INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

T.803

(11/2002)

SERIES T: TERMINALS FOR TELEMATIC SERVICES

Information technology – JPEG 2000 image coding system: Conformance testing

ITU-T Recommendation T.803

INTERNATIONAL STANDARD ISO/IEC 15444-4
ITU-T RECOMMENDATION T.803

Information technology – JPEG 2000 image coding system: Conformance testing

Summary

ITU-T Rec. T.800 | ISO/IEC 15444-1 is a specification that describes an image compression system that allows great flexibility, not only for the compression of images but also for access into the codestream. The codestream provides a number of mechanisms for locating and extracting portions of the compressed image data for the purpose of retransmission, storage, display, or editing. This access allows storage and retrieval of compressed image data appropriate for a given application without decoding.

This Recommendation | International Standard provides the framework, concepts, and methodology for testing and the criteria to be achieved to claim compliance to ITU-T Rec. T.800 | ISO/IEC 15444-1. The objective of standardization in this field is to promote interoperability between JPEG 2000 encoders and decoders and to test these systems for compliance to this Specification. Compliance testing is the testing of a candidate product for the existence of specific characteristics required by a standard. It involves testing the capabilities of an implementation against both the compliance requirements in the relevant standard and the statement of the implementation's capability.

The purpose of this Recommendation | International Standard is to define a common test methodology, to provide a framework for specifying abstract test suites (ATS), and to define the procedures to be followed during compliance testing.

Any organization contemplating the use of test methods defined in this Recommendation | International Standard should carefully consider the constraints on their applicability. Compliance testing does not include robustness testing, acceptance testing, and performance testing.

Source

ITU-T Recommendation T.803 was prepared by ITU-T Study Group 16 (2001-2004) and approved on 29 November 2002. An identical text is also published as ISO/IEC 15444-4.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications. 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 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

LIST OF FIGURES

LIST OF TABLES

INTERNATIONAL STANDARD
ITU-T RECOMMENDATION

Information technology – JPEG 2000 image coding system: Conformance testing

1 Scope

This Recommendation | International Standard specifies the framework, concepts, methodology for testing, and criteria to be achieved to claim compliance to ITU-T Rec. T.800 | ISO/IEC 15444-1. It provides a framework for specifying abstract test suites and for defining the procedures to be followed during compliance testing.

This Recommendation | International Standard:

- specifies compliance testing procedures for encoding and decoding using JPEG 2000 Part 1 (ITU-T Rec. T.800 | ISO/IEC 15444-1);
- specifies codestreams, decoded images, and error metrics to be used with the testing procedures;
- specifies abstract test suites;
- provides guidance for creating an encoder compliance test.

This Recommendation | International Standard does not include the following tests:

Acceptance testing: the process of determining whether an implementation satisfies acceptance criteria and enables the user to determine whether or not to accept the implementation. This includes the planning and execution of several kinds of tests (e.g., functionality, quality, and speed performance testing) that demonstrate that the implementation satisfies the user requirements.

Performance testing: measures the performance characteristics of an Implementation Under Test (IUT) such as its throughput, responsiveness, etc., under various conditions.

Robustness testing: the process of determining how well an implementation processes data which contains errors.

2 Normative references

The following references document the technical details and legal codestream syntax that serve as a basis for this Recommendation | International Standard. At the time of publication, the editions indicated were valid. All Recommendations and Standards are subject to revision, and parties to agreements based on this Recommendation | International Standard are encouraged to investigate the possibility of applying the most recent edition of the Recommendations and Standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau of the ITU maintains a list of currently valid ITU-T Recommendations.

- ITU-T Recommendation T.800 (2002) | ISO/IEC 15444-1:2003, *Information technology – JPEG 2000 image coding system: Core coding system.*

3 Definitions

For the purposes of this Recommendation | International Standard, the following definitions apply. The definitions from ITU-T Rec. T.800 | ISO/IEC 15444-1 clause 3 also apply to this Recommendation | International Standard.

3.1 **abstract test suite**: Generic compliance testing concepts and procedures for a given requirement.

3.2 **arithmetic coder**: An entropy coder that converts variable length strings to variable length codes (encoding) and vice versa (decoding).

3.3 **big endian**: An order of bytes with the most significant byte first.

3.4 **bit**: A contraction of the term "binary digit"; a unit of information represented by a zero or a one.

3.5 **bit-depth**: The number of bits required to represent an original component of an image.

3.6 **bit-plane**: A two-dimensional array of bits. In this Recommendation | International Standard, a bit-plane refers to all the bits of the same magnitude in all coefficients or samples. This could refer to a bit-plane in a component, tile-component, code-block, region of interest, or other.

3.7 **bitstream**: The actual sequence of bits resulting from the coding of a sequence of symbols. It does not include the markers or marker segments in the main and tile-part headers or the EOC marker. It does include any packet headers and in stream markers and marker segments not found within the main or tile-part headers.

3.8 **box**: A portion of the file format defined by a length and unique box type. Boxes of some types may contain other boxes.

3.9 **byte**: Eight bits.

3.10 **Cclass**: Defines a level of performance for a decoder. Also provides guidance for encoders to produce codestreams that are easily decodable by compliant decoders.

3.11 **code-block**: A rectangular grouping of coefficients from the same sub-band of a tile-component.

3.12 **coder**: An embodiment of either an encoding or decoding process.

3.13 **codestream**: A collection of one or more bitstreams and the main header, tile-part headers, and the EOC required for their decoding and expansion into image data. This is the image data in a compressed form with all of the signalling needed to decode. This does not include the file format.

3.14 **coding pass**: A procedure accessing coefficients in a code-block where the context and bit are determined. Typically there are three different coding passes for each bitplane, each coefficient will be represented in exactly one of the three passes. For an encoder a coding pass examines coefficients and augments a bitstream. For a decoder a coding pass reads a bitstream and updates coefficients.

3.15 **coefficient**: The values that are the result of a transformation.

3.16 **component**: A two-dimensional array of samples. An image typically consists of several components (e.g., red, green, and blue).

3.17 **compressed image data**: Part or all of a codestream. Can also refer to a collection of bitstreams in part or all of a codestream.

3.18 **compliance**: Fulfilment of the specified requirements, as defined in this Specification, for a given Profile and Cclass.

3.19 **compliance test procedure**: The process of assessing compliance.

3.20 **context**: Function of coefficients previously decoded and used to condition the decoding of the present coefficient.

3.21 **decoder**: An embodiment of a decoding process, and optionally a colour transformation process.

3.22 **decoding process**: A process that takes as its input all or part of a codestream and outputs all or part of a reconstructed image.

3.23 **decomposition level**: A collection of wavelet sub-bands where each coefficient has the same spatial impact or span with respect to the source component samples. These include all sub-bands of the same two-dimensional sub-band decomposition. For the last decomposition level, the LL sub-band is also included.

3.24 **Discrete Wavelet Transformation (DWT)**: A transformation that iteratively transforms one signal into two or more filtered and decimated signals corresponding to different frequency bands. This transformation operates on spatially discrete samples.

3.25 **encoder**: An embodiment of an encoding process, and optionally a colour transformation process.

3.26 **encoding process**: A process that takes as its input all or part of a source image data and outputs a codestream.

3.27 **executable test suite**: Set of executable test cases that support the abstract test cases.

3.28 **file format**: A codestream and additional support data and information not explicitly required for the decoding of the codestream. Examples of such support data include text fields providing titling, security and historical information, data to support placement of multiple codestreams within a given data file, and data to support exchange between platforms or conversion to other file formats.

3.29 **fully decode**: Applying ITU-T Rec. T.800 | ISO/IEC 15444-1 to produce an image from a codestream where all coded data in the codestream has been used to produce the image.

3.30 **guard bits**: Additional most significant bits that have been added to sample data.

3.31 **header**: Either a part of the codestream that contains only markers and marker segments (main header and tile part header) or the signalling part of a packet (packet header).

3.32 **image**: The set of all components.

3.33 **image data**: The component samples making up an image. Image data can refer to either the source image data or the reconstructed image data.

3.34 **implementation**: A realization of a specification.

3.35 **Implementation Compliance Statement (ICS)**: Statement of specification options and the extent to which they have been implemented by an implementation under test.

3.36 **Implementation Under Test (IUT)**: An implementation that is being evaluated for compliance.

3.37 **irreversible**: A transformation, progression, system, quantization, or other process that, due to systemic or quantization error, prevents lossless recovery.

3.38 **JP2 file**: The name of a file in the file format described in this Specification. Structurally, a JP2 file is a contiguous sequence of boxes.

3.39 **JPEG**: Joint Photographic Experts Group – The joint ISO/ITU committee responsible for developing standards for continuous-tone still picture coding. It also refers to the standards produced by this committee: ITU-T Rec. T.81 | ISO/IEC 10918-1, ITU-T Rec. T.83 | ISO/IEC 10918-2, ITU-T Rec. T.84 | ISO/IEC 10918-3 and ITU-T Rec. T.87 | ISO/IEC 14495-1.

3.40 **LL sub-band**: The sub-band obtained by forward horizontal low-pass filtering and vertical low-pass filtering. This sub-band contributes to reconstruction with inverse vertical low-pass filtering and horizontal low-pass filtering.

3.41 **layer**: A collection of compressed image data from coding passes of one, or more, code-blocks of a tile-component. Layers have an order for encoding and decoding that must be preserved.

3.42 **lossless**: A descriptive term for the effect of the overall encoding and decoding processes in which the output of the decoding process is identical to the input to the encoding process. Distortion-free restoration can be assured. All of the coding processes or steps used for encoding and decoding are reversible.

3.43 **lossy**: A descriptive term for the effect of the overall encoding and decoding processes in which the output of the decoding process is not identical to the input to the encoding process. There is distortion (measured mathematically). At least one of the coding processes or steps used for encoding and decoding is irreversible.

3.44 **main header**: A group of markers and marker segments at the beginning of the codestream that describe the image parameters and coding parameters that can apply to every tile and tile-component.

3.45 **marker**: A two-byte code in which the first byte is hexadecimal FF (0xFF) and the second byte is a value between 1 (0x01) and hexadecimal FE (0xFE).

3.46 **marker segment**: A marker and associated (not empty) set of parameters.

3.47 **packet**: A part of the codestream comprising a packet header and the compressed image data from one layer of one precinct of one resolution level of one tile-component.

3.48 **packet header**: Portion of the packet that contains signalling necessary for decoding that packet.

3.49 **parser**: Reads and identifies components of the codestream down to the code-block level.

3.50 **partial decoding**: Producing an image from a subset of an entire codestream.

3.51 **precinct**: A rectangular region of a transformed tile-component, within each resolution level, used for limiting the size of packets.

3.52 **precision**: Number of bits allocated to a particular sample, coefficient, or other binary numerical representation.

3.53 **progression**: The order of a codestream where the decoding of each successive bit contributes to a "better" reconstruction of the image. What metrics make the reconstruction "better" is a function of the application. Some examples of progression are increasing resolution or improved sample fidelity.

3.54 **profile**: A subset of technology, from ITU-T Rec. T.800 | ISO/IEC 15444-1, that meets the needs of a given application with limits on parameters within a selected technology. This is a codestream limitation.

3.55 **quantization**: A method of reducing the precision of the individual coefficients to reduce the number of bits used to represent them. This is equivalent to division while compressing and multiplying while decompressing. Quantization can be achieved by an explicit operation with a given quantization value (scalar quantization) or by dropping (truncating) coding passes from the codestream.

3.56 **reconstructed image**: An image that is the output of a decoder.

3.57 **reference grid**: A regular rectangular array of points used to define other rectangular arrays of data. The reference grid is used to determine the number of samples in tile-components for example.

3.58 **Region Of Interest (ROI)**: A collection of coefficients that are considered of particular relevance by some user-defined measure.

3.59 **reversible**: A transformation, progression, system, or other process that does not suffer systemic or quantization error and therefore allows for lossless signal recovery.

3.60 **reversible filter**: A particular filter pair used in the wavelet transformation which allows lossless compression.

3.61 **sample**: One element in the two-dimensional array that comprises a component.

3.62 **selective arithmetic coding bypass**: A coding style where some of the code-block passes are not coded by the arithmetic coder. Instead, the bits to be coded are appended directly to the bitstream without coding.

3.63 **shift**: Multiplication or division of a number by powers of two. Division of an integer via shift implies truncation toward minus infinity of the non-integer portion.

3.64 **sign bit**: A bit that indicates whether a number is positive (zero value) or negative (one value).

3.65 **sign-magnitude notation**: A binary representation of an integer where the distance from the origin is expressed with a positive number and the direction from the origin (positive or negative) is expressed with a separate single sign bit.

3.66 **source image**: An image used as input to an encoder.

3.67 **sub-band**: A group of transform coefficients resulting from the same sequence of low-pass and high-pass filtering operations, both vertically and horizontally.

3.68 **testing**: The process of evaluating compliance.

3.69 **tile**: A rectangular array of points on the reference grid, registered with an offset from the reference grid origin and defined by a width and height.

3.70 **tile-component**: All the samples of a given component in a tile.

