Errata Notice on Schema Locations

March 17, 2022

This standard makes use of namespace locations with a form of http://www.scte.org/schemas/xyz/*, where “xyz” is the location of the specific schema being referenced. Due to limitations on the current SCTE website, those specific locations are not available.

To find such schemas:

1. Go to the standards download page at https://www.scte.org/standards/library/catalog/
2. Search for the standard number (xyz in the above example)
3. Select the document from the table
4. Scroll to the “Supporting Documentation” section of the document webpage.

The schema will be listed within the Supporting Documentation section.

This notice will be removed once the exact namespace values are functional.

SCTE · ISBE

STANDARDS

Digital Video Subcommittee

AMERICAN NATIONAL STANDARD

ANSI/SCTE 130-3 2020

Digital Program Insertion-Advertising Systems Interfaces

Part 3

Ad Management Service (ADM) Interface

NOTICE

The Society of Cable Telecommunications Engineers (SCTE) / International Society of Broadband Experts (ISBE) Standards and Operational Practices (hereafter called "documents") are intended to serve the public interest by providing specifications, test methods and procedures that promote uniformity of product, interchangeability, best practices and ultimately the long-term reliability of broadband communications facilities. These documents shall not in any way preclude any member or non-member of SCTE•ISBE from manufacturing or selling products not conforming to such documents, nor shall the existence of such standards preclude their voluntary use by those other than SCTE•ISBE members.

SCTE•ISBE assumes no obligations or liability whatsoever to any party who may adopt the documents. Such adopting party assumes all risks associated with adoption of these documents, and accepts full responsibility for any damage and/or claims arising from the adoption of such documents.

Attention is called to the possibility that implementation of this document may require the use of subject matter covered by patent rights. By publication of this document, no position is taken with respect to the existence or validity of any patent rights in connection therewith. SCTE•ISBE shall not be responsible for identifying patents for which a license may be required or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

Patent holders who believe that they hold patents which are essential to the implementation of this document have been requested to provide information about those patents and any related licensing terms and conditions. Any such declarations made before or after publication of this document are available on the SCTE•ISBE web site at http://www.scte.org.

140 Philips Road
Exton, PA 19341

TABLE OF CONTENTS

List of Figures

Title **Page Number**

List of Tables

Title **Page Number**

Digital Program Insertion–Advertising Systems Interfaces Part 3–Ad Management Service (ADM) Interface

1.0 SCOPE

This document in conjunction with the SCTE 130 Part 3 Extensible Markup Language (XML) schema document (i.e., the XSD document) defines the XML messages expressing placement opportunities, placement decisions, and placement related event data typically exchanged between an Ad Management Service (ADM) and an Ad Decision Service (ADS). Additionally, this document and the accompanying schema document describe the auxiliary XML messages, elements, and attributes supporting the primary message exchanges.

2.0 REFERENCES

2.1 Normative references

The following documents contain provisions, which, through reference in this text, constitute provisions of the standard. At the time of Subcommittee approval, the editions indicated were valid. All standards are subject to revision; and while parties to any agreement based on this standard are encouraged to investigate the possibility of applying the most recent editions of the documents listed below, they are reminded that newer editions of those documents *may* not be compatible with the referenced version.

[SCTE130-2]	SCTE 130-2 2020: Digital Program Insertion—Advertising Systems Interfaces Part 2 – Core Data Elements
[SCTE130-7]	SCTE 130-7 2015: Digital Program Insertion – Advertising Systems Interfaces Part 7 – Message Transport.
[SCTE130-8]	SCTE 130-8 2020: Digital Program Insertion —Advertising Systems Interfaces Part 8—General Information Service
[W3C-XSD]	XML Schema Part 1: Structures Second Edition

All normative references found in SCTE 130 Part 2 are normatively included and apply to this document. See [SCTE130-2] for additional information.

2.2 Informative References

The following documents *may* provide valuable information to the reader but are not required when complying with this standard.

[SCTE236]	ANSI/SCTE 236—Content Metadata.
[SCTE130-1]	SCTE 130-1: Digital Program Insertion—Advertising Systems Interfaces Part 1 – Overview.
[SCTE130-4]	SCTE 130-4: Digital Program Insertion—Advertising Systems Interfaces Part 4 – Content Information Service.
[SCTE130-5]	SCTE 130-5: Digital Program Insertion—Advertising Systems Interfaces Part 5 –Placement Opportunity Information Service.
[SCTE130-6]	SCTE 130-6: Digital Program Insertion—Advertising Systems Interfaces Part 6 – Subscriber Information Service.
[SCTE35]	ANSI/SCTE 35—Digital Program Insertion Cueing Message for Cable.
[RFC2326]	RFC 2326, “Real Time Streaming Protocol (RTSP)”, April 1998, www.ietf.org/rfc/rfc2326.txt.
[SCTE118-2]	ANSI/SCTE 118-2: Program-Specific Ad Insertion-Content Provider to Traffic Communication Applications Data Model

3.0 COMPLIANCE NOTATION

shall	This word or the adjective "***required***" means that the item is an absolute requirement of this document.
shall not	This phrase means that the item is an absolute prohibition of this document.
forbidden	This word means the value specified ***shall*** never be used.
should	This word or the adjective "*recommended*" means that there may exist valid reasons in particular circumstances to ignore this item, but the full implications should be understood and the case carefully weighted before choosing a different course.
should not	This phrase means that there may exist valid reasons in particular circumstances when the listed behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.
may	This word or the adjective "*optional*" means that this item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because it enhances the product, for example; another vendor may omit the same item.
deprecated	Use is permissible for legacy purposes only. Deprecated features may be removed from future versions of this document. Implementations should avoid use of deprecated features.

4.0 DEFINITION OF TERMS

Throughout this standard the terms below have specific meanings. Because some of the terms are defined in other SCTE documents having very specific technical meanings, the reader is referred to the original source for their definition. For terms defined by this standard, brief definitions are given below.

All SCTE 130 Part 2, SCTE 130 Part 7 and SCTE 130 Part 8 definitions are normatively included herein. See [[SCTE130-2], [SCTE130-7] and [SCTE130-8] for additional information.

This document additionally defines the following terms:

ADM interface: The ADM to ADS message communication link defined by SCTE 130 Part 3 (i.e., this document and its companion XML schema document).

interstitial: A placement opportunity occurring between entertainment assets (e.g., placement opportunities that exist between music videos in a playlist). An interstitial placement opportunity is not associated with the entertainment assets that *may* be part of a session containing interstitial placement opportunities. An interstitial placement opportunity does not preclude the entertainment asset preceding the interstitial placement opportunity from having post-roll placement opportunities or the entertainment asset following from having pre-roll placement opportunities.

mid-roll: A placement opportunity occurring within an entertainment asset.

placement: The decision resulting from a placement opportunity which *may* include a content binding and a set of constraints.

placement management: The aggregated operational sequence of placement messaging and placement update messaging.

placement operations: Placement management aggregated with placement status messaging.

placement messaging: The PlacementRequest / PlacementResponse message exchange sequence.

placement status messaging: The PlacementStatusNotification / PlacementStatusAcknowledgement message exchange sequence.

placement update messaging: The PlacementUpdateNotification / PlacementUpdateAcknowledgement message exchange sequence.

placement opportunity: A potentially constrained location relative to digital content where ad insertion or content alterations *may* occur. The alterations *may* include insertions, replacements, or deletions of content in whole or in part. These locations, which contain the opportunity for content insertion, have traditionally been referred to as avails [SCTE35] for linear video content; however, placement opportunity refers to address and time locations where content *may* be placed, regardless of platform.

placement service: The distinctive information (i.e., characteristic set) identifying an entertainment content flow where advanced advertising techniques *may* apply.

post-roll: A placement opportunity following the play out of an entertainment asset.

pre-roll: A placement opportunity preceding an entertainment asset. (Note: Definition differs from SCTE 35. See [SCTE35].)

registration-established service channel: A service channel duration commencing with a successful placement service registration and continuing until either participating logical service completes terminating an active service channel. (Note: This document refines the SCTE 130 Part 2 definition by qualifying the registration as a placement service registration. See [SCTE130-2] for the base definitions.)

5.0 ABBREVIATIONS

All SCTE 130 Part 2, SCTE 130 Part 7 and SCTE 130 Part 8 abbreviations are normatively included herein. See [SCTE130-2], [SCTE130-7] and [SCTE130-8] for additional information.

This document additionally uses the following abbreviations:

NPT: Normal Play Time

PODM: Placement Opportunity Data Model

SCTE: Society of Cable Telecommunications Engineers

UTC: Coordinated Universal Time

VOD: Video on Demand

6.0 INTRODUCTION

The Ad Management Service (ADM) and the Ad Decision Service (ADS) are core elements in a standardized SCTE 130 advanced advertising system. Both the ADM and the ADS ***shall*** be SCTE 130 compliant logical services implementing all messages and message semantics as defined in SCTE 130 Part 2 and this document. See [SCTE130-2] for additional Part 2 information.

The ADM originates messages articulating ad insertion opportunities (i.e., placement opportunities) with the primary communication consumer being the ADS. The ADS determines how advertising content is combined with non-advertising (entertainment) content assets (i.e., placement decisions). The ADM completes the opportunity/decision lifecycle via reporting activity events (i.e., placement status). The ADM/ADS communication interface is defined herein and it is referred to as the ADM interface throughout the remainder of this specification. The ADM interface facilitates both preconfigured ad decisions as well as real-time placement selection.

Figure 1 illustrates a very basic example advertising system utilizing only an ADM and an ADS. Figure 1 represents one possible configuration and many other combinations beyond the illustration are possible.

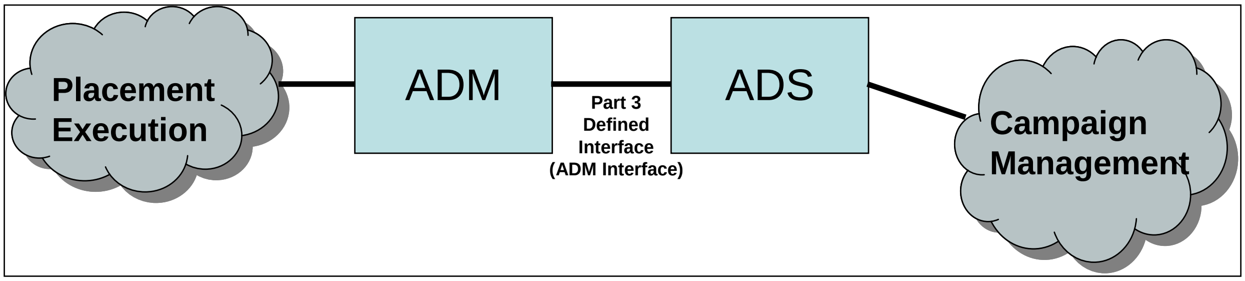


Figure 1: A Very Basic SCTE 130 Advertising System

In Figure 1, the placement system, represented by the Placement Execution cloud, is responsible for opportunity determination and placement execution (e.g., content insertion/splicing). Communication between the ADM and the Placement Execution cloud is outside the scope of this specification. The ADM, via the ADM interface, communicates placement opportunities and placement status to the ADS and the ADS returns placement decisions as a response to placement opportunities. The Campaign Management cloud symbolizes the system providing the information necessary for the ADS to render ad decisions. Communication between the ADS and the Campaign Management cloud is outside the scope of this specification. The internal implementation and operation of both the ADM and the ADS are outside the scope of this specification.

The ADM and the ADS *may* be clients of any SCTE 130 defined logical service as illustrated in Figure 2.

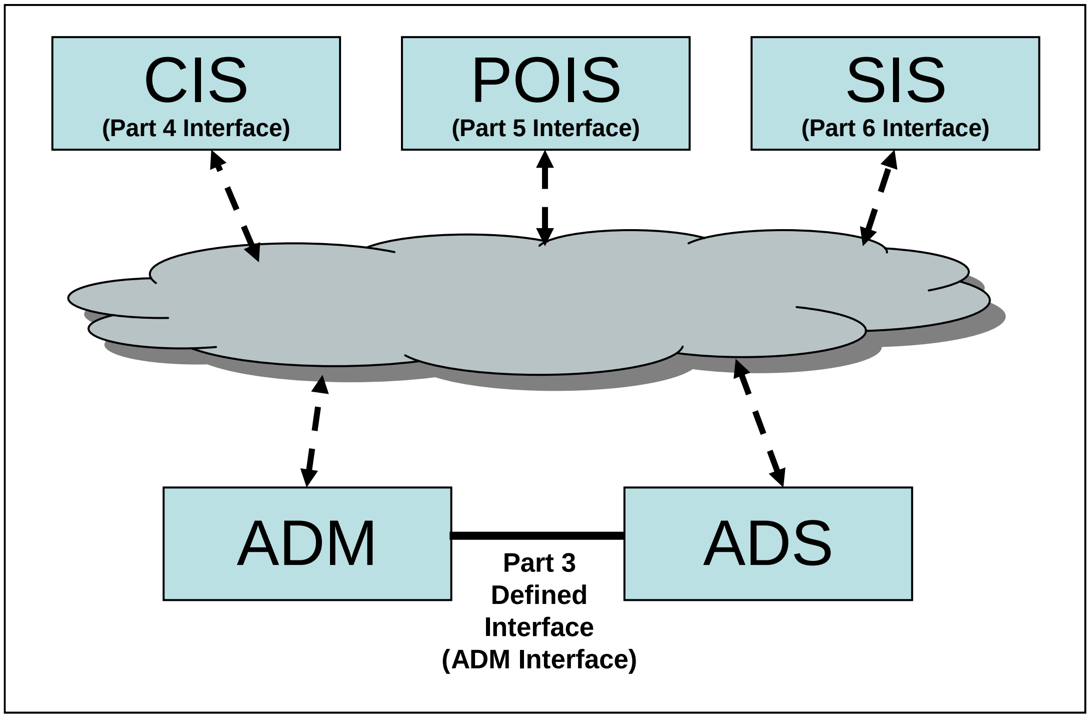


Figure 2: SCTE 130 Logical Service Relationships

The Content Information Service (CIS) provides content asset metadata and availability information for both advertising and non-advertising content. See [SCTE130-4] for additional information. The Placement Opportunity Information Service (POIS) *may* supply placement opportunity descriptions. See [SCTE130-5] for additional information. The Subscriber Information Service (SIS) *may* provide relevant subscriber information such as advertising focus refinement parameters. See [SCTE130-6] for additional information. With the exception of the ADM interface (i.e., the ADM/ADS communication path), all Figure 2 elements are outside the scope of this specification.

An ADM *may* support multiple, simultaneously active ADS as each ADS *may* provide different placement decision services. An ADS *may* provide the same decision services or different services to multiple ADM simultaneously. Figure 3 illustrates these possible high-level interconnects.

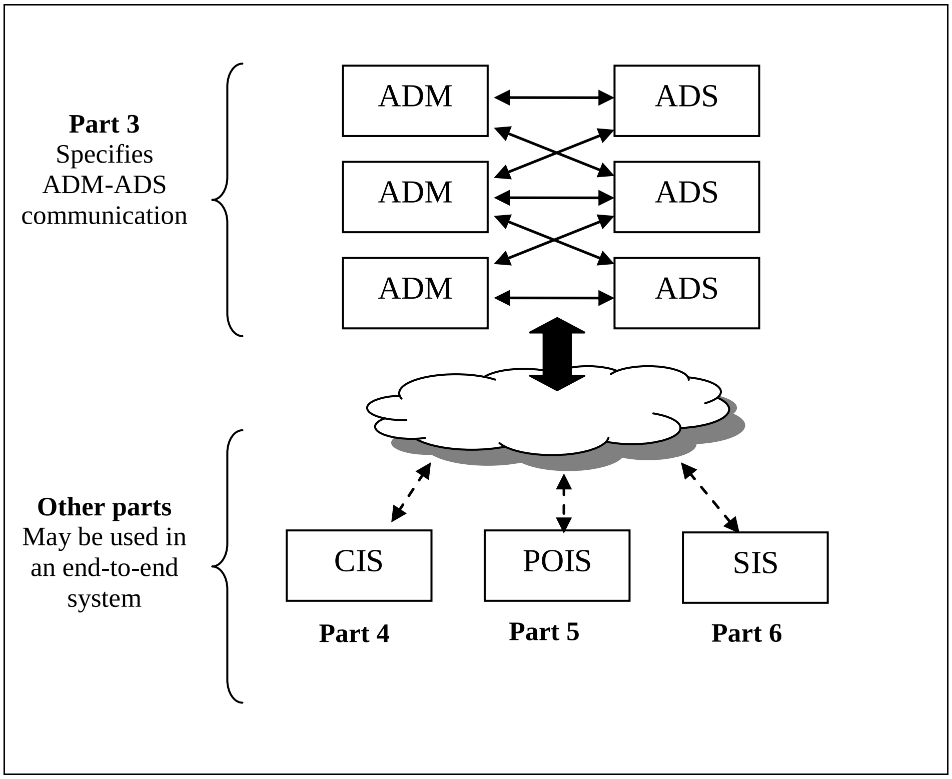


Figure 3: ADM/ADS Interconnects

Figure 4 illustrates a more sophisticated example SCTE 130 advertising system leveraging multiple SCTE 130 logical services. Figure 4 represents one possible configuration and many other combinations beyond the illustration are possible.

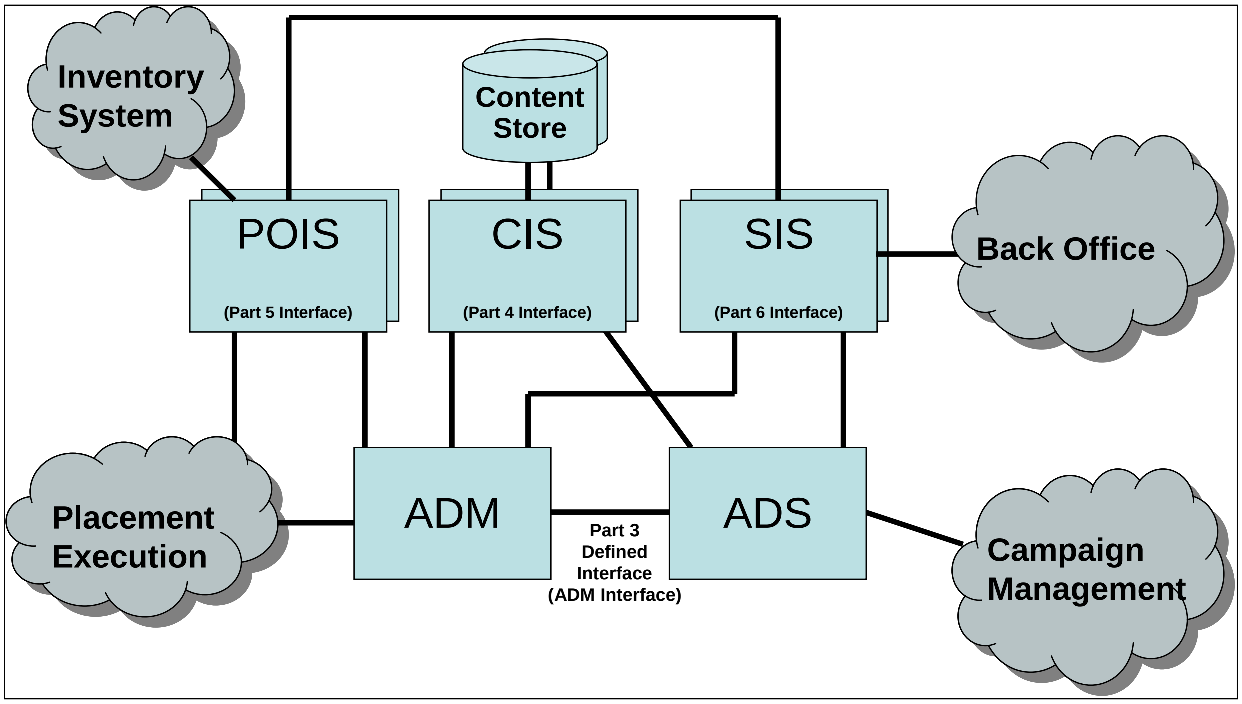


Figure 4: SCTE 130 Advertising System Example

In this example, the Placement Execution cloud and/or the ADM block *may* consult the POIS for placement opportunity information with the POIS fronting the Inventory System cloud. Additionally, the ADS *may* communicate with the POIS to acquire the same placement opportunity information for its usage. The Inventory System cloud symbolizes the advertising opportunity scheduling systems. Either or both the ADM and ADS *may* communicate with the CIS to obtain content asset metadata and ad asset availability information. The ADM, ADS, and/or the POIS *may* communicate with the SIS in order to obtain relevant subscriber information such as advertising focus refinement parameters and placement opportunity enablement. This SIS fronts the Back Office cloud which represents the Operation Support Systems (OSS) and Business Support Systems (BSS). With the exception of the ADM interface, all Figure 4 elements are outside the scope of this specification.

6.1 Document Organization

This document provides a structured, logical approach to the SCTE 130 ADM interface. Each subsequent section focuses on a particular specification aspect and this document’s presentation order provides a top-down, conceptual introductory understanding necessary to implement the ADM interface.

Section 7 explains this document’s notational conventions and Section 8 identifies the processing conventions. Section 9 defines the XML namespace usage and the applicable XML semantics. Section 10 introduces the ADM interface messaging concepts.

Section 11 defines the ADM interface logical service management messages and Section 12 defines the placement operations messages. Section 11 and subsequent sections often *may* reference XML attributes and elements defined later in this document. The reader *may* occasionally need to reference other document sections or SCTE 130 Part 2 [SCTE130-2] in order to find the complete explanation of a syntactic component.

Section 13 presents the ADM interface's XML attributes and complex types followed by Section 14 which defines the specification's individual XML elements. Attributes, complex types, and elements in Section 13 and Section 14 are presented in alphabetical order. Appendix A provides the normative status code values, their meanings, and their message usage applicability followed by Appendix B which is an informative example.

6.2 Placement Service

The term placement service refers to the distinctive information identifying an entertainment content flow where advanced advertising techniques *may* apply. Typically, the ADM is provisioned to be aware of the entertainment content for which it ***shall*** offer placement opportunities and this configuration generally occurs via some external mechanism which is outside the scope of this specification. Example entertainment content includes a traditional linear broadcast network feed, a zoned form of the broadcast network, or a specific provider's collection of on-demand video assets to name just a few types.

The actual placement service definition is a compound information grouping utilizing two XML elements—the SystemContext element and the Service element. See Section 14.52 and Section 14.44 respectively for additional information. A placement service is identifiable via the Service element's @id attribute. Placement service organization is facilitated via the ServiceDescription element. See Section 14.45 for additional information. The combination of elements and their sub-elements provide for a rich placement service expression capability.

The ADM *may* be queried for its list of provisioned placement services via the list ADM services message exchange. See Section 11.2 for additional information. Based upon the list of offered placement services, the ADS registers for the placement services for which it desires to provide placement decisions. See Section 11.3 for additional information regarding registration. The ADS registration *may* be more or less general than the details provided by the list ADM services message exchange.

The remainder of this document defines the ADM interface.

7.0 NOTATIONAL CONVENTIONS

7.1 Normative XML Schema

See [SCTE130-2] for information.

7.2 Document Conventions

This specification utilizes the same document conventions as SCTE 130 Part 2. See [SCTE130-2] for conventions and XML schema illustration nomenclature explanations.

This specification utilizes XML substitution groups for additional extensibility. XML substitution groups designate elements as substitutes for other element declarations without changing the original schema documents. Within this document, substitutable elements are graphically identified using the following illustrative technique.

