
Information technology — MPEG systems technologies —

Part 1:

Binary MPEG format for XML

**AMENDMENT 2: Conservation of prefixes
and extensions on encoding of wild cards**

Technologies de l'information — Technologies des systèmes MPEG —

Partie 1: Format binaire de MPEG pour XML

*AMENDEMENT 2: Conservation des préfixes et des extensions pour
l'encodage des caractères de remplacement*

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2008

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

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 in Switzerland

Foreword

ISO (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.

Amendment 2 to ISO/IEC 23001-1:2006 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

Information technology — MPEG systems technologies —

Part 1: Binary MPEG format for XML

AMENDMENT 2: Conservation of prefixes and extensions on encoding of wild cards

In 6.2.2, replace:

DecoderInit () {	Number of bits	Mnemonic
SystemsProfileLevelIndication	8+	vluimsbf8
...		

with:

DecoderInit () {	Number of bits	Mnemonic
SystemsProfileLevelIndication	8+	vluimsbf8
UnitSizeCode	3	bslbf
NoAdvancedFeatures	1	bslbf
ReservedBits	4	bslbf
If (! NoAdvancedFeatures) {		
AdvancedFeatureFlags_Length	8+	vluimsbf8
/** FeatureFlags **/		
InsertFlag	1	bslbf
AdvancedOptimisedDecodersFlag	1	bslbf
AdditionalSchemaFlag	1	bslbf
AdditionalSchemaUpdatesOnlyFlag	1	bslbf
FragmentReferenceFlag	1	bslbf
MPCOnlyFlag	1	bslbf
HierarchyBasedSubstitutionCodingFlag	1	bslbf
ContextPathTableFlag	1	bslbf
PrefixTableFlag	1	bslbf
ReservedBitsZero /**for additional Flags **/	24	bslbf
If (PrefixTableFlag) {		
PrefixTable ()		
}		
ReservedBitsZero	AdvancedFeatureFlags_Lent	

	h*8-33-PrefixTable_length*8	
}/** FeatureFlags end **/		
/** Start FUUConfig **/		
If (! AdditionalSchemaUpdatesOnlyFlag) {		
NumberOfSchemas	8+	vluimsbf8
for (k=0; k< NumberOfSchemas; k++) {		
SchemaURI_Length[k]	8+	vluimsbf8
SchemaURI[k]	8* SchemaURI_Length[k]	bslbf
LocationHint_Length[k]	8+	vluimsbf8
LocationHint[k]	8* LocationHint_Length[k]	bslbf
NumberOfTypeCodecs[k]	8+	vluimsbf8
for (i=0; i< NumberOfTypeCodecs[k]; i++) {		
TypeCodecURI_Length[k][i]	8+	vluimsbf8
TypeCodecURI[k][i]	8* TypeCodecURI_Length[k][i]	bslbf
NumberOfTypes[k][i]	8+	vluimsbf8
for (j=0; j< NumberOfTypes[k][i]; j++) {		
TypeIdentificationCode[k][i][j]	8+	vluimsbf8
}		
}		
}		
If (ContextPathTableFlag) {		
ContextPathTable()		
}		
/** FUUConfig - Advanced optimised decoder framework **/		
If (AdvancedOptimisedDecodersFlag) {		
NumOfAdvancedOptimisedDecoderTypes	8+	vluimsbf8
for (i=0; i< NumOfAdvancedOptimisedDecoderTypes; i++) {		
AdvancedOptimisedDecoderTypeURI_Length[i]	8+	vluimsbf8
AdvancedOptimisedDecoderTypeURI[i]	8* AdvancedOptimisedDecoderTypeURI_Length[i]	bslbf
}		
AdvancedOptimisedDecodersConfig ()		
}		
/** FUUConfig - Fragment reference framework **/		
If (FragmentReferenceFlag) {		
NumOfSupportedFragmentReferenceFormat	8	uimsbf
for (i=0;i< NumOfSupportedFragmentReferenceFormat;i++) {		
SupportedFragmentReferenceFormat[i]	8	bslbf
}		

}		
}		
/** end FUUConfig **/		
If (AdditionalSchemaFlag) {		
AdditionalSchemaConfig ()		
}		
/** Initial document **/		
If (!AdditionalSchemaUpdateOnlyFlag) {		
InitialDocument_Length	8+	vluimsbf8
InitialDocument()		
}		
}		

Add the following paragraph:

PrefixTable {		
NumberOfPrefixes	8+	vluimsbf8
for (k=0; k< NumberOfPrefixes; k++) {		
SchemaURI_Length[k]	8+	vluimsbf8
SchemaURI[k]	8* SchemaURI_Lengt h[k]	bslbf
Prefix_Length[k]	8+	vluimsbf8
Prefix[k]	8* [Prefix_Length[k]]	bslbf
}		
}		

Add the following paragraph:

Semantics

Name	Definition
PrefixTable_length	Indicates length in Bytes of PrefixTable.
PrefixTableFlag	Indicates whether a prefix table is given in the <code>decoderInit</code> .
NumberOfPrefixes	Specifies the number of prefixes defined in the PrefixTable.
Prefix_Length[k]	Indicates the size in bytes of the Prefix[k]. A value of zero is forbidden.
Prefix[k]	This is the UTF-8 representation of the prefix associated to SchemaURI[k] in the PrefixTable. The decoder SHOULD bind the namespace names in the current document to the prefixes given in the PrefixTable if present. The default namespace, used to construct unprefixed elements, will be declared as a null string "".

If different prefixes are associated to a same namespace name in the

PrefixTable, or a given namespace name is associated to different prefixes, the decoder should only use the first defined prefix/namespace binding in the output document.

NOTE A document author wishing to use prefix conservation in order to maintain, after decoding, the validity of QNames/XPath expressions in the content of an element should only use one global prefix/namespace name mapping for those prefix and namespace names that are used in the QNames/XPath expressions. However, in order to ensure in all cases the validity of those expressions after decoding, the optimized decoder for QName expressions specified in subclause 8.8 should be used.

In 6.2.2, replace:

Table 1 — Index Table for SystemsProfileLevelIndication

<i>Index</i>	<i>Systems Profile and Level</i>
0	no profile specified
1 – 127	Reserved for ISO Use

with:

Table 1 — Index Table for SystemsProfileLevelIndication

<i>Index</i>	<i>Systems Profile and Level</i>
0	no profile specified
1 – 124	Reserved for ISO Use
125	ETSI TS 102 323/102 529 (DVB GBS TV-A and DVB IPTV BCG)
126	ETSI TS 102 034 (DVB IPTV SD&S)
127	ETSI TS 102 471 (DVB IPDC ESG)
128-	User specific

In 7.5.2.4.5.2 (AnyElementDecoding Syntax), replace:

AnyElementDecoding() {	Number of bits	Mnemonic
GlobalElementSchemaID	$\text{ceil}(\log_2(\text{NumberOfSchemas} + \text{NumberOfAdditionalSchemas}))$	uimsbf
AnyElement_Length	5+	vluimsbf
Any_SBC_Operand_Selector	5+	vluimsbf
If (inPayloadDecoding()) {		
Element(ChildrenSchemaMode, theAnyType)		
}		
}		

with: