INTERNATIONAL STANDARD

ISO/IEC 15444-9

First edition
2005-12-01

AMENDMENT 4
2010-12-15

Information technology — JPEG 2000 image coding system: Interactivity tools, APIs and protocols

AMENDMENT 4: JPIP server and client profiles

Technologies de l'information — Système de codage d'images JPEG 2000: Outils d'interactivité, interfaces de programmes d'application et protocoles

AMENDEMENT 4: Profils du serveur JPIP et du client

Reference number
ISO/IEC 15444-9:2005/Amd.4:2010(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published by ISO in 2011

Published in Switzerland

CONTENTS

Electronic attachment = JPIP test data and scripts

Foreword

IISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 15444-9:2005/Amd.4 was prepared jointly by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information* in collaboration with ITU-T. The identical text is published as Rec. ITU-T.808(2005)/Amd.4 (05/2010).

INTERNATIONAL STANDARD
RECOMMENDATION ITU-T

Information technology – JPEG 2000 image coding system: Interactivity tools, APIs and protocols

Amendment 4

JPIP server and client profiles

1) Clause 2, Normative references

Add the following reference to clause 2:

– Recommendation ITU-T T.803 (2002) | ISO/IEC 15444-4:2004, *Information technology – JPEG 2000 image coding system: Conformance testing.*

2) Clause 5.1

a) Replace the definition of `TOKEN` *by:*

```
TOKEN      =  1*(ALPHA / DIGIT / "." / "_" / "-")
```

b) Add the following token at the end of the ABNF-rules:

```
IDTOKEN    =  1*(TOKEN / ";")
```


3) Clause C.2.4

Replace the definition of the `target-id` *by the following:*

```
target-id = IDTOKEN
```

4) Clause C.3.2

Replace the definition of the `channel-id` *by the following:*

```
channel-id = IDTOKEN
```

5) Clause L.1

Add to the list of fields in L.1 the following:

```
;================================
; C.4.5 Frame Size for Variable Dimension Data (fvsiz)
;================================
fvsiz = "fvsiz" "=" 1#UINT ["," round-direction]
round-direction = "round-up" / "round-down" / "closest"
;================================
; C.4.7 Offset for Variable Dimension Data (rvoff)
;================================
rvoff = "rvoff" "=" 1#UINT
;================================
```

```
; C.4.8 Region Size for Variable Dimension Data(rvsiz)
;================================
rvsiz = "rvsiz" "=" 1#UINT
```

6) Clause L.2

a) *Replace the list of* `jpip-response` `headers` *by the following:*

```
jpip-response-header =
                    / JPIP-tid                  ; D.2.2
                    / JPIP-cnew                 ; D.2.3
                    / JPIP-qid                  ; D.2.4
                    / JPIP-fsiz                 ; D.2.5
                    / JPIP-rsiz                 ; D.2.6
                    / JPIP-roff                 ; D.2.7
                    / JPIP-fvsiz                ; D.2.8
                    / JPIP-rvsiz                ; D.2.9
                    / JPIP-rvoff                ; D.2.10
                    / JPIP-comps                ; D.2.11
                    / JPIP-stream               ; D.2.12
                    / JPIP-context              ; D.2.13
                    / JPIP-roi                  ; D.2.14
                    / JPIP-layers               ; D.2.15
                    / JPIP-srate                ; D.2.16
                    / JPIP-metareq              ; D.2.17
                    / JPIP-len                  ; D.2.18
                    / JPIP-quality              ; D.2.19
                    / JPIP-type                 ; D.2.20
                    / JPIP-mset                 ; D.2.21
                    / JPIP-cap                  ; D.2.22
                    / JPIP-pref                 ; D.2.23
                    / JPIP-align                ; D.2.24
                    / JPIP-subtarget            ; D.2.25
```


b) *Add respectively the following items to the list of JPIP Response BNF:*

```
;================================
; D.2.8 Frame Size for Variable Dimension Data (JPIP-fvsiz)
;================================
JPIP-fvsiz = "JPIP-fvsiz" ":" LWSP 1#UINT
;================================
; D.2.9 Region Size for Variable Dimension Data(JPIP-rvsiz)
;================================
```

```
JPIP-rvsiz = "JPIP-rvsiz" ":" LWSP 1#UINT

;================================

; D.2.10 Offset for Variable Dimension Data (JPIP-rvoff)

;================================

JPIP-rvoff = "JPIP-rvoff" ":" LWSP 1#UINT

;================================

; D.2.24 Alignment (JPIP-align)

;================================

JPIP-align = "JPIP-align" ":" LWSP "yes" / "no"

;================================

; D.2.25 Subtarget (JPIP-subtarget)

;================================

JPIP-subtarget = "JPIP-subtarget" ":" LWSP byte-range / src-codestream-specs
```

7) Clause 6.2

a) *Replace the third point of the second paragraph by the following:*

– Annex C defines the client request syntax. The client shall produce compliant requests and the server shall be able to parse, interpret and respond to all compliant requests.

b) *Add the following to 6.2:*

– Server and client conformance is further structured into profiles and variants. Profiles define which fields servers must support and implement beyond simply parsing and interpreting. Variants define the operating modes and features of the JPIP standard a client and server use to transmit data. Clients and servers must provide a common subset of variants in order to interoperate. See Annex J for details about conformance and testing for conformance.

8) Clause 7

Change clause 7 to:

Conformance with this Recommendation | International Standard by a client means that the client's JPIP requests are well structured, valid and conformant to the JPIP client requests as defined by this Recommendation | International Standard, and that it is able to parse the JPIP responses defined by this Recommendation | International Standard.

Conformance with this Recommendation | International Standard by a server means that the server's JPIP responses are well structured, valid and conformant to the JPIP server response signalling as defined by this Recommendation | International Standard, and is able to parse the JPIP requests defined by this Recommendation | International Standard. Servers shall parse and interpret all normative request types and shall respond to all compliant requests. Compliance to a profile requires servers furthermore to support and implement all mandatory fields within that profile to the extent defined in Annex J.

Conformance, profiles and conformance testing methodologies of this Recommendation | International Standard are defined in Annex J.

It is expected that server applications may reduce efficiency by sending additional data not explicitly requested for or redundant data depending on the network quality-of-service. Such implementation decisions are application specific and provide the JPIP system with high utility.

9) Clause 3.3

Add the following definitions:

3.3.24 **profile**: Conformance is structured according to profiles; a profile defines the set of request fields that a server is expected to support and implement and a client communicating with a server in this profile may issue expecting the

server to support them. A server is conforming to a profile if it supports and implements all request fields within this profile to the extent defined in Annex J.

3.3.25 variant: Variants define the operating modes and features of the JPIP standard that a client and a server use to exchange requests and data. Clients and servers must provide a common subset of variants in order to interoperate.

10) Clause 5.1

In 5.1, delete the definition of ENCODED-CHAR, *and insert the following definition instead, and replace all occurrences of* ENCODED-CHAR *by* OCTAL-ENCODED-CHAR*:*

```
OCTAL-ENCODED-CHAR = "\" QUADDIG OCTDIG OCTDIG
QUADDIG = "0" / "1" / "2" / "3"
OCTDIG = "0" / "1" / "2" / "3" / "4" / "5" / "6" / "7"
```