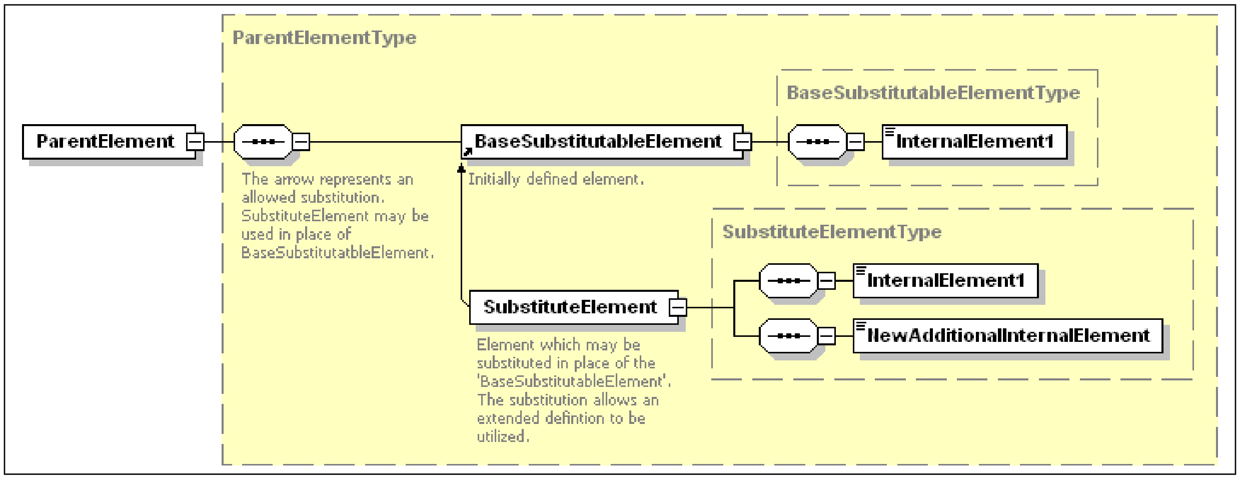

ParentElementType
BaseSubstitutableElementType
ParentElement
BaseSubstitutableElement
InternalElement1
The arrow represents an allowed substitution. SubstituteElement may be used in place of BaseSubstitutatbleElement.
Initially defined element.
SubstituteElementType
SubstituteElement
InternalElement1
NewAdditionalInternalElement
Element which may be substituted in place of the 'BaseSubstitutableElement'. The substitution allows an extended defintion to be utilized.

Figure 5: Substitution Group Schema Convention

In Figure 5, the element referred to as "SubstituteElement" *may* be used in place of the element named "BaseSubstitutableElement" provided the XML namespace declarations are included in the document as per [W3C-XSD]. The diagram's illustrative arrow signals the reader of the possible element substitution.

8.0 PROCESSING CONVENTIONS

8.1 Unknown/Unrecognized/Unsupported XML Elements and Attributes

See [SCTE130-2] for information.

9.0 XML NAMESPACES

This document uses the '*adm*' prefix 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.

Standard	XML Schema Prefix	XML Schema Elements	Value
2020 (latest)	core (SCTE 130-2)	Schema namespace	http://www.scte.org/schemas/130-2/2008a/core[1]
		Schema version attribute	20200321
		Schema filename	SCTE_130-2_core_20200321.xsd
	adm (This doc.)	Schema namespace	http://www.scte.org/schemas/130-3/2013/adm[2]
		Schema version attribute	20200326
		Schema filename	SCTE_130-3_adm_20200326.xsd
	podm (This doc. Appendix D)	Schema namespace	http://www.scte.org/schemas/130-3/2013/adm/podm[3]
		Schema version attribute	20200326
		Schema filename	SCTE_130-3_podm_20200326.xsd

[1] While this specification has a ratified year of 2020, the XML schema/XSD namespace has a year of 2008a, which is the year the XSD and this specification's syntax was initially ratified. All subsequent changes have been backwards compatible and thus, the namespace has not changed.

[2] While this specification has a ratified year of 2020, the XML schema/XSD namespace has a year of 2013, which is the year the XSD and this specification's syntax was initially ratified. All subsequent changes have been backwards compatible and thus, the namespace has not changed.

[3] See note 2.

	podm_query (This doc. Appendix D)	Schema namespace	http://www.scte.org/schemas/130-3/2013/adm/podm[4]
		Schema version attribute	20200326
		Schema filename	SCTE_130-3_podm_query_20200326.xsd
	gis (SCTE 130-8)	Schema namespace	http://www.scte.org/schemas/130-8/2011/gis[5]
		Schema version attribute	20200325
		Schema filename	SCTE_130-8_gis_20200325.xsd
	admwsdl (This doc. Appendix C)	Schema namespace	http://www.scte.org/wsdl/130-3/2013/adm[6]
		Schema version attribute	n/a
		WSDL filename	adm_20200326.wsdl

[4] While this specification has a ratified year of 2020, the XML schema/XSD namespace has a year of 2013, which is the year the XSD and this specification's syntax was initially ratified. All subsequent changes have been backwards compatible and thus, the namespace has not changed.

[5] While this specification has a ratified year of 2020, the XML schema/XSD namespace has a year of 2011, which is the year the XSD and this specification's syntax was initially ratified. All subsequent changes have been backwards compatible and thus, the namespace has not changed.

[6] While this specification has a ratified year of 2020, the WSDL namespace has a year of 2013, which is the year the WSDL and this specification's syntax was initially ratified. All subsequent changes have been backwards compatible and thus, the namespace has not changed.

	adswsdl (This doc. Appendix C)	Schema namespace	http://www.scte.org/wsdl/130-3/2013/ads[7]
		Schema version attribute	n/a
		WSDL filename	ads_20200326.xsd
	XML foundation. See [XMLSchema P1 from SCTE130-2 normative references].	xsd	http://www.w3.org/2001/XMLSchema

Table 1: XML Namespace Declarations

9.1 @version Attribute

For this specification's defined messages, the message's @version attribute ***shall*** be set to the value "1.4" corresponding to this document's revision. For those messages defined by SCTE 130 Part 2, the @version value ***shall*** be set to the value defined and specified in SCTE 130 Part 2. See [SCTE130-2] for additional information.

The @version attribute in conjunction with the namespace identifier (see Table 1) form the formal specification version identification.

See Section D.4.1 for further clarification of PODM versioning in this release and backward compatability for this version and previous versions including specific element usage limitations applicable to version "1.4".

10.0 ADM INTERFACE MESSAGE INTRODUCTION

The following topics are covered by SCTE 130 Part 2 and this specification considers all aspects defined therein to be normative and applicable herein. See [SCTE130-2] for information specific to this list and for expanded associated topics.

- Message format

[7] While this specification has a ratified year of 2020, the WSDL namespace has a year of 2013, which is the year the WSDL and this specification's syntax was initially ratified. All subsequent changes have been backwards compatible and thus, the namespace has not changed.

- XML message carriage
- Transport mechanisms
- Message error handling
- High availability

The ADM interface defined herein ***shall*** include all messages defined in SCTE 130 Part 2 and all ADM interface messages shall be compliant with SCTE 130 Part 2 [SCTE130-2]. Table 2 enumerates the ADM interface messages defined herein along with the message originators (i.e., the message sources). Both the ADM and the ADS ***shall*** be compliant with all messages listed in Table 2 and all Table 2 message's @version attribute ***shall*** comply with Section 9.1.

Message	**Source**	**Registration Required**
SCTE 130 Part 3 Defined Messages		
ListADMServicesRequest	ADS	
ListADMServicesResponse	ADM	
ADSRegistrationRequest	ADS	
ADSRegistrationResponse	ADM	
ListADSRegistrationRequest	ADS	
ListADSRegistrationResponse	ADM	
ADSDeregisterRequest	ADS	X
ADSDeregisterResponse	ADM	X
ADSDeregistrationNotification	ADM	X
ADSDeregistrationAcknowledgement	ADS	X
PlacementRequest	ADM	X
PlacementResponse	ADS	X
PlacementUpdateNotification	ADS	X
PlacementUpdateAcknowledgement	ADM	X
PlacementStatusNotification	ADM	X
PlacementStatusAcknowledgement	ADS	X
SCTE 130 Part 2 Defined Messages		
core:ServiceStatusNotification	ADM or ADS	
core:ServiceStatusAcknowledgement	ADM or ADS	
core:ServiceCheckRequest	ADM or ADS	
core:ServiceCheckResponse	ADM or ADS	

Table 2: SCTE 130 Part 3 Messages and Message Sources

Typically, a communication channel, referred to as service channel (see [SCTE130-2] for additional information) is established between two logical services in order to enable one service to deliver messages to another logical service. The service channel comprises both the logical and physical characteristics and the service channel lifecycle is dependent on the message type.

There are two common service channel lifecycles for this specification's messages. The first service channel lifecycle is limited to the duration of the single message pair exchange (i.e., a send and receive). The second service channel lifecycle duration commences with a successful placement service registration and continues until either participating logical service completes terminating the active service channel. This second service channel lifecycle is referred to herein as a registration-established service channel.

In Table 2, the column "Registration Required" indicates whether the specific message requires a registration-established service channel in order to be sent or received. A successful and active registration message exchange creating a registration-established service channel ***shall*** occur before any of these messages ***shall*** be sent and these messages ***shall*** only be exchanged for the active service channel duration. Registration-established service channel termination ***shall*** prohibit these messages from being exchanged on the concluded service channel.

Once successfully constructed, the registration-established service channel ***shall*** remain active until terminated via deregistration. Normal service channel termination ***shall*** occur via a deregistration message exchange. Abnormal termination (e.g., a system failure, etc.) should attempt to use the normal deregistration sequences. However, these situations *may* require alternative deregistration mechanisms outside the scope this specification.

Deregistration *may* be initiated by either the ADS or the ADM using an ADSDeregisterRequest or an ADSDeregistrationNotification message respectively. Physical and/or transport layer connections *may* transition without impacting the service channel active state. That is, the registration-established service channel maintains its operational state independent of the transport persistence. For example, if TCP/IP is used as the transport protocol, the TCP/IP connection *may* be terminated and re-established multiple times during which time the registration-established service channel is considered to remain persistent.

The ADM interface message exchanges are grouped into the following general activity categories:

Logical service management which covers:

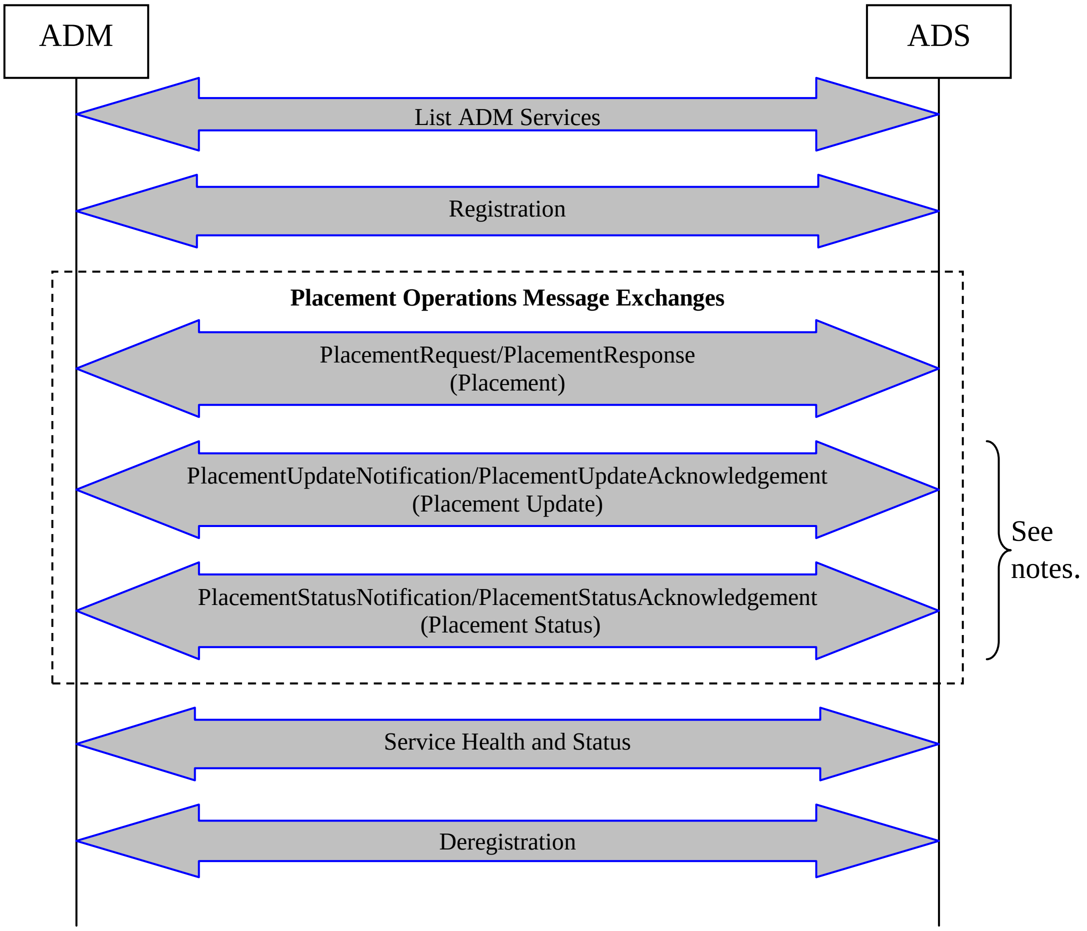
- Communication management including placement service offering descriptions, registration, and deregistration
- Status management including logical service health checks, logical service status notifications, and active registration listing

Placement operations which includes:

- Placement management including placement opportunity descriptions, placement decisions which *may* include placements and placement updates

- Placement status consisting of placement event reporting which “close the loop” on placement decisions

Figure 6 illustrates a high-level overview of the typical message exchange sequences between an ADS and an ADM.


ADM
ADS
List ADM Services
Registration
Placement Operations Message Exchanges
PlacementRequest/PlacementResponse
(Placement)
PlacementUpdateNotification/PlacementUpdateAcknowledgement
(Placement Update)
PlacementStatusNotification/PlacementStatusAcknowledgement
(Placement Status)
See
notes.
Service Health and Status
Deregistration

Notes: The placement message exchange may be followed by any number of placement update message exchanges. Additionally, there may be any number of placement status message exchanges and these message exchanges may occur at any time. Typically, the placement status exchanges commence following the initial placement message exchange and there is no guarantee to their relative interleaving with respect to the other message exchanges.

Figure 6: Message Exchange Overview

The ADS *may* obtain the list of offered ADM placement services, capabilities and message exchange endpoints using the ListADMServices message exchange. The ADS registers its placement decision intentions using the registration message exchange and terminates its placement service relationship using the deregistration message exchange. At any time, either logical service *may* notify or check the service health and status of the peer.

The placement operations message exchanges encompass placement messaging (i.e., PlacementRequest/PlacementResponse messages), placement update messaging (i.e., PlacementUpdateNotification/PlacementUpdateAcknowledgement messages), and placement status messaging (i.e., PlacementStatusNotification/PlacementStatusAcknowledgement messages). These messages exchanges describe placement opportunities, placement decisions, placement decision updates, and placement reporting for the placement decisions.

11.0 ADM INTERFACE—LOGICAL SERVICE MANAGEMENT MESSAGES

Excluding placement operations messages, this specification defines the following message exchanges for the purpose of logical service provisioning and status management:

- List offered placement services—a detailed list of ADM provided placement services, ADM capabilities, and ADM message exchange endpoints.
- Registration—ADS placement service identification and establishment of the service channel(s).
- Registration status—a report of the established placement services and their service channels.
- Deregistration—the termination of placement services and their established service channels.
- Logical service health—the abilities to verify that a service channel, logical service and transport connection are alive and operational and to provide logical service status notifications. The capabilities leverage the SCTE 130 Part 2 core:ServiceStatus and core:ServiceCheck message exchanges as defined therein. See [SCTE130-2] for additional information.

The following sections detail this specification’s logical service management messages.

11.1 ADM Logical Service Discovery

Initial ADM logical service discovery is outside the scope of this specification. SCTE 130 Part 7 *may* provide details. See [SCTE130-7] for additional information.

With the exception of the ADM and its ListADMServicesRequest message exchange endpoint discovery, the remaining endpoints *may* be directly provided by both the ADM and ADS (i.e., they do not require separate discovery mechanisms and *may* be bootstrapped using the messages defined herein). See Sections 11.2 and 11.3 for additional information.

11.2 List ADM Placement Services

Any logical service *may* query an ADM at any time for its offered placement services, capabilities, and message exchange endpoint addresses. The querying entity is not required to be registered prior to making this query and the query *may* occur at anytime. This message exchange duration defines this service channel lifecycle. Typically, the message exchange occurs before registration and its associated registration-established service channel establishment. Figure 7 illustrates the list ADM placement services message exchange.

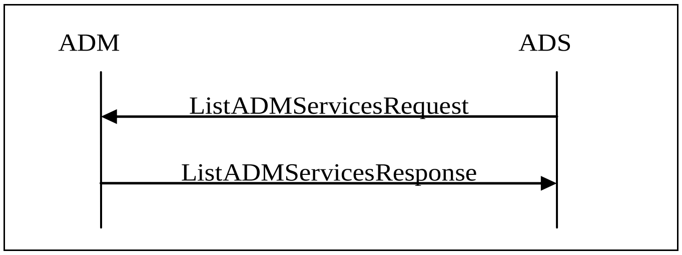


Figure 7: List ADM Placement Services Message Exchange

An ADS initiates the message exchange using the ListADMServicesRequest message and the ADM ***shall*** respond with a ListADMServicesResponse message. The ADM response includes the following:

- The ADM offered placement service list
- The ADM capabilities
- The ADM message exchange endpoint list

How an ADM determines its offered placement service list and capabilities is outside the scope of this specification.

The ADM message exchange endpoint list *may* supply the ADS with ADM endpoints for individual message processing entry. Typically, the list is populated with the endpoint set necessary for bootstrapping ADS to ADM communications. With the exception of the discovery of the ListADMServicesRequest endpoint, the remaining ADM/ADS message exchange endpoints *may* be provided using a combination of the ListADMServicesResponse message and the registration message exchange (i.e., ADSRegistrationRequest/ADSRegistrationResponse messages).

The list ADM offered placement services message schemas are described in the following sections.

11.2.1 ListADMServicesRequest Message Schema

Figure 8 illustrates the list ADM placement services request message schema.

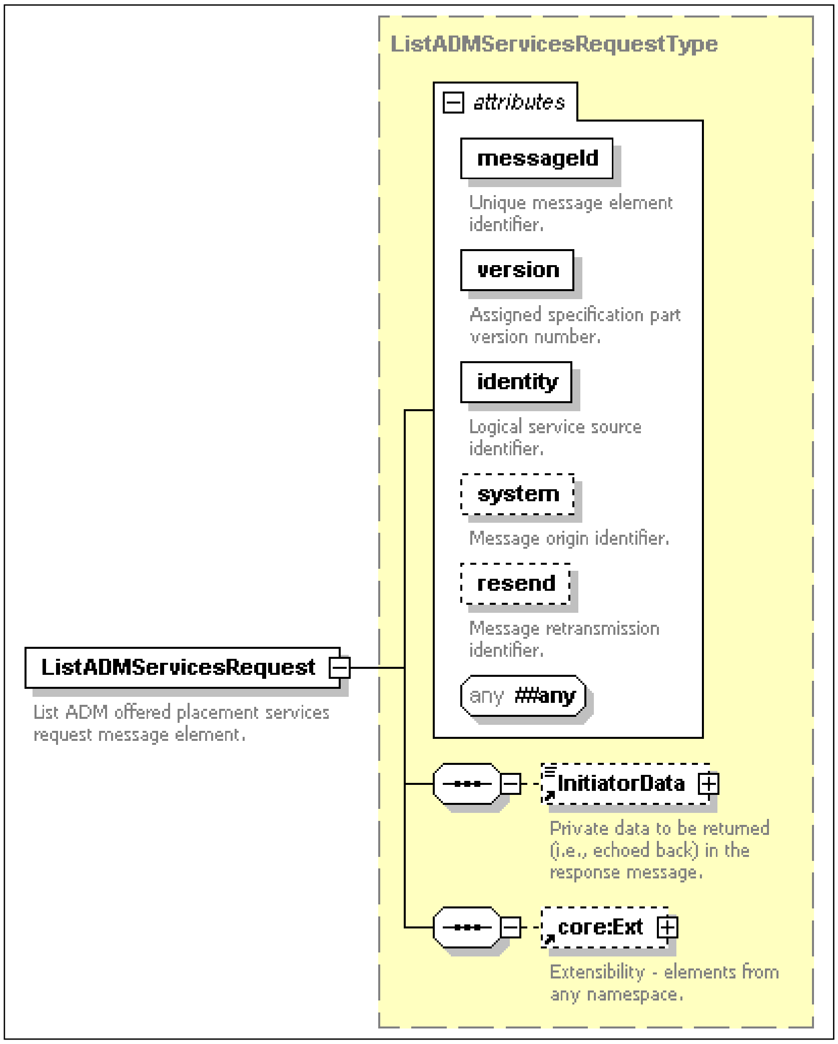

ListADMServicesRequestType
attributes
messageId
Unique message element identifier.
version
Assigned specification part version number.
identity
Logical service source identifier.
system
Message origin identifier.
resend
Message retransmission identifier.
any ##any
ListADMServicesRequest
List ADM offered placement services request message element.
InitiatorData
Private data to be returned (i.e., echoed back) in the response message.
core:Ext
Extensibility - elements from any namespace.

Figure 8: ListADMServicesRequest Message Schema

11.2.1.1 Semantic Definitions for the ListADMServicesRequest Message

@messageId [Required, core:messageIdAttrType]—The message identifier. See [SCTE130-2] for additional information.

@version [Required, core:versionAttrType]—The ADM message interface version. See Section 9.1 for additional information.

@identity [Required, core:identityAttrType]—The origin logical service identifier. See [SCTE130-2] for additional information.

@system [Optional, core:systemAttrType]—The message origin identifier. See [SCTE130-2] for additional information.

@resend [Optional, core:resendAttrType]—The message is a retransmit of a previous ListADMServicesRequest message. The attribute’s value is the original message’s @messageId value. See [SCTE130-2] for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:InitiatorData [Optional]—The core:InitiatorData element contains implementation specific private data which ***shall*** be returned in the ListADMServicesResponse message. See [SCTE130-2] for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

The ListADMServicesRequest element’s value *may* be empty.

11.2.2 ListADMServicesResponse Message Schema

Figure 9 illustrates the list ADM placement services response message schema describing the placement services offered by the ADM.

