



INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

J.380.2

(11/2011)

SERIES J: CABLE NETWORKS AND TRANSMISSION
OF TELEVISION, SOUND PROGRAMME AND OTHER
MULTIMEDIA SIGNALS

Digital transmission of television signals

Digital Program Insertion–Advertising Systems Interfaces: Part 2 - Core data elements

CAUTION !

PREPUBLISHED RECOMMENDATION

This prepublication is an unedited version of a recently approved Recommendation. It will be replaced by the published version after editing. Therefore, there will be differences between this prepublication and the published version.

FOREWORD

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

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

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

NOTE

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

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

INTELLECTUAL PROPERTY RIGHTS

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

As of the date of approval of this Recommendation, ITU [had/had not] received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at <http://www.itu.int/ITU-T/ipr/>.

© ITU 2011

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

Recommendation ITU-T J.380.2 (ex J.dpi-asi-2)

Digital Program Insertion–Advertising Systems Interfaces: Part 2 - Core data elements

Summary

This document describes the Digital Program Insertion Advertising Systems Interfaces' core messaging and data types using extensible mark-up language (XML), XML Namespaces, and XML Schema.

Keywords

(TBD)

...

TABLE OF CONTENTS

Document Organization	7
1 Scope	8
2 References.....	8
3 Definitions	9
3.1 Terms defined elsewhere	9
3.2 Terms defined in this Recommendation.....	9
4 Abbreviations and acronyms	9
5 Conventions	10
5.1 Normative XML Schema	10
5.2 Document Conventions	10
6 XML Namespaces	12
6.1 Unknown/Unrecognized/Unsupported XML Elements and Attributes	12
6.2 Element Order	12
6.3 Language Identification.....	12
7 J.380 Message Basics	12
7.1 Common Schema for All Messages	14
7.1.1 Semantic Definitions for the J.380 Message Common Attributes.....	14
7.2 Request and Notification Messages Base Schema	15
7.2.1 Semantic Definitions for the Request and Notification Messages Base Schema	16
7.3 Response and Acknowledgement Messages Base Schema.....	17
7.3.1 Semantic Definitions for the Response and Acknowledgement Messages Base Schema.....	18
7.4 J.380 Message Characterizations.....	19
7.4.1 Transport Mechanisms	19
7.4.2 Message Order	20
7.4.3 Multiple Messages	20
7.4.4 Message Timeliness	20
7.4.5 Message Specification Versioning.....	20
7.4.6 Message Error Handling	20
7.4.7 High Availability and Message Retransmission	21
7.4.8 List Registration	24
7.5 Addressing.....	24
7.5.1 Internet Protocol version 4 (IPv4) Address.....	24
7.5.2 Internet Protocol version 6 (Ipv6) Address.....	24

	7.5.3	Ipv4 and Ipv6 Port Identifier	25
	7.5.4	IEEE Media Access Control (MAC) Address	26
8	J.380.2	Messages	26
	8.1	Service Check Messages	27
	8.1.1	ServiceCheckRequest Message Schema	27
	8.1.2	ServiceCheckResponse Message Schema.....	29
	8.1.3	Service Check Message Examples.....	30
	8.2	Service Status Messages.....	32
	8.2.1	ServiceStatusNotification Message Schema	32
	8.2.2	ServiceStatusAcknowledgement Message Schema	34
	8.2.3	Service Status Message Examples	36
9	J.380	Core Attribute Types and Elements	38
	9.1	Semantic Definitions for J.380 Core Types	38
	9.1.1	dateTimeTimezoneType Simple Type	38
	9.1.2	nonEmptyStringType Simple Type.....	38
	9.2	Semantic Definitions for J.380 Core Attribute Types.....	38
	9.2.1	identityAttrType Attribute Type	38
	9.2.2	idAttrType Attribute Type	39
	9.2.3	mediaAvailableAttrType Attribute Type	39
	9.2.4	messageIdAttrType Attribute Type.....	39
	9.2.5	messageRefAttrType Attribute Type	40
	9.2.6	registrationRefAttrType Attribute Type	40
	9.2.7	resendAttrType Attribute Type	40
	9.2.8	systemAttrType Attribute Type	40
	9.2.9	versionAttrType Attribute Type.....	40
	9.3	Address Element.....	40
	9.3.1	Address Element Schema.....	40
	9.3.2	Address Element Examples.....	41
	9.4	AdType Element.....	41
	9.4.1	AdType Element Schema.....	41
	9.4.2	AdType Element Examples.....	42
	9.5	AssetRef Element	42
	9.5.1	AssetRef Element Schema	42
	9.5.2	AssetRef Element Examples	43
	9.6	Callout Element	43
	9.6.1	Callout Element Schema	44
	9.6.2	Callout Element Examples	45
	9.7	Content Element	47
	9.7.1	Content Element Schema	47

9.7.2	Content Element Examples	50
9.8	ContentDataModel Element	50
9.8.1	ContentDataModel Element Schema	50
9.8.2	ContentDataModel Element Examples	51
9.9	ContentLocation Element	51
9.9.1	ContentLocation Element Schema	51
9.9.2	ContentLocation Element Examples	52
9.10	CurrentDateTime Element	53
9.10.1	CurrentDateTime Element Schema	53
9.10.2	CurrentDateTime Element Examples	53
9.11	Duration Element	53
9.11.1	Duration Element Schema	53
9.11.2	Duration Element Examples	54
9.12	Ext Element	54
9.12.1	Ext Element Schema	54
9.12.2	Ext Element Examples	54
9.13	ExternalStatusCode Element	54
9.13.1	ExternalStatusCode Schema	54
9.13.2	ExternalStatusCode Element Examples	56
9.14	InitiatorData Element	56
9.14.1	InitiatorData Element Schema	56
9.14.2	InitiatorData Element Examples	56
9.15	Note Element	56
9.15.1	Note Element Schema	57
9.15.2	Note Element Examples	57
9.16	Program Element	57
9.16.1	Program Element Schema	57
9.16.2	Program Element Examples	58
9.17	SegmentationUpid Element	58
9.17.1	SegmentationUpid Element Schema	58
9.17.2	SegmentationUpid Element Examples	60
9.18	SpotRef	60
9.18.1	SpotRef Element Schema	60
9.18.2	SpotRef Element Examples	62
9.19	StatusCode Element	62
9.19.1	StatusCode Element Schema	62
9.19.2	StatusCode Element Examples	64
9.20	Tracking Element	64
9.20.1	Tracking Element Schema	65
9.20.2	Tracking Element Examples	65

