | 2011-05-14 | 16 | 11.1002/1000/11287 |
| 3.10 | ITU-T H.222.0 (2006) Amd. 6 | 2011-05-14 | 16 | 11.1002/1000/11288 |
| 4.0 | ITU-T H.222.0 | 2012-06-29 | 16 | 11.1002/1000/11655 |
| 4.1 | ITU-T H.222.0 (2012) Amd. 1 | 2014-01-13 | 16 | 11.1002/1000/12054 |
| 4.2 | ITU-T H.222.0 (2012) Amd. 2 | 2014-01-13 | 16 | 11.1002/1000/12055 |
| 4.3 | ITU-T H.222.0 (2012) Amd. 3 | 2014-01-13 | 16 | 11.1002/1000/12056 |
| 4.4 | ITU-T H.222.0 (2012) Amd. 4 | 2014-01-13 | 16 | 11.1002/1000/12057 |
| 4.5 | ITU-T H.222.0 (2012) Amd. 5 | 2014-10-14 | 16 | 11.1002/1000/12306 |
| 5.0 | ITU-T H.222.0 | 2014-10-14 | 16 | 11.1002/1000/12359 |
| 5.1 | ITU-T H.222.0 (2014) Amd. 1 | 2015-04-29 | 16 | 11.1002/1000/12452 |
| 5.2 | ITU-T H.222.0 (2014) Amd. 1 Cor. 1 | 2015-11-29 | 16 | 11.1002/1000/12625 |
| 5.3 | ITU-T H.222.0 (2014) Amd. 2 | 2015-12-14 | 16 | 11.1002/1000/12632 |
| 5.4 | ITU-T H.222.0 (2014) Amd. 3 | 2015-12-14 | 16 | 11.1002/1000/12633 |
| 5.5 | ITU-T H.222.0 (2014) Amd. 1 Cor. 2 | 2016-07-14 | 16 | 11.1002/1000/12899 |
| 5.5 | ITU-T H.222.0 (2014) Cor. 1 | 2016-07-14 | 16 | 11.1002/1000/12903 |
| 5.7 | ITU-T H.222.0 (2014) Amd. 4 | 2016-07-14 | 16 | 11.1002/1000/12900 |
| 5.8 | ITU-T H.222.0 (2014) Amd. 5 | 2016-07-14 | 16 | 11.1002/1000/12901 |
| 5.9 | ITU-T H.222.0 (2014) Amd. 6 | 2016-07-14 | 16 | 11.1002/1000/12902 |
| 5.10 | ITU-T H.222.0 (2014) Amd. 3 Cor. 1 | 2017-03-01 | 16 | 11.1002/1000/13184 |
| 5.10 | ITU-T H.222.0 (2014) Cor. 2 | 2017-03-01 | 16 | 11.1002/1000/13188 |
| 5.12 | ITU-T H.222.0 (2014) Amd. 7 | 2017-03-01 | 16 | 11.1002/1000/13186 |
| 5.13 | ITU-T H.222.0 (2014) Amd. 8 | 2017-03-01 | 16 | 11.1002/1000/13187 |
| 6.0 | ITU-T H.222.0 | 2017-03-01 | 16 | 11.1002/1000/13269 |
| 6.0 | ITU-T H.222.0 (2017) Amd. 1 | 2017-12-14 | 16 | 11.1002/1000/13431 |

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. 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 Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 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.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. 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, ITU had received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at <http://www.itu.int/ITU-T/ipr/>.

© ITU 2018

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

CONTENTS

| | <i>Page</i> |
|---|-------------|
| 1) Clause 2.1.69 | 1 |
| 2) New clauses 2.1.70, 2.1.71, 2.1.73, 2.1.74 | 1 |
| 3) Clause 2.6.80 | 1 |
| 4) Clause 2.6.81 | 2 |
| 5) Annex S, clause S.2 | 5 |
| 6) Annex S, new clauses S.3 and S.4 | 5 |
| 7) Annex S, new clause S.5 | 6 |
| 8) Annex S, clause S.6 | 9 |
| 9) Annex S, clause S.8.1 | 9 |
| 10) Annex S, Table S.2 | 10 |

INTERNATIONAL STANDARD
ITU-T RECOMMENDATIONInformation technology – Generic coding of moving pictures and
associated audio information – Part 1: Systems

Amendment 1

Ultra-low latency and 4K and higher resolution support for
transport of JPEG 2000 video

1) Clause 2.1.69

Replace 2.1.69 with the following:

2.1.69 JPEG 2000 (J2K) video access unit: The JPEG 2000 codestream or codestreams comprising a decodable and randomly accessible (portion of) image, preceded by all the parameters required to decode the access unit and display the decoded data.

2) New clauses 2.1.70, 2.1.71, 2.1.73, 2.1.74

Add new 2.1.70, 2.1.71, 2.1.73, 2.1.74 and update other subclause numbering accordingly:

2.1.70 J2K block: The JPEG 2000 codestream or codestreams corresponding to a rectangular portion of a video frame, as detailed in S.3.