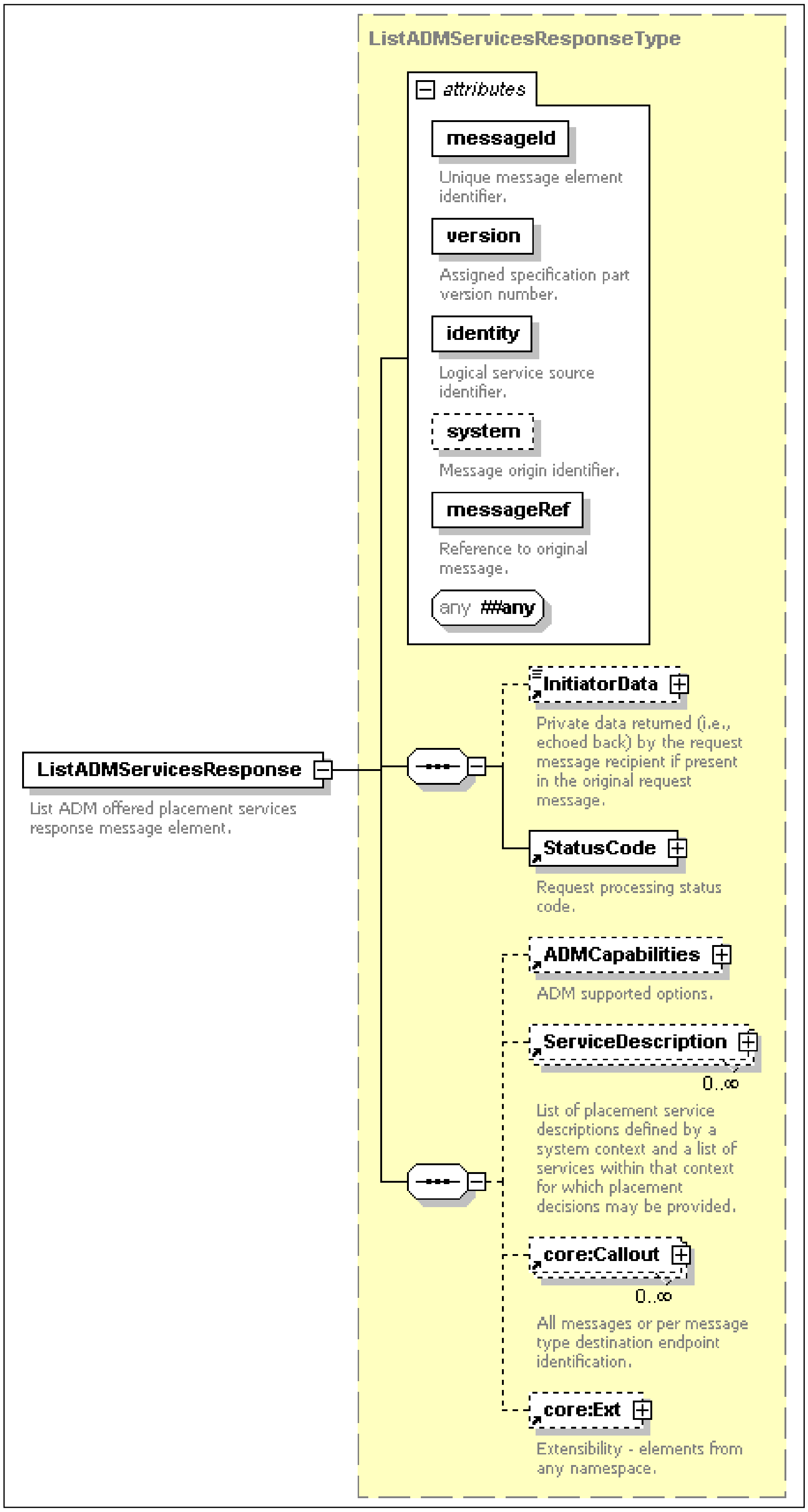

ListADMServicesResponseType
attributes
messageId
Unique message element identifier.
version
Assigned specification part version number.
identity
Logical service source identifier.
system
Message origin identifier.
messageRef
Reference to original message.
any ##any
InitiatorData
Private data returned (i.e., echoed back) by the request message recipient if present in the original request message.
StatusCode
Request processing status code.
ListADMServicesResponse
List ADM offered placement services response message element.
ADMCapabilities
ADM supported options.
ServiceDescription
0..∞
List of placement service descriptions defined by a system context and a list of services within that context for which placement decisions may be provided.
core:Callout
0..∞
All messages or per message type destination endpoint identification.
core:Ext
Extensibility - elements from any namespace.

Figure 9: ListADMServicesResponse Message Schema

11.2.2.1 Semantic Definitions for the ListADMServicesResponse Message

@messageId [Required, core:messageIdAttrType]—The message identifier. See [SCTE130-2] for additional information.

@version [Required, core:versionAttrType]—The ADM message interface version. See Section 9.1 for additional information.

@identity [Required, core:identityAttrType]—The origin logical service identifier. See [SCTE130-2] for additional information.

@system [Optional, core:systemAttrType]—The message origin identifier. See [SCTE130-2] for additional information.

@messageRef [Required, core:messageRefAttrType]—A reference to the paired ListADMServicesRequest message element initiating this message exchange. The value ***shall*** be the ListADMServicesRequest message's @messageId attribute value. See [SCTE130-2] for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:InitiatorData [Optional]—The core:InitiatorData element ***shall*** be an exact copy of the ListADMServicesRequest message's core:InitiatorData element and ***shall*** only be present if found in the paired request message. See [SCTE130-2] for additional information.

core:StatusCode [Required]—An applicable status code specific to the ListADMServicesRequest message processing. See [SCTE130-2] and Appendix A for additional information.

ADMCapabilities [Optional]—ADM supported options and limitations. Element omission ***shall*** indicate the ADM is using the element's defined default values. See Section 14.1 for additional information.

ServiceDescription [Optional]—Zero or more ServiceDescription elements where each ServiceDescription element uniquely describes a collection of ADM offered placement services. See Section 14.45 for additional information. If the core:StatusCode element does not contain a success value, the ServiceDescription elements *may* be omitted. If an ADM is offline, then no elements *may* be present. The element sequence order ***shall no***t convey information.

core:Callout [Optional]—Zero or more core:Callout elements specifying the ADM message exchange endpoint(s). See [SCTE130-2] for additional information. Zero core:Callout elements *may* be returned for either an error condition or if the ADM endpoint discovery is occurring via a different mechanism.

A single message exchange endpoint *may* be defined to handle all ADM destined messaging identified in Table 3. If a single endpoint is handling all messaging, there should be only one core:Callout element present in the ListADMServicesResponse message and the core:Callout element’s @message attribute ***shall no***t be used. This core:Callout element is referred to as the default endpoint.

If independent endpoints are desired, the core:Callout element’s @message attribute ***shall*** be used and each core:Callout element ***shall*** contain a value from Table 3. The core:Callout element’s @message attribute’s value ***shall*** appear exactly as shown in Table 3.

Value	Description
ServiceStatusNotification	Destination for ServiceStatusNotification messages.
ADSRegistrationRequest	Destination for ADSRegistrationRequest messages.
ADSDeregisterRequest	Destination for ADSDeregisterRequest messages.
ListADSRegistrationRequest	Destination for ListADSRegistrationRequest messages.
…	User defined message endpoints which are outside of this specification’s scope. The string ***shall*** be prefixed with the text “private:”.

Table 3: ListADMServicesResponse/core:Callout/@Message Attribute Values

If independent message exchange endpoints are desired for only a subset of endpoints, the default endpoint *may* be used in conjunction with one or more additional core:Callout elements containing values from Table 3. All message endpoints listed in Table 3 that are not specifically referenced by an @message attribute ***shall*** be available through the default endpoint (assuming one is supplied). This behavior allows an ADM to provide message exchange endpoints for one or more core:Callout endpoints while utilizing a single, general purpose endpoint for all other messaging.

If no default endpoint is supplied and only a subset of the Table 3 endpoints are provided in the ListADMServicesResponse message, the unlisted endpoints ***shall*** be discovered by a different, unspecified mechanism which is outside the scope of this specification.

Within the core:Callout element, the ADM *may* provide one or more core:Address elements. The processing rules for these elements are not specified here and are outside the scope of this specification.

Each core:Address element contained within the core:Callout element ***shall*** support the core:ServiceCheck message exchange as per [SCTE130-2]. When a Callout element contains more than one Address element, at least one Address element specified reception endpoint ***shall*** successfully respond to the ServiceCheckRequest message as required by [SCTE130-2].

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

11.2.3 List ADM Placement Services Examples

Example 1 illustrates a successful list ADM placement services request and response (Example 2) message exchange for an ADM offering a single entry representing “all” services (thus, a generic service description element). Note: The string “All ADM Offered Services” is an implementation choice specific to this example. This specification’s usage of this string in conjunction with the concept of all placement services is purely coincidental and it ***shall no***t be considered normative when representing said concept.

```
<ListADMServicesRequest messageId="29D61093-6E81-4A5B-A1F6-8544AA7F29A0"
version="1.4" identity="ADSLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F5" system="ads"/>
```

Example 1

Example 2 declares the ADM capabilities and indicates the ADM is using a default endpoint for all message processing. This behavior is indicated by the presence of a single core:Callout element and the core:Callout element omitting the @message attribute. It also declares support for PODM namespace version 1.2.

```
<ListADMServicesResponse messageId="29D61093-6E81-4A5B-A1F6-8544AA7F29A1"
version="1.4" identity="ADMLogicalService_1170FEF4-C19B-450E-B624-
```

```
421B23F525F4" system="adm" messageRef="29D61093-6E81-4A5B-A1F6-
8544AA7F29A0">
  <core:StatusCode class="0"/>
  <ADMCapabilities placementStatusOnlyRegistration="false">
    < NameSpaceVersion
      namespace="http://www.scte.org/schemas/130-3/2013/adm/podm"
      version="1.2"/>
  </ADMCapabilities>
  <ServiceDescription>
    <SystemContext/>
    <Service id="All ADM Offered Services"/>
  </ServiceDescription>
  <core:Callout>
    <core:Address
type="HTTP">http://173.35.23.33:80/SCTE130MessageProcessor/default</core:Address>
  </core:Callout>
</ListADMServicesResponse>
```

Example 2

Example 3 illustrates a successful response message, assuming the Example 1 request message, for an ADM offering placement services for a service group identified as “Den”. The ADM supplies a default endpoint used for all message exchanges except the ADSRegistrationRequest and the ADSDeregisterRequest message exchanges for which it is supplying unique endpoints. It also declares support for PODM namespace version 1.1.

```
<ListADMServicesResponse messageId="29D61093-6E81-4A5B-A1F6-8544AA7F29A1"
version="1.4" identity="ADMLogicalService_1170FEF4-C19B-450E-B624-
```

```
421B23F525F4" system="adm" messageRef="29D61093-6E81-4A5B-A1F6-
8544AA7F29A0">
  <core:StatusCode class="0"/>
  <ADMCapabilities placementStatusOnlyRegistration="false">
    < NameSpaceVersion
      namespace="http://www.scte.org/schemas/130-3/2013/adm/podm"
      version="1.1"/>
  </ADMCapabilities >
  <ServiceDescription>
    <SystemContext>
      <ServiceGroup>Den</ServiceGroup>
    </SystemContext>
    <Service id="ServiceDen"/>
  </ServiceDescription>
  <core:Callout>
    <core:Address
type="HTTP">http://173.35.23.33:80/SCTE130MessageProcessor/default</core:Address>
  </core:Callout>
  <core:Callout message="ADSRegistrationRequest">
    <core:Address
type="HTTP">http://173.35.23.33:80/SCTE130MessageProcessor/ADSRegistrationReque
st</core:Address>
  </core:Callout>
  <core:Callout message="ADSDeregisterRequest">
    <core:Address
type="HTTP">http://173.35.23.33:80/SCTE130MessageProcessor/ADSDeregisterRequest
</core:Address>
  </core:Callout>
</ListADMServicesResponse>
```

Example 3

Example 4 is a success response message, assuming the Example 1 request message, for an ADM offering on-demand placement services for the provider’s moviego.com and movies_r_us.com. For moviego.com, the ADM is handling pre-roll and post-roll placement opportunities using an “interactive” placement decision policy. For movies_r_us.com, the ADM is only handling local pre-roll placement opportunities for the specific product type using a “default” placement decision policy. The ADM is using a single, default endpoint for all message processing. It also declares support for PODM namespace version 1.1 and 1.2. A subsequent ADSRegistrationMessage can choose the PODM namespace to utilize.

```
<ListADMServicesResponse messageId="29D61093-6E81-4A5B-A1F6-8544AA7F29A1"
version="1.4" identity="ADMLogicalService_1170FEF4-C19B-450E-B624-
```

```
421B23F525F4" system="adm" messageRef="29D61093-6E81-4A5B-A1F6-
8544AA7F29A0">
  <core:StatusCode class="0"/>
  <ADMCapabilities placementStatusOnlyRegistration="false">
    < NameSpaceVersion
      namespace="http://www.scte.org/schemas/130-3/2013/adm/podm"
     version="1.1"/>
    < NameSpaceVersion
      namespace="http://www.scte.org/schemas/130-3/2013/adm/podm"
      version="1.2"/>
  </ADMCapabilities >
  <ServiceDescription>
    <SystemContext/>
    <Service id="Service1" opportunityType="preRoll" decisionPolicy="interactive">
      <ContentProvider providerID="moviego.com">
        <core:ContentDataModel type="SCTE236>URI</core:ContentDataModel>
      </ContentProvider>
    </Service>
    <Service id="Service2" opportunityType="postRoll" decisionPolicy="interactive">
      <ContentProvider providerID="moviego.com">
        <core:ContentDataModel type="SCTE236">URI</core:ContentDataModel>
      </ContentProvider>
    </Service>
    <Service id="Service3" opportunityType="preRoll" decisionPolicy="default">
      <ContentProvider providerID="movies_r_us.com">
        <core:ContentDataModel type="SCTE236">URI</core:ContentDataModel>
      </ContentProvider>
      <ProductType>FOD</ProductType>
      <Scope>Local</Scope>
    </Service>
  </ServiceDescription>
  <core:Callout>
    <core:Address>www.adshark.com/allMessages</core:Address>
  </core:Callout>
</ListADMServicesResponse>
```

Example 4

Example 5 is a successful response message, assuming the Example 1 request message, for an ADM offering traditional linear replacement services for two different zones of a specific network feed. The decision policy is “preload” for both listed services and the ADM is using unique entry points for every service channel message endpoint. It also declares support for PODM namespace version 1.2. It also declares support for a non SCTE 130 namespace from www.foo.com. This declaration informs the ADS that it can use private extensions supported by version 1.0 of the specified www.foo.com namespace.

This declaration informs the ADS that it can use private extensions supported by the specified www.bar.com namespace.

```
<ListADMServicesResponse messageId="29D61093-6E81-4A5B-A1F6-8544AA7F29A1"
version="1.4" identity="ADMLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F4" system="adm" messageRef="29D61093-6E81-4A5B-A1F6-
8544AA7F29A0">
  <core:StatusCode class="0"/>
  <ADMCapabilities placementStatusOnlyRegistration="true">
    < NameSpaceVersion
      namespace="http://www.scte.org/schemas/130-3/2013/adm/podm"
      version="1.2"/>
    < NameSpaceVersion
      namespace="http://www.foo.com/schemas/namespace/foo"
    version="1.0"/>
    < NameSpaceVersion
      namespace="http://www.bar.com/schemas/namespace/bar"/>
  </ADMCapabilities>
  <ServiceDescription>
    <SystemContext>
      <Network>ABC</Network>
      <Zone>Z3</Zone>
    </SystemContext>
    <Service id="ABC Network Zone3" decisionPolicy="preload"/>
  </ServiceDescription>
  <ServiceDescription>
    <SystemContext>
      <Network>ABC</Network>
      <Zone>Z4</Zone>
    </SystemContext>
    <Service id="ABC Network Zone4" decisionPolicy="preload"/>
  </ServiceDescription>
  <core:Callout message="ServiceStatusNotification">
    <core:Address type="HTTP">http://175.33.53.33:7381/scte130/ssn</core:Address>
  </core:Callout>
  <core:Callout message="ADSRegistrationRequest">
    <core:Address type="HTTP">http://175.33.53.33:7382/scte130/arr</core:Address>
  </core:Callout>
  <core:Callout message="ADSDeregisterRequest">
    <core:Address type="HTTP">http://175.33.53.33:7383/scte130/adr</core:Address>
  </core:Callout>
  <core:Callout message="ListADSRegistrationRequest">
    <core:Address type="HTTP">http://175.33.53.33:7384/scte130/larr</core:Address>
  </core:Callout>
</ListADMServicesResponse>
```

Example 5

Example 6 illustrates a failure response message, assuming the Example 1 request message, for an ADM not currently offering any services.

```
<ListADMServicesResponse messageId="29D61093-6E81-4A5B-A1F6-8544AA7F29A1"
version="1.4" identity="ADMLogicalService" system="adm" messageRef="29D61093-
6E81-4A5B-A1F6-8544AA7F29A0">
 <core:StatusCode class="1" detail="11">
   <core:Note>ResourceNotFound. No provisioned ADM services.</core:Note>
  </core:StatusCode>
</ListADMServicesResponse>
```

Example 6

11.3 Registration

A registration-established service channel, as defined in Section 10.0, is created between two logical services using the registration message exchange. Typically, this service channel is used for placement operations messaging. An ADS service contacts an ADM by sending an ADSRegistrationRequest message and the ADM ***shall*** respond with an ADSRegistrationResponse message. Figure 10 illustrates this message exchange.

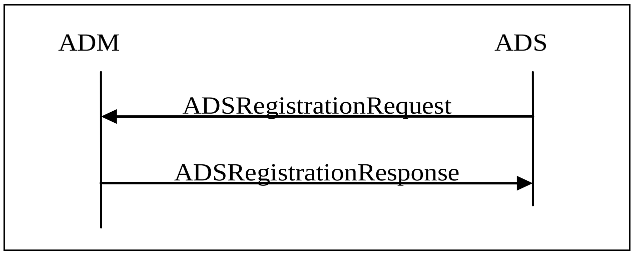


Figure 10: Registration Message Exchange

During the registration request message processing, the ADM *may* optionally verify each ADS supplied endpoint via a SCTE 130 Part 2 core:ServiceCheck message exchange before returning an ADSRegistrationResponse message. (See [SCTE130-2] for additional information regarding the core:ServiceCheck message exchange.) The ADS ***shall*** be prepared to receive the core:ServiceCheckRequest message on the registration supplied endpoints before receiving a response else the ADM *may* fail the registration request (i.e., the ADM *may* test each core:Address element supplied in each core:Callout element). When a core:Callout element contains more than one core:Address element, at least one core:Address element specified reception endpoint ***shall*** successfully respond to the core:ServiceCheckRequest message. The ADM *may* reject the registration request message if all core:Address elements of a core:Callout element fail to successfully respond. An ADM ***shall not*** reject the registration message if only a subset of each core:Callout element's core:Address elements successfully respond to the core:ServiceCheckRequest message.

The ADS *may* provide unique service channel endpoint identifiers for the placement operation message exchanges including specific destinations for PlacementRequest messages and PlacementStatusNotification messages. In the ADM response message, the ADM *may* provide the PlacementUpdateNotification message callout.

ADM implementations *may* choose to limit the number of supported service channel connections. An ADM ***shall no***t be required to support independently registered callbacks for each offered placement service identified by the ListADMServicesResponse message's ServiceDescription elements. ADM implementations are encouraged to support at least as many service channel connections as the count of offered placement services itemized in the ListADMServicesResponse message. An ADM implementation ***shall no***t be required to accept an ADS registration solely for status reporting. An ADM *may* optionally offer this feature by indicating support within the ADMCapabilities element and allowing an ADSRegistrationRequest message containing a core:Callout element which specifies only a PlacementStatusNotification endpoint.

ADM registration conflict determination and resolution are outside the scope of this specification.

During registration, the ADS specifies the placement services for which it is providing placement decisions and it declares the endpoint information for where associated PlacementRequest and PlacementStatusNotification messages are to be sent. These endpoints *may* be the same value or they *may* be unique. The endpoints *may* be different addresses on the same system or they *may* be endpoints on physically separated systems as illustrated in Figure 11. This figure is one possible example implementation with many other abstractions possible and not represented by this single illustration.

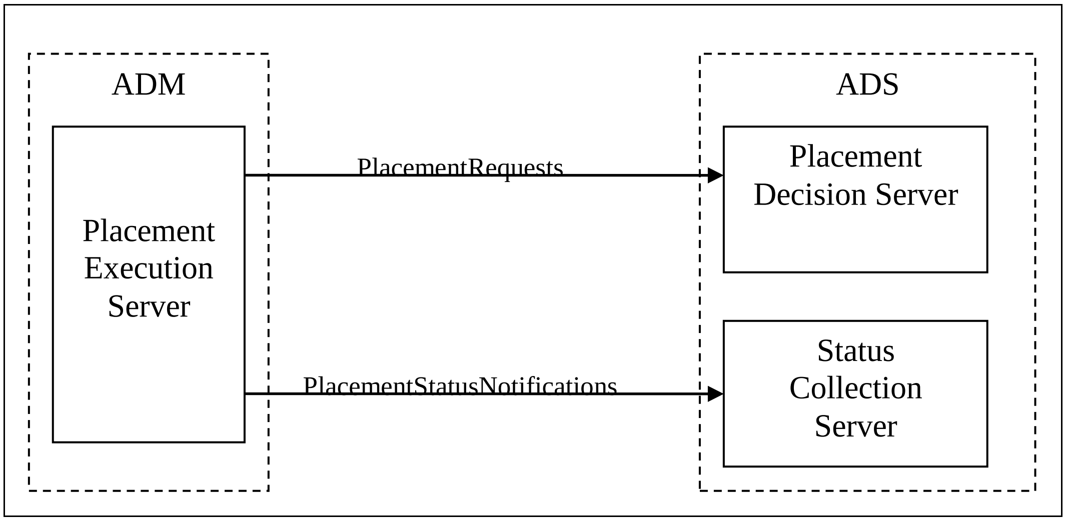

ADM
Placement Execution Server
PlacementRequests
PlacementStatusNotifications
ADS
Placement Decision Server
Status Collection Server

Figure 11: ADS Registering Independent Endpoints

The ADSRegistrationRequest message ***shall*** always be identified using the original ADSRegistrationRequest message’s @messageId attribute value which in a retransmitted registration message ***shall*** be the @resend attribute’s value. This original message’s @messageId attribute value is subsequently used for registration listing and deregistration message exchange per SCTE 130 Part 2. See [SCTE130-2] for additional information.

The ADM registration message schemas are described in the following sections.

11.3.1 ADSRegistrationRequest Message Schema

Figure 12 illustrates the ADSRegistrationRequest message schema.

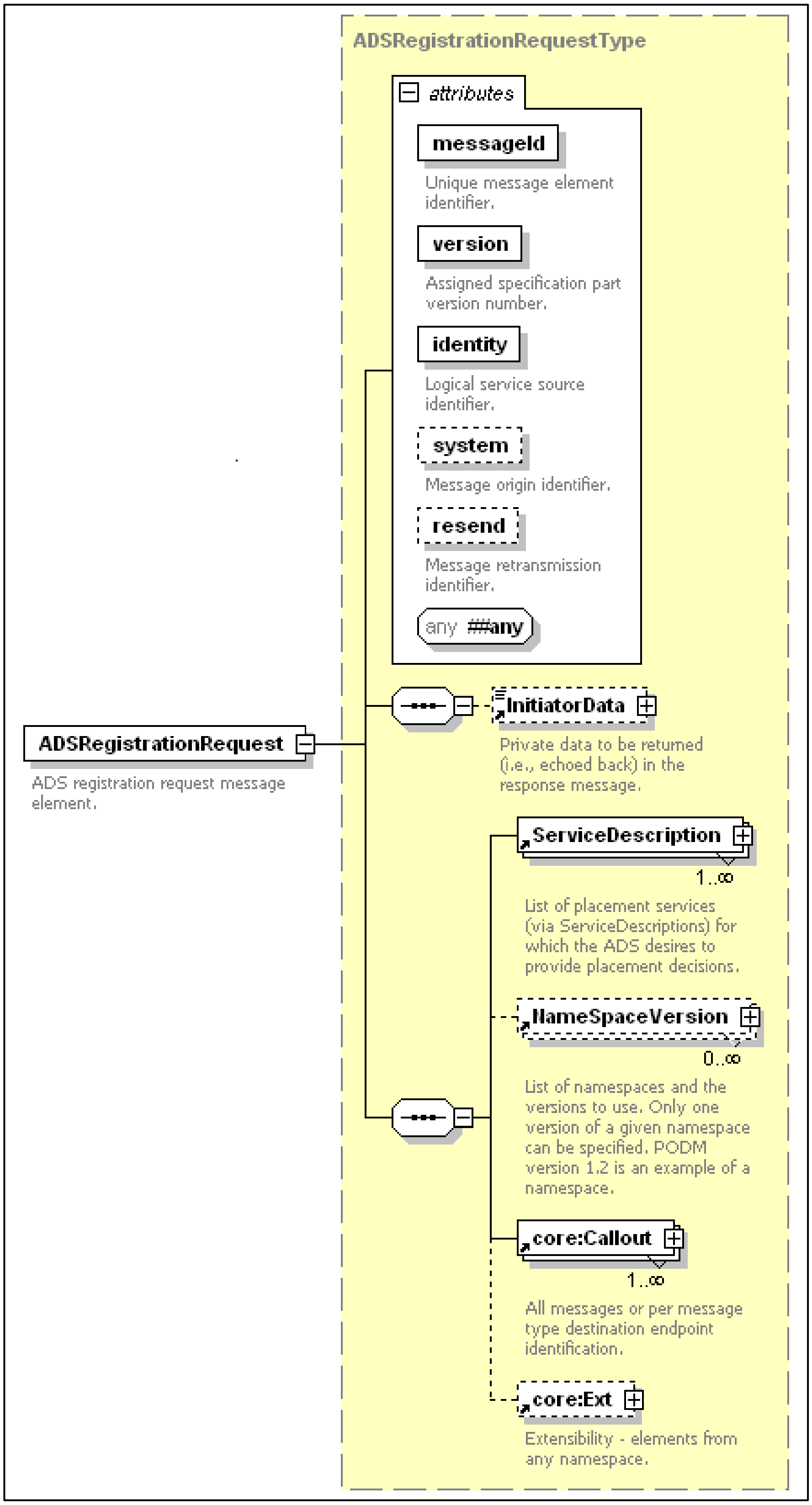

ADSRegistrationRequestType
attributes
messageId
Unique message element identifier.
version
Assigned specification part version number.
identity
Logical service source identifier.
system
Message origin identifier.
resend
Message retransmission identifier.
any ##any
InitiatorData
Private data to be returned (i.e., echoed back) in the response message.
ADSRegistrationRequest
ADS registration request message element.
ServiceDescription
1..∞
List of placement services (via ServiceDescriptions) for which the ADS desires to provide placement decisions.
NameSpaceVersion
0..∞
List of namespaces and the versions to use. Only one version of a given namespace can be specified. PODM version 1.2 is an example of a namespace.
core:Callout
1..∞
All messages or per message type destination endpoint identification.
core:Ext
Extensibility - elements from any namespace.