3.71 **tile-part**: A portion of the codestream with compressed image data for some, or all, of a tile. The tile-part may include one or more packets that make up the coded tile.

3.72 **tile-part header**: A group of markers and marker segments at the beginning of each tile-part in the codestream that describe the tile-part coding parameters.

3.73 **transformation**: A mathematical mapping from one signal space to another.

3.74 **transform coefficient**: A value that is the result of a transformation.

4 Abbreviations

For the purposes of this Recommendation | International Standard, the following abbreviations apply. The abbreviations defined in ITU-T Rec. T.800 | ISO/IEC 15444-1 subclause 4.1 also apply to this Recommendation | International Standard.

ATS	Abstract Test Suite
CCITT	International Telegraph and Telephone Consultative Committee, now ITU-T
ETS	Executable Test Suite
ICC	International Colour Consortium
ICT	Irreversible Component Transform
ICS	Implementation Compliance Statement
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization

ITU	International Telecommunication Union
ITU-T	International Telecommunication Union – Telecommunication Standardization Sector (formerly the CCITT)
IUT	Implementation Under Test
JPEG	Joint Photographic Experts Group
MSE	Mean Squared Error
RCT	Reversible Component Transform
ROI	Region of Interest
TCS	Test Codestream

5 Symbols

For the purposes of this Recommendation | International Standard, the following symbols apply. The symbols defined in ITU-T Rec. T.800 | ISO/IEC 15444-1 subclause 4.2 also apply to this Recommendation | International Standard.

0x----	Denotes a hexadecimal number
B	Bitdepth accuracy for reversible 5-3
C	Components guaranteed to be decoded
COC	Coding style component marker
COD	Coding style default marker
COM	Comment marker
CRG	Component registration marker
EPH	End of packet header marker
EOC	End of codestream marker
H	Image height guarantee
L	Layer guarantees
L_{body}	Code data buffering guarantee
M	Decoded bitplane guarantee
N_{cb}	Code-block parsing guarantee
N_{comp}	Component parsing guarantee
P	Irreversible 9-7 precision guarantee
PLM	Packet length, main header marker
PLT	Packet length, tile-part header marker
POC	Progression order change marker
PPM	Packed packet headers, main header marker
PPT	Packed packet headers, tile-part header marker
QCC	Quantization component marker
QCD	Quantization default marker
RGN	Region of interest marker
SIZ	Image and tile size marker
SOC	Start of codestream marker
SOP	Start of packet marker
SOD	Start of data marker
SOT	Start of tile-part marker
T_L	Transform level guarantees
TLM	Tile-part lengths marker
W	Image width guarantee

6 General description

Perhaps the most distinctive feature of JPEG 2000 is its emphasis on and support for scalability. An existing codestream may be accessed at a reduced resolution, at a reduced quality (higher compression), at a reduced number of components, and even over a reduced spatial region. Moreover, this Recommendation | International Standard supports a rich family of information progression sequences whereby the information may be reordered without introducing additional distortion. This enables a single compressed codestream to serve the needs of a diverse range of applications.

JPEG 2000 encoders may employ only a fraction of the features supported by ITU-T Rec. T.800 | ISO/IEC 15444-1. Likewise, some decoders will not support all the features supported by this Recommendation | International Standard. It is impossible to provide test cases for all possible combinations of tools that an encoder or decoder may choose to implement. This Recommendation | International Standard provides abstract test procedures for JPEG 2000 encoders and decoders. A developer may designate the features that have been implemented and determine a set of test cases that applies to those features. For the greatest level of interoperability, there are explicit decoder test procedures. These tests are run for a particular Profile (defined in ITU-T Rec. T.800 | ISO/IEC 15444-1) and a particular compliance class defined herein. Passing the explicit tests allows a decoder to be labelled "Profile-x Cclass-y Compliant".

Even with the explicit decoder tests, it is expected that some decoders may not decode all of the information that was originally incorporated into the codestream by an encoder. This is the only way to truly exploit the scalability of ITU-T Rec. T.800 | ISO/IEC 15444-1. It is desirable to allow decoders to ignore information that is not of interest to their target application. While this flexibility is one of the strengths of JPEG 2000, it also renders inappropriate some of the conventional compliance testing methodologies that have been applied to non-scalable or less scalable compression standards.

Many approaches to compliance could be taken. At one extreme, decoder implementations might be allowed to decode any portion of the codestream that is of interest to them. At the other extreme, they might be required to correctly decode the entire codestream. The first approach offers content providers and consumers no guarantee concerning the quality of the resulting imagery. The other approach is also inappropriate because it offers the implementor no guarantee concerning the resources that may be required for decoding, and in many cases the codestream may contain information that is of no interest to the application.

This Recommendation | International Standard describes compliance for JPEG 2000 decoders in terms of a system of guarantees. These guarantees serve to discourage encoders from producing codestreams that will be exceedingly difficult or impossible for a decoder to process, to encourage decoders to provide quality images from any reasonable codestream, and to encourage use of the flexibility and scalability of JPEG 2000 codestreams.

Profiles define a subset of technology, from ITU-T Rec. T.800 | ISO/IEC 15444-1: JPEG 2000, that meets the needs of a given application with limits on parameters within a selected technology. Profiles limit bitstreams. Decoders define capabilities for all bitstreams within a profile. Encoders achieve quality guarantees for particular decoders by encoding bitstreams which meet a particular profile definition. *Compliance classes* (Cclass) define guarantees of a given level of image quality for a decoder and guidance for encoders to produce codestreams that are easily decodable by compliant decoders.

Essentially, if a JPEG 2000 encoder produces a codestream with certain properties, then a decoder of a certain Cclass will be capable of producing an image with some defined level of quality. The compliance class of a decoder is based solely on passing certain tests. The tests in this Recommendation | International Standard are designed to require a compliant decoder to be capable of decoding all codestreams with a set of defined properties.

6.1 Profiles and compliance classes

Two profiles are defined in ITU-T Rec. T.800 | ISO/IEC 15444-1, labelled Profile 0 and Profile 1. The two profiles describe bitstream constraints for an ITU-T Rec. T.800 | ISO/IEC 15444-1 encoder. Profile 0 is a subset of Profile 1. Hence, any implementation capable of decoding Profile 1 test streams shall be capable of passing the compliance tests for Profile 0 of the same Cclass.

Three compliance classes (Cclass) are defined in Annex A. These Cclasses define levels of image quality guarantees for decoders and guidance for encoders to produce codestreams that are easily decodable by compliant decoders. Cclass guarantees increase with the increasing Cclass numbers.

6.2 Decoders

Compliant implementations of the decoder are not required to decode each codestream in its entirety but are required to guarantee performance up to one Cclass for some profile. These guarantees are directly connected with the resources required by a decoder. They may be interpreted as a contract by the implementation to recover, decode and transform a well-defined minimal subset of the information contained in any codestream. This contract is described in a manner that scales with the Cclass. The contract may be exploited by content providers to optimize recovered image quality over a family of decoders according to their known Cclasses.

For a given profile, decoder guarantees are expressed in terms of several parameters including decoded image dimensions, H (height) and W (width), and a number of components, C, for the Cclass. The parameters are not dependent on the codestream that is actually being decoded. Annex A defines the parameters and the classes for which compliance claims may be made and tested. Annex E allows a decoder to define guarantees that are greater than that of the defined Cclasses.

6.3 Encoders and codestreams

ITU-T Rec. T.800 | ISO/IEC 15444-1 describes two restricted profiles (Profile 0 and Profile 1) that provide guarantees concerning the parameter ranges and information placement in a codestream. Since codestream limitations may also adversely affect scalability and interoperability, the smallest possible number of limitations are imposed by these profiles. Annex F allows an encoder to define guarantees of codestreams produced by the encoder.

Encoders may also be required to conform to certain guarantees in particular application areas of interest that are outside the scope of this Recommendation | International Standard. As an example, a medical image application may require the encoder to guarantee lossless performance up to a given image size.

6.4 Implementation compliance statement

Evaluation of compliance for a particular implementation may require a statement of the options that have been implemented. This will allow the implementation to be tested for compliance against only the relevant requirements.

Such a statement is called an ICS. This statement shall contain only options within the framework of requirements specified in ITU-T Rec. T.800 | ISO/IEC 15444-1. Examples of these can be found in Annex E for decoders and Annex F for encoders.

6.5 Abstract test suites

The Abstract Test Suites (ATS) define general tests for sub-systems of ITU-T Rec. T.800 | ISO/IEC 15444-1. Each ATS includes the following parts and are defined in Annex C.

a) Test purpose: What the test requirement is.

b) Test Method: The procedures to be followed for the given ATS.

c) Reference: The portion of the ISO document that is being tested by the given ATS.

6.6 Encoder compliance testing procedure

The informative procedures for testing encoders are defined in Annex D and are complemented by the information that is gathered from a completed ICS (Annex F). These procedures are informative since ITU-T Rec. T.800 | ISO/IEC 15444-1 is a codestream and decoder compliance standard.

6.7 Decoder compliance testing procedure

The procedures for testing decoders are defined in Annex B and use the ETS that are defined in Annex C. These procedures and ETS will allow an IUT to evaluate compliance to each profile and Cclass.

7 Conventions

The compliance files including test codestreams, jp2 files, reference decoded images, and descriptive files are supplied in the form of a compressed file. File locations given in this Recommendation | International Standard are expressed relative to the top level of the directory tree. A Unix style file structure and delimiters are assumed.

This Recommendation | International Standard contains instructions for the use of these files. No support can be provided by ISO | ITU-T beyond that offered in this Recommendation | International Standard and through links on the official JPEG web site, http://www.jpeg.org.

8 Copyright

These compliance files were originally developed by the parties indicated in the file **COPYRIGHT.** In particular, the original developers of these files and their respective companies, the editors and their companies, and ISO/IEC have disclaimed liability for any proposed use of these files or modifications thereof.

9 Compliance files availability and updates

The compliance test images released with this Recommendation | International Standard are the latest tested versions available at the date at which the text was approved. Later versions of the compliance test images may be made available after the publication of this Recommendation | International Standard. These may be found through links that are maintained at http://www.jpeg.org/software.

Annex A

Decoder compliance classes

(This annex does not form an integral part of this Recommendation | International Standard)

This annex describes the compliance classes for ITU-T Rec. T.800 | ISO/IEC 15444-1. The classes and parameters are described to provide assistance in designing a compliant decoder. Actual compliance is determined by the testing procedures in Annex B and the codestreams, reference images, and tolerances in Annex C. The definitions of compliance classes in this annex are useful for the design of an encoder. The parameters may correspond to particular parts of an implementation.

A.1 Compliance class parameter definitions

Because of resource limitations, implementations of JPEG 2000 sometimes will not be able to decode a codestream in its entirety. This subclause defines various parameters for which a specific implementation might be limited. A set of values for every parameter defines a compliance class. Thus an implementation of a particular Cclass must guarantee resources as defined in all the parameters.

A.1.1 Profile: codestream guarantees

Profiles provide limits on the codestream syntax parameters. Two profiles are defined in ITU-T Rec. T.800 | ISO/IEC 15444-1, labelled Profile 0 and Profile 1. There is also an implied profile that has no restrictions above the descriptive capability of Annex A of ITU-T Rec. T.800 | ISO/IEC 15444-1. Testing arbitrary codestreams requires almost unlimited computational resources. Thus, all defined tests pertain to a specific profile and will only use codestreams compliant to that profile.

A.1.2 *H*, *W*, *C*: Image size guarantees

Decoders may be limited in the size of the output image that they are capable of producing, due to physical display characteristics or memory limitations. H, W, and C are respectively the largest height, width, and number of components that are required to be decoded for a decoder in the compliance class. Codestreams containing more samples than the H, W, and C for a Cclass must still be decoded. Compliance for these codestreams is based on the ability to decode at the largest resolution smaller than or equal to that specified by the decoder's Cclass. This largest resolution size is determined by the component in the codestream with the smallest sampling factor relative to the reference grid. For images with different subsampling on different components, the smallest subsampling values are used to determine the region that must be decoded, and the corresponding region of each component shall be decoded.

For Cclass 0, a decoder must be able to decode the largest resolution level that is no larger than 128 samples in either image dimension and that requires no more than three levels of IDWT processing. For Cclass 1, a decoder must be able to decode the largest resolution level that is no larger than 2048 samples in either image dimension and that requires no more than seven levels of IDWT processing.