9.21	URI Element	65
9.21.1	URI Element Schema	65
9.21.2	URI Element Examples	66

LIST OF FIGURES

Figure 1: Schema Illustration Explanation.....	11
Figure 2: Basic Communication Overview.....	13
Figure 3: Service Channel Illustrated.....	13
Figure 4: J.380 Message Common Attribute Schema.....	14
Figure 5: Request Message Base Schema	15
Figure 6: Notification Message Base Schema	16
Figure 7: Response Message Base Schema	17
Figure 8: Acknowledgement Message Base Schema	18
Figure 9: @messageRef and InitiatorData Paired Message Linkage.....	19
Figure 10: Intermediary Logical Service	20
Figure 11: Example Resend Message Sequence.....	22
Figure 12: Service Check Message Exchange	27
Figure 13: ServiceCheckRequest Message Schema	28
Figure 14: ServiceCheckResponse Message Schema	29
Figure 15: Service Status Message Exchange.....	32
Figure 16: ServiceStatusNotification Message Schema	33
Figure 17: ServiceStatusAcknowledgement Message Schema	35
Figure 18: Basic @identity Attribute Mapping	38
Figure 19: Complex @identity Attribute Mappings	39
Figure 20: Address Element Schema	41
Figure 21: AdType Element Schema	42
Figure 22: AssetRef Element Schema	43
Figure 23: Callout Element Schema	45
Figure 24: Content Element Schema	48
Figure 25: ContentDataModel Element Schema	50
Figure 26: ContentLocation Element Schema	52
Figure 27: CurrentDateTime Element Schema	53
Figure 28: Duration Element Schema	53
Figure 29: Ext Element Schema.....	54
Figure 30: ExternalStatusCode Element Schema	55
Figure 31: InitiatorData Element Schema.....	56
Figure 32: Note Element Schema	57
Figure 33: Program Element Schema	57
Figure 34: SegmentationUpid Element Schema	59
Figure 35: SpotRef Element Schema	61
Figure 36: StatusCode Element Schema	63
Figure 37: Tracking Element Schema.....	65
Figure 38: URI Element Schema	65

LIST OF TABLES

Table 1: XML Namespace Declarations	12
---	----

Table 2: ITU-T J.380.2 Messages 27

Table 3: ContentDataModel Element's @type Attribute Defined Values 51

Table 4: ExternalStatusCode Element's @source Attribute Values 55

Table 5: StatusCode Element @class Attribute Values..... 63

Table 6: StatusCode Element @detail Attribute Values..... 67

LIST OF EXAMPLES

Example 1 24

Example 2 25

Example 3 25

Example 4 26

Example 5 30

Example 6 30

Example 7 30

Example 8 30

Example 9 31

Example 10 35

Example 11 35

Example 12 35

Example 13 35

Example 14 43

Example 15 44

Example 16 44

Draft new Recommendation ITU-T J.380.2 (ex J.dpi-asi-2)

Digital Program Insertion–Advertising Systems Interfaces: Part 2 - Core data elements

Introduction

This document defines the following:

- The extensible J.380 message structure.
- A set of required messages that shall be implemented by all J.380 compliant logical services.
- A common set of XML elements (i.e., core elements) which may appear in any J.380 message or within any J.380 defined element.
- Common XML attributes used by the core data elements which may also be used by any other element.

All externally defined J.380 messages shall contain the appropriate common message attributes as defined herein.

This specification also defines requirements for the J.380 message transport without specifying the actual transport implementation. [b-ITU-T J.380.7] is expected to define the normative transport specification.

Document Organization

This document provides a structured, logical approach to the core aspects of the J.380 sub-series. Each subsequent Clause focuses on a particular specification aspect and the document's presentation order provides a top-down, conceptual introduction necessary to implement J.380.

Clause 5 explains the document's notational conventions. Clause 6 defines the XML namespace usage and the applicable XML semantics. Clause 7 introduces the message concepts and the core XML attributes which are the foundation of every J.380 message. This Clause also defines the message exchange characteristics common to all J.380 message interactions.

Clause 7 and its subsequent Clauses often may reference XML attributes and elements defined later in the document. The reader may occasionally need to reference other Clauses of this document in order to find the complete explanation of a syntactic component.

Clause 8 defines messages that shall be supported by every J.380 logical service. Clause 9 presents the core XML attributes followed by the principal specification XML elements with each group presented in alphabetical order. Annex A concludes the document with the normative status code values, their meanings, and their message usage applicability.

1 Scope

ITU-T J.380.2 describes the Digital Program Insertion Advertising Systems Interfaces' core messaging and data types using extensible markup language (XML), XML Namespaces, and XML Schema. Core messaging includes the extensible message schemas, the common J.380 message attributes, and the required J.380 messages. The core data types are XML attributes and XML elements which may be used in any J.380.x message element or within a J.380.x element definition.

2 References

The following ITU-T Recommendations and other references contain provisions, which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published.

The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

- [XML] W3C Recommendation, "*Extensible Markup Language (XML) 1.0 (Fourth Edition)*", Tim Bray, et al, 16 August 2006, <http://www.w3.org/TR/2006/REC-xml-20060816/>.
- [XMLNamespaces] W3C Recommendation, "Namespaces In XML (Second Edition)", Tim Bray, et al, 16 August 2006, <http://www.w3.org/TR/2006/REC-xml-names-20060816/>.
- [XMLSchemaP1] W3C Recommendation, "XML Schema Part 1: Structures (Second Edition)", H. Thompson, et al, 28 October 2004, <http://www.w3.org/TR/xmlschema-1/>.
- [XMLSchemaP2] W3C Recommendation, "XML Schema Part 2: Datatypes (Second Edition)", P. Biron, et al, 28 October 2004, <http://www.w3.org/TR/xmlschema-2/>.
- [XMLInfoSet] W3C Recommendation, "XML InfoSet (Second Edition)", John Cowan, Richard Tobin, 4 February 2004, <http://www.w3.org/TR/2004/REC-xml-infoset-20040204/>.
- [SCTE35] ANSI/SCTE 35 2007—Digital Program Insertion Cueing Message for Cable.
- [SCTE118-3] ANSI/SCTE 118-3 2006—Program Specific Ad Insertion—Traffic System to Ad Insertion System File Format Specification.
- [RFC3986] RFC 3986, "Uniform Resource Identifier (URI): Generic Syntax", January 2005, www.ietf.org/rfc/rfc3986.txt.
- [IEEE802] IEEE Std 802-2001 - IEEE Standard for Local and Metropolitan Area Networks: Overview and Architecture-IEEE Computer Society, approved 6 December 2001, <http://standards.ieee.org/getieee802/802.html>
- [SCTE130-2/core schema] Part 2 core schema: <http://www.scte.org/schemas/130-2/2008a/core>

3 Definitions

3.1 Terms defined elsewhere

None.

3.2 Terms defined in this Recommendation

3.2.1 content: The video, audio, and data streams taken together as a single identifiable unit. Content may refer to the original entertainment (programming) content, an ad spot, an interactive or enhanced application asset, or any other similar asset.

3.2.2 default endpoint: The endpoint where messages are delivered in the absence of a message specific endpoint designation.

3.2.3 endpoint: An address, a Uniform Resource Identifier (URI), or a specific location where a logical service function or functions shall be found and consumed via message exchange.

3.2.4 event: A general term indicating something has happened or occurred.

3.2.5 logical service: A well-defined, self-contained set of functions accessible via one or more endpoints. The logical service has some type of underlying computer system that supports message communication.

3.2.6 message: The unit of communication between two logical services.

3.2.7 program: A time-bounded collection of video, audio, and data streams.

3.2.8 registration-established service channel: A service channel duration commencing with a successful registration and continuing until termination through deregistration.

3.2.9 scope of uniqueness: Uniqueness is context relative and for this specification's purpose shall be defined by one of following: global, service channel or element.

3.2.10 global uniqueness: Global or universally unique and at no other time shall the item be compromised, reused, or otherwise taken to have more than one meaning. Enforcement of uniqueness as well as creation of globally unique identifiers is outside the scope of this specification and RFC 4122 is recommended. See [b-IETF RFC 4122] for additional information.

3.2.11 service channel uniqueness: Uniqueness scoped by the @identity attribute and the service channel and at no other time shall the item be compromised, reused, or otherwise taken to have more than one meaning. XML messages shall be service channel unique and a message shall not be compromised or reused for the duration of the service channel. Service channel uniqueness is relative only to the endpoints where the message exchange is occurring and within the identity domain of the two endpoints involved in the exchange. Enforcement of uniqueness as well as creation of identity unique identifiers is outside the scope of this specification and IETF RFC 4122 is recommended. See [b-IETF RFC 4122] for additional information.

3.2.12 element uniqueness: Generally, XML elements shall be unique according to existing XML compliance where the element's distinctiveness is unambiguous and unique relative to its immediate spatial relationship to other elements.

3.2.13 service channel: A message communication path between two logical services.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

ADS	Ad Decision Service.
ADM	Ad Management Service
CIS	Content Information Service
HA	High Availability
POIS	Placement Opportunity Information Service
SIS	Subscriber Information Service
UPID	Unique Program Identifier—See [SCTE35].
URI	Uniform Resource Identifier—See [RFC3986].
UUID	Universally Unique Identifier—See [IETF RFC 4122].

5 Conventions

5.1 Normative XML Schema

Descriptions of messages, elements, and attributes are normative and, when combined with the normative XML schema document (provided separately), comprise the full normative specification. Non-normative schema illustrations and instance examples are included herein for informational purposes only. Any real or implied usage, semantics, or structure indicated by the schema illustrations and examples shall not be considered part of the specification.

No messages representing the interfaces defined in the schema are considered conformant unless they are valid according to the schema document. Additionally, other J.380 standard normative parts may impose additional rules or restrictions that shall be adhered to in order for the messages to be considered conformant to those parts.

In the case where the written normative specification (for example, this document) and the normative schema document (i.e., the separately provided XML 'xsd' file) conflict, the written normative specification shall take precedence over the XML schema document.

The inclusion of a normative XML schema document does not require or imply the specific use of the schema nor a requirement that a message be validated.

5.2 Document Conventions

XML elements may be listed in the format "element" or "prefix:element" without the double-quotes where "prefix" denotes the appropriate namespace defined in Clause 6 and "element" denotes the name of the element. A slash "/" denotes that a child XML element or XML attribute follows.

XML attributes are listed in the format "@attribute" or "@prefix:attribute" without the double-quotes where "prefix" denotes the appropriate namespace defined in Clause 6. An "@" character without the double-quotes denotes an XML attribute as opposed to an XML element, and "attribute" denotes the name of the attribute.