Figure 12: ADSRegistrationRequest Message Schema

11.3.1.1 Semantic Definitions for the ADSRegistrationRequest Message

@messageId [Required, core:messageIdAttrType]—The message identifier. See [SCTE130-2] for additional information.

@version [Required, core:versionAttrType]—The ADM message interface version. See Section 9.1 for additional information.

@identity [Required, core:identityAttrType]—The origin logical service identifier. See [SCTE130-2] for additional information. An ADS ***shall*** use this attribute to identify all their placement services and their registration-established service channels for registration listing and deregistration purposes.

@system [Optional, core:systemAttrType]—The message origin identifier. See [SCTE130-2] for additional information.

@resend [Optional, core:resendAttrType]—The message is a retransmit of a previous ADSRegistrationRequest message. The attribute's value is the original message's @messageId value. See [SCTE130-2] for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:InitiatorData [Optional]—The core:InitiatorData element contains implementation specific private data which ***shall*** be returned in the ADSRegistrationResponse message. See [SCTE130-2] for additional information.

ServiceDescription [Required]—One or more placement service descriptions for which the registering entity desires interaction. Generally, the expected registration-established placement service interaction is to supply placement decisions and placement status message exchanges. As an alternative, a placement status only registered service *may* optionally be supported by an ADM. Typically, each ServiceDescription element should exactly match a ServiceDescription element as provided in the ListADMServicesResponse message described in Section 11.2.2.

An ADS *may* restrict a placement service's scope by adding elements to a ListADMServicesResponse message's ServiceDescription element's children. For example, if an ADM returns a ServiceDescription element contain the Network element with the value "ESPN," the ADS *may* register a ServiceDescription element containing a Network element with the value "ESPN" and a Scope element with the value "Local".

An ADS *may* also expand a placement serivce's scope by removing elements from a ListADMServicesResponse message's

ServiceDescription element's children. For example, if an ADM returns a ServiceDescription element containing a Network element with the value "ESPN and a Service/@oppotunityBinding value of "preRoll," the ADS *may* register a ServiceDescription element containing only a Network element having the value "ESPN."

An ADM *may* optionally accept ADSRegistrationRequest messages which are not exact matches to the ServiceDescription elements returned in the ListADMServicesResponse message. The ServiceDescription element sequence ***shall*** be processed in document order.

NameSpaceVersion [Optional] – One or more NameSpaceVersion elements specifying the namespaces to use on the exchange end point(s). The version attribute is optional but ***shall*** be supplied if returned for a given namespace in the ListADMServicesResponse. The list of supported namespaces is returned in the ListADMServicesResponse (see Section 11.2). See Section 14.20 for additional information.

An implementation *may* support registration for multiple versions of a given namespace. The specific end point behavior if registering for multiple versions of a given namespace is out of scope of this standard.

core:Callout [Required]—One or more core:Callout elements specifying the ADS message exchange endpoint(s). See [SCTE130-2] for additional information. A single message exchange endpoint *may* be defined to handle all ADS destined messaging. If a single message exchange endpoint is handling all messaging (i.e., a default endpoint), there should be only one core:Callout element present and the core:Callout element's @message attribute ***shall no***t be used.

If independent endpoints are desired, the core:Callout element's @message attribute ***shall*** be used and each core:Callout element ***shall*** contain a value from Table 4. The core:Callout element's @message attribute's value ***shall*** appear exactly as shown in Table 4.

Value	Description
ServiceStatusNotification	Destination for ServiceStatusNotification messages.
PlacementRequest	Destination for PlacementRequest messages.
PlacementStatusNotification	Destination for PlacementStatusNotification messages.
ADSDeregistrationNotification	Destination for ADSDeregistrationNotification messages.
…	User defined message endpoints which are outside of this specification's scope. The string ***shall*** be prefixed with the text "private:".

Table 4: ADSRegistrationRequest/core:Callout/@Message Attribute Values

If independent endpoints are desired for a subset of message exchange endpoints, the default endpoint *may* be used with one or more additional core:Callout elements containing values from Table 4. All messages that are not specifically referenced by an @message attribute ***shall*** be available through the default endpoint.

If no default endpoint is supplied and only a subset of the Table 4 endpoints are provided in the ADSRegistrationRequest message, the unlisted endpoints ***shall*** be discovered by a different, unspecified mechanism which is outside the scope of this specification.

Within the core:Callout element, the ADS *may* provide one or more core:Address elements. The processing rules for these elements are not specified here and are outside the scope of this specification.

Each core:Address element contained within the core:Callout element ***shall*** support the core:ServiceCheck message exchange as per [SCTE130-2]. When a Callout element contains more than one Address element, at least one Address element specified reception endpoint ***shall*** successfully respond to the ServiceCheckRequest message as required by [SCTE130-2] and an ADM ***shall no***t reject the registration message if only a subset of the Address elements successfully respond to the ServiceCheckRequest message.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

11.3.2 ADSRegistrationResponse Message Schema

Figure 13 illustrates the ADSRegistrationResponse message schema.

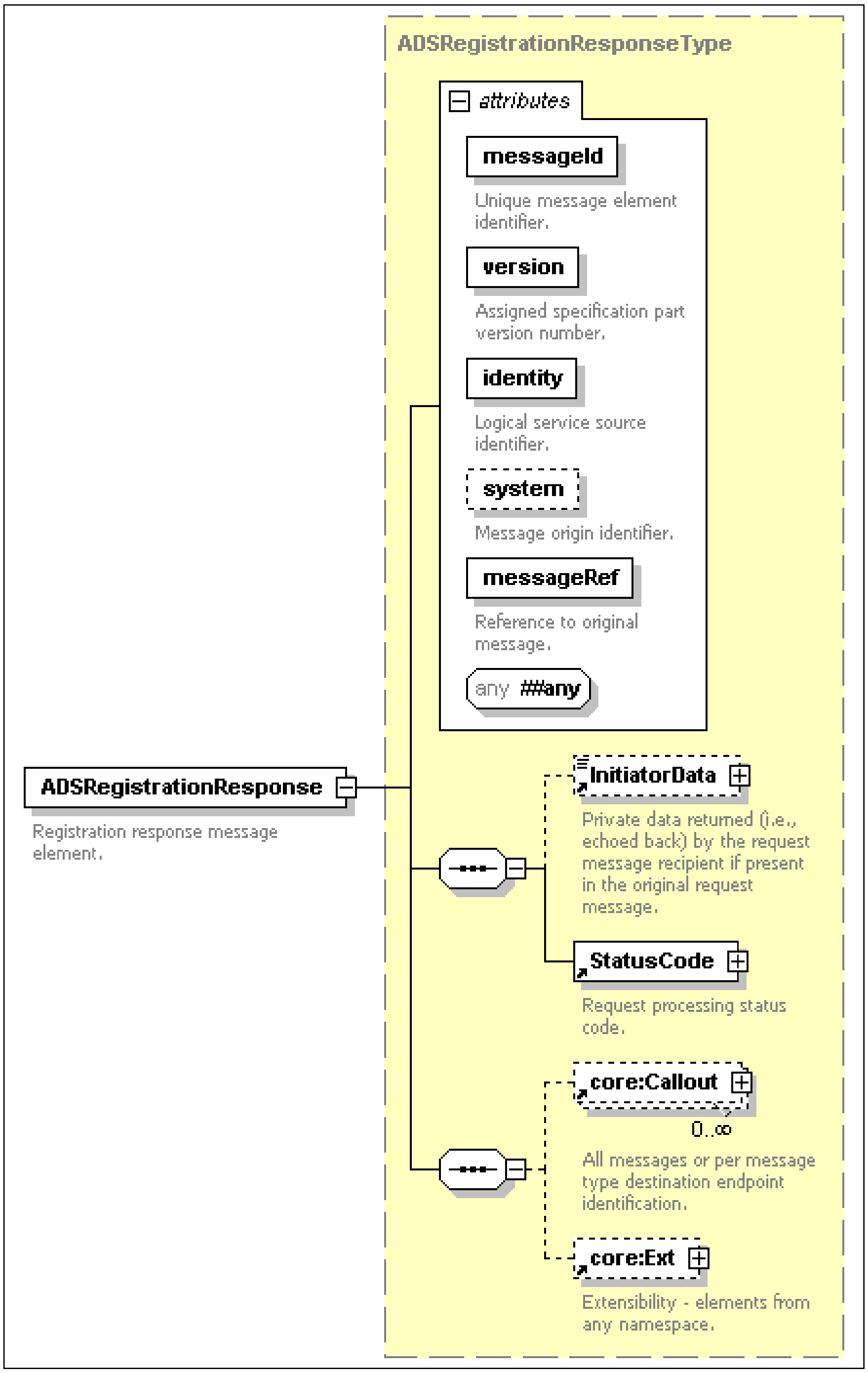

ADSRegistrationResponseType
attributes
messageId
Unique message element identifier.
version
Assigned specification part version number.
identity
Logical service source identifier.
system
Message origin identifier.
messageRef
Reference to original message.
any ##any
ADSRegistrationResponse
Registration response message element.
InitiatorData
Private data returned (i.e., echoed back) by the request message recipient if present in the original request message.
StatusCode
Request processing status code.
core:Callout
0..∞
All messages or per message type destination endpoint identification.
core:Ext
Extensibility - elements from any namespace.

Figure 13: ADSRegistrationResponse Message Schema

11.3.2.1 Semantic Definitions for the ADSRegistrationResponse Message

@messageId [Required, core:messageIdAttrType]—The message identifier. See [SCTE130-2] for additional information.

@version [Required, core:versionAttrType]—The ADM message interface version. See Section 9.1 for additional information.

@identity [Required, core:identityAttrType]—The origin logical service identifier. See [SCTE130-2] for additional information.

@system [Optional, core:systemAttrType]—The message origin identifier. See [SCTE130-2] for additional information.

@messageRef [Required, core:messageRefAttrType]—A reference to the paired ADSRegistrationRequest message element initiating this message exchange. The value ***shall*** be the ADSRegistrationRequest message's @messageId attribute value. See [SCTE130-2] for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:InitiatorData [Optional]—The core:InitiatorData element ***shall*** be an exact copy of the ADSRegistrationRequest message's core:InitiatorData element and ***shall*** only be present if found in the paired request message. See [SCTE130-2] for additional information.

core:StatusCode [Required]—An applicable status code specific to the ADSRegistrationRequest message processing. See [SCTE130-2] and Appendix A for additional information.

If the core:StatusCode element indicates an error, no registration for any placement service (i.e., for any ServiceDescription element) ***shall*** be deemed to have occurred (i.e., no transaction).

core:Callout [Optional]—Zero or more core:Callout elements specifying the ADM message exchange endpoint(s). See [SCTE130-2] for additional information. Zero core:Callout elements *may* be returned for either an error condition or if the ADM message exchange endpoint discovery is occurring via a different mechanism.

A single message exchange endpoint *may* be defined to handle all ADM destined messaging identified in Table 5. If a single endpoint is handling all messaging, there should be only one core:Callout element present in the ADSRegistrationResponse message and the core:Callout element's @message attribute ***shall no***t be used. If an ADM desires to use the same default endpoint as the ListADMServicesResponse message, then the default endpoint ***shall*** be identically repeated in this message and consequently, it applies additionally to all endpoint destination listed in Table 5. The ADM *may* optionally use a different default endpoint for the message exchange endpoints identified in Table 5. In this case, the default endpoints are mutually exclusive as the message exchange endpoint list is disjoint between the ListADMServicesResponse and ADSRegistrationResponse messages.

If independent endpoints are desired, the core:Callout element’s @message attribute ***shall*** be used in each Callout element and it ***shall*** contain a value from Table 5. The core:Callout element’s @message attribute’s value ***shall*** appear exactly as shown in Table 5.

Value	Description
PlacementUpdateNotification	Destination for PlacementUpdateNotification messages.
…	User defined message endpoint outside of the scope of this specification. The string ***shall*** be prefixed with the text “private:”.

Table 5: ADSRegistrationResponse/core:Callout/@Message Attribute Values

If independent message exchange endpoints are desired for only a subset of Table 5 endpoints, the default message exchange endpoint *may* be used in conjunction with one or more additional core:Callout elements containing values from Table 5. All message endpoints listed in Table 5 that are not specifically referenced by an @message attribute ***shall*** be available through the default endpoint (assuming one is supplied).

If no default endpoint is supplied and only a subset of the Table 5 endpoints are provided in the ADSRegistrationResponse message, the unlisted endpoints ***shall*** be discovered by a different, unspecified mechanism which is outside the scope of this specification.

Within the core:Callout element, the ADM *may* provide one or more core:Address elements. The processing rules for these elements are not specified here and are outside the scope of this specification.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

11.3.3 Registration Examples

Example 7 illustrates an ADSRegistrationRequest message and Example 8 illustrates a successful ADSRegistrationResponse message. Example 7 indicates the ADS is handling all placement decisions. The ADSRegistrationRequest message’s single core:Callout element omits the @message attribute indicating all messages are to utilize the same endpoint (i.e., the default endpoint). The ADS provides two Address elements where the first address *may* be interpreted as the primary message exchange endpoint and the second endpoint *may* be utilized as a backup or secondary destination. This endpoint interpretation is outside the scope of this specification.

In Example 7, the ADS decided to identify this registration-established placement service as “All Services” and to utilize PODM version 1.2 for these services. This value is ADS determined and was developed specifically for this example only. This specification’s usage of this string in conjunction with the concept of all placement services ***shall no***t be considered normative when representing said concept.

```
<ADSRegistrationRequest messageId="35EC24A4-9D72-11DB-96CA-005056C00000"
version="1.4" system="ads" identity="ADSLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F5">
  <ServiceDescription>
    <SystemContext/>
    <Service id="All Services"/>
  </ServiceDescription>
  <NameSpaceVersion
            namespace="http://www.scte.org/schemas/130-3/2013/adm/podm"
            version="1.2"/>
  <core:Callout>
    <core:Address type="HTTP">http://192.168.201.7:80/SCTE130Msgs</core:Address>
    <core:Address type="HTTP">http://192.168.208.9:80/SCTE130Msgs</core:Address>
  </core:Callout>
</ADSRegistrationRequest>
```

Example 7

In Example 8, the ADM accepts the registration and responds with a duplicate default endpoint as Example 2. Consequently, all message exchanges are routed through this message exchange endpoint.

```
<ADSRegistrationResponse messageId="35EC24A4-9D72-11DB-96CA-005056C00002"
version="1.4" identity="ADMLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F4" system="adm" messageRef="35EC24A4-9D72-11DB-96CA-
005056C00000">
  <core:StatusCode class="0"/>
  <core:Callout>
    <core:Address
type="HTTP">http://173.35.23.33:80/SCTE130MessageProcessor/default</core:Address>
  </core:Callout>
</ADSRegistrationResponse>
```

Example 8

Example 9 illustrates a registration request message where the ADS registers as a decision provider for nationally scoped, interstitial placement opportunities for two product types. The registration also handles all pre-roll and post-roll placement opportunities with local scope. The ADS registers independent PlacementRequest and PlacementStatusNotification endpoints via the two core:Callout elements. All other message exchanges use the supplied default endpoint. This registration request also indicates the end points will utilize PODM version 1.1.

```
<ADSRegistrationRequest messageId="35EC24A4-9D72-11DB-96CA-005056C00006"
version="1.4" identity="ADSLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F5" system="ads">
  <ServiceDescription>
    <SystemContext/>
    <Service id="Service1" opportunityType="interstitial">
      <ContentProvider providerID="espn.com"/>
      <ProductType>MOD</ProductType>
      <Scope>National</Scope>
    </Service>
    <Service id="Service2" opportunityType="interstitial">
      <ContentProvider providerID="espn.com"/>
      <ProductType>FOD</ProductType>
      <Scope>National</Scope>
    </Service>
    <Service id="Service3" opportunityType="preRoll">
      <Scope>Local</Scope>
    </Service>
    <Service id="Service4" opportunityType="postRoll">
      <Scope>Local</Scope>
    </Service>
    </ServiceDescription>
    <NameSpaceVersion
            namespace="http://www.scte.org/schemas/130-3/2013/adm/podm"
            version="1.1"/>
  <core:Callout message="PlacementRequest">
    <core:Address>adselector1.ads.com</core:Address>
  </core:Callout>
  <core:Callout message="PlacementStatusNotification">
    <core:Address>adstatus1.ads.com</core:Address>
  </core:Callout>
  <!--Default handler for all other messages.-->
  <core:Callout>
    <core:Address>adselectorgeneral.ads.com</core:Address>
  </core:Callout>
</ADSRegistrationRequest>
```

Example 9

Example 10 illustrates the ADM response where the ADM specifies a message exchange endpoint specific for PlacementUpdateNotification messages.

```
<ADSRegistrationResponse messageId="35EC24A4-9D72-11DB-96CA-005056C00112"
version="1.4" identity="ADMLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F4" system="adm" messageRef="5EC24A4-9D72-11DB-96CA-
005056C00006">
  <core:StatusCode class="0"/>
  <core:Callout message="PlacementUpdateNotification">
    <core:Address type="HTTP">http://192.168.3.33:80/SCTE130api/PUN</core:Address>
  </core:Callout>
</ADSRegistrationResponse>
```

Example 10

Example 11 illustrates a registration request which has errantly attempted to register for the same placement services as in Example 9. The response message (Example 12) indicates the registration failure. This is an example of a duplicate registration request where an ADM has decided to reject duplicates. An ADM *may* decide to accept duplicates. This implementation specific behavior is outside the scope of this specification.

```
<ADSRegistrationRequest messageId="35EC24A4-9D72-11DB-96CA-005056C00006"
version="1.4" identity="ADSLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F5" system="ads">
  <ServiceDescription>
    <SystemContext/>
    <Service id="Service3" opportunityType="preRoll">
      <Scope>Local</Scope>
    </Service>
    <Service id="Service4" opportunityType="postRoll">
      <Scope>Local</Scope>
    </Service>
    </ServiceDescription>
    <NameSpaceVersion
          namespace="http://www.scte.org/schemas/130-3/2013/adm/podm"
          version="1.2"/>
  <core:Callout>
    <core:Address>adselector1.ads.com</core:Address>
    <core:Address>adselector2.ads.com</core:Address>
  </core:Callout>
</ADSRegistrationRequest>
```

Example 11

```
<ADSRegistrationResponse messageId="35EC24A4-9D72-11DB-96CA-005056C00008"
version="1.4" identity="ADMLogicalService_1170FEF4-C19B-450E-B624-
```

```
421B23F525F4" system="adm" messageRef="35EC24A4-9D72-11DB-96CA-
005056C00006">
  <core:StatusCode class="1" detail="3">
    <core:Note>Registration overlap.</core:Note>
    <core:Note>The resource is already in use.</core:Note>
  </core:StatusCode>
</ADSRegistrationResponse>
```

Example 12

11.4 List ADS Registrations

The active registrations *may* be viewed using the ListADSRegistrationRequest message. The ADM ***shall*** respond with a ListADSRegistrationResponse message. This message exchange duration defines this service channel lifecycle. Figure 14 illustrates the list ADS registration request and response message exchange and the individual message schema descriptions follow in subsequent sections.

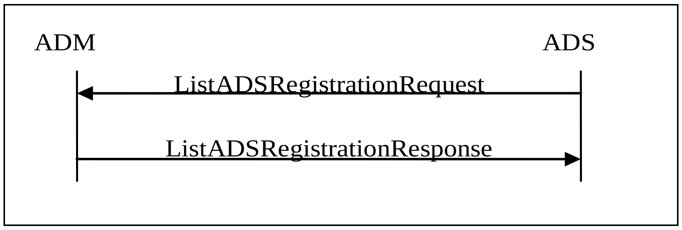


Figure 14: ADM Registration Status Message Exchange

There are two list registration granularity levels:

Per logical service (i.e., via the @identity attribute)

Per registration message

An ADM *may* restrict access to registration information by returning a status code equating to “not authorized” for the list registration request on a per logical service basis. The per registration message granularity level ***shall*** be supported by all compliant ADMs.

The message exchange ***shall*** be conformant with SCTE 130 Part 2 list registration requirements. See [SCTE130-2] for additional information

11.4.1 ListADSRegistrationRequest Message Schema

Figure 15 illustrates the ListADSRegistrationRequest Message Schema.

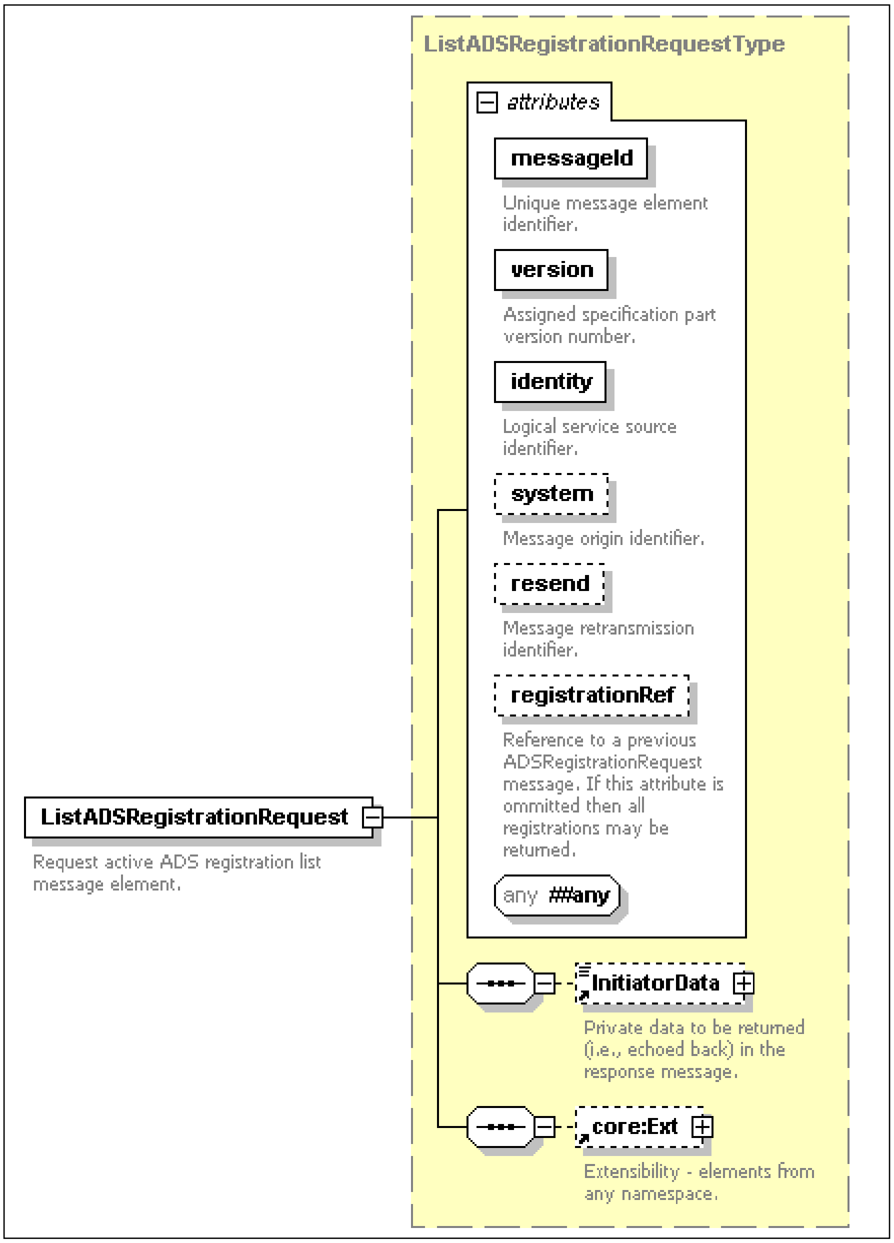

ListADSRegistrationRequestType
attributes
messageId
Unique message element identifier.
version
Assigned specification part version number.
identity
Logical service source identifier.
system
Message origin identifier.
resend
Message retransmission identifier.
registrationRef
Reference to a previous ADSRegistrationRequest message. If this attribute is ommitted then all registrations may be returned.
any ##any
ListADSRegistrationRequest
Request active ADS registration list message element.
InitiatorData
Private data to be returned (i.e., echoed back) in the response message.
core:Ext
Extensibility - elements from any namespace.

Figure 15: ListADSRegistrationRequest Message Schema

The ListADSRegistrationRequest message returns all active registrations associated with a specific logical service source as identified by the @identity attribute when the @registrationRef attribute is omitted. If only a specific registration is to be returned, both the @identity and the @registrationRef attributes ***shall*** be included. The following table defines the inclusion relationships.

@identity	@registrationRef	Description	Access Restrictions Permitted
Included	Omitted	All registrations for matching the supplied @identity value.	Yes
Included	Included	A specific registration matching the @identity and @registrationRef values.	No

Table 6: List ADS Registration Inquiry Granularity Control

The access restrictions permitted column indicates whether the status code equivalent to not authorized *may* optionally be returned.

11.4.1.1 Semantic Definitions for the ListADSRegistrationRequest Message

@messageId [Required, core:messageIdAttrType]—The message identifier. See [SCTE130-2] for additional information.

@version [Required, core:versionAttrType]—The ADM message interface version. See Section 9.1 for additional information.

@identity [Required, core:identityAttrType]—The origin logical service identifier. A logical service *may* use this attribute to identify all related registration-established placement services and their service channels for registration listing. See [SCTE130-2] along with Table 6 for supplementary usage details

@system [Optional, core:systemAttrType]—The message origin identifier. See [SCTE130-2] for additional information.

@resend [Optional, core:resendAttrType]—The message is a retransmit of a previous ListADSRegistrationRequest message. The attribute's value is the original message's @messageId value. See [SCTE130-2] for additional information.

@registrationRef [Optional, core:registrationRefAttrType]—A reference to an original ADSRegistrationRequest message element for which the registration information is to be returned. The value ***shall*** always be the original ADSRegistrationRequest message's @messageId attribute value which in a retransmitted registration message ***shall*** be the @resend attribute's value. See [SCTE130-2] for additional information and refer to Table 6 for supplementary usage details.

@##any [Optional]—Additional attributes from any namespace.

core:InitiatorData [Optional]—The core:InitiatorData element contains implementation specific private data which ***shall*** be returned in the

ListADSRegistrationResponse message. See [SCTE130-2] for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

The ListADSRegistrationRequest element's value *may* be empty.

11.4.2 ListADSRegistrationResponse Message Schema

Figure 16 illustrates the ListADSRegistrationResponse element schema describing the active ADS service channel registrations.

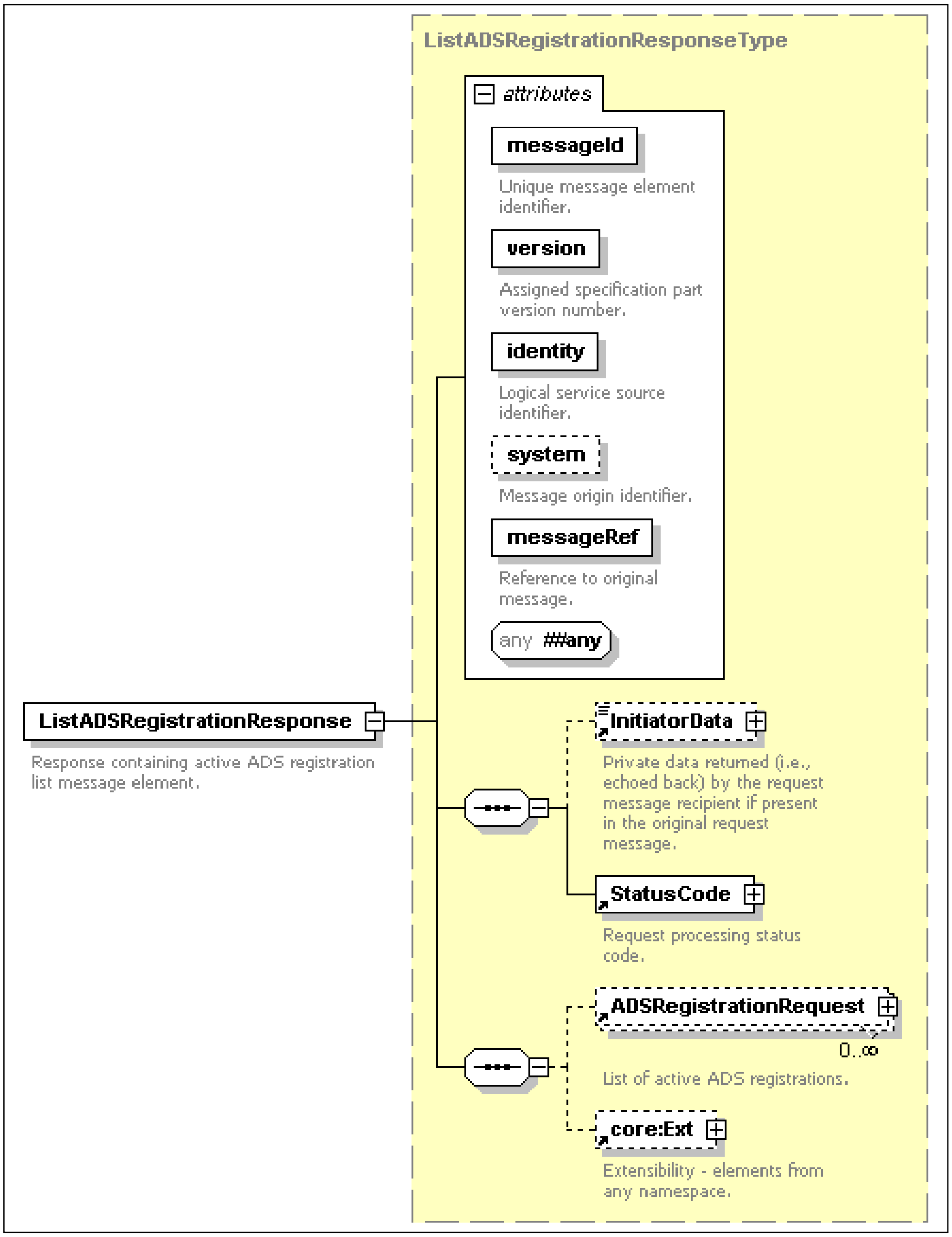

ListADSRegistrationResponseType
attributes
messageId
Unique message element identifier.
version
Assigned specification part version number.
identity
Logical service source identifier.
system
Message origin identifier.
messageRef
Reference to original message.
any ##any
ListADSRegistrationResponse
Response containing active ADS registration list message element.
InitiatorData
Private data returned (i.e., echoed back) by the request message recipient if present in the original request message.
StatusCode
Request processing status code.
ADSRegistrationRequest
0..∞
List of active ADS registrations.
core:Ext
Extensibility - elements from any namespace.

Figure 16: ListADSRegistrationResponse Message Schema

11.4.2.1 Semantic Definitions for the ListADSRegistrationResponse Message

@messageId [Required, core:messageIdAttrType]—The message identifier. See [SCTE130-2] for additional information.

@version [Required, core:versionAttrType]—The ADM message interface version. See Section 9.1 for additional information.

@identity [Required, core:identityAttrType]—The origin logical service identifier. See [SCTE130-2] for additional information.

@system [Optional, core:systemAttrType]—The message origin identifier. See [SCTE130-2] for additional information.

@messageRef [Required, messageRefAttrType]—A reference to the paired ListADSRegistrationRequest message element initiating this message exchange. The value ***shall*** be the ListADSRegistrationRequest message's @messageId attribute value. See [SCTE130-2] for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:InitiatorData [Optional]—The core:InitiatorData element ***shall*** be an exact copy of the ListADSRegistrationRequest message's core:InitiatorData element and ***shall*** only be present if found in the paired request message. See [SCTE130-2] for additional information.

core:StatusCode [Required]—An applicable status code specific to the ListADSRegistrationRequest message processing. See [SCTE130-2] and Appendix A for additional information.

ADSRegistrationRequest [Optional]—Zero or more ADSRegistrationRequest message elements. The ADSRegistrationRequest element ***shall*** be a recoded copy of the accepted registration message. The message element order does not convey any information (e.g., element order does not reflect registration order). See [SCTE130-2] for compliance requirements and Section 11.3.1 for additional details regarding the ADSRegistrationRequest message schema.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

11.4.3 ADS Registration Status Examples

Example 13 and Example 14 illustrates the list all ADS registrations request and response messages respectively when there are no active ADS registrations.

```
<ListADSRegistrationRequest messageId="29D61093-6E81-4A5B-A1F6-
8544AA7F29A0" version="1.4" identity="ADSLogicalService_1170FEF4-C19B-450E-
B624-421B23F525F5" system="ads"/>
```

Example 13

```
<ListADSRegistrationResponse messageId="29D61093-6E81-4A5B-A1F6-
8544AA7F29A5" version="1.4" identity="ADMLogicalService_1170FEF4-C19B-450E-
B624-421B23F525F4" system="adm" messageRef="29D61093-6E81-4A5B-A1F6-
8544AA7F29A0">
  <core:StatusCode class="0">
    <core:Note>There are currently no ADS registrations.</core:Note>
  </core:StatusCode>
</ListADSRegistrationResponse>
```

Example 14

Example 15 assumes Example 13’s list all request message was issued. The returned data includes the ADS Registration section’s Example 9 and Example 7 messages assuming the two messages are the only active registrations. The included element order carries no meaning.

```
<ListADSRegistrationResponse messageId="29D61093-6E81-4A5B-A1F6-
8544AA7F29A5" version="1.4" identity="ADMLogicalService_1170FEF4-C19B-450E-
```

```
B624-421B23F525F4" system="adm" messageRef="29D61093-6E81-4A5B-A1F6-
8544AA7F29A0">
  <core:StatusCode class="0"/>
  <ADSRegistrationRequest messageId="35EC24A4-9D72-11DB-96CA-005056C00006"
version="1.4" identity="ADSLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F5" system="ads">
    <ServiceDescription>
      <SystemContext/>
      <Service id="Service1" opportunityType="interstitial">
        <ContentProvider providerID="espn.com"/>
        <ProductType>MOD</ProductType>
        <Scope>National</Scope>
      </Service>
      <Service id="Service2" opportunityType="interstitial">
        <ContentProvider providerID="espn.com"/>
        <ProductType>FOD</ProductType>
        <Scope>National</Scope>
      </Service>
      <Service id="Service3" opportunityType="preRoll">
        <Scope>Local</Scope>
      </Service>
      <Service id="Service4" opportunityType="postRoll">
        <Scope>Local</Scope>
      </Service>
      </ServiceDescription>
      <NameSpaceVersion
        namespace="http://www.scte.org/schemas/130-3/2013/adm/podm"
        version="1.2"/>
    <core:Callout message="PlacementRequest">
      <core:Address>adselector1.ads.com</core:Address>
    </core:Callout>
    <core:Callout message="PlacementStatusNotification">
      <core:Address>adstatus1.ads.com</core:Address>
    </core:Callout>
    <!--Default handler for all other messages.-->
    <core:Callout>
      <core:Address>adselectorgeneral.ads.com</core:Address>
    </core:Callout>
  </ADSRegistrationRequest>
  <ADSRegistrationRequest messageId="35EC24A4-9D72-11DB-96CA-005056C00000"
version="1.4" system="ads" identity="ADSLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F5">
    <ServiceDescription>
      <SystemContext/>
      <Service id="All Services"/>
    </ServiceDescription>
    <core:Callout>
      <core:Address type="HTTP">http://192.168.201.7:80/SCTE130Msgs</core:Address>
      <core:Address type="HTTP">http://192.168.208.9:80/SCTE130Msgs</core:Address>
    </core:Callout>
  </ADSRegistrationRequest>
</ListADSRegistrationResponse>
```

Example 15

Example 16 illustrates a list request message for a single registration message. The example requests the ADS Registration section’s Example 7 registration. The response message provided in Example 17 contains the ADS Registration section’s Example 7 ADSRegistrationRequest message.

```
<ListADSRegistrationRequest messageId="29D61093-6E81-4A5B-A1F6-
8544AA7F29A1" version="1.4" system="ads" identity="ADSLogicalService_1170FEF4-
C19B-450E-B624-421B23F525F5" registrationRef="35EC24A4-9D72-11DB-96CA-
005056C00000"/>
```

Example 16

```
<ListADSRegistrationResponse messageId="29D61093-6E81-4A5B-A1F6-
8544AA7F29A5" version="1.4" identity="ADMLogicalService_1170FEF4-C19B-450E-
B624-421B23F525F4" system="adm" messageRef="29D61093-6E81-4A5B-A1F6-
8544AA7F29A1">
  <core:StatusCode class="0"/>
  <ADSRegistrationRequest messageId="35EC24A4-9D72-11DB-96CA-005056C00000"
version="1.4" system="ads" identity="ADSLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F5">
    <ServiceDescription>
      <SystemContext/>
      <Service id="All Services"/>
      </ServiceDescription>
      <NameSpaceVersion
        namespace="http://www.scte.org/schemas/130-3/2013/adm/podm"
        version=“1.1”/>
    <core:Callout>
      <core:Address type="HTTP">http://192.168.201.7:80/SCTE130Msgs</core:Address>
      <core:Address type="HTTP">http://192.168.208.9:80/SCTE130Msgs</core:Address>
    </core:Callout>
  </ADSRegistrationRequest>
</ListADSRegistrationResponse>
```

Example 17

Example 18 illustrates a list request message for a specific logical service source using the @identity attribute. This ADM implementation has chosen to return a not authorized response as demonstrated in Example 19. An ADM implementation *may* accept this request and then return the appropriate data. This implementation specific behavior is outside the scope of this specification.

```
<ListADSRegistrationRequest messageId="29D61093-6E81-4A5B-A1F6-
8544AA7F29A3" version="1.4" system="ads" identity="ADSLogicalService_1170FEF4-
C19B-450E-B624-421B23F525F5"/>
```

Example 18

```
<ListADSRegistrationResponse messageId="29D61093-6E81-4A5B-A1F6-
8544AA7F29A4" version="1.4" identity="ADMLogicalService" system="adm"
messageRef="29D61093-6E81-4A5B-A1F6-8544AA7F29A3">
  <core:StatusCode class="1" detail="13">
    <core:Note>Not authorized.</core:Note>
    <core:Note>The ADM does not allow list functionality using the identity attribute
only.</core:Note>
  </core:StatusCode>
</ListADSRegistrationResponse>
```

Example 19

11.5 Deregistration

An ADS *may* cancel an active registration (i.e., terminate an active registration-established service channel) using the ADSDeregisterRequest message. The ADM ***shall*** acknowledge the request by sending an ADSDeregisterResponse message.

If an ADM terminates a registration for any reason other than an ADSDeregisterRequest message, it should send an ADSDeregistrationNotification message indicating the end of the registration-established service channel. The ADS ***shall*** confirm receipt of this notification message by issuing an ADSDeregistrationAcknowledgement message.

A peer logical service ***shall no***t initiate a deregistration sequence for the same registration-established service channel after receiving a deregistration (i.e., for the same registration-established service channel, the ADM ***shall no***t send an ADSDeregistrationNotification message after the receipt of an ADSDeregisterRequest nor ***shall*** the ADS send a ADSDeregisterRequest message after receiving a ADSDeregistrationNotification message). Note: This restriction does not prevent a possible simultaneous deregistration for the same registration-established service channel.

Figure 17 illustrates this message exchange followed by the schema descriptions.

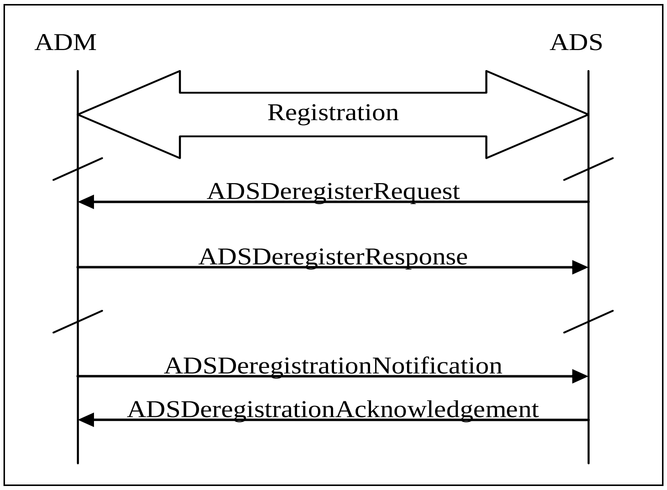

ADM
ADS
Registration
ADSDeregisterRequest
ADSDeregisterResponse
ADSDeregistrationNotification
ADSDeregistrationAcknowledgement

Figure 17: Deregistration Message Exchange

There are two ADS or ADM initiated deregistration granularity levels:

- Per logical service (i.e., via the @identity attribute)
- Per registration message

An ADM *may* restrict deregistration access to a per registration message only by returning a status code equating to “not authorized” for a deregistration request on a per logical service basis. All implementations ***shall*** support the per registration message granularity level.

Deregistration cancels all active PlacementRequest, PlacementResponse, and PlacementUpdateNotification messages and any forthcoming or undelivered PlacementStatusNotification messages.

When an ADS initiates a deregistration message exchange, the ADS ***shall*** be prepared to receive and process any ADM originated messages on the deregistering service channel until it receives the ADSDeregisterResponse message. During the time between the ADSDeregisterRequest and the ADSDeregisterResponse messages, the ADM *may* initiate any message exchanges it deems necessary in order to close the service channel. For example, the ADM *may* deliver pending PlacementStatusNotification messages. SCTE 130 Part 7 [SCTE130-7] *may* specify a maximum deregister message processing timeout value and any other similar values.

If an ADM initiates the deregistration sequence via the ADSDeregistrationNotification message, the ADS ***shall no***t initiate any message exchanges associated with this deregistering service channel as identified in Table 2 with the “Source” column containing the value ADS and the “Registration Required” column being checked following receipt of the ADSDeregistrationNotification message. The ADS ***shall*** return the ADSDeregistrationAcknowledgement message when processing is complete.

The deregistration message schemas are described in the following sections.

11.5.1 ADSDeregisterRequest Message Schema

Figure 18 illustrates the ADSDeregisterRequest element schema.

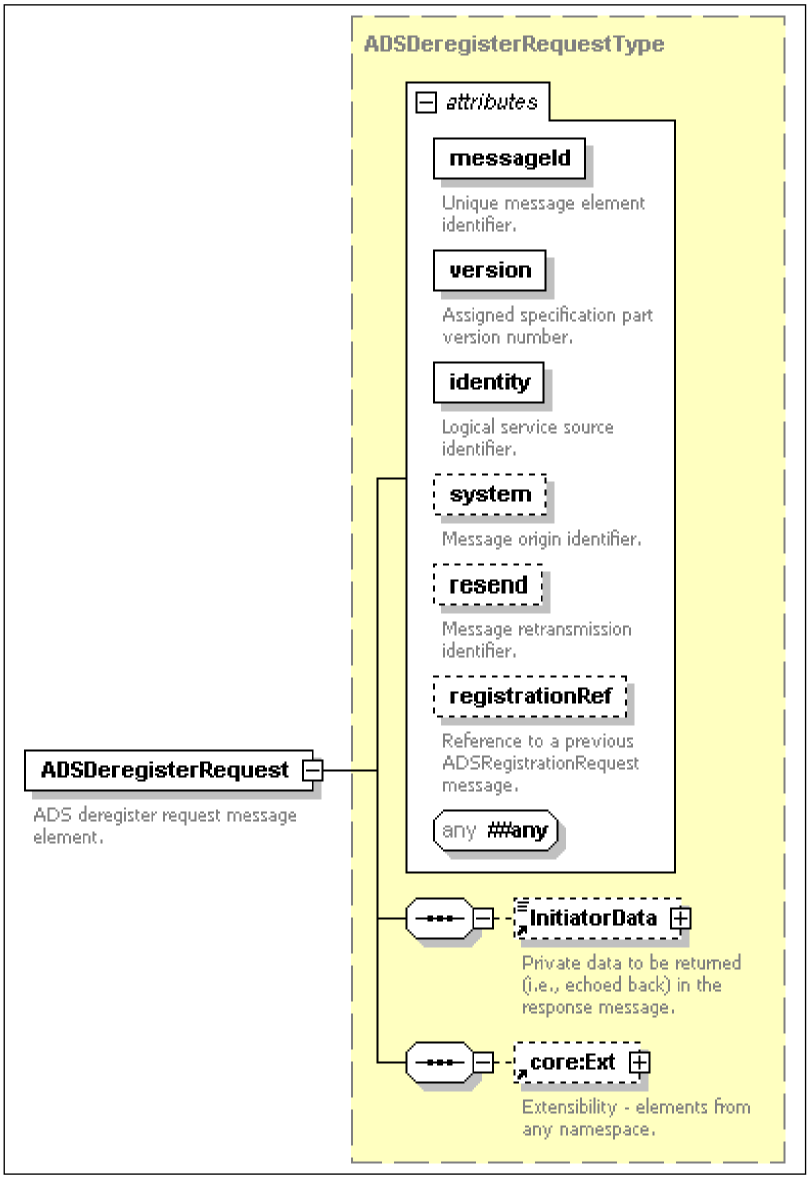

ADSDeregisterRequestType
attributes
messageId
Unique message element identifier.
version
Assigned specification part version number.
identity
Logical service source identifier.
system
Message origin identifier.
resend
Message retransmission identifier.
registrationRef
Reference to a previous ADSRegistrationRequest message.
any ##any
ADSDeregisterRequest
ADS deregister request message element.
InitiatorData
Private data to be returned (i.e., echoed back) in the response message.
core:Ext
Extensibility - elements from any namespace.

Figure 18: ADSDeregisterRequest Message Schema

The ADSDeregisterRequest message terminates all current registrations for a specific logical service source when only the @identity attribute is included and the @registrationRef attribute is omitted. If only a specific registration is to be terminated, both the @identity and the @registrationRef attributes ***shall*** be included. The following table illustrates the defined inclusion relationships.

@identity	@registrationRef	Description	Access Restrictions Permitted
Included	Omitted	All registrations matching the supplied @identity value.	Yes
Included	Included	A specific registration matching the @identity and @registrationRef values.	No

Table 7: ADS Deregistration Granularity Control

The access restrictions permitted column indicates whether the status code equivalent to “not authorized” *may* optionally be returned.

11.5.1.1 Semantic Definitions for the ADSDeregisterRequest Message

@messageId [Required, core:messageIdAttrType]—The message identifier. See [SCTE130-2] for additional information.

@version [Required, core:versionAttrType]—The ADM message interface version. See Section 9.1 for additional information

@identity [Required, core:identityAttrType]—The origin logical service identifier. A logical service *may* use this attribute to delete all related registration-established placement services and their service channels. See [SCTE130-2] along with Table 7 for additional information.

@system [Optional, core:systemAttrType]—The message origin identifier. See [SCTE130-2] for additional information.

@resend [Optional, core:resendAttrType]—The message is a retransmit of a previous ADSDeregisterRequest message. The attribute’s value is the original message’s @messageId value. See [SCTE130-2] for additional information.

@registrationRef [Optional, core:registrationRefAttrType]—A reference to an original ADSRegistrationRequest message element for which the sender is now deregistering the service channel. The value ***shall*** always be the original ADSRegistrationRequest message’s @messageId attribute value which in a retransmitted registration message ***shall*** be the @resend attribute’s value. See [SCTE130-2] for additional information and refer to Table 7 for supplementary usage details.