Equations A-1 and A-2 express these restrictions for a single tile image. The minimization with respect to *XR*siz over all components selects the component with the smallest sampling relative to the reference grid. The maximum $r \leq T_L$ that satisfies both conditions is the number of levels that must be decoded. T_L is defined in A.1.9. Xsiz, Ysiz, XOsiz, and YOsiz come from the SIZ marker segment defined in ITU-T Rec. T.800 | ISO/IEC 15444-1. N_L is the number of wavelet transforms used in the image as specified in the COD or COC marker segments. (If a non-negative r does not exist to satisfy both conditions, then no decoder obligation exists.)

$$\left\lceil \frac{X_{siz}}{\min\limits_{\forall i}(XR_{siz}(i)) \cdot 2^{N_L - r}} \right\rceil - \left\lceil \frac{XO_{siz}}{\min\limits_{\forall i}(XR_{siz}(i)) \cdot 2^{N_L - r}} \right\rceil \le W \tag{A-1}$$

$$\left\lceil \frac{Y_{siz}}{\min\limits_{\forall i}(YR_{siz}(i)) \cdot 2^{N_L - r}} \right\rceil - \left\lceil \frac{YO_{siz}}{\min\limits_{\forall i}(YR_{siz}(i)) \cdot 2^{N_L - r}} \right\rceil \le H \tag{A-2}$$

For images with multiple tiles, a tile with top left and bottom right coordinates (tx_0, ty_0), (tx_1, ty_1) on the reference grid, and number of decompositions levels N_L, is decodable at resolution $r \le T_L$ subject to the restrictions in Equations A-3 and A-4.

$$\left\lceil \frac{tx_1}{\min\limits_{\forall i}(XR_{siz}(i)) \cdot 2^{N_L - r}} \right\rceil - \left\lceil \frac{tx_0}{\min\limits_{\forall i}(XR_{siz}(i)) \cdot 2^{N_L - r}} \right\rceil \le W \tag{A-3}$$

$$\left\lceil \frac{ty_1}{\min\limits_{\forall i}(YR_{siz}(i)) \cdot 2^{N_L - r}} \right\rceil - \left\lceil \frac{ty_0}{\min\limits_{\forall i}(YR_{siz}(i)) \cdot 2^{N_L - r}} \right\rceil \le H \tag{A-4}$$

See Annex B of ITU-T Rec. T.800 | ISO/IEC 15444-1 for exact definition of tx_0, ty_0, tx_1, and ty_1 including limitations to SIZ marker parameters. If a non-negative r does not exist to satisfy both conditions for any tile, then no decoder obligation exists. Otherwise, one or more tiles shall be decoded, where the total decoded width need not exceed W and the total decoded height need not exceed H. There is no obligation to decode partial tile areas.

NOTE – For the purposes of compliance testing, all possible combination of these parameters will NOT be provided as test reference images. It may be necessary for the Implementation Under Test to provide the ability to set the decode resolution and decoded tile position. However, some post processing will be permitted as described in Annex B.

A decoder claiming compliance at some Cclass with image dimensions $H \times W$ and number of components C must also be capable of decoding any image with width less than or equal to W, height less than or equal to H, and number of components less than or equal to C. For each Cclass, the minimum values for H, W, and C are specified in Table A.1.

A.1.3 N_{cb}: Code-block parsing guarantee

Decoders need not decode compressed bits that cannot be recovered from the codestream due to excessive parser memory being required. An upper bound for the parser state memory required to reach a point x in the codestream may be determined from the total number of code-blocks for which state information must be kept, the total number of precincts for which a packet has been encountered, and the total number of components of the codestream.

At position x in the codestream, $N_{cb}(x)$ is defined as the total number of code-blocks in every precinct where the first header byte of at least one received packet for the precinct lies outside the range 0x80 to 0x8F.

Decoders are permitted to stop parsing the codestream at the point, x, once $N_{cb}(x) > N_{cb}$, where N_{cb} is defined for each compliance class. Decoders are permitted to stop parsing the codestream once packet headers with more than N_{cb} code-blocks have been encountered. Code-blocks in packets prior to the packet with the N_{cb}th code-block shall be decoded up to the limits of other parameters in the compliance class.

NOTE – Packets headers with the first bit set to 0 are defined as empty. The above definition adds all the code-blocks associated with such precincts to N_{cb} for these empty packets because a decoder requires more memory for these packets than for packets starting in the listed range.

A.1.4 N_{comp}: Component parsing guarantee

Decoders could be required to buffer information about each component for many thousands of components just to parse a codestream. To limit the required memory, decoders are permitted to stop parsing the codestream at a point, x, once the following condition is reached:

$$C_{\max}(x) > N_{comp}$$

where $C_{\max}(x)$ is defined as the largest component index for which a packet has been encountered up to point x regardless of the emptiness or the relevance of the packet.

Code-blocks in packets prior to the above stop condition shall be decoded up to the limits of other parameters in the compliance class.

A.1.5 L_{body}: Coded data buffering guarantee

The parser state memory described in A.1.3 is required to parse packets regardless of whether their code-blocks are relevant to the dimensions and number of components for which compliance is being claimed. For those code-blocks that are relevant, the implementation is required to store the recovered packet bytes. These are the code bytes that are processed by the block decoder (Annexes C and D, ITU-T Rec. T.800 | ISO/IEC 15444-1).

After a given number of decoded codestream bytes, x, the quantity $L_{body}(x)$ is defined as the total number of packet bytes that have been encountered so far in packets whose precincts are relevant to the dimensions and components for which compliance is being claimed. Although some implementations may be able to decode some of these packet bytes incrementally, L_{body} represents an upper bound on the number of packet bytes that must be stored by the decoder prior to decoding. If the number of relevant packet bytes exceeds L_{body}, then the Implementation Under Test (IUT) is allowed to stop reading the codestream and to decode the code-blocks obtained up to the limits of other parameters in the compliance class.

A.1.6 M: Decoded Bitplane guarantee

The decoder shall decode all of the packet bytes recovered by the parser in accordance with the requirements described above. This obligation is limited to the most significant M bit-planes of each code-block. Specifically, the block decoder must correctly decode the first $3(M - P_b) - 2$ coding passes, if available, of any relevant code-block, b, where P_b is the number of zero-valued most significant bit-planes signalled in the relevant packet header as described in Annex B of ITU-T Rec. T.800 | ISO/IEC 15444-1. The decoder is free to decode any number of additional coding passes for any code- block. Codestreams with large values for the number of guard bits will have a larger number of zero-valued most significant bit-planes, and thus a decoder of any given Cclass will decode fewer useful bit-planes. Likewise, codestreams with large values for the shift in the RGN marker segment may have fewer bit-planes decoded.

A.1.7 P: 9-7I Precision guarantee

Codestreams that make use of the irreversible 9-7 discrete wavelet transform will require dequantization, the 9-7 inverse discrete wavelet transform, and potentially the inverse irreversible component transform (ICT). The precision values for the wavelet transform are chosen to allow high quality imagery at various bit-depths, e.g., 8, 12, or 16 bits per sample. However, for Cclass 0, the accuracy of the 9-7I filter required is set such that it is possible to be compliant by decoding and inverse quantizing and performing a 5-3I (irreversible 5-3) inverse wavelet transform. This allows lower cost decoders to be used for the lowest compliance class only. For higher compliance classes, using the 5-3 filter in place of the 9-7 filter will not be sufficient to pass the compliance tests.

Using the 5-3 inverse wavelet transform to decode imagery compressed with the 9-7 wavelet introduces signal dependent noise. For example errors are highest around edges in the imagery. Because induced errors are signal dependent, there is no "precision" specified for the implementation of the wavelet transform for Cclass 0. Instead, the bounds on accuracy of the 9-7 transform have been set for each Cclass 0 reference image to allow an implementation to use the 5-3I inverse wavelet filter. Using the 5-3I inverse wavelet transform instead of a 9-7I filter does not relieve a decoder of the requirement to perform inverse quantization.

For compliance classes other than Cclass 0, the precision guarantee in Table A.1 refers to the implementation's minimum word size that will achieve the target MSE values for the test streams.

To facilitate end-to-end testing for compliance, dequantization may be performed using mid-point rounding. That is, the value of r in Equation G-6 of ITU-T Rec. T.800 | ISO/IEC 15444-1 can be $r = 1/2$. Implementations under test may provide the option of using different values for the reconstruction parameter, r; however, if the value $r = 1/2$ is supported and employed for compliance testing this will typically increase the ease of passing.

A.1.8 B: 5-3R Precision guarantee

A decoder is expected to implement the reversible 5-3R IDWT exactly, for component bit-depths of B bits/sample or less, as specified in the SIZ marker segment (see Annex A of ITU-T Rec. T.800 | ISO/IEC 15444-1). If a codestream employs the reversible component transform (RCT) and the IUT claims compliance at 3 or more components, it must be able to perform both the 5-3R IDWT and the inverse RCT exactly for bit-depths of B bits/sample or less.

A.1.9 T_L: Transform level guarantee

For each Cclass, a decoder is expected to be able to synthesize a minimum number of levels of the IDWT, T_L. For codestreams that contain more than T_L decomposition levels, the decoded image from a compliant decoder in a given Cclass may include only the top resolution levels.

A.1.10 L: Layer guarantee

For each Cclass, a decoder is expected to decode a minimum number of layers, L, in a codestream. For codestreams that contain more than L layers, the decoded image from a compliant decode in a given Cclass may include only the top L layers. This relieves compliant decoders from the burden of decoding inefficient codestreams with an excessive number of layers.

A.1.11 Progressions

For all Cclasses, a decoder is expected to decode all possible progressions as specified in the COD marker segment. If a POC marker segment is used in a codestream, a Cclass 0 decoder shall decode packets associated with the first progression order specified in the POC marker segment for that tile. Additional packets in the tile may be skipped. For all other Cclasses, packets may be skipped only due to other limitations (e.g., N_{cb} and L_{body}) and there is no explicit limitation on the number of progression order changes that may occur.

A.1.12 Tile-parts

Codestreams may contain multiple tile-parts for each tile. Profile 0 codestreams require all initial tile-parts to appear in spatial order in the codestream before other tile parts. Cclass 0 decoders may ignore tile-parts beyond the first even if N_{cb} or L_{body} has not been reached. For all other Cclasses, tile-parts may be skipped only due to other limitations (e.g., N_{cb} and L_{body}) and there is no explicit limitation on the number of tile-parts that may occur.

A.1.13 Precincts

Tiles may contain several precincts. Cclass 0 decoders need only decode the first precinct in each sub-band of each tile. Note that Profile 0 codestreams are designed to have only one precinct in each sub-band up to the resolution level decoded by a Cclass 0 decoder. Other profiles do not have this guarantee. Other Cclass decoders shall decode all relevant precincts.

A.2 Compliance class definitions

Table A.1 defines three compliance classes in terms of the parameters of this annex.

Table A.1 – Definitions of compliance classes (Cclass)

Parameter	Cclass 0	Cclass 1	Cclass 2
$W \times H$ (Size)	128×128	2048×2048	16384×16384
C (Components)	1	4	256
N_{cb}	$(HW/1024 + 32)C = 48$	$(HW/256 + 128)C = 66\,048$	$(HW/256 + 128)C = 268\,468\,224$
N_{comp}	64	256	16 384
L_{body}	8192 bytes	2^{23} bytes	2^{30} bytes
M	11	15	30
P	Low enough to allow 5-3I decoding of 9-7I data	16-bit fixed point implementation	32-bit single precision floats
B	8	12	16
T_L	3	7	12
L	15	255	65535
Progressions	All "basic" progressions in COD, only need decode first progression per tile	Limited only by number of levels, layers and components	Limited only by number of levels, layers and components
Tile Parts	Decode only first tile-part per tile	Decode all tile-parts up to N_{cb} or L_{body} limits	Decode all tile-parts up to N_{cb} or L_{body} limits
Precincts	Decode 1st precinct per sub-band	Decode all precincts up to N_{cb} or L_{body} limits	Decode all precincts up to N_{cb} or L_{body} limits

A.3 Lossless encoding and decoding

The minimum compliance point, Cclass 0, guarantees sufficient resources to ensure truly lossless decoding to a bit-depth of at least 8 bits per sample. However, this does not mean that lossless performance will be achieved, even if the codestream contains a lossless representation of the image. A compliant Cclass 0 decoder may fail to reproduce a perfectly reconstructed 8-bit version of a losslessly compressed image if the codestream contains a large amount of irrelevant information (e.g., extra image components or resolutions that are not targeted by the particular decoder implementation under consideration). Extra irrelevant information may cause a compliant decoder to stop reading the codestream due to L_{body} or N_{cb} limits being reached before all relevant information has been recovered.

Again, lossless decompression may not be achieved, even if the codestream contains a lossless representation of 8-bit imagery, if the compressor employed an unnecessarily large number of guard bits or unnecessarily large sub-band dynamic range parameters, ε_b (Annex E, ITU-T Rec. T.800 | ISO/IEC 15444-1), for some sub-bands, or if ROI information (Annex H, ITU-T Rec. T.800 | ISO/IEC 15444-1) was included in the codestream. The compressor is at liberty to make such choices and their potential impact on decoders at any Cclass should be considered.

Annex B

Decoder compliance testing procedures

(This annex forms an integral part of this Recommendation | International Standard)

B.1 General

This annex defines the procedure to follow to determine if a decoder is compliant to a particular compliance class (Cclass) for a particular profile. The Cclass is a level of quality achieved by a decoder. A decoder of a higher Cclass is required to achieve a higher quality image for the same codestream. A decoder compliant with a certain Cclass may have capabilities beyond those required for compliance to that Cclass but not sufficient for compliance with the next highest Cclass. In such a case, it may be necessary to alter the outputs of the decoder in order to test its compliance with the lower Cclass. This annex defines procedures for such alterations.