NOTE – Usage of J2K blocks requires J2K block mode (defined in 2.1.71) to be enabled in the J2K video descriptor. Such usage facilitates the support of 4k and higher resolutions.

2.1.71 J2K block mode: Optional mode defined in S.3, dividing each frame of a J2K video stream in a certain amount of rectangular blocks, each encoded as an independent J2K block (defined in 2.1.70).

2.1.73 J2K stripe: The JPEG 2000 codestream or codestreams comprising a decodable horizontally divided portion of an image, as detailed in S.4.

NOTE – Usage of J2K stripes requires J2K stripe mode (defined in 2.1.74) to be enabled in the J2K video descriptor. Such usage enables transport of a J2K video stream with a low end-to-end latency.

2.1.74 J2K stripe mode: Optional mode defined in S.4, dividing the (portion of) image transported in a J2K video access unit in a succession of horizontal stripes, each encoded as an independent J2K stripe (defined in 2.1.73).

3) Clause 2.6.80

Replace Table 2-99 with the following:

Table 2-99 – J2K video descriptor

| Syntax | No. of bits | Mnemonic |
|--|-------------|---------------|
| J2K_video_descriptor() { | | |
| descriptor_tag | 8 | uimsbf |
| descriptor_length | 8 | uimsbf |
| extended_capability_flag | 1 | bslbf |
| profile_and_level | 15 | bslbf |
| horizontal_size | 32 | uimsbf |
| vertical_size | 32 | uimsbf |
| max_bit_rate | 32 | uimsbf |
| max_buffer_size | 32 | uimsbf |
| DEN_frame_rate | 16 | bslbf |
| NUM_frame_rate | 16 | bslbf |
| if (extended_capability_flag == '1') { | | |

Table 2-99 – J2K video descriptor

| Syntax | No. of bits | Mnemonic |
|---|-------------|----------|
| <code>stripe_flag</code> | 1 | bslbf |
| <code>block_flag</code> | 1 | bslbf |
| <code>mdm_flag</code> | 1 | bslbf |
| <code>reserved (all bits to be set to '0')</code> | 5 | bslbf |
| <code>} else {</code> | | |
| <code>color_specification</code> | 8 | bslbf |
| <code>}</code> | | |
| <code>still_mode</code> | 1 | bslbf |
| <code>interlaced_video</code> | 1 | bslbf |
| <code>reserved</code> | 6 | bslbf |
| <code>if (extended_capability_flag == '1') {</code> | | |
| <code>colour_primaries</code> | 8 | uimsbf |
| <code>transfer_characteristics</code> | 8 | uimsbf |
| <code>matrix_coefficients</code> | 8 | uimsbf |
| <code>video_full_range_flag</code> | 1 | bslbf |
| <code>reserved</code> | 7 | bslbf |
| <code>if (stripe_flag == '1') {</code> | | |
| <code>strp_max_idx</code> | 8 | uimsbf |
| <code>strp_height</code> | 16 | uimsbf |
| <code>}</code> | | |
| <code>if (block_flag == '1') {</code> | | |
| <code>full_horizontal_size</code> | 32 | uimsbf |
| <code>full_vertical_size</code> | 32 | uimsbf |
| <code>blk_width</code> | 16 | uimsbf |
| <code>blk_height</code> | 16 | uimsbf |
| <code>max_blk_idx_h</code> | 8 | uimsbf |
| <code>max_blk_idx_v</code> | 8 | uimsbf |
| <code>blk_idx_h</code> | 8 | uimsbf |
| <code>blk_idx_v</code> | 8 | uimsbf |
| <code>}</code> | | |
| <code>if (mdm_flag == '1') {</code> | | |
| <code>X_c0, Y_c0, X_c1, Y_c1, X_c2, Y_c2</code> | 16x6 | uimsbf |
| <code>X_wp</code> | 16 | uimsbf |
| <code>Y_wp</code> | 16 | uimsbf |
| <code>L_max</code> | 32 | uimsbf |
| <code>L_min</code> | 32 | uimsbf |
| <code>MaxCLL</code> | 16 | uimsbf |
| <code>MaxFALL</code> | 16 | uimsbf |
| <code>}</code> | | |
| <code>}</code> | | |
| <code>for (i=0; i<N; i++) {</code> | | |
| <code>private_data_byte</code> | 8 | bslbf |
| <code>}</code> | | |
| <code>}</code> | | |

4) Clause 2.6.81

Replace 2.6.81 with the following:

2.6.81 Semantics of fields in J2K video descriptor

extended_capability_flag – This 1-bit field indicates that the J2K video stream uses extended color specification (through three bytes defining the chromaticity parameters, as described below), and that it might have one or several of the following capabilities enabled: stripes (through the J2K stripe mode), blocks (through the J2K block mode), or inclusion of mastering display metadata. The exact list of enabled capabilities is set through subsequent flags in the video descriptor (see below).

profile_and_level – This 15-bit field shall correspond to the 15 least significant bits of the 2-bytes Rsiz value included in all J2K codestream main headers of this J2K video stream. Rsiz values that are defined in Table A.10 of Rec. ITU-T T.800 | ISO/IEC 15444-1 and do set to '0' their most significant bit are allowed.