@##any [Optional]—Additional attributes from any namespace.

core:InitiatorData [Optional]—The core:InitiatorData element contains implementation specific private data which ***shall*** be returned in the ADSDeregisterResponse message. See [SCTE130-2] for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

The ADSDeregisterRequest element's value *may* be empty.

11.5.2 ADSDeregisterResponse Message Schema

Figure 19 illustrates the ADSDeregisterResponse element schema.

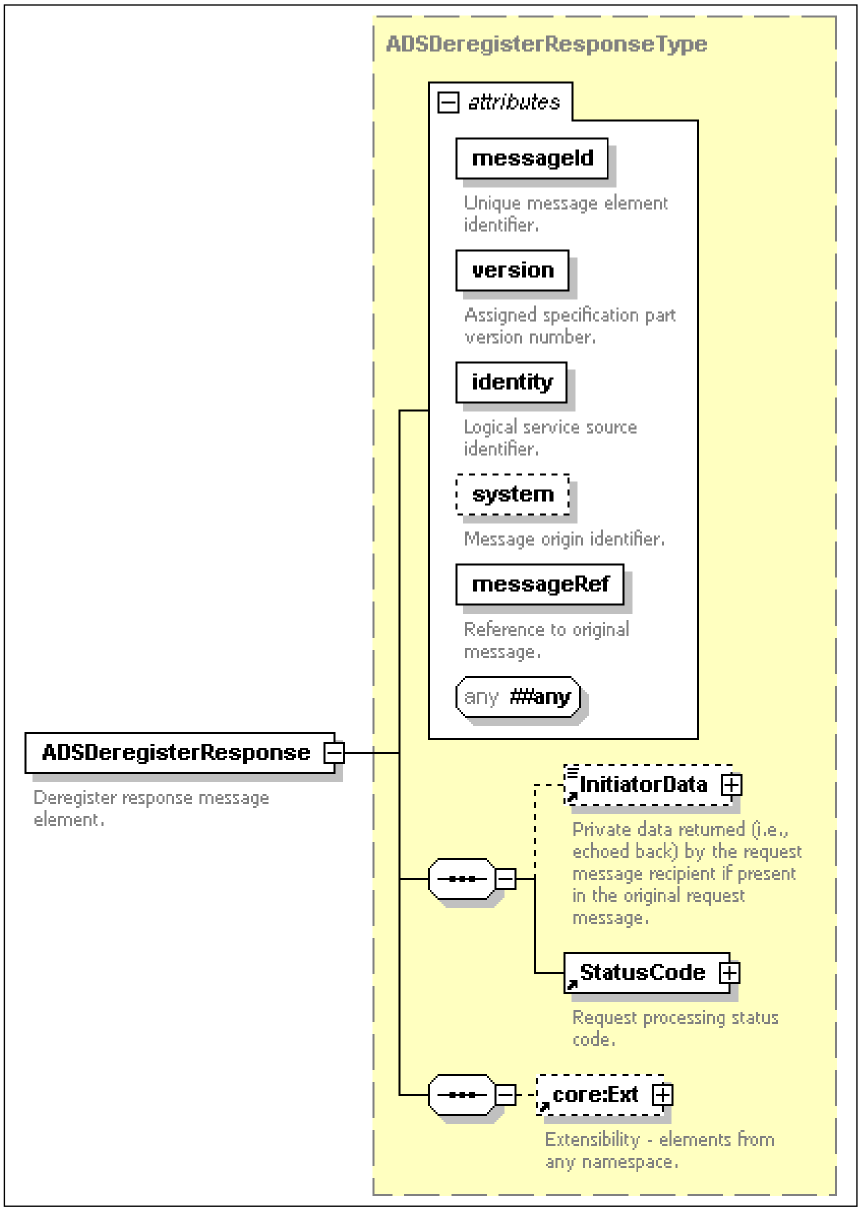

ADSDeregisterResponseType
attributes
messageId
Unique message element identifier.
version
Assigned specification part version number.
identity
Logical service source identifier.
system
Message origin identifier.
messageRef
Reference to original message.
any ##any
ADSDeregisterResponse
Deregister response message element.
InitiatorData
Private data returned (i.e., echoed back) by the request message recipient if present in the original request message.
StatusCode
Request processing status code.
core:Ext
Extensibility - elements from any namespace.

Figure 19: ADSDeregisterResponse Message Schema

11.5.2.1 Semantic Definitions for the ADSDeregisterResponse Message

@messageId [Required, core:messageIdAttrType]—The message identifier. See [SCTE130-2] for additional information.

@version [Required, core:versionAttrType]—The ADM message interface version. See Section 9.1 for additional information.

@identity [Required, core:identityAttrType]—The origin logical service identifier. See [SCTE130-2] for additional information.

@system [Optional, core:systemAttrType]—The message origin identifier. See [SCTE130-2] for additional information.

@messageRef [Required, core:messageRefAttrType]—A reference to the paired ADSDeregisterRequest message element initiating this message exchange. The value ***shall*** be the ADSDeregisterRequest message's @messageId attribute value. See [SCTE130-2] for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:InitiatorData [Optional]—The core:InitiatorData element ***shall*** be an exact copy of the ADSDeregisterRequest message's core:InitiatorData element and ***shall*** only be present if found in the paired request message. See [SCTE130-2] for additional information.

core:StatusCode [Required]—An applicable status code specific to the ADSDeregisterRequest message processing. See [SCTE130-2] and Appendix A for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

11.5.3 ADS Deregistration Request/Response Examples

Example 20 illustrates an attempt to deregister a logical service source via the @identity attribute. Example 21 shows the ADM rejecting the request from Example 20. This ADM implementation has chosen to return a not authorized response. An ADM implementation *may* accept this request and subsequently terminate all registrations for the specified @identity attribute. This implementation specific behavior is outside the scope of this specification.

```
<ADSDeregisterRequest messageId="8F34DD15-3A3C-4F2C-923A-DE7DA3665FF0"
version="1.4" identity="ADSLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F5" system="ads"/>
```

Example 20

```
<ADSDeregisterResponse messageId="8F34DD15-3A3C-4F2C-923A-DE7DA3665FF5"
version="1.4" identity="ADMLogicalService_1170FEF4-C19B-450E-B624-
```

```
421B23F525F4" system="adm" messageRef="8F34DD15-3A3C-4F2C-923A-
DE7DA3665FF0">
  <core:StatusCode class="1" detail="13">
    <core:Note>Not authorized. The ADM does not allow deregistration by the identity
attribute only.</core:Note>
  </core:StatusCode>
</ADSDeregisterResponse>
```

Example 21

Example 22 illustrates a successful deregistration of a specific active ADSRegistrationRequest message. Example 23 shows the response message.

```
<ADSDeregisterRequest messageId="8F34DD15-3A3C-4F2C-923A-DE7DA36655F0"
version="1.4" identity="ADSLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F5" system="ads" registrationRef="35EC24A4-9D72-11DB-96CA-
005056C0000"/>
```

Example 22

```
<ADSDeregisterResponse messageId="8F34DD15-3A3C-4F2C-923A-DE7DA3665FF5"
version="1.4" identity="ADMLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F4" system="adm" messageRef="8F34DD15-3A3C-4F2C-923A-
DE7DA3665FF0">
  <core:StatusCode class="0"/>
</ADSDeregisterResponse>
```

Example 23

Example 24 illustrates an unsuccessful attempt to deregister a logical service source via the @identity attribute only. Example 25 shows the response message where the ADM allows the action but was unable to locate any matching registrations.

```
<ADSDeregisterRequest messageId="8F34DD15-3A3C-4F2C-923A-DE7DA3665EF0"
version="1.4" identity="BigGun" system="ads"/>
```

Example 24

```
<ADSDeregisterResponse messageId="8F34DD15-3A3C-4F2C-923A-DE7DA3665FF5"
version="1.4" identity="ADMLogicalService_1170FEF4-C19B-450E-B624-
```

```
421B23F525F4" system="adm" messageRef="8F34DD15-3A3C-4F2C-923A-
DE7DA3665EF0">
  <core:StatusCode class="1" detail="5">
    <core:Note>Ambiguous details.</core:Note>
    <core:Note> Failed to locate any active registrations with the
identity="BigGun".</core:Note>
  </core:StatusCode>
</ADSDeregisterResponse>
```

Example 25

11.5.4 ADSDeregistrationNotification Message Schema

Figure 20 illustrates the ADSDeregistrationNotification element’s schema.

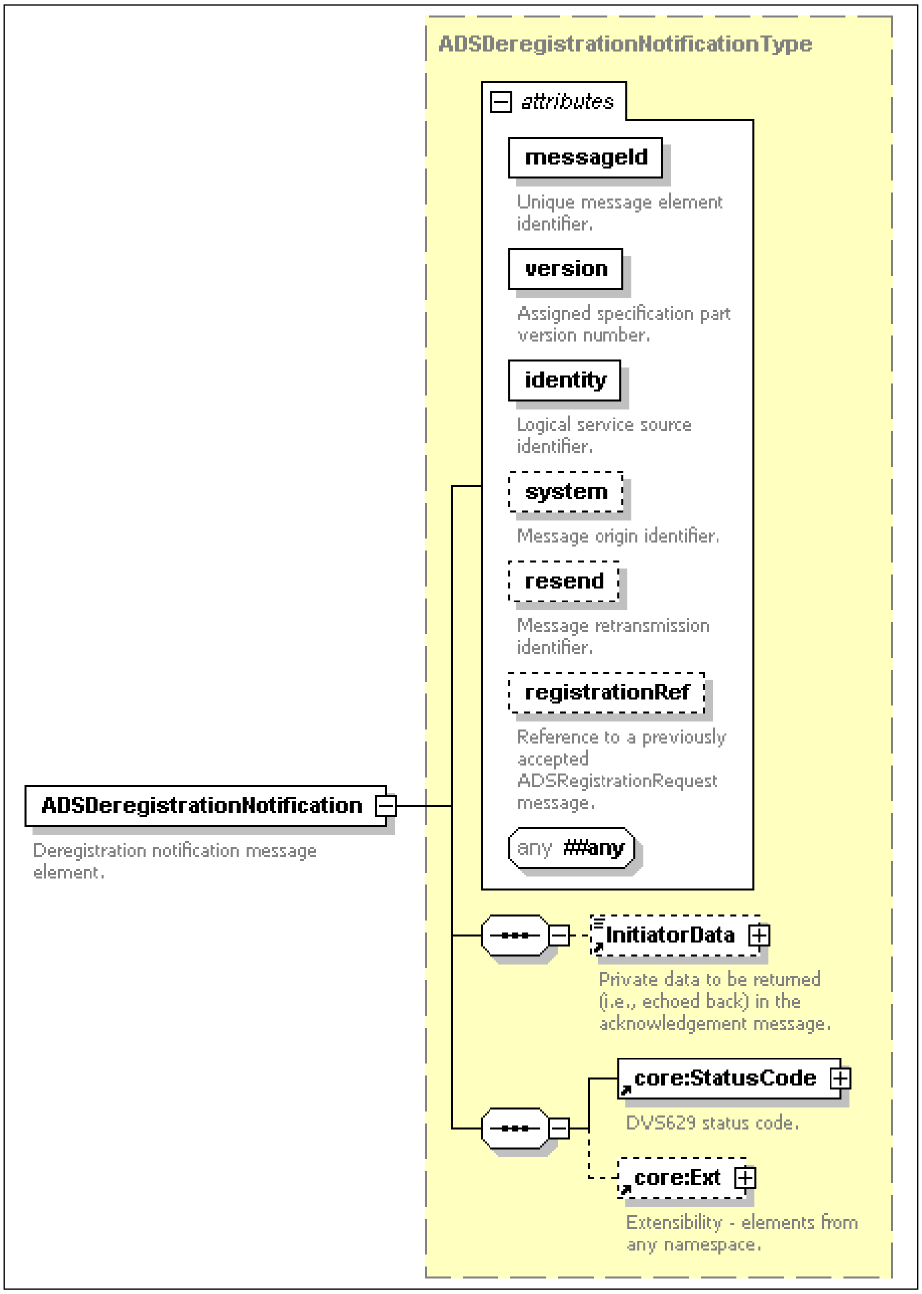

ADSDeregistrationNotificationType
attributes
messageId
Unique message element identifier.
version
Assigned specification part version number.
identity
Logical service source identifier.
system
Message origin identifier.
resend
Message retransmission identifier.
registrationRef
Reference to a previously accepted ADSRegistrationRequest message.
any ##any
ADSDeregistrationNotification
Deregistration notification message element.
InitiatorData
Private data to be returned (i.e., echoed back) in the acknowledgement message.
core:StatusCode
DVS629 status code.
core:Ext
Extensibility - elements from any namespace.

Figure 20: ADSDeregistrationNotification Message Schema

11.5.4.1 Semantic Definitions for the ADSDeregistrationNotification Message

@messageId [Required, core:messageIdAttrType]—The message identifier. See [SCTE130-2] for additional information.

@version [Required, core:versionAttrType]—The ADM message interface version. See Section 9.1 for additional information.

@identity [Required, core:identityAttrType]—The origin logical service identifier. See [SCTE130-2] for additional information.

@system [Optional, core:systemAttrType]—The message origin identifier. See [SCTE130-2] for additional information.

@resend [Optional, core:resendAttrType]—The message is a retransmit of a previous ADSDeregistrationNotification message. The attribute's value is the original message's @messageId value. See [SCTE130-2] for additional information.

@registrationRef [Optional, core:registrationRefAttrType]—A reference to an original ADSRegistrationRequest message element for which the ADM is providing this deregistration notice. The value ***shall*** always be the original ADSRegistrationRequest message's @messageId attribute value which in a retransmitted registration message ***shall*** be the @resend attribute's value. See [SCTE130-2] for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:InitiatorData [Optional]—The core:InitiatorData element contains implementation specific private data which ***shall*** be returned in the ADSDeregistrationAcknowledgement message. See [SCTE130-2] for additional information.

core:StatusCode [Required]—An applicable status code specific to the ADSDeregistrationNotification message's purpose. See [SCTE130-2] and Appendix A for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

11.5.5 ADSDeregistrationAcknowledgement Message Schema

Figure 21 illustrates the ADSDeregistrationAcknowledgement element's schema.

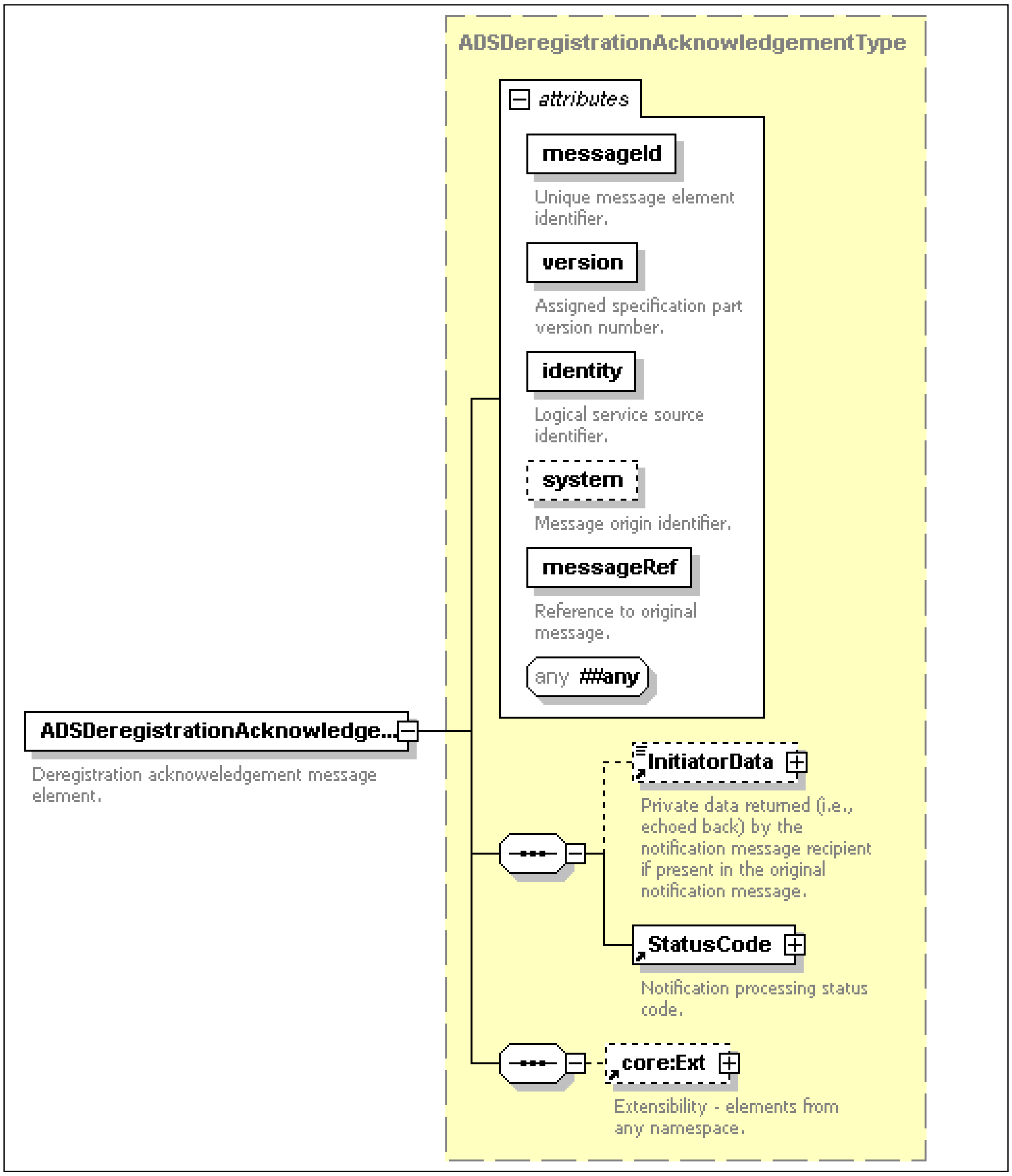

ADSDeregistrationAcknowledgementType
attributes
messageId
Unique message element identifier.
version
Assigned specification part version number.
identity
Logical service source identifier.
system
Message origin identifier.
messageRef
Reference to original message.
any ##any
ADSDeregistrationAcknowledge...
Deregistration acknoweledgement message element.
InitiatorData
Private data returned (i.e., echoed back) by the notification message recipient if present in the original notification message.
StatusCode
Notification processing status code.
core:Ext
Extensibility - elements from any namespace.

Figure 21: ADSDeregistrationAcknowledgement Message Schema

11.5.5.1 Semantic Definitions for the ADSDeregistrationAcknowledgement Message

@messageId [Required, core:messageIdAttrType]—The message identifier. See [SCTE130-2] for additional information.

@version [Required, core:versionAttrType]—The ADM message interface version. See Section 9.1 for additional information.

@identity [Required, core:identityAttrType]—The origin logical service identifier. See [SCTE130-2] for additional information.

@system [Optional, core:systemAttrType]—The message origin identifier. See [SCTE130-2] for additional information.

@messageRef [Required, core:messageRefAttrType]—A reference to the paired ADSDeregistrationNotification message element initiating this message exchange. The value ***shall*** be the ADSDeregistrationNotification message's @messageId attribute value. See [SCTE130-2] for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:InitiatorData [Optional]—The core:InitiatorData element ***shall*** be an exact copy of the ADSDeregistrationNotification message's core:InitiatorData element and ***shall*** only be present if found in the paired request message. See [SCTE130-2] for additional information.

core:StatusCode [Required]—An applicable status code specific to the ADSDeregistrationAcknowledgement message processing. See [SCTE130-2] and Appendix A for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

11.5.6 ADS Deregistration Notification/Acknowledgment Examples

Example 26 illustrates a successful message exchange when an ADM logical service is being taken out of service. The ADM is notifying the ADS of the deregistration. Example 26 is the notification message and Example 27 is the acknowledgement message.

```
<ADSDeregistrationNotification messageId="8F34DD15-3A3C-4F2C-923A-
DE7DA3665FF5" version="1.4" identity="ADMLogicalService_1170FEF4-C19B-450E-
B624-421B23F525F4" system="adm">
  <core:StatusCode class="3">
    <core:Note>The ADM is terminating the registration as it is being taken out of
service.</core:Note>
  </core:StatusCode>
</ADSDeregistrationNotification>
```

Example 26

```
<ADSDeregistrationAcknowledgement messageId="8F34DD15-3A3C-4F2C-923A-
DE7DA3665FF0" version="1.4" identity="ADSLogicalService_1170FEF4-C19B-450E-
```

```
B624-421B23F525F5" system="ads" messageRef="8F34DD15-3A3C-4F2C-923A-
DE7DA3665FF5">
  <core:StatusCode class="0"/>
</ADSDeregistrationAcknowledgement>
```

Example 27

11.6 Service Check Support

The ADM and ADS ***shall*** both support the ServiceCheck message exchange, which includes the core:ServiceCheckRequest and core:ServiceCheckResponse messages, as defined by SCTE 130 Part 2. See [SCTE130-2] for additional information.

11.7 Service Status Support

The ADM and ADS ***shall*** both support the ServiceStatus message exchange, which includes the core:ServiceStatusNotification and core:ServiceStatusAcknowledgement messages, as defined by SCTE 130 Part 2. See [SCTE130-2] for additional information.

12.0 ADM INTERFACE—PLACEMENT OPERATIONS MESSAGES

Messages communicating placement opportunities, placement decisions, placement updates, and placement event reporting are collectively referred to as placement operations. Placement operations assume a registration-established service channel has been successfully completed. See Section 11.3 for details regarding registration.

Placement messaging refers to the specific exchange of a PlacementRequest message supplying placement opportunity descriptions and a PlacementResponse message containing placement decisions. Placement update messaging encompasses placement decision updating along with placement abandonment signaling (i.e., cancellation). Placement event reporting is accomplished via the placement status message exchanges.

The following sections detail the individual placement operations message exchanges.

12.1 Placement Messaging

Placement messaging refers to an ADM sent PlacementRequest message to which an ADS responds with a PlacementResponse message. The PlacementRequest message communicates placement opportunities and ***shall*** be sent to the ADS supplied callback address included in the ADSRegistrationRequest message if specified.

Placement messaging ***shall*** be initiated by an ADM. An ADM determines the presence of an ad placement opportunity and communicates information such as specific targeting criteria, entertainment content identification, etc., as part of a PlacementRequest message. The ADS ***shall*** respond with a PlacementResponse message whose contents typically are placement decisions. The placement decisions *may* contain placements where each placement defines an action. Each placement *may* also contain a content binding and a placement constraint set.

Placement opportunity detection and its characteristic determination along with placement decision implementation are all outside the scope of this specification. This specification does not preclude or prefer any method or combination thereof.

Figure 22 illustrates the basic placement message exchange sequence and a high-level overview of the possible subsequent message exchanges.

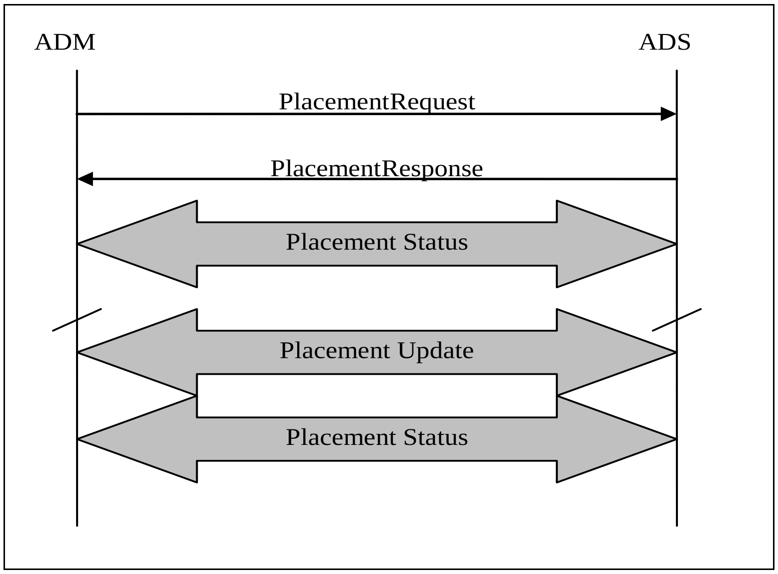


Figure 22: Placement Messaging Message Exchange

A placement opportunity description, implemented as an XML PlacementOpportunity element, is a potentially constrained location relative to digital content where ad insertion or content alterations *may* occur. To a classic linear technologist, a placement opportunity might be called an avail or a (commercial) break. The optional placement opportunity constraints *may* be requirements, limitations, or guidance specific to the placement opportunity. The alterations *may* include insertions, replacements, or deletions of content in whole or in part. Example placement opportunity constraints include minimum, maximum or fixed total ad content durations. In traditional linear broadcast, for example, the placement opportunity typically has a fixed placement opportunity duration constraint such as a total ad insertion time of 30, 60, or 120 seconds.