When describing concrete XML schemas, an element wildcard (<xsd:any/>) is represented by the notation ##any. The ##any element refers to all namespaces including the namespace defined by this document. Any J.380 or alternative namespace element may be included.

The constrained wildcard element <xsd:any namespace="##other"/> is represented by ##other. The ##other element refers to all other namespaces except the namespace defined by this document. Any ITU-T J.380.2 namespace element may be included.

An attribute wildcard (<xsd:anyAttribute/>) is represented by the notation @##any (or 'any ##any' within the schema diagrams). The ##any attribute indicates that any J.380 defined or alternatively defined attribute may be included.

Figure 1 explains the schema illustration technique used throughout the J.380 specifications. Symbols and their meanings are explained within Figure 1.

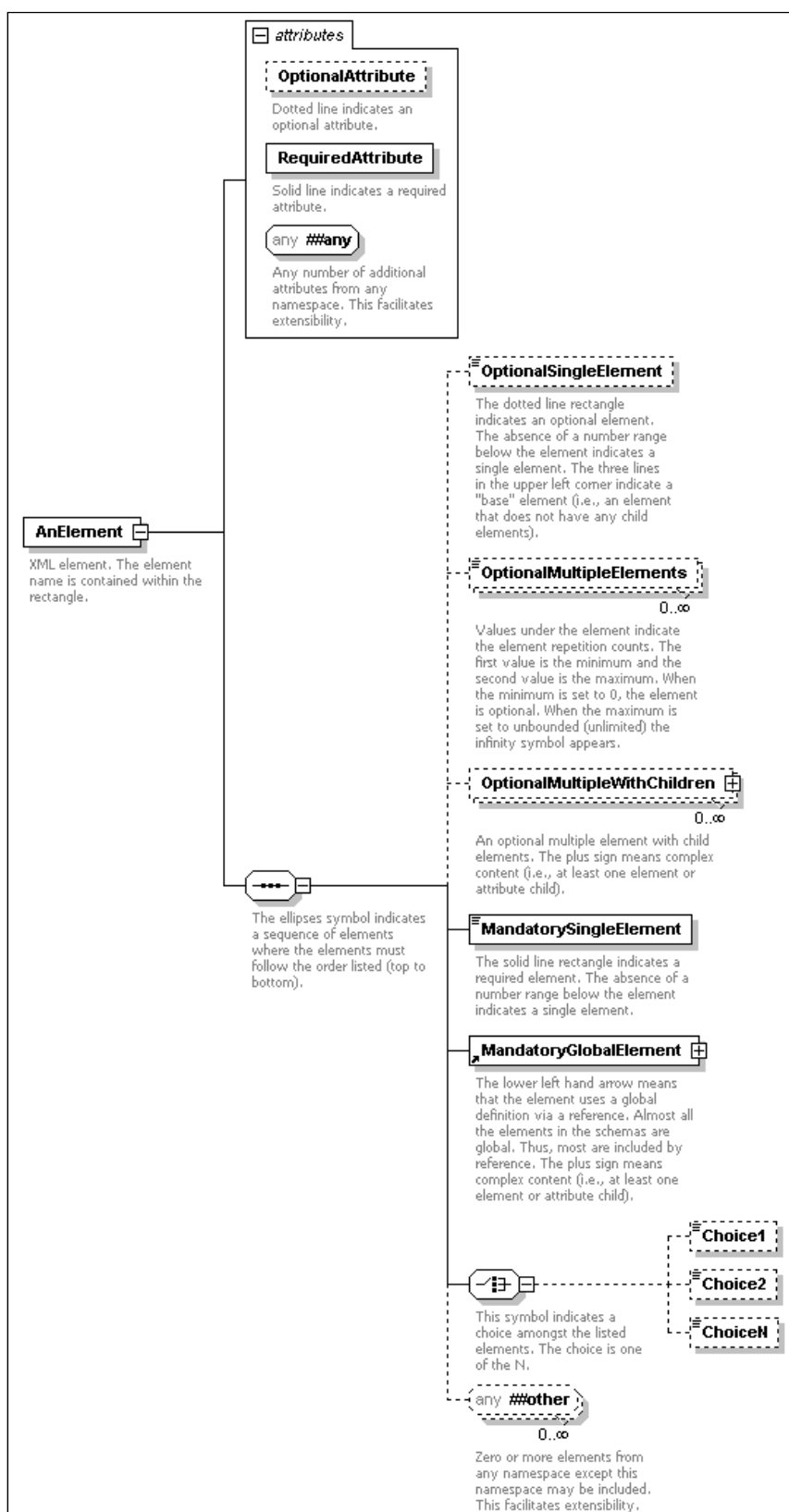


Figure 1: Schema Illustration Explanation

6 XML Namespaces

This document uses the prefix 'core' for the interface associated with this specification's XML namespace URI. This URI shall be used by all implementations applying this specification. Table 1 lists the prefix, the corresponding namespace, and a description of the defining specification.

Table 1: XML Namespace Declarations

Prefix	Namespace	Description
core	http://www.scte.org/schemas/130-2/2008a/core	ITU-T J.380.2 (i.e., this document)
xsd	http://www.w3.org/2001/XMLSchema	XML foundation. See [XMLSchemaP1].

The namespace is formatted per the SCTE recommendation and includes the J.380 version through the inclusion of a date which may minimally be a specification year identifier.

6.1 Unknown/Unrecognized/Unsupported XML Elements and Attributes

Generally, unknown, unrecognized or unsupported XML elements and attributes contained within J.380 messages and elements should be ignored during message processing. Specifically, these are elements or attributes which the implementation does not understand or expect. The logical service should generate an appropriate response message for any element or attribute in this categorization. For example, the return message might contain an informational Note element. Ideally, validation should be utilized to identify and report prohibited constructs through the appropriate mechanisms. Elements and attributes which are prohibited by a namespace should generate an appropriate error response message.