B.2 Decoder test procedure

The procedure defined herein will determine if a decoder is compliant at a chosen profile and Cclass.

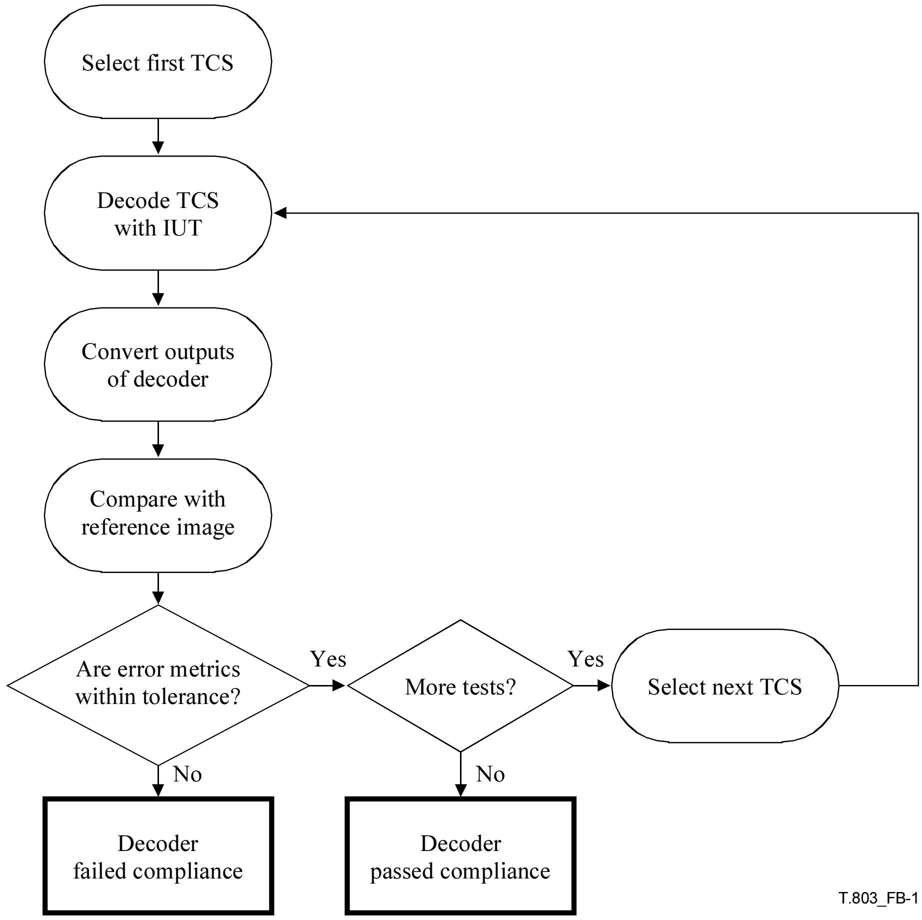

Figure B.1 – Decoder compliance test flow chart

The following steps for testing the set of codestreams are shown in flow chart in Figure B.1:

a) each codestream is decoded using the decoder under test;

b) the decoded outputs are format converted if necessary;

c) the difference between the decoded outputs and reference images are measured; and

d) the measurements are compared with the limits for the particular image and compliance class.

Each of these steps is defined in more detail in a later subclause of this annex. Failure to meet the tolerance limits for a single image results in the decoder failing to be compliant at the profile and Cclass being tested.

B.2.1 Files for testing

A particular ETS defines the codestreams, output images, and error tolerances. This is done in Annex C for four compliance points.

B.2.2 Decoder settings

Decoders may have mechanisms for supporting various decompression settings. These may be set in the most advantageous way to achieve compliance. For example, a decoder with a "fast mode" and an "accurate mode" may be set to the "accurate mode" for compliance. These settings should be noted in any statement of compliance. Settings that allow the output resolution or spatial region of the reference decoded images to be matched may be changed for each decoded image. The same user controlled settings for accuracy or quantization reconstruction factor of wavelet coefficients shall be used for all test codestreams.

For each reference decoded image it is useful to know the output resolution and spatial extent; this is provided with the ETS. All reference decoded images begin at the upper left corner of the TCS image area and include decoded tiles up to the width and height declared in these tables. The reduction in resolution from the full TCS resolution is provided in these tables, as well. For Cclass 0, when several reference decoded images are provided for a single test codestream, the decoder output need only be compared with one of the reference image choices.

B.2.3 Output file format conversion

The reference decoded images are provided in a specific format defined below. In order to compare decoded images from the decoder under test with these images, several conversions may be necessary. These conversions are done as post-processing steps outside of the decoder solely for determining compliance. There is no requirement for a compliant decoder to perform these processes as part of its normal operation. These conversions shall not introduce a quality change (either loss or gain) except as required by the specific conversions described in the following subclauses.

B.2.3.1 Order-dependent conversions

The following post-processing conversions, if required, shall be performed in the indicated order and prior to the order-independent conversions.

B.2.3.1.1 Scaling

Decoders may produce a high resolution image even when testing for a low Cclass. For example, a full 256×256 component from Profile 0 test codestream 3 might be produced by a decoder attempting to pass Cclass 0. The reference image for Cclass 0 is only 128×128. Thus, the decoded image must be scaled to 128×128. Components should be scaled by applying the forward wavelet transform (5-3 or 9-7 according to which was used to decode the component) and keeping only the required low pass sub-band. It is preferable but not required for decoders to be able to decode any desired resolution. The reference images were created by decoding the number of wavelet decomposition levels specified in the tables for each test.

B.2.3.1.2 Multiple component transform

For Cclass 0 only, a decoder can decode a single component. For multiple component images this will be the first codestream component, which may correspond to the luminance of the multi-component image. A decoder that decodes 3 components from the image and performs the inverse colour transform must use post-processing to extract a single-component image to be compared with the reference image. This post-processing consists of a forward colour transform and extraction of the first component of the output. If the components were transformed with the 9-7 filter, the forward ICT from ITU-T Rec. T.800 | ISO/IEC 15444-1 Equation G-9 shall be used. If the components were transformed with the 5-3 reversible filter, the forward RCT from ITU-T Rec. T.800 | ISO/IEC 15444-1 Equation G-3 shall be used.

B.2.3.1.3 Conversion to integer

A decoder producing floating point output samples must have the samples converted to the nearest integer.

B.2.3.1.4 Clipping

Because of quantization and potential differences in the forward and inverse wavelet transforms, decoded component samples may be outside their nominal range. They shall be clipped as follows for unsigned components:

$$x_{clipped} = \begin{cases} 0, & x_{unclipped} < 0 \\ x_{unclipped}, & 0 \le x_{unclipped} < 2^{D} \\ 2^{D} - 1, & 2^{D} \le x_{unclipped} \end{cases} \tag{B-1}$$

where D is the bit-depth of the component. For signed components, clipping is defined as:

$$x_{clipped} = \begin{cases} -2^{D-1}, & x_{unclipped} < -2^{D-1} \\ x_{unclipped}, & -2^{D-1} \le x_{unclipped} < 2^{D-1} \\ 2^{D-1} - 1, & 2^{D-1} \le x_{unclipped} \end{cases} \tag{B-2}$$

where D is the bit-depth of the component.

B.2.3.1.5 Bit-depth scaling

The decoded images shall be scaled to the smaller of the image component bit-depth and the compliance class bit-depth. For example, if a decoder decodes a 12-bit component into the most significant bits of a 16-bit word, the component shall be scaled to 8 bits per sample for a Cclass 0 decoder test. If the decoder produces an 8-bit component for a codestream with a 4-bit component, the component shall be scaled to 4-bits per sample. The scaling shall be implemented by simple arithmetic shifts. This can be expressed as:

$$x_{scaled} = \left\lfloor \frac{x_{unscaled}}{2^{d-r}} \right\rfloor \tag{B-3}$$

where d is the nominal bit-depth of the decoded component and r is the bit-depth of the reference image.

B.2.3.1.6 Subsampling

All reference components contain the number of samples expected from ITU-T Rec. T.800 | ISO/IEC 15444-1 Equation B-2, restricted to *H* and *W* for the Cclass. For testing purposes, it is preferable for a decoder to not upsample the image (e.g., by replication by $XRsiz^c$ or $YRsiz^c$ to create a component the size of the reference grid). If such upsampling cannot be turned off, a post-processing step to subsample the upsampled image shall be used to return the component to the appropriate size.

B.2.3.1.7 Cropping

Decoders may have the capability to decode the entire image when presented with a test codestream, and in some cases this may yield an image that is larger than the *H* and *W* requirements for the Cclass being tested. In such cases, the decoded image shall be cropped for testing purposes, leaving the upper left corner to be the size of the appropriate reference image for the Cclass being tested. Cropping is accomplished by discarding all samples outside of the dimensions of the reference component.

NOTE – For a decoder of a given Cclass, there may be several possible ways to produce an output image that satisfies the *H* and *W* requirements for that Cclass. The possibilities will depend upon the specific capabilities of the decoder, the size of the image and its tiles, and the number of decomposition levels present in each tile. The reference images for each codestream are produced in one of two ways when possible. First, the decoder processes the first image tile to its maximum resolution consistent with the size constraint for the given Cclass. If possible, additional tiles are included at the same resolution to bring the final output image size as near as possible to the Cclass *H* and *W* size limits. (Note that this implies the tiles are produced at full resolution.) Second, the decoder processes all tiles of the image at a common resolution consistent with the Cclass *H* and *W* size limits. This latter strategy produces something similar to a thumbnail or overview image, typically at a resolution reduced from that of the original image.

B.2.3.2 Order-independent conversions

The following post-processing conversions, if required, shall be performed after any order-dependent conversions. These conversions may be applied in any convenient order.

B.2.3.2.1 Component de-interleaving

Decoded images in interleaved format (e.g., one red sample followed by one green sample followed by one blue sample) shall be converted to one file per component.

B.2.3.2.2 Endianness

All reference images are stored in big endian (i.e., with most significant byte first). If the decoded image is produced in little endian format then the bytes shall be swapped for comparison purposes.

B.2.3.2.3 Sign extend to byte boundary

All reference images are stored with either 8 or 16 bits per sample. For example, 4- and 12-bit component samples shall be padded in the most significant bit positions with an additional 4 bits. These bits shall be "1" for negative samples in signed images and "0" for all other samples. The sign extension must properly account for the endianness of the samples.

B.2.4 Compare decoded and formatted components with reference components

Once the decoded image has been properly formatted it is compared with the reference image. The reference components for each Cclass are listed in the tables defining the ETS. The decoded components must be compared separately for all components for which there is a reference decoded image. For each component, the mean squared error (MSE) shall be determined as follows:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{x}_i)^2 \qquad \text{(B-4)}$$

where N is the number of samples in the component (reference width times reference height), x_i is a sample from the reference image, and $\hat{x}_i$ is the corresponding sample from the decoded and formatted image.

The Peak error, p, shall be determined as follows:

$$p = \underset{i=1}{\overset{N}{MAX}}\left|x_i - \hat{x}_i\right| \qquad \text{(B-5)}$$

B.2.5 Compare error metrics with specification

Each ETS defines a peak error and MSE error for each test codestream and reference decoded image. These are defined in Annex C.

For the Cclass0 tests, only the first codestream image component is decoded. Any ICT or RCT indicated in the codestream is not performed. When multiple reference files are provided, the decoder output shall be tested against at least one of these references.

Other Cclasses require decoding multiple components. Separate reference images are provided for each component which must be decoded, with the suffix '-*c*' where the *c* indicates the component number. The IUT must make comparisons for all the component reference files listed in tables defining the ETS.

B.2.6 Reference components file format

This subclause describes the file format, called PGX, of the reference images used for comparison with the output of the decoder under test. The decoder under test is not required to produce this particular file format. Any necessary conversion, as specified in B.2.3, may be applied. The reference decoded files are stored with one file per component. There is also a header line that describes the size of the components.

NOTE – In some operating systems the PGX test files provided may be split into a header only file and a raw data file with no header using the commands 'head -1 file.pgx >header.txt' and 'tail +1 file.pgx >file.raw'.

B.2.6.1 Header format

The header used by the reference images consists of a single text line of the form:

PG <byte order> [+|-]<bit-depth> <width> <height> <newline>

Each of the fields is separated by one space character, 0x20.

<byte order> literal "ML" (0x4D, 0x4C)

NOTE – Some PGX writers will use the literal "LM" to indicate least significant byte first. This is not used by the reference images with this Recommendation | International Standard.

[+|-]<bit-depth> If this field begins with an ASCII digit or a '+' character, the image data is unsigned; if the field begins with a '-' character the data is signed. The ASCII value indicates the bit-depth. There may be a space (0x20) between the sign ('+' or '-') and the bit-depth.

<width> the number of samples horizontally for the image. There are no offsets or subsampling with this format.

<height> the number of samples vertically for the image. There are no offsets or subsampling with this format.

<newline> may consist of the two bytes 0x0d, 0x0a, or the single character 0x0a, the character 0x0a shall be the last before the beginning of the raw data and will always be present.

Example:

PG ML +8 128 128

B.2.6.2 Data format

The binary data appears immediately after the 0x0a byte in the header. The data in the raw file is stored most significant byte first using 1, 2, or 4 bytes per sample. One byte is used for component depths of 1 to 8 bits, two bytes are used for component depths of 9 to 16 bits, and four bytes are used for depths of 17 to 32 bits. Signed data is stored in 2's complement, with sign extension throughout the 1, 2, or 4 bytes (e.g., –1 is stored as 0xFFFF for a 12-bit signed image).

Annex C

Compliance tests

(This annex forms an integral part of this Recommendation | International Standard)

This annex specifies the abstract test suites and executable test suites that will be used in the compliance test procedures from Annex B.

C.1 Abstract test suite (informative)

The Abstract Test Suite (ATS) defines the general tests for components of ITU-T Rec. T.800 | ISO/IEC 15444-1. The lists in this subclause are used to define the decoder ETS, and could be used to develop additional encoder or decoder ETSs.

C.1.1 Syntax and compressed data order

a) Test purpose: To test the ability of an implementation to encode and decode codestreams with optional markers, marker values, and markers in different locations in the codestream.

b) Test method: Using lossless codestreams, encode or decode several different codestreams with variations of markers, marker values, and marker locations. Parameters and existence of markers may be limited by the profile being tested. Specific test items include:
   - location of markers in codestream: optional markers in different positions of the codestream;
   - priority of markers: markers in the codestream that override previous markers;
   - proper use of the pointer markers: TLM, PLT, PLM correct;
   - image offsets: XOsiz and YOsiz with several values including odd, even, large, and small;
   - tile dimensions: several size tiles including odd, even, very small tiles;
   - tile offsets: XTOsiz and YTOsiz with several values including odd and even;
   - component subsampling: XRsiz and YRsiz with several values including odd and even;
   - code-blocks dimensions: all values;
   - progression orders: 5 progression orders, progression order changes;
   - packet headers: include SOP and EPH, number of layers, different locations of packet headers (main header, tile-part header, codestream), bit stuffing;
   - precincts: having several values of precinct sizes (including different for each sub-band and a precinct that is smaller than the code-block sizes), precinct and tile boundaries equivalent at low resolution, different at high resolution;
   - precincts containing no data, resolution levels containing no valid coefficient, precincts that have no coefficients in several/all bands;
   - packet-non-empty bit of the packet header;
   - sub-bands: with no precincts;
   - tile-parts: placing the tile-parts in different locations in the codestream, POC in 2nd tile-part.

c) Reference: ITU-T Rec. T.800 | ISO/IEC 15444-1 Annex A: Codestream syntax and Annex B: Image and compressed image data ordering.

C.1.2 Arithmetic entropy encoding

a) Test purpose: To evaluate the correct operation of the implementation of the arithmetic encoder within the IUT.

b) Test method: Losslessly compressed images using different combinations of options. Can change within a single image by changing the options in each tile. Encode, or decode several different codestreams with different arithmetic entropy coding parameters. Specific test options include:
   - arithmetic coder bypass;
   - context reset on coding pass boundaries;
   - termination on each coding pass;
   - vertically causal context;
   - predictable termination;
   - segmentation symbols;
   - combinations of options.

c) Reference: ITU-T Rec. T.800 | ISO/IEC 15444-1 Annex C: Arithmetic entropy coding.

C.1.3 Coefficient bit modelling

a) Test purpose: To test the correct operation of the coefficient bit modelling of the arithmetic encoding of the IUT.

b) Test method: Encode, or decode several different codestreams with all possible neighbouring location contexts. Test all contexts occurring in all sub-band types and with and without vertically causal contexts.

c) Reference: ITU-T Rec. T.800 | ISO/IEC 15444-1 Annex D: Coefficient bit modelling.

C.1.4 Quantization

a) Test purpose: To test the accuracy of the quantization implementation of the IUT.

b) Test method: Encode or decode several different codestreams with zero levels of decomposition in the wavelet transform so that the quantization is the only parameter being tested. The accuracy is tested with MSE and max error. Specific items to test include:
   - exponent and mantissa;
   - guard bits;
   - dequantization offset value;
   - derived and explicit quantization.

c) Reference: ITU-T Rec. T.800 | ISO/IEC 15444-1 Annex F: Discrete wavelet transformation of tile-components.

C.1.5 Discrete wavelet transform

a) Test purpose: To test the accuracy of the implementation of the two discrete wavelet transforms.

b) Test method: Encode or decode several different codestreams with the two filters, varying the tile size, and number of levels, and test the accuracy of each of the filters. The accuracy of the 9-7 filter shall be defined by MSE and max error while the 5-3 filter shall have no difference. Specific items to test include:
   - precision of 9-7;
   - reversibility of 5-3;
   - number of decomposition levels, including 0;
   - different offset conditions and tile sizes, very small sub-bands down to 1×1, empty sub-bands, one column sub-bands, one-row sub-bands;
   - saturation conditions.

c) Reference: ITU-T Rec. T.800 | ISO/IEC 15444-1 Annex F: Discrete wavelet transformation of tile-components.

C.1.6 DC level shift and multiple component transform

a) Test purpose: To test the ability of the implementation to achieve the DC level shift and the accuracy of the two multiple component transforms.

b) Test method: Encode or decode several different codestreams with IUT using different combinations of options:
   - component depth;
   - component samples that are signed and unsigned;
   - precision of ICT;
   - correctness of RCT.

c) Reference: ITU-T Rec. T.800 | ISO/IEC 15444-1 Annex G: DC level shifting and multiple component transformations.

C.1.7 Region of interest

a) Test purpose: To test the correct implementation of ROI in the IUT.

b) Test method: Encode or decode several different images with regions of different size, number, shift value, different in each component:
   - different ROI in each component;
   - RGN marker in main and tile-part header, test for proper treatment of priorities;
   - shift value;
   - partial decoding of an image with ROI.

c) Reference: ITU-T Rec. T.800 | ISO/IEC 15444-1 Annex H: Coding of images with regions of interest.

C.1.8 File format

a) Test purpose: To test the ability of the implementation to accurately represent the JPEG 2000 compressed data within the JP2 file format.

b) Test method: Encode or decode several different codestreams with different parameters of the file format.

c) Reference: ITU-T Rec. T.800 | ISO/IEC 15444-1 Annex I: JP2 file format syntax.

C.2 Executable test suite (ETS)

The executable test suites are the embodiment of the ATS. Commonly several ATSs are embodied in one ETS. This subclause defines four ETSs, namely Cclass 0 and Cclass 1 decoders for Profile 0 and Profile 1 codestreams. There are 16 test codestreams defined for Profile 0 and 7 test codestreams defined for Profile 1 (Profile 1 compliance requires both Profile 0 and Profile 1 test codestreams). The tolerance values are defined as a function of Cclass for each image and may be more difficult to achieve as Cclass increases for each image. Additional ETSs may be made available after the publication of this Recommendation | International Standard, see http://www.jpeg.org/software for the latest set of ETSs.

Each ETS consists of codestreams, reference decoded images, and tolerance values for MSE and peak error. In addition, some information is provided about the test codestreams which may aid correction of a non-compliant decoder. This information consists of a table listing many of the markers in the codestream, the offset into the codestream where those markers occur, and the value of the parameters of those markers.

C.2.1 Class 0 Profile 0

C.2.1.1 Codestreams

The test codestreams for this ETS are in the directory `codestreams_profile0`. There are 16 codestreams, with names of the form `p0_##.j2k`, where '##' is the codestream number.

C.2.1.2 Reference decoded images

The reference decoded images for this ETS are in the directory `reference_class0_profile0`. The filenames are of the form `c0p0_##[rN].pgx`, where '##' is the codestream number, and N is the number of inverse wavelet transforms that have been skipped in the decoding of the codestream. A zero indicates a full resolution image (typically this is cropped to the upper left corner). A one indicates an image with resolution reduced by 2^1 in each dimension.

C.2.1.3 Tolerances

The maximum allowable MSE and peak errors are listed in Table C.1, along with the size of the reference decoded images.

Table C.1 – Class 0 Profile 0 reference images and allowable errors

TCS	Reference file	Resolution reduction	Signed	Depth	Width	Height	Peak	MSE
p0_01.j2k	c0p0_01.pgx	0	+	8	128	128	0	0
p0_02.j2k	c0p0_02.pgx	0	+	8	64	126	0	0
p0_03.j2k	c0p0_03r0.pgx	0	–	4	128	128	0	0
	c0p0_03r1.pgx	1	–	4	128	128	0	0
p0_04.j2k	c0p0_04.pgx	3	+	8	80	60	33	55.8
p0_05.j2k	c0p0_05.pgx	3	+	8	128	128	54	68
p0_06.j2k	c0p0_06.pgx	3	+	8	65	17	109	743
p0_07.j2k	c0p0_07.pgx	0	–	8	128	128	10	0.34
p0_08.j2k	c0p0_08.pgx	5	–	8	17	96	7	6.72
p0_09.j2k	c0p0_09.pgx	2	+	8	5	10	4	1.47
p0_10.j2k	c0p0_10.pgx	0	+	8	64	64	10	2.84
p0_11.j2k	c0p0_11.pgx	0	+	8	128	1	0	0
p0_12.j2k	c0p0_12.pgx	0	+	8	3	5	0	0
p0_13.j2k	c0p0_13.pgx	0	+	8	1	1	0	0
p0_14.j2k	c0p0_14.pgx	2	+	8	13	13	0	0
p0_15.j2k	c0p0_15r0.pgx	0	–	4	128	128	0	0
	c0p0_15r1.pgx	1	–	4	128	128	0	0
p0_16.j2k	c0p0_16.pgx	0	+	8	128	128	0	0

C.2.1.4 Test codestream descriptions (informative)

Table C.2 describes the features being tested by the Profile 0 codestreams. This information is provided solely to help correction of non-compliant decoders when accuracy is not guaranteed.

Table C.2 – Items tested by Profile 0 codestreams

Codestream	Tests
1	5-3 wavelet, 64 × 64 code-blocks, MQ-coder, context model
2	Component subsampling, multiple layers, termination every coding pass, predictable termination, segmentation symbols, COD, QCD, EPH, SOP, and 0xFF30 marker segments, 32 × 32 code-blocks
3	Multiple tiles, signed data, 4-bit/component data, QCC, POC, CRG, TLM, and RGN marker segments
4	Multiple components, termination every coding pass, 9-7 wavelet, precinct sizes in COD, irreversible component transform, scalar expound quantization
5	Different subsampling for different components, different wavelet filters and parameters for different components
6	12-bit component samples, RGN in main and tile
7	Large number of tiles
8	Large image
9	9-7 wavelet transform overflow
10	Image source is pseudo-random, subsampling by 4, 0 guard bits, reversible colour transform, tile-parts
11	1 sample high image, 0 decomposition level test, segmentation symbols
12	Special wavelet transform cases
13	Large number of components
14	5-3 wavelet transform saturation
15	RGN, POC, Signed, QCC, COM
16	Empty packet header bit

The directory `descriptions_profile0` contains information about every test codestream. The information is of the type shown in Table C.3.

Table C.3 – Profile 0 codestream 0 contents

Offset	Item	Value
0	SOC (Start of codestream)	
2	SIZ (Image and tile size)	
	Required Capabilities	JPEG 2000 profile 0
	Reference Grid Size	128 × 128
	Image Offset	0 × 0
	Reference Tile Size	128 × 128
	Reference Tile Offset	0 × 0
	Components	1
	Component #0 Depth	8
	Component #0 Signed	no
	Component #0 Sample Separation	1 × 1
45	QCD (Quantization default)	
	Quantization Type	None
	Guard Bits	2
	Exponent #0	8
	Exponent #1	9
	Exponent #2	9
	Exponent #3	10
	Exponent #4	9
	Exponent #5	9
	Exponent #6	10
	Exponent #7	9
	Exponent #8	9
	Exponent #9	10
60	COD (Coding style default)	
	Default Precincts of $2^{15} \times 2^{15}$	Yes
	SOP Marker Segments	No
	EPH Marker Segments	No
	All Flags	00000000
	Progression Order	Resolution level-layer-component-position
	Layers	1
	Multiple Component Transformation	None
	Decomposition Levels	3
	Code-block size	64 × 64
	Selective Arithmetic Coding Bypass	No
	Reset Context Probabilities	No
	Termination on Each Coding Pass	No
	Vertically Causal Context	No
	Predictable Termination	No
	Segmentation Symbols	No
	Wavelet Transformation	5-3 reversible
74	SOT (Start of tile-part)	
	Tile	0
	Length	7314
	Index	0
	Tile-Parts	1
7388	EOC (End of codestream)	

C.2.2 Class 0 Profile 1

C.2.2.1 Codestreams

The test codestreams for this ETS are in the directory **`codestreams_profile1`**. The codestream names have the form **`p1_##.j2k`**, where '##' is the codestream number. In addition to passing the test for all these codestreams, a decoder must also pass the Class 0 Profile 0 test to pass the Class 0 Profile 1 test.

C.2.2.2 Reference decoded images

The reference decoded images for this ETS are in the directory **`reference_class0_profile1`**. The filenames are of the form **`c0p1_##[rN].pgx`**, where '##' is the codestream number, '*N*' is the number of inverse wavelet transforms that have been skipped in the decoding of the codestream.