NOTE – the combination of the `extend_capability_flag` and the `profile_and_level` field ensures backward and forward compatibility with legacy devices conforming to previous versions of this Recommendation | International Standard. Having the `extended_capability_flag` set to '1' leads indeed to a 16-bit value outside the range accepted by previous versions of this Recommendation | International Standard. This way, J2K video streams with extended capabilities can be unequivocally identified by both legacy and new devices.

horizontal_size – This 32-bit field indicates the horizontal size of the frame (for progressive) or field (for interlaced) comprised in each J2K access unit. If J2K block mode is enabled, this frame or field corresponds to a spatial rectangular block of the entire video frame or field. It shall be coded the same as the `Xsiz` parameter found in all J2K codestream main headers of this J2K video stream, as defined in Annex A of Rec. ITU-T T.800 | ISO/IEC 15444-1.

vertical_size – This 32-bit field indicates the vertical size of the frame (for progressive) or field (for interlaced) comprised in each J2K access unit. If J2K block mode is enabled, this frame or field corresponds to a spatial rectangular block of the entire video frame or field. If J2K stripe mode is disabled, it shall be coded the same as the `Ysiz` parameter found in all J2K codestream main headers of this J2K video stream. If J2K stripe mode is enabled, it shall be coded as the sum of the `Ysiz` parameters found in all J2K codestreams composing the frame (for progressive) or a field (for interlaced) comprised in each J2K access unit. `Ysiz` parameters are defined in Annex A of Rec. ITU-T T.800 | ISO/IEC 15444-1.

max_bit_rate – This field may be coded the same as the `brat_max_br` field specified in Table S.1 and shall not exceed the maximum compressed bit rate value for the profile and level specified in Table S.2. This field shall be set appropriately and signalled when `profile_and_level` = '000 0011 0000 0111', where no maximum bit rate is specified.

max_buffer_size – This field shall not exceed the Maximum buffer size value for the profile and level specified in Table S.2. When `profile_and_level` = '000 0011 0000 0111', the `max_buffer_size` shall be set appropriately and shall not exceed (`max_bit_rate`/1.60E5), where `max_bit_rate` is expressed in bit/s.

DEN_frame_rate – This field shall be coded the same as `frat_denominator` field specified in Table S.1 (see Annex S).

NUM_frame_rate – This field shall be coded the same as `frat_numerator` field specified in Table S.1 (see Annex S).

NOTE – J2K frame rate is derived from the `DEN_frame_rate` and `NUM_frame_rate` values. Table 2-100 lists examples of typical broadcast frame rates with associated values of `DEN_frame_rate` and `NUM_frame_rate`.

Table 2-100 – Example frame rates based on `DEN_frame_rate` and `NUM_frame_rate` values

| DEN_frame_rate | NUM_frame_rate | Frame rate ratio (decimal representation) | Frame rate |
|---------------------|---------------------|--|------------|
| 0000 0000 0000 0000 | | | Forbidden |
| 0000 0011 1110 1001 | 0101 1101 1100 0000 | 24 000 / 1001 | 23.976 |
| 0000 0000 0000 0001 | 0000 0000 0001 1000 | 24 / 1 | 24.0 |
| 0000 0000 0000 0001 | 0000 0000 0001 1001 | 25 / 1 | 25.0 |
| 0000 0011 1110 1001 | 0111 0101 0011 0000 | 30 000 / 1001 | 29.97 |
| 0000 0000 0000 0001 | 0000 0000 0001 1110 | 30 / 1 | 30.0 |
| 0000 0000 0000 0001 | 0000 0000 0011 0010 | 50 / 1 | 50.0 |
| 0000 0011 1110 1001 | 1110 1010 0110 0000 | 60 000 / 1001 | 59.94 |
| 0000 0000 0000 0001 | 0000 0000 0011 1100 | 60 / 1 | 60.00 |

stripe_flag – This 1-bit field is included only if the `extended_capability_flag` is set to '1'. It indicates whether the J2K video stream has J2K stripe mode enabled. When this flag is set to '1' the J2K access unit elementary stream header (see Table S.1) shall not include the syntax element `j2k_tcod`, shall include the syntax element `j2k_strp`, and the corresponding J2K access unit shall be made of a succession of J2K stripes. When this flag is set to '0', the J2K access unit elementary stream header shall include the syntax element `j2k_tcod`, shall not include the syntax element `j2k_strp`, and the corresponding J2K access unit shall be made of one J2K codestream in case of progressive content and two J2K codestreams in case of interlaced content.

block_flag – This 1-bit field is included only if the `extended_capability_flag` is set to '1'. When set to '1', it indicates that the J2K video stream has J2K block mode enabled, meaning that this J2K video stream actually corresponds to a spatial rectangular block of the full video stream. Subdivision of each frame into rectangular independent blocks is further defined in Section S.3. When set to '0', then the associated J2K video stream shall not have J2K block mode enabled.