The following two examples illustrate possible placement opportunity detection mechanisms though many other methods are possible. In a traditional linear broadcast system for example, an SCTE 35 splice_info_section() containing a splice_insert() splice command type *may* signal the placement opportunity start. See [SCTE35] for additional information. Alternatively, in an on-demand system, the SCTE 130 Placement Opportunity Information Service (POIS) *may* be consulted at the entertainment content start (i.e., session setup) for the subscriber specific placement opportunities and their characteristics.

There are five traditional content relative placement opportunity locations as illustrated in Figure 23 and Figure 24.

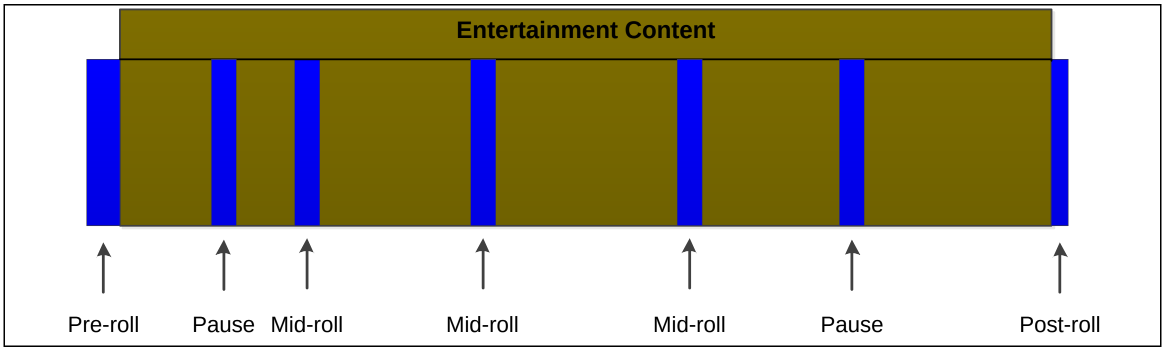


Figure 23: Traditional Placement Opportunity Locations Within An Asset

The pre-roll, mid-roll and post-roll placement opportunities are typically associated with any form of on-demand content including but not limited to video on-demand (VOD), network personal video recorder (nPVR), or time-shifted content. The pre-roll placement opportunity precedes the entertainment content while the post-roll location follows the entertainment content.

A mid-roll placement opportunity occurs within the context of the entertainment content. Traditionally, a mid-roll placement opportunity has been associated with classic linear broadcast. However, this placement opportunity type is not limited to this context.

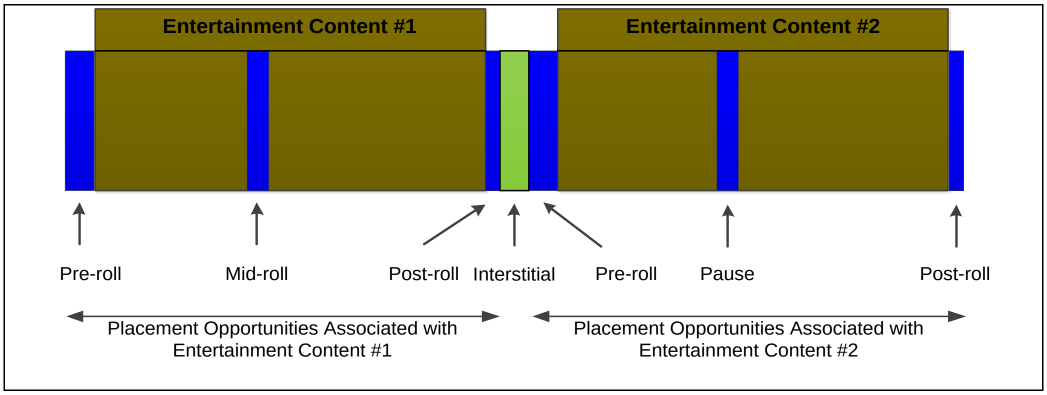

Entertainment Content #1
Entertainment Content #2
Pre-roll
Mid-roll
Post-roll
Interstitial
Pre-roll
Pause
Post-roll
Placement Opportunities Associated with Entertainment Content #1
Placement Opportunities Associated with Entertainment Content #2

Figure 24: Traditional Placement Opportunity Locations Between Assets

An interstitial placement opportunity occurs between entertainment content. Typically, this placement opportunity type is applied in playlist applications where multiple entertainment assets are presented in sequential fashion (e.g., music video playlist). However, this placement opportunity type is not limited to this context.

Typically, pre-roll, mid-roll, post-roll, and interstitial placement opportunity existence is predetermined. A mid-roll placement opportunity is generally planned (or scheduled) though the exact location *may* be determined in real-time (e.g., ad placement in a sporting event).

A pause placement opportunity is generally the result of a subscriber pressing the pause button. The pause placement opportunity, or any trick mode viewer event creating a placement opportunity of that respective type, is an example of a spontaneous placement opportunity. A spontaneous placement opportunity is an unplanned or unscheduled opening for zero or more ad placements to take place. The placement opportunity’s location relative to the entertainment content is variable and inconsistent (though it *may* be somewhat predictable – for example, when a classic linear broadcast interstitial commercial avail is encountered).

Spontaneous placement opportunities are never guaranteed to occur and typically, there is no limit to the number of possible placement opportunities that *may* transpire with respect to the entertainment content. Spontaneous placement opportunities *may* not be possible in some systems. For example, a classic linear broadcast system does not have any notion of a spontaneous placement opportunity. However, a non-live delivery of linear broadcast does allow for spontaneous placement opportunities.

Both planned and spontaneous placement opportunity expressions are supported by this specification using the @opportunityType attribute. Support for each placement opportunity type is optional and implementation dependent. Thus, all placement opportunity types *may* be supported and no one placement type ***shall*** be mandated.

A placement opportunity *may* have a placement decision owner. For example, a placement opportunity *may* be owned by a distributor, a provider, or a third party. Placement opportunity ownership determination is outside the scope of this specification. An example mechanism supporting this concept is the SCTE 35 [SCTE35] segmentation_descriptor() which *may* convey a placement opportunity ownership identifier. The supplied identifier value *may* assist in decision making or in selecting the appropriate PlacementRequest message ADS destination by matching a placement service registration.

The ADM ***shall*** deliver the placement opportunity to the appropriate registration-established placement service ADS.

A placement opportunity *may* be subdivided into an unrestricted number of subunits for refinement or constraint purposes. The individual subunit is referred to as a positional placement opportunity and implemented as an XML PlacementControl element. If, when, and how a placement opportunity is subdivided is outside the scope of this specification. The classic linear broadcast technologist *may* equate a positional placement opportunity to a slot. Figure 25 illustrates an example subdivision.

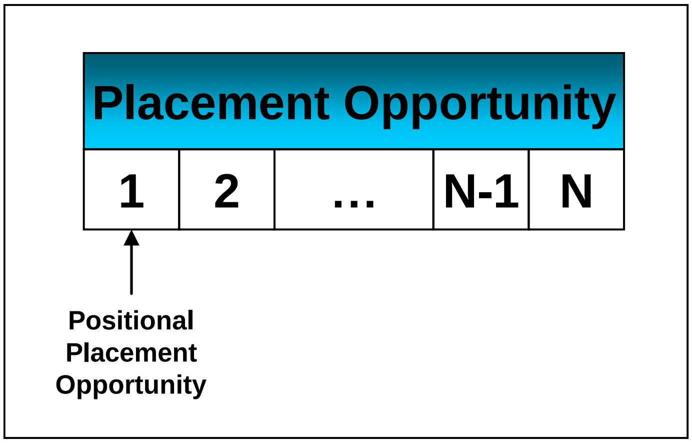


Figure 25: Placement Opportunity Positional Subdivision

Positional subunits are identified by a position number with the first subunit being assigned the value one and each subsequent subunit incrementing the positional value by one. Specific positions with the placement opportunity *may* be identified in the XML document using the @position attribute.

Each positional placement opportunity *may* have a different duration and the granularity of a positional placement opportunity typically maps to the current (or planned) ad asset occupying the temporal location. For example, a single placement opportunity might be subdivided into four positional placement opportunities as illustrated in Figure 26. Each of these subunits might correspond to a 30-second ad with the entire placement opportunity having a two minute duration.

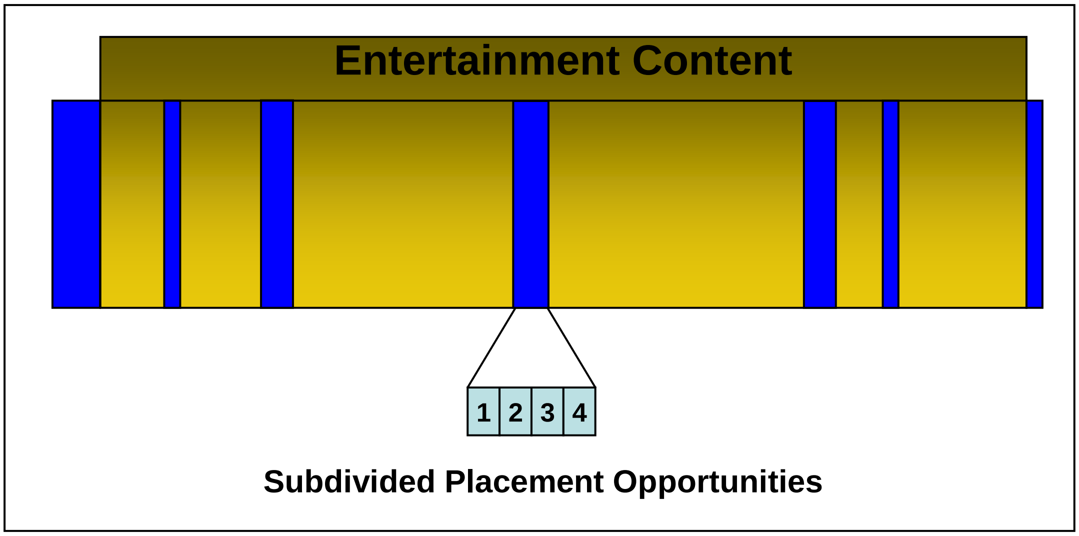


Figure 26: Subdivided Placement Opportunity

The positional placement opportunity provides the capability to identify the content currently occupying the location. The content boundaries, for example, *may* be identified using the SCTE 35 segmentation_descriptor() or some alternate means. See

[SCTE35] for additional information. Determination of currently occupying content and its identification within an entertainment asset are outside the scope of this specification.

The ADS response to a placement opportunity is a placement decision implemented as an XML PlacementDecision element. A placement decision is formalized as zero or more placements. A placement, implemented as an XML Placement element, embodies a decisive action that *may* include a content binding and a constraint set. The placement action denotes a behavior to be performed such as fill (add), replace, delete, or fixed (do nothing). A content binding is a formal assignment of an ad asset to a temporal delivery position and this binding *may* include an ADS assigned tracking identifier. The tracking identifier, implemented as an XML Tracking element, facilitates ad placement reporting. The constraint set defines controls to be additionally applied to the placement. For the classic linear technologist, a placement generally equates to a spot.

Positional placement opportunities and placements represented by the PlacementControl and Placement elements respectively ***shall*** be processed consecutively per their document order (i.e., their XML document position/location top to bottom) except when either the @position attribute or the @placementControlRef attribute indicates a different or non-sequential ordering. Thus, for example, consecutive Placement elements contained in a PlacementDecision element ***shall*** be delivered in the listed document appearance order assuming no @position attribute or other order control. It is recommended that the listed document order always correlate to the expected display sequence for each placement opportunity, positional placement opportunity, placement decision and placement. For example, the first placement opportunity should always come first in the PlacementRequest message, followed by the second placement opportunity, etc. And the PlacementResponse message’s placement decisions should be ordered the same way. The individual placements should be listed in display order in the XML document as well.

Figure 27 illustrates the high-level XML construct relationships contained in the PlacementRequest and the PlacementResponse message. The diagram illustrates the PlacementOpportunity, PlacementControl, PlacementDecision and Placement elements with respect to the entertainment and ad content.

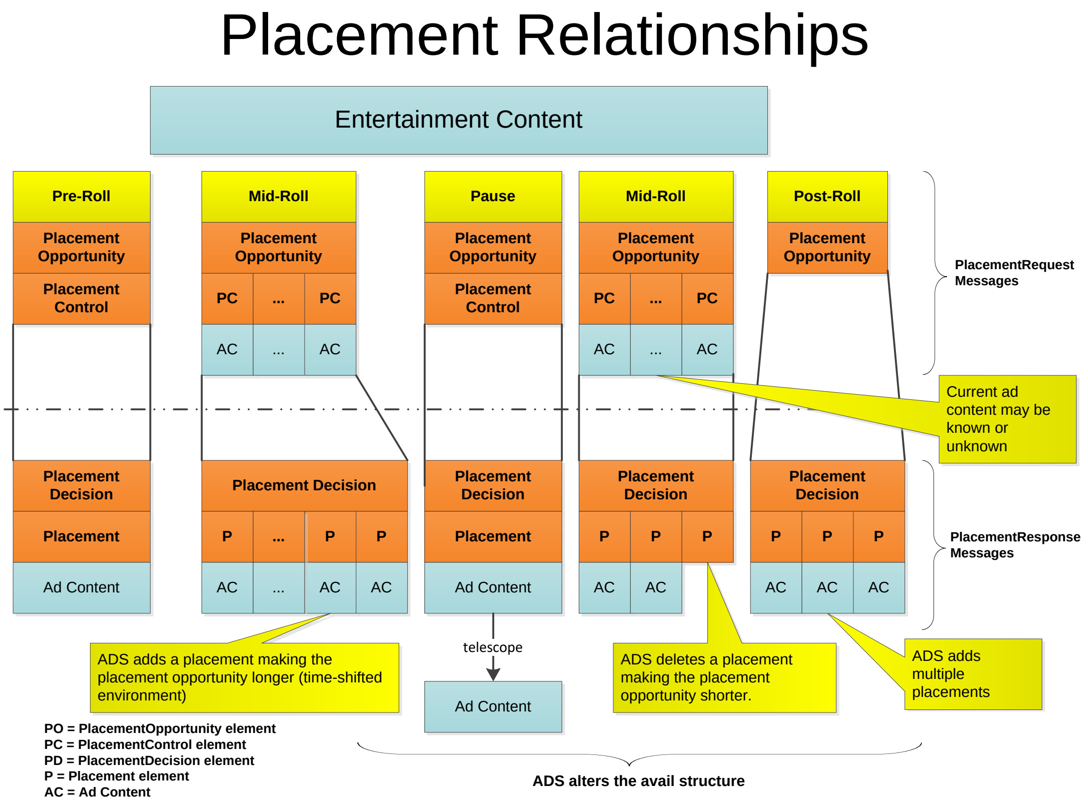


Figure 27: PlacementRequest/PlacementResponse to Content Relationship

The PlacementRequest message transports placement opportunities to an ADS using the PlacementOpportunity elements enabling the ADS to make its placement decisions. Each placement opportunity *may* be subdivided into positional placement opportunities described using the PlacementControl elements. Thus, a placement opportunity *may* contain zero or more PlacementControl elements.

Each placement opportunity within the PlacementRequest message ***shall*** be element unique by having a unique @id attribute. One or more PlacementOpportunity elements *may* be carried within the PlacementRequest message. The choice of how to partition and group placement opportunities are implementation decisions outside the scope of this specification. A wide variety of segmentation possibilities are supported by this specification.

Multiple PlacementRequest messages containing any number of PlacementOpportunity elements, which in turn *may* contain any number of PlacementControl elements, *may* be used to request ADS placement decisions.

An ADM communicates each positional placement opportunity, when known, in a PlacementControl element. Again, the choice of how to partition and cluster the PlacementControl element groups is an implementation decision outside the scope of this specification. Though most segmentation schemes are supported by this specification, it is highly recommended that all PlacementOpportunity and PlacementControl elements for a single placement service registration be consolidated and transported together.

Once the ADS receives a PlacementRequest message, it proceeds to make the appropriate ad placement decisions based on the requirements, guidance, and targeting criteria provided in the ADM PlacementRequest message. The ADS returns its decisions using the PlacementResponse message and the ADS, under certain circumstances, *may* subsequently change the decision using the PlacementUpdateNotification message. See Section 12.2 for additional information.

The PlacementResponse message utilizes a similar design pattern as the PlacementRequest message. Each PlacementDecision element parallels the PlacementOpportunity element and the Placement element functions similarly to the PlacementControl element. The ADS uses the Placement element (rather than the PlacementControl element) to communicate the decided placement behavior, content binding and optional associated Tracking element.

The ADS returned PlacementResponse message (or the PlacementUpdateNotification message – see Section 12.2 for details) *may* alter a placement opportunity structure based upon the Placement element values. National, regional, and local scoped content *may* all be added, replaced, or deleted at any granularity level and it is possible to fully remove either a complete placement opportunity or a positional placement opportunity. When and to what extent an ADS *may* change the placement opportunity structure is outside the scope of this specification.

The Placement element’s @action attribute specifies the placement behavior. A few example @action attribute behavior verbs include: fixed (i.e., no change), fill (insert or add), delete, and replace. See Section 14.26 for the complete list. Each action is specific to a single (positional) placement. A few examples of possible placement decisions are as follows:

- Do nothing (fixed)
- Forms of fill
    - Add additional placements to the placement opportunity growing the total placement duration.
    - Add additional (positional) placements to the placement opportunity without growing the placement duration. (i.e., more ads each with a shorter duration and the total time remains the same as the original placement opportunity duration.)
    - Add additional placements to the placement opportunity and shrink the placement duration (use more but shorter ads).
    - Insert with new placements where none currently exist (i.e., create a new placement opportunity and fill it).

- Delete positional placement opportunities shortening the placement duration. An ADS *may* completely remove all positional placement opportunities, which deletes the entire placement opportunity.
- Replace the ad with another ad of the same or different characteristics, etc.

The ADM originated PlacementRequest message contains the following high-level items:

- An optional system context description containing refinements to the matching registered placement service description. For example, properties unique to a VOD session, zone context, or channel information.
- Optional placement service descriptions containing refinements to the matched registered placement service description. For example, placement ownership refinement.
- An optional entertainment content description for which all the placement opportunities are either surrounding, contained within, or relative to.
- An optional client description that includes placement refinement data (also known as addressability information or targeting criteria). For example, the client's time of day, zip code or terminal address.
- An optional opaque data element, the ADMData element, which ***shall*** be returned by the ADS in the PlacementResponse message and any subsequent PlacementUpdateNotification messages if present in the PlacementRequest message. For example, the opaque data *may* support high resiliency features.
- A list of placement opportunities which each *may* be further subdivided into positional placement opportunities. Each placement opportunity *may* optionally contain an entertainment content description which, if present, is only applicable to the specific placement opportunity. The placement opportunity and the positional placement opportunity descriptions *may* include the desired ad type, characteristics such as min and max duration, min and max placement count, and desired placement scope. The positional placement opportunity *may* additionally include a description of the ad content currently occupying the position including the current ad type and the planned or recommended ADM placement action.

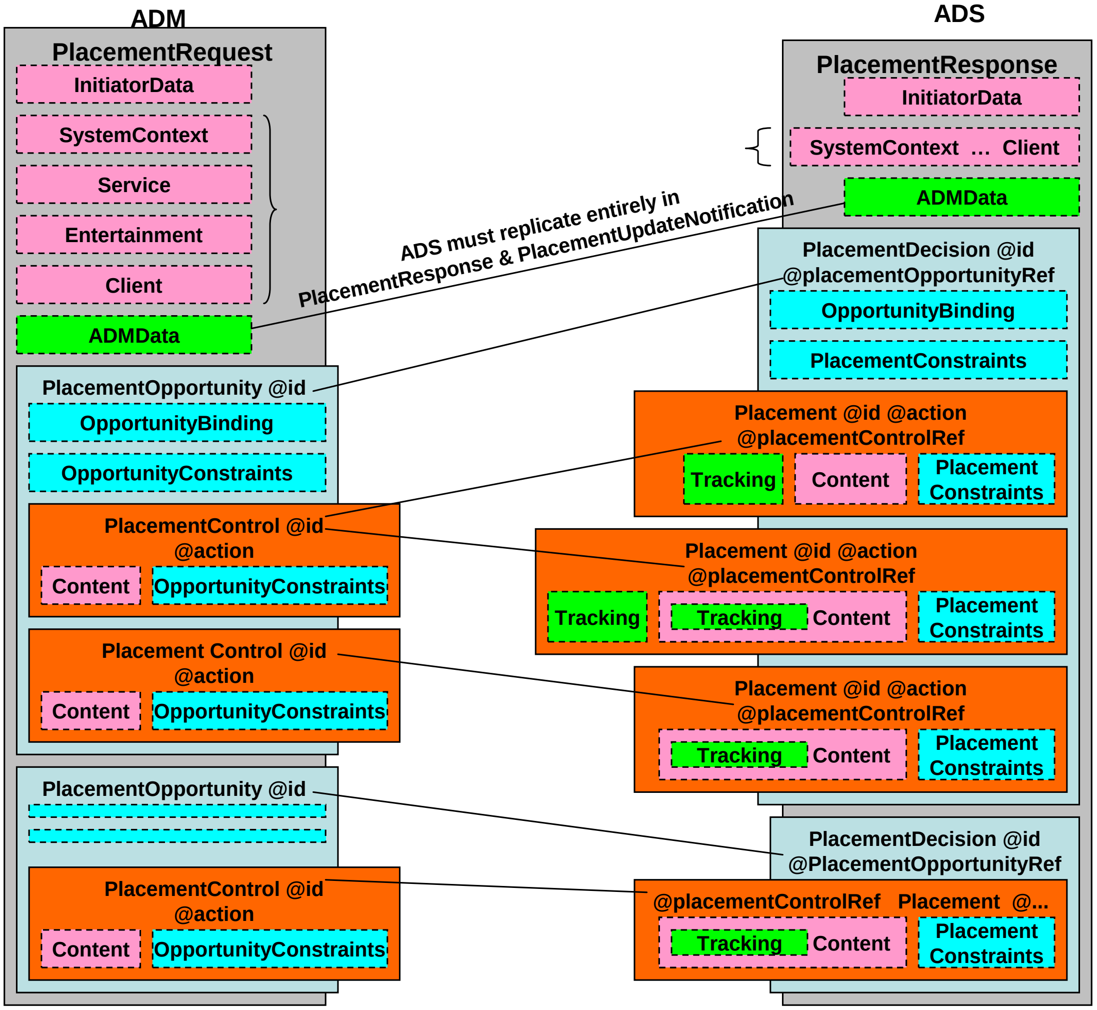
The placement response messages contain the following high-level items:

- An optional system context description containing refinements to the matching registered placement service description.
- Optional placement service descriptions containing refinements to the matched registered placement service description.
- An optional reference to all the original programming/entertainment content description.

- An optional opaque data element, the ADMData element, if present in the PlacementRequest message.
- An optional client description that again facilitates a self-describing response.
- A list of placement decisions each potentially containing a set of placements. Each placement decision *may* optionally contain an entertainment content description which, if present, is only applicable to the specific placement decision. A placement ***shall*** define the action (e.g., insertion, addition, replacement, deletion, etc.), and it *may* fully describe the ad including the ad type, a reference to the content, and a Tracking element for associated event reporting. It is via these placement decisions and the Placement elements which the ADS *may* change or alter the placement opportunity structure by using previously described verbs.