C.2.2.3 Tolerances

The maximum allowable MSE and peak errors are listed in Table C.4, along with the size of the reference decoded images.

Table C.4 – Class 0 Profile 1 reference images and allowable errors

TCS	Reference file	Resolution reduction	Depth	Width	Height	Peak	MSE
p1_01.j2k	c0p1_01.pgx	0	8	61	99	0	0
p1_02.j2k	c0p1_02.pgx	3	8	80	60	35	74.0
p1_03.j2k	c0p1_03.pgx	3	8	128	128	28	18.8
p1_04.j2k	c0p1_04r0.pgx	0	8	128	128	2	0.550
	c0p1_04r3.pgx	3	8	128	128	128	2042
p1_05.j2k	c0p1_05.pgx	4	8	32	32	128	16384
p1_06.j2k	c0p1_06.pgx	1	8	6	6	128	16384
p1_07.j2k	c0p1_07.pgx	0	8	2	12	0	0

C.2.2.4 Additional information

Table C.5 describes the key features being tested by the Profile 1 codestreams. This information is provided solely to help correction of non-compliant decoders when accuracy is not guaranteed.

Table C.5 – Items tested by Profile 1 codestreams

Codestream #	Tests
1	Image offsets, tile offsets
2	Reset context probabilities, vertically causal contexts, precinct sizes, PPT marker segment
3	PPM marker segment
4	QCD marker segment in tile header
5	Odd-sized tile, non-square code-block size, multiple PPM marker segments
6	Small tile size
7	Small precincts, packet inclusion

C.2.3 Class 1 Profile 0

C.2.3.1 Codestreams

The test codestreams for this ETS are in the directory **`codestreams_profile0`**. There are 16 codestreams, with names of the form **`p0_##.j2k`**, where '##' is the codestream number. These are the same test codestreams as for Cclass 0 Profile 0, but they must be decoded to higher quality.

C.2.3.2 Reference decoded images

The reference decoded images for this ETS are in the directory **`reference_class1_profile0`**. The filenames are of the form **`c0p0_##[rN]-C.pgx`**, where '##' is the codestream number, and '*N*' is the number of inverse wavelet transforms that have been skipped in the decoding of the codestream, and '*C*' is the component number. A decoder shall produce all components listed, it may produce more.

C.2.3.3 Tolerances

The maximum allowable MSE and peak errors are listed in Table C.6, along with the size of the reference decoded images.

Table C.6 – Class 1 Profile 0 reference files and maximum error

TCS	Reference filename	Resolution reduction	Signed	Depth	Width	Height	Peak	MSE
p0_01.j2k	c1p0_01-0.pgx	0	+	8	128	128	0	0
p0_02.j2k	c1p0_02-0.pgx	0	+	8	64	126	0	0
p0_03.j2k	c1p0_03-0.pgx	0	–	4	256	256	0	0
p0_04.j2k	c1p0_04-0.pgx	0	+	8	640	480	5	0.776
	c1p0_04-1.pgx	0	+	8	640	480	4	0.626
	c1p0_04-2.pgx	0	+	8	640	480	6	1.070
p0_05.j2k	c1p0_05-0.pgx	0	+	8	1024	1024	2	0.302
	c1p0_05-1.pgx	0	+	8	1024	1024	2	0.307
	c1p0_05-2.pgx	0	+	8	512	512	2	0.269
	c1p0_05-3.pgx	0	+	8	512	512	0	0
p0_06.j2k	c1p0_06-0.pgx	0	+	12	513	129	635	11287
	c1p0_06-1.pgx	0	+	12	257	129	403	6124
	c1p0_06-2.pgx	0	+	12	513	65	378	3968
	c1p0_06-3.pgx	0	+	12	257	65	0	0
p0_07.j2k	c1p0_07-0.pgx	0	–	12	2048	2048	0	0
	c1p0_07-1.pgx	0	–	12	2048	2048	0	0
	c1p0_07-2.pgx	0	–	12	2048	2048	0	0
p0_08.j2k	c1p0_08-0.pgx	1	–	12	257	1536	0	0
	c1p0_08-1.pgx	1	–	12	257	1536	0	0
	c1p0_08-2.pgx	1	–	12	257	1536	0	0
p0_09.j2k	c1p0_09-0.pgx	0	+	8	17	37	0	0
p0_10.j2k	c1p0_10-0.pgx	0	+	8	64	64	0	0
	c1p0_10-1.pgx	0	+	8	64	64	0	0
	c1p0_10-2.pgx	0	+	8	64	64	0	0
p0_11.j2k	c1p0_11-0.pgx	0	+	8	128	1	0	0
p0_12.j2k	c1p0_12-0.pgx	0	+	8	3	5	0	0
p0_13.j2k	c1p0_13-0.pgx	0	+	8	1	1	0	0
	c1p0_13-1.pgx	0	+	8	1	1	0	0
	c1p0_13-2.pgx	0	+	8	1	1	0	0
	c1p0_13-3.pgx	0	+	8	1	1	0	0
p0_14.j2k	c1p0_14-0.pgx	0	+	8	49	149	0	0
	c1p0_14-1.pgx	0	+	8	49	49	0	0
	c1p0_14-2.pgx	0	+	8	49	49	0	0
p0_15.j2k	c1p0_15-0.pgx	0	–	4	256	256	0	0
p0_16.j2k	c1p0_16-0.pgx	0	+	8	128	128	0	0

C.2.4 Class 1 Profile 1

C.2.4.1 Codestreams

The test codestreams for this ETS are in the directory **`codestreams_profile1`**. The codestream names have the form **`p1_##.j2k`**, where '##' is the codestream number. In addition to passing the test for all these codestreams, a decoder must also pass the Class 1 Profile 0 test to pass the Class 1 Profile 1 test.

C.2.4.2 Reference decoded images

The reference decoded images for this ETS are in the directory **`reference_class1_profile1`**. The filenames are of the form **`c1p1_##[rN]-C.pgx`**, where '##' is the codestream number, and '*N*' is the number of inverse wavelet transforms that have been skipped in the decoding of the codestream, and '*C*' is the component number. A decoder shall produce all components listed.

C.2.4.3 Tolerances

The maximum allowable MSE and peak errors are listed in Table C.7, along with the size of the reference decoded images.

Table C.7 – Class 1 Profile 1 reference images and allowable error

TCS	Reference filename	Resolution reduction	Depth	Width	Height	Peak	MSE
p1_01.j2k	c1p1_01-0.pgx	0	8	61	99	0	0
p1_02.j2k	c1p1_02-0.pgx	0	8	640	480	5	0.765
	c1p1_02-1.pgx	0	8	640	480	4	0.616
	c1p1_02-2.pgx	0	8	640	480	6	1.051
p1_03.j2k	c1p1_03-0.pgx	0	8	1024	1024	2	0.300
	c1p1_03-1.pgx	0	8	1024	1024	2	0.210
	c1p1_03-2.pgx	0	8	512	512	1	0.200
	c1p1_03-3.pgx	0	8	512	512	0	0
p1_04.j2k	c1p1_04-0.pgx	0	12	1024	1024	624	3080
p1_05.j2k	c1p1_05-0.pgx	0	8	512	512	40	8.458
	c1p1_05-1.pgx	0	8	512	512	40	9.716
	c1p1_05-2.pgx	0	8	512	512	40	10.154
p1_06.j2k	c1p1_06-0.pgx	0	8	12	12	2	0.600
	c1p1_06-1.pgx	0	8	12	12	2	0.600
	c1p1_06-2.pgx	0	8	12	12	2	0.600
p1_07.j2k	c1p1_07-0.pgx	0	8	2	12	0	0
	c1p1_07-1.pgx	0	8	8	12	0	0

Annex D

Encoder compliance test procedure

(This annex does not form an integral part of this Recommendation | International Standard)

This annex defines testing compliance of an encoder. The encoder may produce any subset of ITU-T Rec. T.800 | ISO/IEC 15444-1, but whatever it does produce must be a legal JPEG 2000 codestream.

D.1 General

It is not a requirement to implement and/or support all possible encoding modes or capabilities of JPEG 2000. However, many modes and capabilities are desirable for many applications. The abstract test suites, as described in Annex C, are used to support the concepts in this subclause. It is impossible to define tests for every encoder variation. Therefore, it is left to the user to define the tests that are appropriate for a given implementation. This subclause gives guidance based on the abstract test procedures, the compliance statement, and the procedures below.

D.2 Reference decoder

The reference decoder is used for the evaluation of compliance of an encoder. The reference decoder is defined in ITU-T Rec. T.804 | ISO/IEC 15444-5. The reference decoder has been developed by the ISO/IEC JTC 1/SC 29/WG 1 committee for the purpose of guidance for implementor and data providers. The reference decoder should be able to decode all encoder developed codestreams that fall within reasonable limits of encoder applications.

D.3 Compliance requirement and acceptance

It is not a requirement for an encoder to produce any specific codestream. However, any codestream that is produced must be a legal JPEG 2000 codestream. A reasonable way that this can be verified is by decoding the codestream with the reference decoder.

This test procedure is adequate for many but not all possible encoder implementations. The reference decoder is limited in some parameters (such as the number of components and component bit-depth) and may not support all possible legal codestreams. On the other hand, the reference decoder may be sufficiently resilient to ignore codestream syntax errors in some cases.

D.4 Encoding compliance test procedure

Figure D.1 shows the flow for the encoder compliance test. The steps for this compliance test are as follows:

a) Select a test image that represents the type of imagery that the encoder is designed to compress. The reference decoded images provided for decoder compliance tests are acceptable but not required.

b) Encode with the encoder under test.

c) Send the codestream to the reference decoder.

d) An encoder is found to be compliant if the reference decoder can fully decode the image.

e) Repeat steps a) through d) for all parameters for which the encoder is designed. These parameters (tile size, number of decomposition levels, bitrates, etc.) should be varied to the extent to which the encoder will be used.

f) Repeat steps a) through e) for several test images, sampling the breadth of imagery types (small image size, large image size, odd image sizes, number of components, components depths, component sampling) the encoder is designed to compress.

If the codestream output by the encoder is embedded within other file formats, it is the responsibility of the tester to strip away any preceding and trailing file information before sending the encoded codestream to the reference decoder.

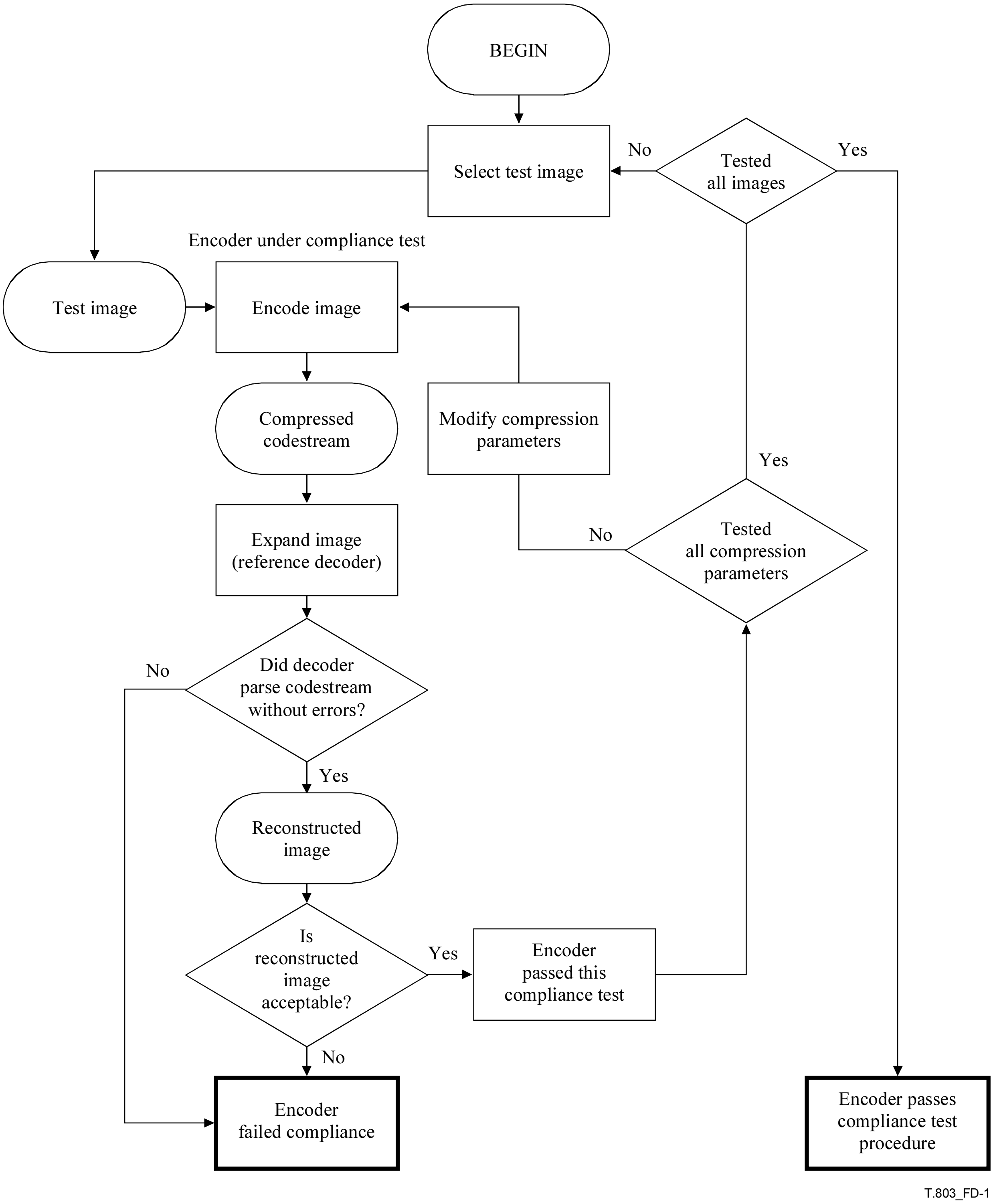

Figure D.1 – Encoder compliance test block diagram

It is up to the implementor to establish requirements for accepting a reconstructed image. For example, a lossless encoder would be expected to produce a reconstructed image identical to the test image. Requirements for a lossy encoder might include obtaining peak error or MSE values in a specified range or similar evaluation.