6.2 Element Order

Element order is constrained by the schemas and must be preserved throughout processing of the XML document. In particular, the order of elements affects the end result of the processing. Consequently, an implementation failing to preserve the order may cause incorrect processing results. Subsequently, the process of producing an abstract XML Information Set (InfoSet) from a concrete XML document, e.g., by parsing it, shall always result in the same abstract InfoSet, with the same element order per XML InfoSet. (See [XMLInfoSet] for additional information.) Any intermediary processing may enhance the XML document but it shall not alter the abstract InfoSet element order (i.e., the XML elements comprising the document shall stay in document order).

6.3 Language Identification

In document processing, it is often useful to identify the natural or formal language in which the content is written. A special attribute named xml:lang may be inserted in documents to specify the language used in the contents and attribute values of any element in an XML document. See [XML] for the allowed values. Typically, the xml:lang attribute is utilized with an xsd:string type.

7 J.380 Message Basics

A logical service is a self-contained set of functions accessible via one or more endpoints where each function is accessed via an XML document exchange. An endpoint identifies where a function may be found and consumed and multiple functions may be found at the same endpoint. Endpoint discovery is outside the scope of this specification and may be addressed by [b-ITU-T J.380.7]. A message is the unit of communication between two logical services. All J.380 identified logical services exchange information based on a common XML schema defined herein.

The XML document root element for all communication exchanges is a specific form of request, response, notification or acknowledgement element referred to as a message. Communication involves a two-way message exchange of a specific paired message set as illustrated in Figure 2. The possible exchanged message sets shall be either a request/response message pair or a notification/acknowledgement message pair. All other message combinations shall not be allowed. When a request or notification message is received by a J.380 logical service, then the appropriately paired response or acknowledgement message shall be returned.

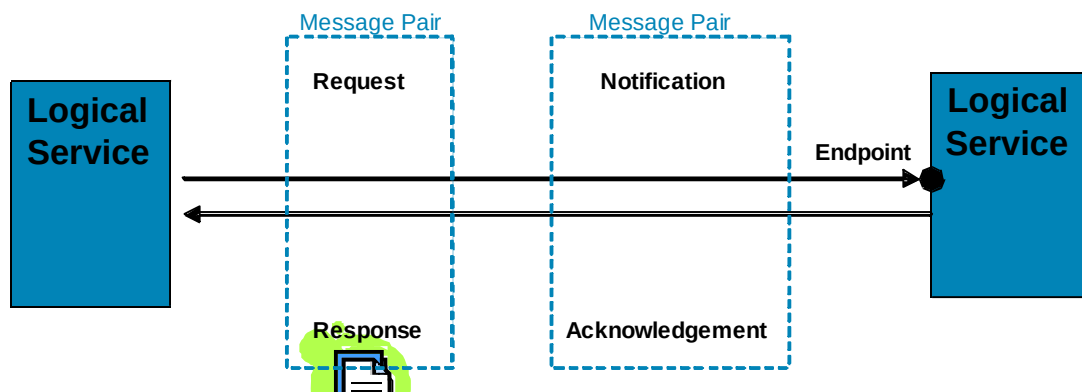


Figure 2: Basic Communication Overview

Figure 3 illustrates a single logical service may be simultaneously communicating with multiple other logical services.

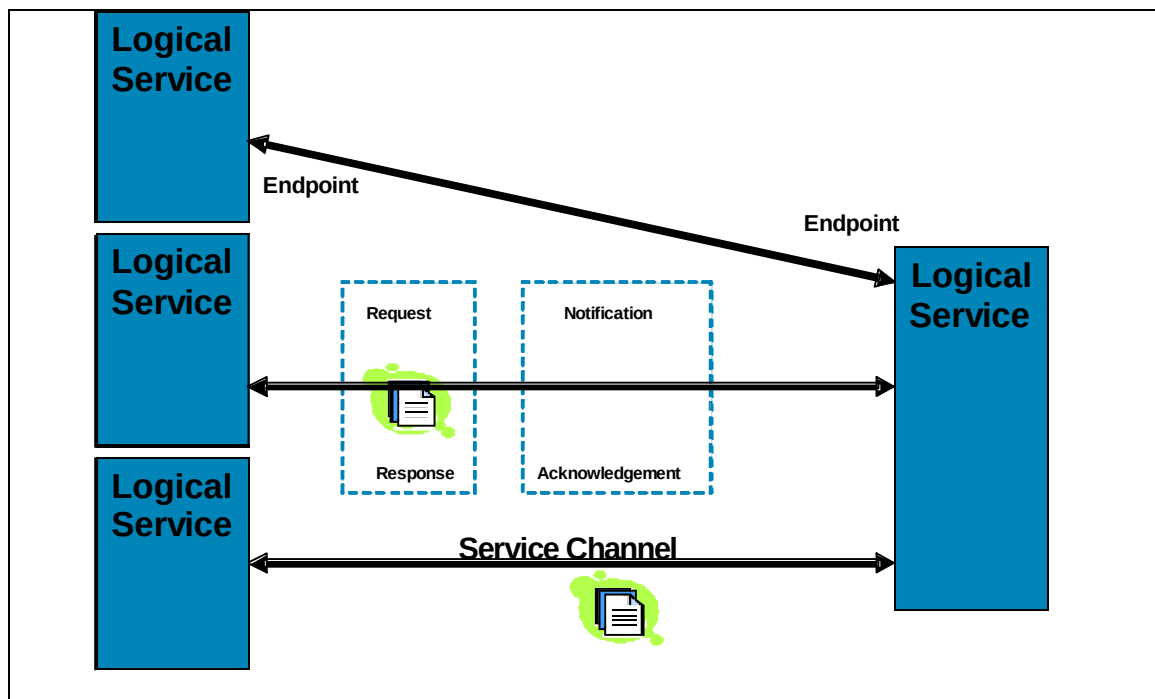


Figure 3: Service Channel Illustrated

A service channel is the message communication path between two logical services which exists for one of the following durations: the duration marked by the first message of a service registration and ending with the last message of service deregistration or the single request/response message transaction where the messages are permitted outside the registration-established service channel scope.