mdm_flag – This 1-bit field is included only if the `extended_capability_flag` is set to '1'. When set to '1', it indicates that the J2K video descriptor contains the characteristics of the Mastering Display Metadata, as described in SMPTE ST 2086:2014 (see below corresponding fields). When set to '0', then the J2K video descriptor shall not contain the characteristics of the Mastering Display Metadata.

color_specification – This 8-bit field is included only if the `extended_capability_flag` is set to '0' and corresponds to the legacy color specification method. It shall be coded the same as the `bcol_colr` 8-bit field of the `j2k_bcol` box as specified in Table S.1 (see Annex S).

still_mode – This 1-bit field, when set to '1', indicates that the J2K video stream may include J2K still pictures. When set to '0', then the associated J2K video stream shall not contain J2K still pictures.

interlaced_video – This 1-bit field indicates whether the J2K video stream contains interlaced video. When this flag is set to '1' the J2K access unit elementary stream header (see Table S.1) shall include the syntax elements `brat_auf2`, `fiel_box_code`, `fiel_fic` and `fiel_fio`. When this flag is set to '0', these syntax elements shall not be present in the J2K access unit elementary stream header.

color primaries, transfer characteristics, matrix coefficients, video_full_range_flag – These four fields (three 8-bit fields and one 1-bit field) are included only if the `extended_capability_flag` is set to '1' and correspond to a color specification method allowing a broader set of color code points than the legacy method (see `color_specification` field above). These fields shall be coded according to the semantics with the same name defined in Rec. ITU-T H.273 | ISO/IEC 23001-8.

strp_max_idx – This 8-bit field is included only if J2K stripe mode is enabled. It shall be in the range 0x01-0xff and indicates the maximum value of the stripe index. It corresponds to the number of stripes in the block/field/frame, minus one. Value 0x00 is forbidden as a minimum of 2 stripes is required (otherwise J2K stripe mode shall be disabled).

strp_height – This 16-bit field is included only if J2K stripe mode is enabled. It indicates the default vertical size of a stripe. Depending on the `vertical_size` field value, the last stripe might have a different height, as detailed in S.4.

full_horizontal_size – This 32-bit field is included only if J2K block mode is enabled. It indicates the horizontal size of the entire video frame of this J2K video stream.

full_vertical_size – This 32-bit field is included only if J2K block mode is enabled. It indicates the vertical size of the entire video frame of this J2K video stream.

blk_width – This 16-bit field is included only if J2K block mode is enabled. It indicates the default width of a J2K block. Depending on the `full_horizontal_size` field value, the last block of a row might have a different width, as detailed in S.3.

blk_height – This 16-bit field is included only if J2K block mode is enabled. It indicates the default height of a J2K block. Depending on the `full_vertical_size` field value, the last block of a column might have a different height, as detailed in S.3.

max_blk_idx_h – This 8-bit field is included only if J2K block mode is enabled and indicates the maximum value of the horizontal block index for this video frame. It corresponds to the total number of blocks in the horizontal direction, minus one.

max_blk_idx_v – This 8-bit field is included only if J2K block mode is enabled and indicates the maximum value of the vertical block index for this video frame. It corresponds to the total number of blocks in the vertical direction, minus one.

blk_idx_h – This 8-bit field is included only if J2K block mode is enabled and indicates the horizontal block index of the current block.

blk_idx_v – This 8-bit field is included only if J2K block mode is enabled and indicates the vertical block index of the current block.

The following fields `X_c0`, `Y_c0`, `X_c1`, `Y_c1`, `X_c2`, `Y_c2`, `X_wp`, `Y_wp`, `L_max` and `L_min` correspond to the fields defined in SMPTE ST2086:2014 "Mastering Display Color Volume Metadata Supporting High Luminance and Wide Color Gamut Images". The fields `MaxFALL` and `MaxCLL` correspond to the fields defined in ANSI/CTA 861-G:2016 "A DTV Profile for Uncompressed High Speed Digital Interfaces". If these 12 fields have unknown values at the time of generating the stream, they shall not be included in the descriptor and the `mdm_flag` shall be set to '0'.

X_c0, Y_c0, X_c1, Y_c1, X_c2, Y_c2 – These 16-bit fields are included only if the `mdm_flag` is set to '1'. They specify the normalized x and y chromaticity coordinates of the colour primary components of the mastering display in increments of 0.00002, according to the CIE 1931 definition of x and y as specified in ISO 11664-1 (see also ISO 11664-3 and CIE 15). For describing mastering displays that use red, green, and blue colour primaries, it is

suggested that index value c0 should correspond to the green primary, c1 should correspond to the blue primary, and c2 should correspond to the red colour primary. The values of these 6 fields shall be in the range of 0 to 50 000, inclusive.

X_wp and **Y_wp** – These 16-bit fields are included only if the *mdm_flag* is set to '1'. They specify the normalized x and y chromaticity coordinates of the white point of the mastering display in normalized increments of 0.00002, according to the CIE 1931 definition of x and y as specified in ISO 11664-1 (see also ISO 11664-3 and CIE 15). The values of **X_wp** and **Y_wp** shall be in the range of 0 to 50 000.

L_max and **L_min** – These 32-bit fields are included only if the *mdm_flag* is set to '1'. They specify the nominal maximum and minimum display luminance, respectively, of the mastering display in units of 0.0001 candelas per square metre. **L_min** shall be less than **L_max**. At minimum luminance, the mastering display is considered to have the same nominal chromaticity as the white point.

MaxCLL – This 16-bit field is included only if the *mdm_flag* is set to '1'. It specifies the Maximum Content Light Level and corresponds to the brightest pixel in the entire stream, in units of 1 cd/m², where 0x0001 represents 1 cd/m² and 0xFFFF represents 65535 cd/m². It shall be calculated according to Annex P Calculation of MaxCLL and MaxFALL section P.1 in ANSI/CTA 861-G:2016 A DTV Profile for Uncompressed High Speed Digital Interfaces.

MaxFALL – This 16-bit field is included only if the *mdm_flag* is set to '1'. It specifies the Maximum Frame Average Light Level and corresponds to the highest frame average brightness per frame in the entire stream, in units of 1 cd/m², where 0x0001 represents 1 cd/m² and 0xFFFF represents 65535 cd/m². It shall be calculated according to Annex P Calculation of MaxCLL and MaxFALL section P.2 in ANSI/CTA 861-G:2016 "A DTV Profile for Uncompressed High Speed Digital Interfaces".

If for some reason the MaxCLL and/or MaxFALL values are unknown, the value 0x0000 shall be used.

5) Annex S, clause S.2

Replace S.2 with the following:

S.2 J2K video access unit, J2K video elementary stream, J2K video sequence and J2K still picture

The J2K video access unit contains the elementary stream (elsm) header created as defined in S.3 concatenated with self-contained Rec. ITU-T T.800 | ISO/IEC 15444-1 codestream(s). The (elsm) header contains all video-related parameters necessary to display the decoded codestream(s). Multiple codestreams may comprise an access unit in the two following situations:

1. When the access unit is an interlaced (portion of) frame.
2. When the J2K stripe mode is enabled, as defined in S.4.

The J2K video elementary stream is a progression of J2K access units and the J2K video sequence is a subset of J2K video elementary stream where all the J2K access units have the same parameters in the (elsm) header.

The J2K still picture (system) consists of a J2K video sequence which contains exactly one J2K access unit. This still picture has an associated PTS and the presentation time of succeeding pictures, if any, is later than that of the still picture by at least two picture periods. The J2K still picture (system) mode is used to support transmission of J2K video access units at a rate much lower than the display frame rate (determined by the difference in PTS values between successive J2K access units). J2K still picture can be used in applications such as 'slide show' and 'stills with Music'.

6) Annex S, new clauses S.3 and S.4

After S.2, add new subclauses S.3 and S.4 and update subsequent subclause numbering accordingly.

S.3 Optional J2K block mode for high resolution support

By default, each video frame is encoded into one self-contained Rec. ITU-T T.800 | ISO/IEC 15444-1 codestream for progressive content, and into two such codestreams for interlaced content (one per field). The codestream(s) are then encapsulated in a J2K access unit, which therefore always contains an entire video frame.

For 4k resolution (3840x2160) and beyond, the J2K block mode can be used. This mode allows to divide each video frame horizontally and vertically in the spatial domain, into multiple video blocks, each block being processed the same way entire frames were processed in the default mode. The main use case for this mechanism is to be able to stream different quadrants or blocks of a video separately. As an example, a 3840x2160p/59.94 video can be divided into four 1920x1080p/59.94 video blocks.

For this mode to be correctly enabled, the video descriptor shall have *extended_capability_flag* and *block_flag* both set to '1' and shall contain the block dimensions (stored in *horizontal_size* and *vertical_size* fields), the frame dimensions (stored in *full_horizontal_size* and *full_vertical_size* fields), the maximum values of the horizontal and vertical block index (*max_blk_idx_h*, *max_blk_idx_v*), and the horizontal and vertical index value for the block being processed by the video descriptor (*blk_idx_h*, *blk_idx_v*). See Table 2-99 for further details on the J2K video descriptor. The blocks shall be constructed so that they have the same horizontal and vertical size, with the exception of the ones located on the last row and the ones located on the last column, which may have a different size. The following method shall be used to derive the width and height of a block:

```

if (blk_idx_h == max_blk_idx_h):
    width = full_horizontal_size - ( horizontal_size * max_blk_idx_h )
otherwise:
    width = horizontal_size

if (blk_idx_v == max_blk_idx_v):
    height = full_vertical_size - ( vertical_size * max_blk_idx_v )
otherwise:
    height = vertical_size

```

When this mode is enabled, each J2K video access unit contains the Rec. ITU-T T.800 | ISO/IEC 15444-1 codestream(s) related to a specific video block only. Corresponding J2K video blocks across video frames compose an independent J2K video elementary stream (with its own J2K video descriptor) that shall have its own PID.

S.4 Optional J2K stripe mode for Ultra-Low Latency

The J2K stripe mode has been defined to allow carriage of JPEG 2000 video in an MPEG-2 transport stream with a sub-frame end-to-end latency. This mode is implemented through an horizontal striping mechanism: the picture embedded in a J2K access unit (whether it be an entire progressive video frame, the two fields of an interlaced video frame, a progressive video block or the two fields of an interlaced video block as defined in S.3) is divided horizontally into an integer number of stripes, each stripe being encoded as a self-contained Rec. ITU-T T.800 | ISO/IEC 15444-1 codestream.

To produce a compliant Ultra-Low Latency (ULL) J2K video stream, the J2K video descriptor (see Table 2-99) shall have the following information set accordingly

- The *extended_capability_flag* and the *stripe_flag* both set to '1'.
- The fields *strp_max_idx* and *strp_height* respectively indicating the number of stripes per frame minus one, and the stripe height.

In the JPEG 2000 Elementary Stream header, they shall ensure the Stripe Coding box (*j2k_strp*) is present, and they shall ensure the Time Coding box (*j2k_tcod*) is not present. Stripes shall be included in raster order, from top to bottom. All stripes shall have the same size, with the exception of the very last (bottom) one, which may have a different height. The stripe height is given by the following formula:

```

if last stripe in the frame/field/block:
    height = vertical_size - (strp_height * strp_max_idx)
otherwise:
    height = strp_height

```

When this mode is enabled, each J2K access unit contains *N* Rec. ITU-T T.800 | ISO/IEC 15444-1 codestreams where:

$$N = (\text{strp_max_idx} + 1) \text{ for progressive content}$$

$$N = 2 * (\text{strp_max_idx} + 1) \text{ for interlaced content}$$

7) Annex S, new clause S.5

Replace old clause S.3 with the following:

S.5 Elementary stream header (elsm) and mapping to PES packets

Each J2K access unit from a J2K video elementary stream shall start with an elementary stream header (elsm header) as detailed in Table S.1.

Table S.1 – J2K Access unit elementary stream header

| Syntax | No. of bits | Mnemonic |
|---|-------------|----------|
| J2K_elsm_header() { | | |
| elsm_box_code '0x656c 736d' | 32 | bslbf |
| // j2k_frat | | |
| frat_box_code '0x6672 6174' | 32 | bslbf |
| frat_denominator | 16 | uimsbf |
| frat_numerator | 16 | uimsbf |
| // j2k_brat | | |
| brat_box_code '0x6272 6174' | 32 | bslbf |
| brat_max_br | 32 | uimsbf |
| brat_auf1 | 32 | uimsbf |
| If (interlaced_video == 1) { | | |
| brat_auf2 | 32 | uimsbf |
| } | | |
| // j2k_fiel | | |
| If (interlaced_video == 1) { | | |
| fiel_box_code '0x6669 656c' | 32 | bslbf |
| fiel_fc | 8 | uimsbf |
| fiel_fio | 8 | uimsbf |
| } | | |
| If (extended_capability_flag == 0 stripe_flag == 0) { | | |
| // j2k_tcod | | |
| tcod_box_code '0x7463 6f64' | 32 | bslbf |
| tcod_hh (0-23) | 8 | uimsbf |
| tcod_mm (0-59) | 8 | uimsbf |
| tcod_ss (0-59) | 8 | uimsbf |
| tcod_ff (1-60) | 8 | uimsbf |
| } else { // extended_capability_flag == 1 && stripe_flag == 1 | | |
| // j2k_strp | | |
| strp_box_code '0x7374 7270' | 32 | bslbf |
| strp_max_idx | 8 | uimsbf |
| frame_vertical_size | 16 | uimsbf |
| reserved | 8 | bsfbf |
| } | | |
| // j2k_bcol | | |
| If (extended_capability_flag == 0) { | | |
| bcol_box_code '0x6263 6f6c' | 32 | bslbf |
| bcol_colcr | 8 | uimsbf |
| reserved | 8 | bsfbf |
| } else { // extended_capability_flag == 1 | | |
| color primaries | 8 | uimsbf |
| transfer characteristics | 8 | uimsbf |
| matrix coefficients | 8 | uimsbf |
| video_full_range_flag | 1 | bsfbf |
| reserved | 23 | bsfbf |
| } | | |
| } | | |

Semantics

j2k_frat – The two fields *frat_denominator* and *frat_numerator* are used to derive the frame rate in frames per second. If either is zero, the frame rate is variable or undefined. Otherwise, the frame rate is expressed by a rational number of the form numerator/denominator. If the frame rate is an integer, the denominator shall be equal to 1. In case of interlaced content (two fields per frame), then the field rate is twice the frame rate.

frat_denominator – This field is defined as a 2-byte big-endian unsigned integer. It specifies the denominator used to calculate the frame rate.

frat_numerator – The field is defined as a 2-byte big-endian unsigned integer. It specifies the numerator used to calculate the frame rate.