Annex E

Decoder Implementation Compliance Statement

(This annex does not form an integral part of this Recommendation | International Standard)

E.1 General

This annex allows the decoder IUT to describe its capabilities or image quality guarantees. Each system will claim a compliance point (profiles, Cclasses, file format) that will be used to define the compliance test procedures. Once it has defined a profile and Cclass, the IUT will be evaluated with all ETS corresponding up to and including the profile and Cclass being tested. Market applications may demand that a given system comply not only to a profile and Cclass, but also to an increase in one or more parameters (i.e., image size, number of components, bit-depth). Subclause E.3 describes the ability of systems to extend beyond the defined Cclasses for each profile.

E.2 Decoder implementation compliance statement

The decoder IUT should select the Cclass to which it supports the given profile. The decoder IUT should also select if it supports the optional JPEG 2000 file format. If an IUT claims compliance to a given Cclass, it is required to meet all expectations of that Cclass as defined in Annex A and the ETSs defined in Annex C. Table E.1 shows the ICS for profiles, Cclasses, and file format. A decoder which claims to be compliant to a given profile must support all profiles up to and including the selected profile at the chosen Cclass. For example, if a system claims to be Profile 1 Cclass 1 compliant then the system must be tested for Profile 0 Cclass 1, and Profile 1 Cclass 1. There is now need to show compliance at Cclass 0 when claiming Cclass 1. However, when ETSs are defined for later Cclasses they may require compliance with Cclass 1. At the time of publication of this Recommendation | International Standard, there is no ETS for Cclass 2. The file format can be supported at any given profile and Cclass. Table E.1 allows an IUT to quickly define level of compliance by marking each empty table entry with either supported or not supported.

Table E.1 – ICS for profiles and Cclass

Cclass	Profile 0	Profile 1
Cclass 0		
Cclass 1		
Cclass 2		

File format	Support
JP2 file format	

E.3 Extended support

This Recommendation | International Standard defines parameters for compliance testing for two profiles and three Cclasses. It is expected that an IUT will extend beyond the minimal parameters defined by these profiles and Cclasses. The following tables allow the IUT to claim capabilities beyond the parameters defined for each Cclass in Table A.1. The most common extensions beyond the defined Cclasses are expected to include the ability to decompress a greater image size, a larger number of components, or a higher component depth. The ability to decode a greater image size or larger number of components is directly related to the $\mathrm{N_{cb}}$ and $\mathrm{L_{body}}$ parameters. Therefore, the N_{cb} and L_{body} must be extended to match the claimed image size and components. For example, a system may want to support more components (e.g., three components) than are required for Profile 1 Cclass 0 but does not wish to extend their system to the next Cclass. In this case, the IUT would indicate in Table E.1 that the IUT is Profile 1 Cclass 0 compliant and would include a compliance claim of three components in Table E.2. The increase to three components will change the values in Table E.2 to the following: Components = 3, N_{cb} = 144, and L_{body} = 24 576 bytes. All other values would be the same for this implementation compliance statement. It is expected that test codestreams, not defined in this Recommendation | International Standard, may be required to test these capabilities.

Table E.2 – Extended capabilities for Cclass 0

Parameter	Cclass 0 minimum	IUT extensions to Cclass 0
Size ($W \times H$)	128×128	
Components	1	
N_{cb}	$(HW/1024 + 32)C = 48$	$(HW/1024 + 32)C =$
N_{comp}	64	
L_{body}	$1/2(HWC) = 8192$ bytes	$1/2(HWC) =$
M	11	
P	Low enough to allow 5-3I decoding of 9-7I data	
B	8	
T_L	3	
L	15	
Progressions	All "basic" progressions in COD, only need decode first progression per tile	
Tile Parts	Decode only first tile-part per tile	
Precincts	Decode 1st precinct per sub-band	

Table E.3 – Extended capabilities for Cclass 1

Parameter	Cclass 1 minimum	IUT extensions to Cclass 1
Size ($W \times H$)	2048×2048	
Components	4	
N_{cb}	$(HW/256 + 128)C = 66\,048$	$(HW/256 + 128)C =$
N_{comp}	256	
L_{body}	$HWC/2 = 2^{23}$ bytes	$HWC/2 =$
M	15	
P	16-bit fixed point implementation	
B	12	
T_L	6	
L	255	
Progressions	Limited only by number of levels, layers and components	
Tile Parts	Decode all tile-parts up to N_{cb} or L_{body} limits	
Precincts	Decode all precincts up to N_{cb} or L_{body} limits	

Table E.4 – Extended capabilities for Cclass 2

Parameter	Cclass 2 minimum	IUT extensions to Cclass 2
Size ($W \times H$)	16384×16384	
Components	256	
N_{cb}	$(HW/256 + 128)C = 268\,468\,224$	$(HW/256 + 128)C =$
N_{comp}	16 384	
L_{body}	$HWC/64 = 2^{30}$	$HWC/64 =$
M	30	
P	32-bit floating point implementation	
B	16	
T_L	12	
L	65 535	

Table E.4 – Extended capabilities for Cclass 2

Parameter	Cclass 2 minimum	IUT extensions to Cclass 2
Progressions	Limited only by number of levels, layers and components	
Tile Parts	Decode all tile-parts up to N_{cb} or L_{body} limits	
Precincts	Decode all precincts up to N_{cb} or L_{body} limits	

Annex F

Encoder Implementation Compliance Statement

(This annex does not form an integral part of this Recommendation | International Standard)

F.1 General

The only requirement for encoder compliance to produce compliant codestreams; any other requirements using quality criteria (such as PSNR or MSE) are not part of this Recommendation | International Standard. There are an unlimited number of possible compliant codestreams. Defining tests for all possible compliant codestream variations is not possible. The compliance testing of an encoder IUT is supported by the Implementation Compliance Statement (ICS) described in this annex, which helps define the compliance testing procedures and parameters in Annex D.

Table F.1 – Encoder implementation marker usage

Marker	Marker value	Image header	Tile header	Bitstream	IUT usage
SOC	0xFF4F	Required			
SOT	0xFF90		Required		
SOD	0xFF93		Required		
EOC	0xFFD9			Required	
SIZ	0xFF51	Required			
COD	0xFF52	Required			
COC	0xFF53				
RGN	0xFF5E				
QCD	0xFF5C	Required			
QCC	0xFF5D				
POC	0xFF5F				
TLM	0xFF55				
PLM	0xFF57				
PLT	0xFF58				
PPM	0xFF60				
PPT	0xFF61				
SOP	0xFF91				
EPH	0xFF92				
CRG	0xFF63				
COM	0xFF64				

F.2 Encoder description

Table F.1 describes required and optional markers that are used by an encoder. Each of the markers is defined as required or optional in each section of an encoded codestream (image header, tile header, bitstream). In Table F.1, white blocks labelled "required" are defined as required markers in that section of the encoded bitstream. The gray boxes are optional and the black boxes are not allowed. It is possible that an encoder may not support all optional markers, but it must adhere to all required markers and marker segments. Table F.1 allows the tester to identify what markers are possible and the IUT usage of each marker. For example, an IUT may use the COC marker but only allow it to be used twice in a given codestream. Tables F.2 through F.7 further define each of the markers and marker segments allowing an encoder IUT to define what capabilities it supports in the encoding of an image. Most of these parameters define the capability and functionality of the encoder in the development of a JPEG 2000 compressed codestream.

It may be possible that an encoder is fully compliant but only allows user control over a selected set of encoder parameters. In this case, the IUT may only want to test those cases. For example, an encoder may support both tiles and precincts but may force the user to use tiles of a given size because of consideration for the Cclass of the decoder system. Each of these markers and marker segments define parameters and capabilities of the encoder IUT.

Compliance testing parameters and ranges may be defined by the ICS. For example, an ICS may define Xsiz and Ysiz to be limited to only power of two square images that are between the size of 256-by-256 and 2048-by-2048. Therefore, in the compliance testing, all square power of two image sizes within the bounds (e.g., 256-by-256, 512-by-512, 1024-by-1024, and 2048-by-2048) should be tested. Values outside these should not be tested since the encoder does not claim to be compliant outside these values. Several different image sizes should be tested if the encoder's ICS claimed the ability to encode any image size between 8-by-8 and 2048-by-2048. A good test set would include test images that are 8-by-8, 8-by-2037, 1025-by-15, 513-by-759, and 2048-by-2048. These image sizes are selected to offer the following challenges for an encoder:

a) odd image sizes;

b) non-square image sizes;

c) images that extend to the limits of both the claimed maximum and minimum.

Table F.2 – Delimiting markers and marker segments

Parameter marker	Marker size	Values	IUT supported values	IUT limitations and usage
SOC	16	0xFF4F		
SOT	16	0xFF90		
Lsot	16	10		
Isot	16	0-65 534		
Psot	32	0 is unknown, 14-(2^{32} –1)		
TPsot	8	0-254		
TNsot	8	0 is unknown 1-255 Number of tile-parts		
SOD	16	0xFF93		
EOC	16	0xFFD9		

Table F.3 – Fixed information marker segments

Parameter marker	Marker size	Values	IUT supported values	IUT limitations and usage
SIZ	16	0xFF51		
Lsiz	16	41-49 190		
Rsiz	16	0000 0000 0000 0000 – 0000 0000 0000 0010		
Xsiz	32	1-(2^{32} – 1)		
Ysiz	32	1-(2^{32} – 1)		
XOsiz	32	0-(2^{32} – 2)		
YOsiz	32	0-(2^{32} – 2)		
XTsiz	32	1-(2^{32} – 1)		
YTsiz	32	1-(2^{32} – 1)		
XTOsiz	32	0-(2^{32} – 2)		
YTOsiz	32	0-(2^{32} – 2)		
Csiz	16	1-16 384		
Ssiz	8	x000 0000 – x0100101		
unsigned signed		0xxx xxxx 1xxx xxxx		
XRsiz	8	1-255		
YRsiz	8	1-255		

Table F.4 – Functional marker segments

Parameter marker	Marker size	Values	IUT supported values	IUT limitations and usage
COD	16	0xFF52		
Lcod	16	12-45		
Scod	8	xxxx x000 – xxxx x111		
SGcod	32	32 bits separated into 8, 16 and 8 bits		
Progression order	8	0000 0000 – 0000 0100		
Number of Layers	16	1-65 535		
Multiple Component Transform	8	0000 0000 or 0000 0001		
SPcod	Variable			
Number of Levels	8	0-32		
Code-block Width	8	xxxx 0000 – xxxx 1000		
Code-block Height	8	xxxx 0000 – xxxx 1000		
Code-block Style	8	xx00 0000 – xx11 1111		
Transformation	8	0000 0000 or 0000 0001		
Precinct size	0	if Scod = xxxx xxx0		
	8	0000 0000 – 1111 1111		

Table F.4 – Functional marker segments

Parameter marker	Marker size	Values	IUT supported values	IUT limitations and usage
COC	16	0xFF53		
Lcoc	16	9-43		
Ccoc if Csiz < 257	8	0-255		
if Csiz ≥ 257	16	0-16 383		
Scoc	8	0000 0000 or 0000 0001		
SPcod		Variable size		
Number of Levels	8	0-32		
Code-block Width	8	xxxx 0000 – xxxx 1000		
Code-block Height	8	xxxx 0000 – xxxx 1000		
Code-block Style	8	xx00 0000 – xx11 1111		
Transformation	8	0000 0000 or 0000 0001		
Precinct size	0	if Scoc = xxxx xxx0		
	8	0000 0000 – 1111 1111		
RGN	16	0xFF5E		
Lrgn	16	5 or 6		
Crgn if Csiz is < 257	8	0-255		
if Csiz ≥ 257	16	0-16 383		
Srgn	8	0000 0000		
SPrgn	8	0-255		
QCD	16	0xFF5C		
Lqcd	16	4-197		
Sqcd	8	0000 0000 – 1110 0010		
SPqcd	Variable			
if Sqcd = xxx0 0000	8	0000 0xxx – 1111 1xxx		
if Sqcd = xxx0 0001 or xxx0 0010	16	0000 0000 0000 0000 – 1111 1111 1111 1111		
QCC	16	0xFF5D		
Lqcc	16	5-199		
Cqcc	Variable			
if Csiz is < 257	8	0-255		
Csiz ≥ 257	16	0-16 383		
Sqcc	8	0000 0000 – 1110 0010		
SPqcc	Variable			
if Sqcc = xxx0 0000	8	0000 0xxx – 1111 1xxx		
if Sqcc = xxx0 0001 or xxx0 0010	16	0000 0000 0000 0000 – 1111 1111 1111 1111		