Once successfully constructed, the registration-established service channel shall remain active until terminated via deregistration. Normal termination of a service channel shall occur via a deregistration message exchange. Abnormal termination (e.g., a system failure, etc.) may require alternative deregistration mechanisms outside the scope of this specification. (Note: ITU-T J.380.7 may provide additional details. See [b-ITU-T J.380.7] for supplementary information.)

Since a J.380 message element is the XML document root element, all J.390 messages shall share a common base syntactic structure as defined in the following Clauses.

7.1 Common Schema for All Messages

All J.380 message XML elements shall share a common attribute definition though individual messages may contain additional attributes. Refer to the individual message schemas for specific details. Figure 4 illustrates the J.380 message common attribute schema.

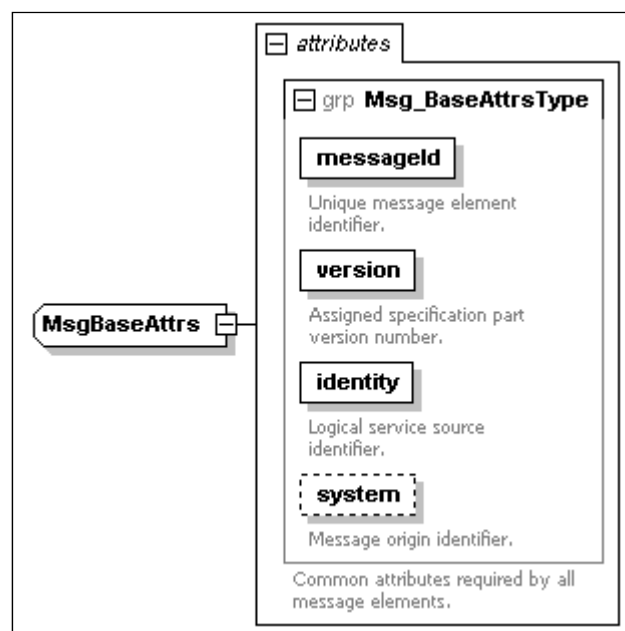


Figure 4: J.380 Message Common Attribute Schema

A single J.380 message shall be identifiable via its @messageId attribute. The individual J.380 message specification version shall be identified via the @version attribute. The logical service message origin shall be specified using the @identity attribute and a further refinement may be provided via the @system attribute.

7.1.1 Semantic Definitions for the J.380 Message Common Attributes

@messageId [Required, messageIdAttrType]—The message identifier. Every J.380 message instance shall have a service channel unique value. The @messageId should be a Universally Unique Identifier as defined by RFC 4122 (see [b-IETF RFC 4122]). See clause 9.2.4 for additional information.

@version [Required, versionAttrType]—The J.380 message specification version number. The value shall be specified by the individual J.380 specification parts. For all messages defined herein, the value shall be "1.1". See clause 9.2.9 for the attribute's type definition.

@identity [Required, identityAttrType]—The origin logical service identifier. A globally unique identifier associated with the logical service. See clause 9.2.1 for additional information.

@system [Optional, systemAttrType]—The message source identifier. See clause 9.2.8 for additional information.

7.2 Request and Notification Messages Base Schema

Figure 5 and Figure 6 illustrate the J.380 request and notification messages base schema.