j2k_brat – The three fields *brat_max_br*, *brat_auf1* and *brat_auf2* specify the maximum bit rate of the elementary stream and the actual access unit size.

brat_max_br – This field is defined as a 4-byte big-endian unsigned integer which specifies a maximum instantaneous bit rate that is not to be exceeded, expressed in bits per second for the elementary stream at the frame rate specified by *j2k_frat* fields. The maximum bit rate shall not exceed the bit rate specified for a given profile and level.

brat_auf1 – This field is defined as a 4-byte big-endian unsigned integer which specifies the size of the contiguous codestream corresponding to the frame in case of progressive content and to the first field in case of interlaced content. This field shall be set to zero if J2K stripe mode is enabled (*stripe_flag* == 1).

brat_auf2 – This field is defined as a 4-byte big-endian unsigned integer which specifies the size of the contiguous codestream corresponding to the second field in case of interlaced content. This field shall be omitted in case of progressive content. This field shall be set to zero in case of interlaced content and J2K stripe mode enabled.

j2k_fiel – The fields *fiel_fic* and *fiel_fio* specify the field order in case of interlaced content. These two fields, together with the field *fiel_box_code* shall only be present in case of interlaced content (*interlaced_video* flag in the *j2k_video_descriptor* set to '1').

fiel_fic – This field is defined as a 1-byte unsigned integer which specifies the number of fields in the access unit. As this field is only present in case of interlaced content, it shall always be set to 2, and is only present for backward compatibility reasons.

fiel_fio – This field is defined as a 1-byte unsigned integer which describes the order of the two fields:

0 field coding unknown;

1 field with the topmost line is stored first in the access unit; fields are in temporal order;

6 field with the topmost line is stored second in the access unit; fields are in temporal order.

j2k_tcod – The fields *tcod_hh*, *tcod_mm*, *tcod_ss*, and *tcod_ff* specify the time code of the access unit. These fields, together with *tcod_box_code* shall only be present if the J2K stripe mode is disabled (*stripe_flag* in the *j2k_video_descriptor* set to '0').

tcod_hh – This field is a 1-byte unsigned integer which specifies the hour. The value ranges from 0 to 23, inclusively.

tcod_mm – This field is a 1-byte unsigned integer which specifies the minutes. The value ranges from 0 to 59, inclusively.

tcod_ss – This field is a 1-byte unsigned integer which specifies the seconds. The value ranges from 0 to 59, inclusively.

tcod_ff – This field is a 1-byte unsigned integer which specifies the frame count. The value ranges from 1 to 60, inclusively.

j2k_strp – The fields *strp_max_idx* and *frame_vertical_size* specify the settings for the striping mechanism. These fields, together with *strp_box_code* and the *reserved* field shall only be present if the J2K stripe mode is enabled (*extended_capability_flag* and *stripe_flag* in the *j2k_video_descriptor* both set to '1').

strp_max_idx – This field is a 1-byte unsigned integer which specifies the number of J2K stripes per frame (for progressive content) or per field (for interlaced content) in this access unit, minus one. This value shall be equal to the *strp_max_idx* value found in the *j2k_video_descriptor*.

frame_vertical_size – This field is a 2-byte unsigned integer which specifies the vertical size of the frame (for progressive) or field (for interlaced) comprised in this J2K access unit. This value shall be equal to the *vertical_size* value found in the *j2k_video_descriptor*.

j2k_bcol – The *j2k_bcol* fields described hereunder specify the color parameters of the access unit. They define one method by which an application can interpret the colourspace of the decompressed access unit. This colour specification is to be applied to the access unit after it has been decompressed and after any reverse decorrelating component transform has been applied to the image data.

bcol_colcr – This field is a 1-byte integer which specifies the corresponding colour standard as defined in Table M.2 of Rec. ITU-T T.800 | ISO/IEC 15444-1. It shall only be present if, in the J2K video descriptor, *extended_capability_flag* is set to '0'.

NOTE – In Table S.1, it should be noted that the *bcol_box_code* is only present when the *bcol_colcr* field is present. It is indeed only present for a purpose of backward compatibility with previous version of this Recommendation | International Standard.

color_primaries, transfer_characteristics, matrix_coefficients, video_full_range_flag – These four fields (three 1-byte integers and one 1-bit flag) shall be coded according to the semantics with the same name defined in Rec. ITU-T H.273 | ISO/IEC 23001-8. They shall only be present if, in the J2K video descriptor *extended_capability_flag* is set to '1'.

Figure S.1 shows the structure and mapping of J2K video access unit into PES packets. JPEG 2000 represents each frame or block (if J2K block mode is enabled) as N Part 1 (Rec. ITU-T T.800 | ISO/IEC 15444-1) contiguous codestreams, where

$N = 1$, for progressive content, J2K stripe mode disabled

2, for interlaced content, J2K stripe mode disabled

$strp_max_idx + 1$, for progressive content, J2K stripe mode enabled

$2*(strp_max_idx + 1)$, for interlaced content, J2K stripe mode enabled

The codestream main header, included within each contiguous codestream, contains all information to decode its image, including the image size and the profile indicator, called a SIZ marker in Rec. ITU-T T.800 | ISO/IEC 15444-1. Preceding the contiguous codestreams, an elementary stream (elsm) header containing video-related information, as shown in Table S.1, shall be present.