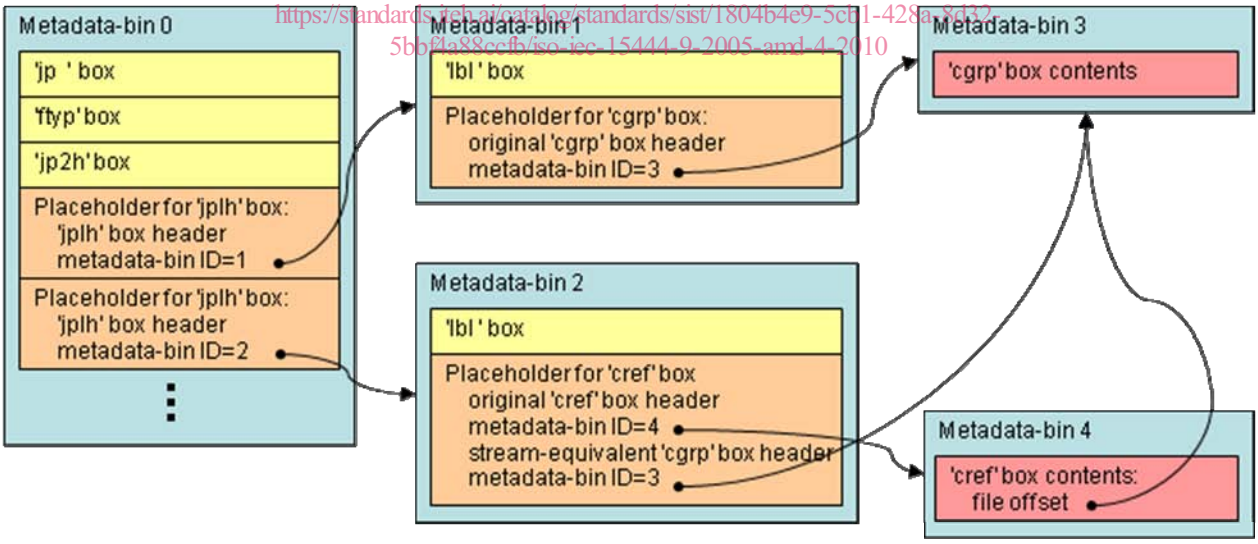
11) Clause 5.2

Replace all occurrences of ENCODED-CHAR *by* OCTAL-ENCODED-CHAR *and replace the first paragraph of 5.2 by the following:*

`box-type` specifies the four characters of the box type. For each character in the box type, if the character is alpha-numeric (A..Z, a..z or 0..9), the character is written directly into the string. If the character is a space (\040), then that character shall be encoded as the underscore character ("_") or by octal encoding. For any other character, a 4-character string is written in its place, consisting of a backslash character ("\") followed by three octal digits representing the value of the character from the box type in octal. The `compatibility-code` is encoded the same way that a `box-type` is encoded.

12) Clause A.3.6.2

At the end of A.3.6.2, replace the example Figure A.10 by the following:



13) Clause A.3.6.3

a) After the last paragraph of A.3.6.3, add a Note:

NOTE – The above definition implies that the placeholder box may be truncated after the last used field, but intermediate fields, even if unused, must be present.

b) In Table A.3, row five – the description of the flag value 01xx *– change the last sentence of the description in the right column to:*

The value of the NCS field shall be treated as if it was set to "1" regardless of the actual value of that field when the field is present.

c) *Add the word "minimal" to the beginning of the description of each cut-off point in Figure A.11.*

14) Clause C.1.1

Replace the last sentence of the paragraph by:

Finally, even with a conforming request, a server might not implement all possible request fields or combinations thereof, but it must parse and interpret all normative request fields and respond appropriately, even if this response is an error. Details of what servers are expected to implement are defined in Annex J.

NOTE – Which responses or methods for signalling error conditions are appropriate depends on the transport layer used. Clause D.1 provides examples for servers using HTTP as transport protocol.

15) Clause C.1.2

a) *Remove the Note before the definition of the* `jpip-request-field` *and add the following:*

The fields in the request shall be sent in compliance with the selected transport protocol. For example, in HTTP, the requests may be part of the query field of a GET request, or the body of a POST request, with individual request fields separated by a "&" character – see Annexes F, G and H for details. In contexts such as these, certain characters found within the BNF syntax or the request parameters may need to be escaped in order to avoid ambiguity. For example, a request field of the form `"target=me&my dog"` should be escaped in an HTTP context, becoming `"target=me%26my%20dog"`, so as to avoid confusion with the "&" used to separate request fields. As another example, `"metareq=[roid/w]"` should be escaped in an HTTP context, becoming `"metareq=%5broid/w%5d"` so as to avoid the use of non-URI character – see IETF RFC 2396 for more on reserved characters, disambiguation and escaping via the hex-hex encoding. Parsers of requests found in such contexts should be prepared to perform hex-hex decoding of each request field.

b) *Replace the* `view-window field` *list by the following:*


```
view-window-field  = fsiz                    ; C.4.2
                   / roff                    ; C.4.3
                   / rsiz                    ; C.4.4
                   / fvsiz                   ; C.4.5
                   / comps                   ; C.4.6
                   / rvoff                   ; C.4.7
                   / rvsiz                   ; C.4.8
                   / stream                  ; C.4.9
                   / context                 ; C.4.10
                   / srate                   ; C.4.11
                   / roi                     ; C.4.12
                   / layers                  ; C.4.13
```

16) Clause C.2.1

Replace the eighth paragraph (before the examples) of C.2.1 by the following:

If the channel ID request field is included in the request, the request need not include Target, Sub-Target or Target ID fields.

17) Clause C.2.3

Replace the body of C.2.3 by:

```
subtarget = "subtarget" "=" byte-range / src-codestream-specs

byte-range = UINT-RANGE

src-codestream-specs = "c" UINT-RANGE
```

This field may be used to qualify the original named resource through the specification of a byte range or a range of codestreams in the original resource. The logical target is to be interpreted as the indicated byte range or a range of