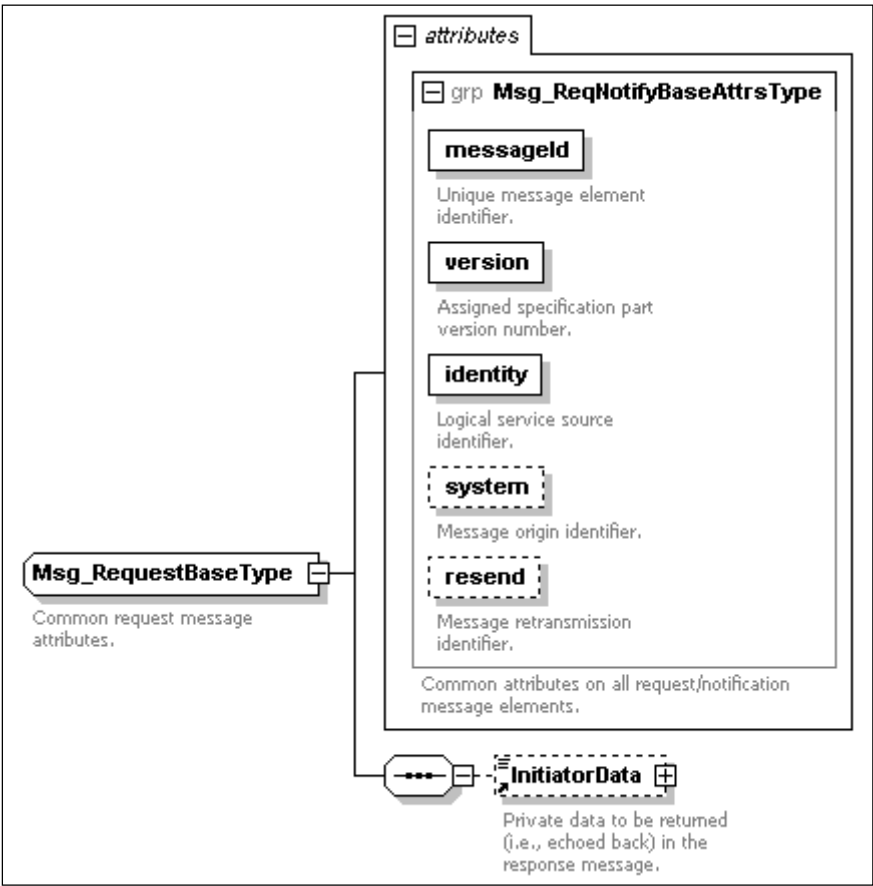


Figure 5: Request Message Base Schema

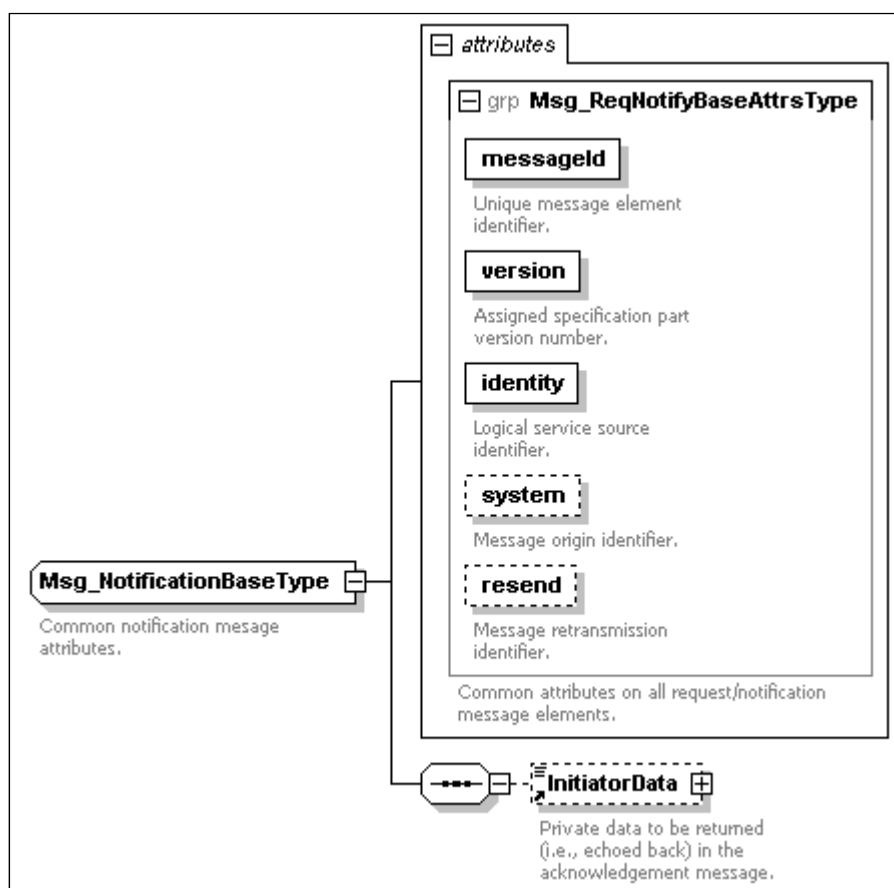


Figure 6: Notification Message Base Schema

7.2.1 Semantic Definitions for the Request and Notification Messages Base Schema

@messageId [Required, messageIdAttrType]—The message identifier. See clause 7.1.1 for additional information.

@version [Required, versionAttrType]—The message specification version. See clause 7.1.1 for additional information.

@identity [Required, identityAttrType]—The origin logical service identifier. See clause 7.1.1 for additional information.

@system [Optional, systemAttrType]—The message source identifier. See clause 7.1.1 for additional information.

@resend [Optional, resendAttrType]—The message identifier of a previously sent original message (i.e., the @messageId of a previous message) for which this message is a retransmit. See clause 7.4.7 and Clause 9.2.7 for additional information.

InitiatorData [Optional]—The InitiatorData element contains implementation specific private data. If the element is present in the request or notification message, the corresponding response or acknowledgement message shall include an exact copy of the original element. Figure 9 illustrates the linkage. If the InitiatorData element is omitted from the original request or notification message, the element shall not be present in the corresponding response or acknowledgement message. See clause 9.14 for additional information.

7.3 Response and Acknowledgement Messages Base Schema

Figure 7 and Figure 8 illustrate the J.380 response and acknowledgement messages base schema.