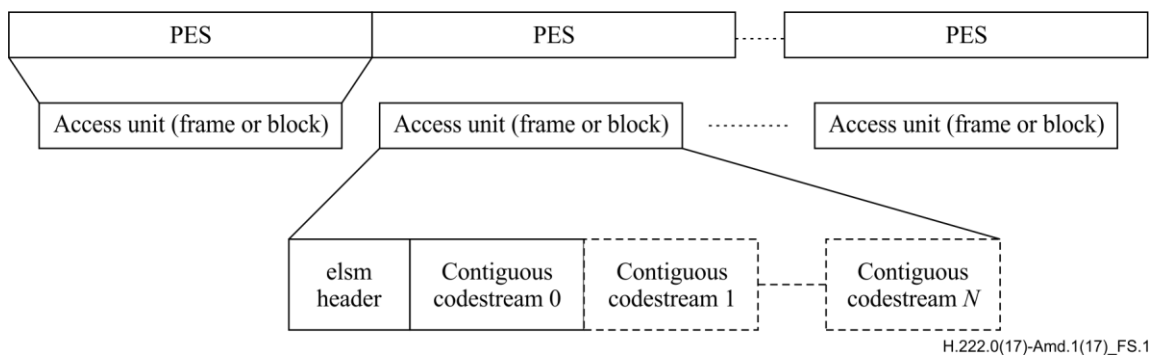


Figure S.1 – Structure and order of JPEG 2000 access units

8) Annex S, clause S.6

Replace the first two items from the list in old clause S.4 with the following:

- 1) Each J2K access unit shall contain an elementary stream header (elsm) defined in Table S.1 followed by N Rec. ITU-T T.800 | ISO/IEC 15444-1 codestream(s) (see S.5).
- 2) Each J2K codestream main header contains a SIZ marker segment that includes a RSIZ capability parameter. Both the SIZ marker segment and RSIZ capability parameter are defined in Annex A of Rec. ITU-T T.800 | ISO/IEC 15444-1. The 15 less significant bits of the RSIZ parameter are equal to the *profile_and_level* parameter of the *j2k_video_descriptor* (see 2.6.81).

9) Annex S, clause S.8.1

Replace old clause S.6.1 with the following:

S.8.1 J2K video elementary stream buffer size

A J2K video elementary stream supplies parameters suitable for determining a standard decoder model buffer size. This CODEC uses variable rate compression. Profiles from Rec. ITU-T T.800 (2015) | ISO/IEC 15444-1:2016 specify operating levels that limit the maximum compressed bit rate of the Rec. ITU-T T.800 | ISO/IEC 15444-1 codestream. Moreover, within access units an elementary stream header contains the present maximum bit rate an encoder must support to decode video without overflow. A decoder may scale the buffer size by reading the maximum bit rate expected by a particular sequence of frames from the elsm header. This header information shall not exceed the limit set by the operating level specified in Rec. ITU-T T.800 (2015) | ISO/IEC 15444-1:2016, Table A.49.

10) Annex S, Table S.2

Replace old Table S.2 with the following:

**Table S.2 – Operating levels and maximum buffer size for JPEG 2000 broadcast profiles
(from Table A.49 in Rec. ITU-T T.800 (2015) | ISO/IEC 15444-1:2016)**

| Levels (Note) | Max. compressed bit rate (Mbit/s) | Maximum buffer size for a 60.0 Hz frame rate, progressive video (MBytes) |
|------------------|-----------------------------------|--|
| Level 0 | Unspecified | Unspecified |
| Level 1 | 200 | 1.25 |
| Level 2 | 200 | 1.25 |
| Level 3 | 200 | 1.25 |
| Level 4 | 400 | 2.5 |
| Level 5 | 800 | 5 |
| Level 6 | 1600 | 10 |
| Level 7 | 3200 | 20 |
| Level 8 | 6400 | 40 |
| Level 9 | 12800 | 80 |
| Level 10 | 25600 | 160 |
| Level 11 | 51200 | 320 |

NOTE – Levels are specified in the JPEG 2000 codestream main header SIZ marker segment, Rsiz capability parameter. Refer to Rec. ITU-T T.800 (2015) | ISO/IEC 15444-1:2016, Table A.10.

SERIES OF ITU-T RECOMMENDATIONS

| | |
|-----------------|---|
| Series A | Organization of the work of ITU-T |
| Series D | Tariff and accounting principles and international telecommunication/ICT economic and policy issues |
| 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 | Cable networks and transmission of television, sound programme and other multimedia signals |
| Series K | Protection against interference |
| Series L | Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant |
| Series M | Telecommunication management, including TMN and network maintenance |
| 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, and associated measurements and tests |
| 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, open system communications and security |
| Series Y | Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities |
| Series Z | Languages and general software aspects for telecommunication systems |