Table F.4 – Functional marker segments

Parameter marker	Marker size	Values	IUT supported values	IUT limitations and usage
POC	16	0xFF5F		
Lpoc	16	9-65 535		
RSpoc	8	0-33		
CSpoc	Variable			
if Csiz is < 257	8	0-255		
if Csiz ≥ 257	16	0-16 383		
LYEpoc	16	0-65 535		
REpoc	8	(RSpoc + 1)-33		
CEpoc	Variable			
if Csiz is < 257	8	(CSpoc + 1)-255, 0		
if Csiz ≥ 257	16	(CSpoc + 1)-16 384, 0		
Ppoc	8	0000 0000 – 0000 0100		

Table F.5 – Pointer marker segments

Parameter marker	Marker size	Values	IUT supported values	IUT limitations and usage
TLM	16	0xFF55		
Ltlm	16	6-65 535		
Zltm	8	0-255		
Sltm	8	x000 xxxx – x111 xxxx		
Ttlm	Variable			
if ST = 0	0	Tiles in order		
if ST = 1	8	0-254		
if ST = 2	16	0-65 534		
Ptlm	Variable			
if SP = 0	16	14-65 535		
if SP = 1	32	14-(2^{32} – 1)		
PLM	16	0xFF57		
Lplm	16	4-65 535		
Zplm	8	0-255		
Nplm	8	0-255		
Iplm	8 (repeated)	0000 0000 – 1111 1111		
PLT	16	0xFF58		
Lplt	16	4-65 535		
Zplt	8	0-255		
Iplt	8 (repeated)	0000 0000 – 1111 1111		

Table F.5 – Pointer marker segments

Parameter marker	Marker size	Values	IUT supported values	IUT limitations and usage
PPM	16	0xFF60		
Lppm	16	7-65 535		
Zppm	8	0-255		
Nppm	32	0-($2^{32} - 1$)		
Ippm	Variable	Packet Headers		
PPT	16	0xFF61		
Lppt	16	4-65 535		
Zppt	8	0-255		
Ippt	Variable	Packet headers		

Table F.6 – In bitstream markers and marker segments

Parameter marker	Marker size	Values	IUT supported values	IUT limitations and usage
SOP	16	0xFF91		
Lsop	16	4		
Nsop	16	0-65 535		
EPH	16	0xFF92		

Table F.7 – Informational marker segments

Parameter marker	Marker size	Values	IUT supported values	IUT limitations and usage
CRG	16	0xFF63		
Lcrg	16	6-65 534		
Xcrg	16	0-65 535		
Ycrg	16	0-65 535		
COM	16	0xFF64		
Lcom	16	5-65 535		
Rcom	16	0 or 1		
Ccom	8	0-255		

Annex G

JP2 File Format Reader Compliance Testing Procedure

(This annex does not form an integral part of this Recommendation | International Standard)

G.1 General

Every compliant JP2 file format reader must be able to fully decode or partially decode the codestream contained within the file within the terms as defined in Annexes B and C. In addition, the reader must also be able to properly interpret the colourspace of the decoded image data as specified within the file format. Minimal compliance for the interpretation of colourspace is defined as the possible upsampling of the decoded image data such that all decoded components are at the same resolution, and the transformation of the decoded image data to the sRGB colourspace for display on a typical computer monitor. While it is well understood that many applications will not convert images directly to the sRGB colourspace, the use of sRGB as a comparison point provides a simple and accurate way to compare output of an application under test with reference output.

While this compliance test is optional with respect to all implementations of the JPEG 2000 standard (as implementation of the JP2 file format is optional for compliant decoders), it is required and normative for all applications that claim compliance with the JP2 file format as defined in Annex I of ITU-T Rec. T.800 | ISO/IEC 15444-1.

This normative compliance test includes the required test files. The compliance test is separated into two parts: first, decoding of the codestream contained within the JP2 file; and second, the interpretation of the colourspace of the decoded image data.

G.2 Compliance requirement and acceptance

A compliant file format reader must first pass the compliance tests for decoding as defined in Annexes B and C. It must also be able to properly interpret the colourspace of each test file. Unlike the compliance tests defined in Annexes B and C, this test does not differentiate between full and partial decode of the image. Any JP2 file format reader that can properly extract the codestream from a JP2 file and decode the codestream to sufficiently pass the tests in Annexes B and C passes the first part of the file format compliance test.

For the purposes of this test, all test images shall be fully decoded in order to minimize the number of reference images that must be tested against. It is assumed that if a file format reader passes the tests in Annexes B and C, then it is capable of properly decoding any codestream contained within a JP2 file.

G.3 Reading a JP2 file compliance test procedure

Figure G.1 shows the flow of the lossless decoder compliance test. A decoder is found to be compliant if the resulting test data, for the tests specified for a particular process, match, within the defined tolerances, to the reference test image.

a) Each test file is decoded using the decoder under test.

b) If the decoded components are not all at the same resolution, all components should be upsampled to the same resolution. While in general applications are not required to upsample the decoded data, and the particular method of interpolation is outside of the scope of this Recommendation | International Standard, this test requires the interpolation of the data in order to bring all images for comparison to a single state.

c) Convert the decoded image data from the source colourspace (as indicated in the Colour Specification boxes within the JP2 file) to the sRGB colourspace. For readers that do not display images (i.e., printers), it is acceptable to convert the images to the desired output colourspace of the device. In this case, the reference output images must also be converted to the same output colourspace for comparison.

d) Compare the reconstructed image (converted image) to the reference sRGB output for that test file by computing the maximum difference between the two images.

e) The peak error is compared with the bound for the test image.

Failure to meet the tolerance limits for a single image results in the decoder failing to be compliant for the file format test.

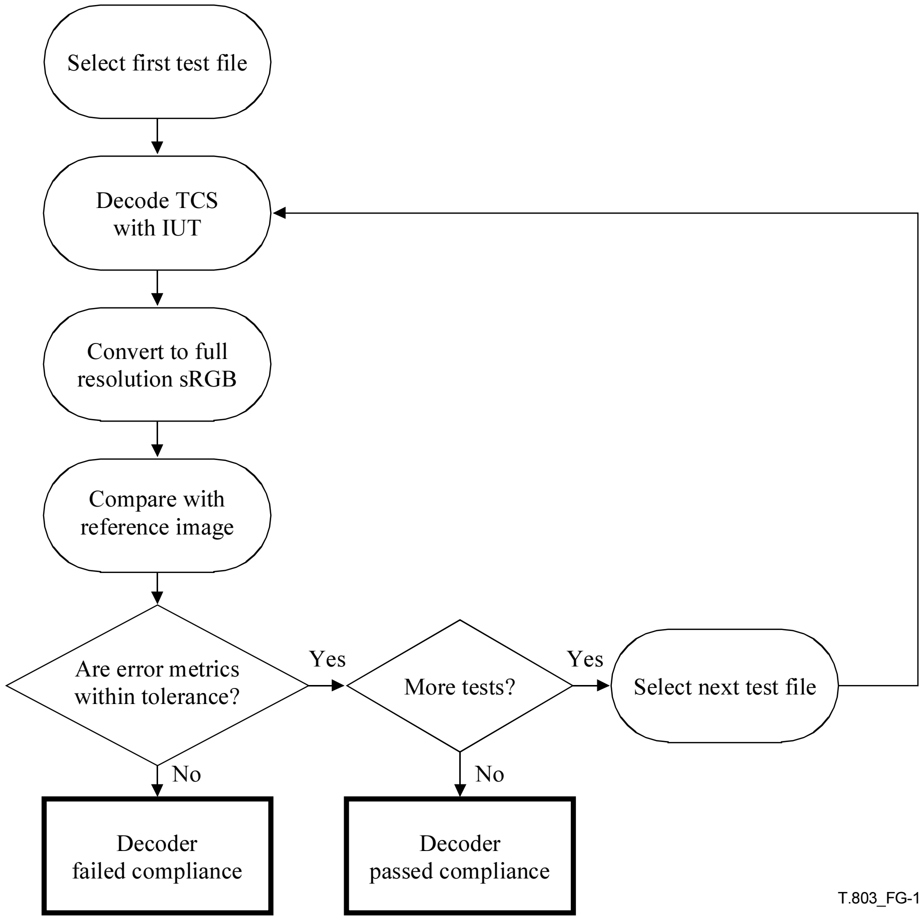

Figure G.1 – JP2 file format reader compliance test block diagram

G.4 JP2 file format test codestreams and images

G.4.1 Test files

The test files for this ETS are in the directory `testfiles_jp2`. The codestream names have the form `file#.jp2`, where '#' is the codestream number.

G.4.2 Reference decoded images

The reference decoded images for this ETS are in the directory `reference_jp2`. The filenames are of the form `jp2_#.tif`, where '#' is the file number. A decoder shall produce all components listed. All codestreams within the files are decoded at full resolution.

G.4.3 Tolerances

The maximum allowable peak errors are listed in Table G.1, along with the size of the reference decoded images.

Table G.1 – JP2 reference images and allowable error

Test file	Reference filename	Components	Depth	Width	Height	Peak
file1.jp2	jp2_1.tif	3	8	768	512	4
file2.jp2	jp2_2.tif	3	8	480	640	4
file3.jp2	jp2_3.tif	3	8	480	640	4
file4.jp2	jp2_4.tif	1	8	768	512	4
file5.jp2	jp2_5.tif	3	8	768	512	4

Table G.1 – JP2 reference images and allowable error

Test file	Reference filename	Components	Depth	Width	Height	Peak
file6.jp2	jp2_6.tif	1	12	768	512	4
file7.jp2	jp2_7.tif	3	16	480	640	4
file8.jp2	jp2_8.tif	1	8	700	400	4
file9.jp2	jp2_9.tif	1	8	768	512	4

G.4.4 Additional information

This information is provided solely to help correction of non-compliant decoders, and is not guaranteed. All the test files conform to Profile 0. In addition:

– **file1.jp2**: Three 8-bit components in the sRGB colourspace. The components are in the standard order and are transformed using the RCT. This file also includes XML metadata. If a reader fails on this file, the likely cause is a problem with parsing of the boxes.

– **file2.jp2**: Three 8-bit components in the sRGB-YCC colourspace. All components are at the full resolution, but are stored in reverse order in the codestream. The JP2 file contains a Channel Definition box that correctly associates each physical component with the correct colour in the sRGB-YCC definition. If a reader fails on this file, the likely causes are an incorrect interpretation of the Channel Definition box or an error in the conversion from sRGB-YCC to sRGB.

– **file3.jp2**: Three 8-bit components in the sRGB-YCC colourspace, with the Cb and Cr components being subsampled 2x in both the horizontal and vertical directions. The components are stored in the standard order. If a reader fails on this file, the likely causes are an error in either resampling or in conversion from sRGB-YCC to sRGB.

– **file4.jp2**: One 8-bit component in the sRGB-grey colourspace. If a reader fails on this file, the likely cause is an error in parsing of the boxes.

– **file5.jp2**: Three 8-bit components in the ROMM-RGB colourspace, encapsulated in a JP2 compatible JPX file. The components have been transformed using the RCT. The colourspace is specified using both a Restricted ICC profile and using the JPX-defined enumerated code for the ROMM-RGB colourspace. If a decoder fails on this file, the likely cause is an incorrect implementation of the Restricted ICC Three-Component transformation.

– **file6.jp2**: One 12-bit component in the sRGB-grey colourspace. If a reader fails on this file, the likely cause is an error in the extraction of 8 bits from the 12-bit codestream.

– **file7.jp2**: Three 16-bit components in the e-sRGB colourspace, encapsulated in a JP2 compatible JPX file. The components have been transformed using the RCT. The colourspace is specified using both a Restricted ICC profile and using the JPX-defined enumerated code for the e-sRGB colourspace. If a reader fails on this file, the likely causes are an error in conversion from the 16-bit data to 8-bit, or an error in implementation of Restricted ICC support.

– **file8.jp2**: One 8-bit component in a gamma 1.8 space. The colourspace is specified using a Restricted ICC profile. If a reader fails on this file, the likely cause is the implementation of the Monochrome Input profile as part of Restricted ICC.

– **file9.jp2**: One 8-bit component, which is used as input to a 256-entry palette that maps the single component to three 8-bit components. The depaletized components are in the sRGB colourspace. If a reader fails on this test, the likely cause is the application of the palette to the single component.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of 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	Cable networks and 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

Printed in Switzerland

Geneva, 2004