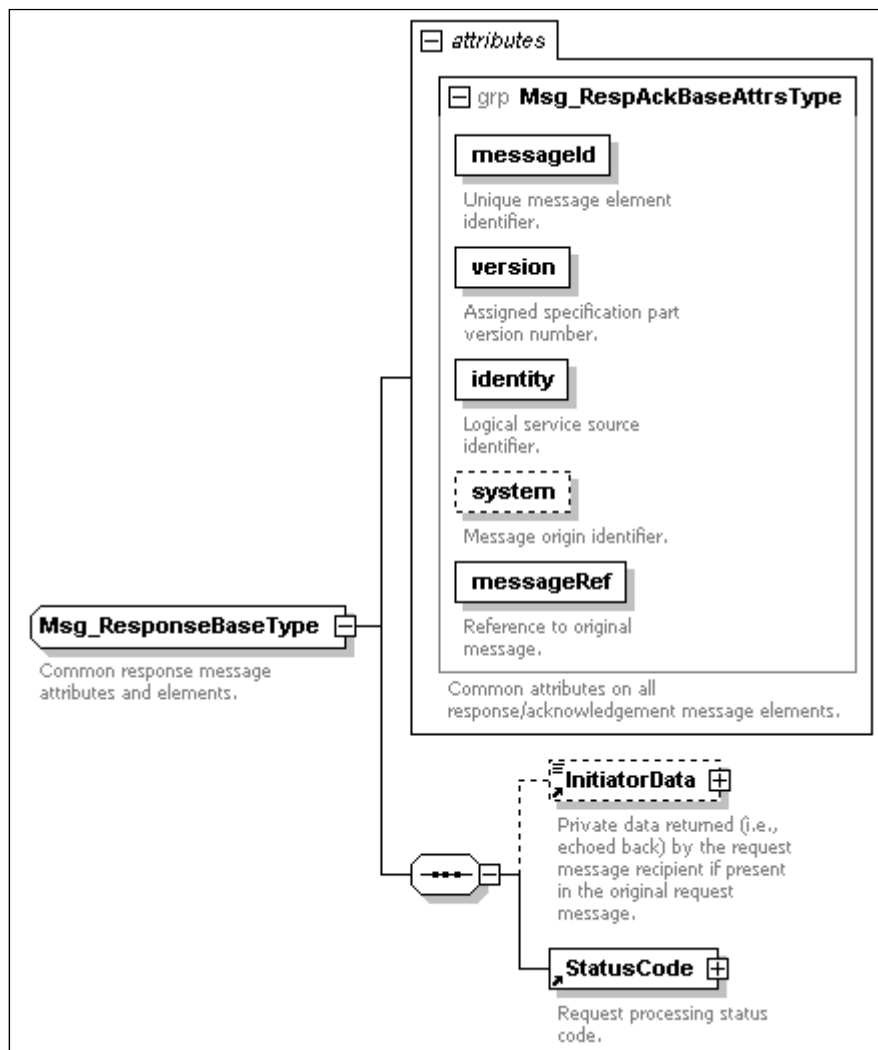


Figure 7: Response Message Base Schema

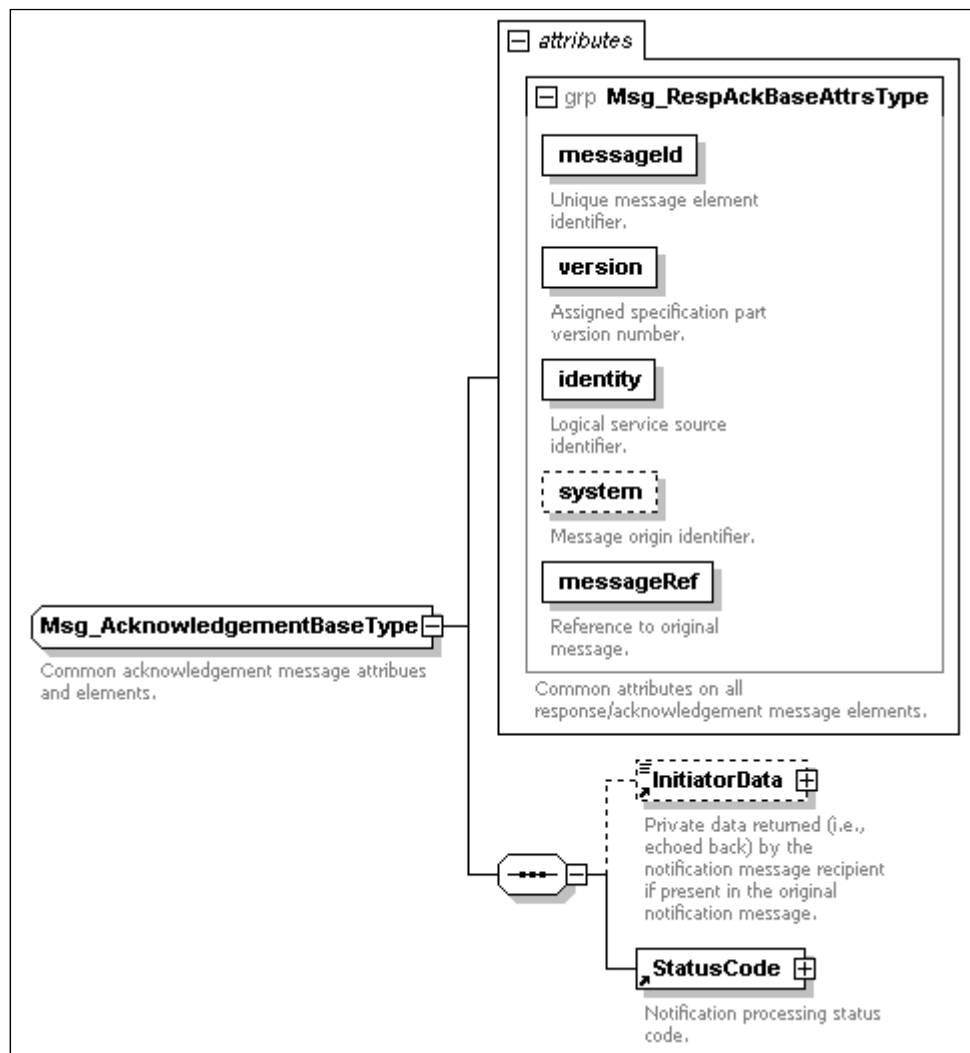


Figure 8: Acknowledgement Message Base Schema

7.3.1 Semantic Definitions for the Response and Acknowledgement Messages Base Schema

@messageId [Required, messageIdAttrType]—The message identifier. See clause 7.1.1 for additional information.

@version [Required, versionAttrType]—The message specification version. See clause 7.1.1 for additional information.

@identity [Required, identityAttrType]—The origin logical service identifier. See clause 7.1.1 for additional information.

@system [Optional, systemAttrType]—The message source identifier. See clause 7.1.1 for additional information.

@messageRef [Required, messageRefAttrType]—An attribute in all response and acknowledgement messages which references the paired request or notification message. The attribute's value is the request or notification message's @messageId attribute value from the paired J.380 message. See clause 9.2.5 for additional information. Figure 9 illustrates the linkage.