For some placement response message optional items, they *may* be included to facilitate a self-describing response.

Figure 28 provides a high-level representation of the major elements comprising the PlacementRequest and PlacementResponse messages. The figure illustrates the required element correspondences, or linkages, typically applicable to an element's reference attribute. Note that Figure 28 does not illustrate all message elements (i.e., it is not comprehensive or all inclusive). See the individual schemas for further details.



Note: The PlacementUpdateNotification message is semantically similar to the PlacementResponse message.

Figure 28: Placement Message High-Level View (not comprehensive)

In Figure 28, the PlacementRequest and PlacementResponse messages utilize the base SCTE 130 message element design. The SystemContext, Service, Entertainment and Client elements are optional in both the PlacementRequest and PlacementResponse messages. These elements are typically used in the PlacementRequest message and are provided (or included) in the PlacementResponse message primarily for debug and message symmetry.

The Entertainment element identifies the entertainment or programming asset referential context and *may* be supplied at either the root message scope or the local element scope, i.e., within a specific PlacementOpportunity element (or PlacementDecision element when contained in the PlacementResponse message). The Entertainment element ***shall*** only be used in one of the two scopes (i.e., the inclusion locations are mutually exclusive). When supplied at the root message scope, as illustrated in Figure 28, the Element applies to all message contained PlacementOpportunity (or PlacementDecision) elements. When specified at the local element scope, the value applies only to the specific including element. Consequently, multiple Entertainment assets *may* be referenced when contained in the PlacementOpportunity or PlacementDecision elements. Additionally, when using the local element scope, the same Entertainment element information ***shall*** be repeated in each PlacementOpportunity or PlacementDecision element as applicable. The Entertainment element ***shall no***t be included at both the root message scope and the local element scope within a single message.

If the PlacementRequest message supplied the optional ADMData element, the ADMData element ***shall*** be returned in the PlacementResponse message and any subsequent PlacementUpdateNotification message. See Section 14.2 for ADMData element details. An example ADM usage *may* be for providing high resiliency services.

In Figure 28, the ADM generated PlacementRequest message contains two PlacementOpportunity elements with each element uniquely identified by the individual PlacementOpportunity element's @id attribute.

The first PlacementOpportunity element (proceeding top down in the figure) contains two PlacementControl elements and each element is uniquely identified by their @id attribute. Each PlacementControl element describes a single positional placement opportunity.

An ADM *may* optionally communicate its suggested or planned behavior using the @action attribute. For example, an ADM operating in a traditional linear environment *may* propose to replace the current positional placement opportunity with new ADS specified ad content. The proposed ADM behavior *may* be overridden by the ADS, but this is an implementation decision outside the scope of this specification. The ADS ***shall*** always respond with an @action attribute included in its Placement element. This action represents the ADS ad placement decision which, for this linear example, *may* be to confirm replacement.

The ADM *may* optionally describe the current ad content occupying the positional placement opportunity by using the core:Content element (assuming the ADM has this information). For example, the information *may* be conveyed via the SCTE-35 segmentation_descriptor(). See [SCTE35] for additional information. The PlacementControl element *may* also enumerate the position within the placement opportunity using the @position attribute. For example, the ADM *may* only ask for position number two to be filled in the placement opportunity if it is not modifying the first slot in a subdivided placement opportunity.

The second PlacementOpportunity element in Figure 28 contains only a single PlacementControl element describing an independent placement opportunity for the same entertainment content asset.

The PlacementResponse message element (or similarly the PlacementUpdateNotification message element) should typically contain the same number of PlacementDecision elements as the original PlacementRequest message's PlacementOpportunity element count. However, the PlacementResponse (or PlacementUpdateNotification) message *may* contain additional PlacementDecision elements if the ADS added additional PlacementDecision elements. There ***shall*** minimally be a one-to-one correspondence where every PlacementOpportunity element contained in the PlacementRequest message ***shall*** be referenced by at least one PlacementDecision element in the response or update message. The PlacementDecision element document order should typically match the PlacementOpportunity element order.

Placement opportunities contained in a PlacementRequest message ***shall*** only be for the recipient ADS. The same PlacementRequest message ***shall no***t be delivered to two or more ADS logical service registrants (i.e., two ADS each having a different @identity attribute value). For example, if an ADM determines that different portions of a placement opportunity are to be handled by different registered ADS, the ADM ***shall*** create two different PlacementRequest messages.

Each PlacementResponse message's PlacementDecision element ***shall*** have an @placementOpportunityRef attribute value referencing an original PlacementOpportunity element's @id attribute. In Figure 28, this pairing is illustrated via the lines connecting the PlacementOpportunity element's @id attribute and PlacementDecision element's @placementOpportunityRef attribute.

Within the PlacementDecision element, the ADS ***shall*** use Placement elements to communicate its individual positional placement decisions. Each Placement element *may* represent a single ad asset. Because the ADS *may* change the (positional) PlacementOpportunity structure, the Placement element should reference an original PlacementControl element via the @placementControlRef attribute if possible. Multiple Placement elements *may* refer to the same PlacementControl element. For example, this *may* occur when two or more placements are used to fill the positional placement opportunity currently occupied by a single ad asset.

If the placement decision or placement expires before the PlacementResponse (or PlacementUpdateNotification) message's lifecycle ends, the ADM behavior is undefined and outside the scope of the specification.

The following sections detail the placement messaging schemas.

12.1.1 PlacementRequest Message Schema

Figure 29 illustrates the PlacementRequest message schema used to announce the ad placement opportunities and request ADS ad placement decisions.

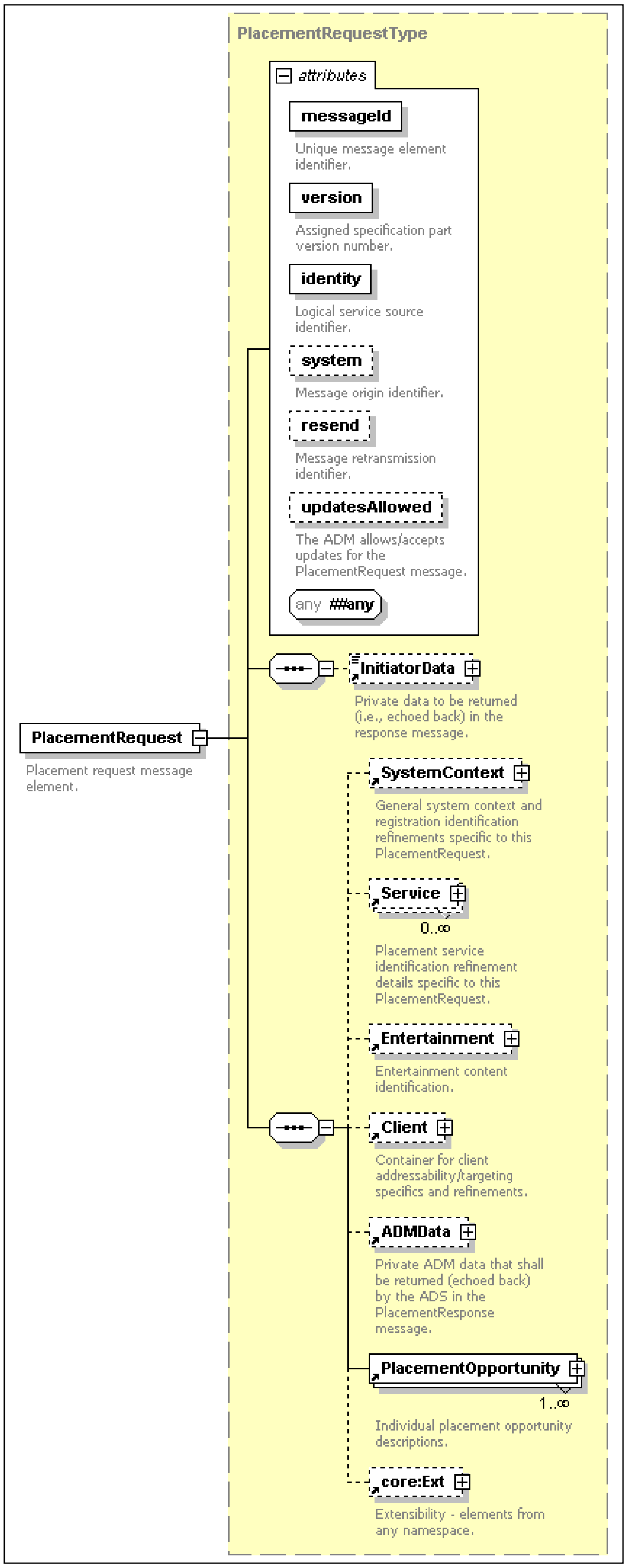

PlacementRequestType
attributes
messageId
Unique message element identifier.
version
Assigned specification part version number.
identity
Logical service source identifier.
system
Message origin identifier.
resend
Message retransmission identifier.
updatesAllowed
The ADM allows/accepts updates for the PlacementRequest message.
any ##any
PlacementRequest
Placement request message element.
InitiatorData
Private data to be returned (i.e., echoed back) in the response message.
SystemContext
General system context and registration identification refinements specific to this PlacementRequest.
Service
0..∞
Placement service identification refinement details specific to this PlacementRequest.
Entertainment
Entertainment content identification.
Client
Container for client addressability/targeting specifics and refinements.
ADMData
Private ADM data that shall be returned (echoed back) by the ADS in the PlacementResponse message.
PlacementOpportunity
1..∞
Individual placement opportunity descriptions.
core:Ext
Extensibility - elements from any namespace.

Figure 29: PlacementRequest Message Schema

12.1.1.1 Semantic Definitions for the PlacementRequest Message

@messageId [Required, core:messageIdAttrType]—The message identifier. See [SCTE130-2] for additional information.

@version [Required, core:versionAttrType]—The ADM message interface version. See Section 9.1 for additional information.

@identity [Required, core:identityAttrType]—The origin logical service identifier. See [SCTE130-2] for additional information.

@system [Optional, core:systemAttrType]—The message origin identifier. See [SCTE130-2] for additional information.

@resend [Optional, core:resendAttrType]—The message is a retransmit of a previous PlacementRequest message. The attribute's value is the original message's @messageId value. See [SCTE130-2] for additional information.

@updatesAllowed [Optional, xsd:boolean]—The value true indicates the ADM ***shall*** accept placement update message exchanges for this placement opportunity set. The value false, which is the default value if the attribute is omitted, indicates the ADM ***shall no***t accept placement update message exchanges for this placement request message. See Section 12.2 for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:InitiatorData [Optional]—The core:InitiatorData element contains implementation specific private data which ***shall*** be returned in the PlacementResponse message. See [SCTE130-2] for additional information.

SystemContext [Optional]—System environment information that typically refines the placement service identification. See Section 14.52 for additional information.

Service [Optional]—Zero or more elements containing message local information typically refining the placement service registration match. See Section 14.44 for additional information.

Entertainment [Optional]—The entertainment (i.e., programming) content for which the placement opportunities are associated. See Section 14.11 for additional information. If this element is present, the Entertainment element contained within any child PlacementOpportunity element ***shall no***t be used (i.e., the entertainment asset identification applies to all of this message's PlacementOpportunity elements and its scope ***shall no***t be overridden). See Section 14.33 for details regarding a

PlacementOpportunity element referencing an individual entertainment content asset which enables a PlacementRequest message to reference multiple entertainment assets via the usage of multiple PlacementOpportunity elements.

Client [Optional]—A container for client specific placement focus refinements (i.e., targeting criteria). See Section 14.4 for additional information.

ADMData [Optional]—Private ADM information. If included, this element ***shall*** be returned by the ADS in the corresponding PlacementResponse message and any associated PlacementUpdateNotification messages. See Section 14.2 for additional information.

PlacementOpportunity [Required]—One or more placement opportunity descriptions. See Section 14.33 for additional information. See x for an additional illustration.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

Figure 28 illustrates the required attribute linkage between a PlacementOpportunity element and its placement service identifier which is provided during registration. Additionally, the figure illustrates how a message contained Service element is linked to and utilized to provide further refinements to a registration.

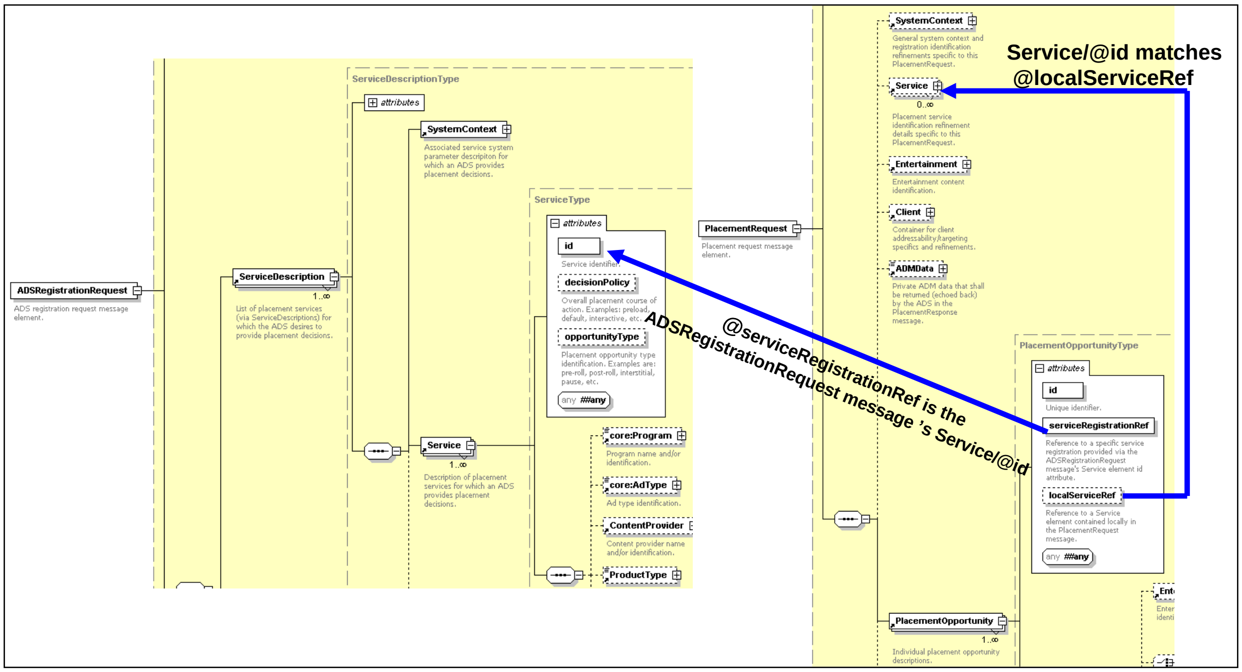

ADSRegistrationRequest
ADS registration request message element.
ServiceDescription
1..∞
List of placement services (via ServiceDescriptions) for which the ADS desires to provide placement decisions.
ServiceDescriptionType
attributes
SystemContext
Associated service system parameter descripiton for which an ADS provides placement decisions.
Service
1..∞
Description of placement services for which an ADS provides placement decisions.
ServiceType
attributes
id
Service identifier.
decisionPolicy
Overall placement course of action. Examples: preload, default, interactive, etc.
opportunityType
Placement opportunity type identification. Examples are: pre-roll, post-roll, interstitial, pause, etc.
any ##any
core:Program
Program name and/or identification.
core:AdType
Ad type identification.
ContentProvider
Content provider name and/or identification.
ProductType
PlacementRequest
Placement request message element.
SystemContext
General system context and registration identification refinements specific to this PlacementRequest.
Service
0..∞
Placement service identification refinement details specific to this PlacementRequest.
Entertainment
Entertainment content identification.
Client
Container for client addressability/targeting specifics and refinements.
ADMData
Private ADM data that shall be returned (echoed back) by the ADS in the PlacementResponse message.
PlacementOpportunityType
attributes
id
Unique identifier.
serviceRegistrationRef
Reference to a specific service registration provided via the ADSRegistrationRequest message's Service element id attribute.
localServiceRef
Reference to a Service element contained locally in the PlacementRequest message.
any ##any
PlacementOpportunity
1..∞
Individual placement opportunity descriptions.
Service/@id matches @localServiceRef
@serviceRegistrationRef is the ADSRegistrationRequest message 's Service/@id

Figure 30: Service\@id Attribute Linkages

12.1.2 PlacementResponse Message Schema

Figure 31 illustrates the PlacementResponse message schema.

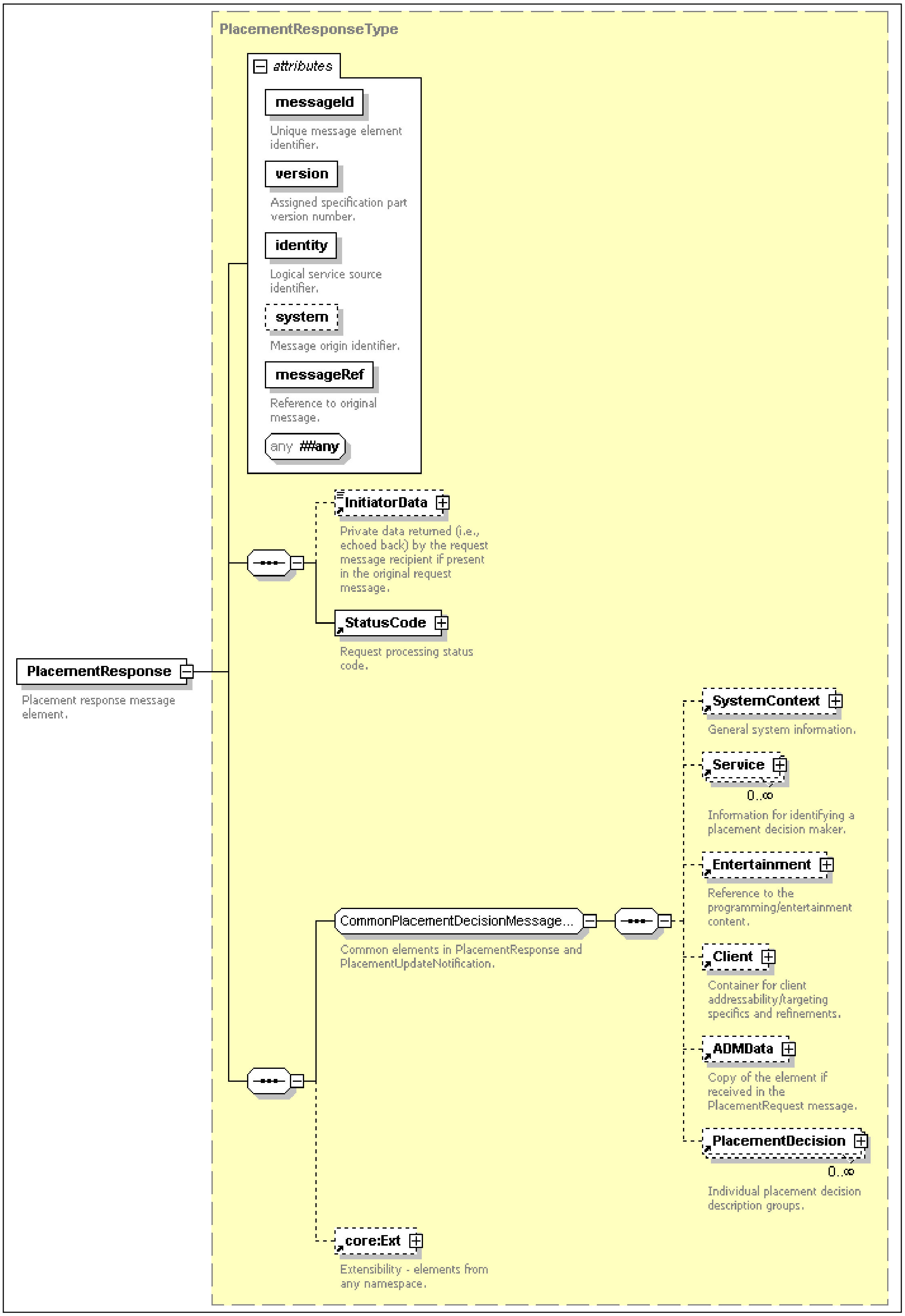

PlacementResponseType
attributes
messageId
Unique message element identifier.
version
Assigned specification part version number.
identity
Logical service source identifier.
system
Message origin identifier.
messageRef
Reference to original message.
any ##any
InitiatorData
Private data returned (i.e., echoed back) by the request message recipient if present in the original request message.
StatusCode
Request processing status code.
PlacementResponse
Placement response message element.
SystemContext
General system information.
Service
0..∞
Information for identifying a placement decision maker.
Entertainment
Reference to the programming/entertainment content.
CommonPlacementDecisionMessage...
Common elements in PlacementResponse and PlacementUpdateNotification.
Client
Container for client addressability/targeting specifics and refinements.
ADMData
Copy of the element if received in the PlacementRequest message.
PlacementDecision
0..∞
Individual placement decision description groups.
core:Ext
Extensibility - elements from any namespace.

Figure 31: PlacementResponse Message Schema

A non-success status code ***shall*** terminate the placement message sequence. In this case, no placement update message exchange ***shall*** subsequently ensue specifically relating to this PlacementRequest message identifier (i.e., the PlacementRequest message's @messageId attribute in the original message or the @resend attribute value in a retransmitted PlacementRequest message).

A placement update message exchange ***shall no***t be initiated until the PlacementResponse message is sent.

12.1.2.1 Semantic Definitions for the PlacementResponse Message

@messageId [Required, core:messageIdAttrType]—The message identifier. See [SCTE130-2] for additional information.

@version [Required, core:versionAttrType]—The ADM message interface version. See Section 9.1 for additional information.

@identity [Required, core:identityAttrType]—The origin logical service identifier. See [SCTE130-2] for additional information.

@system [Optional, core:systemAttrType]—The message origin identifier. See [SCTE130-2] for additional information.

@messageRef [Required, core:messageRefAttrType]—A reference to the paired PlacementRequest message element initiating this message exchange. The value ***shall*** be the PlacementRequest message's @messageId attribute value. See [SCTE130-2] for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:InitiatorData [Optional]—The core:InitiatorData element ***shall*** be an exact copy of the PlacementRequest message's core:InitiatorData element and ***shall*** only be present if found in the paired request message. See [SCTE130-2] for additional information.

core:StatusCode [Required]—An applicable status code specific to the PlacementRequest message processing. See [SCTE130-2] and Appendix A for additional information.

SystemContext [Optional]—System environment information that typically is a copy of the original information. See Section 14.52 for additional information.

Service [Optional]—Zero or more elements containing message local information. Typically, this is a copy of the original information. See Section 14.44 for additional information.

Entertainment [Optional]—The entertainment (i.e., programming) content for which the placement opportunities are associated. See Section 14.11 for additional information. Typically, this is a copy of the original information. If this element is present, the Entertainment element contained within any child PlacementDecision element ***shall no***t be used (i.e., the entertainment asset identification applies to all of this message's PlacementDecision elements and its scope ***shall no***t be overridden). See Section 14.31 for details regarding a PlacementDecision element referencing an individual entertainment content asset which enables a PlacementResponse message to reference multiple entertainment assets via the usage of multiple PlacementDecision elements.

Client [Optional]—A container for client specific placement focus refinements (i.e., targeting criteria). Typically, this is a copy of the original information. See Section 14.4 for additional information.

ADMData [Optional]—An identical copy of the ADMData element as it appeared in the PlacementRequest message. This element ***shall*** only be present if the ADMData element was present in the corresponding PlacementRequest message. See Section 14.2 for additional information.

PlacementDecision [Optional]—Zero or more PlacementDecision elements with each element containing one or more Placement elements. See Section 14.31 for additional information. The Placement elements define the individual positional placement decisions. Each PlacementDecision element typically should correspond to a PlacementOpportunity element. If a processing error occurred, no elements *may* be present.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

12.1.3 Placement Messaging Examples

Example 28 illustrates a VOD pre-roll and post-roll placement request and response (Example 29). The pre-roll placement opportunity is constrained by time and the second placement opportunity, the post-roll, is restricted by placement count. The pre-roll placement opportunity requires a telescoping ad placement. The request provides placement focus (targeting) criteria of ZipPlus4 and gender equal to male. The PlacementResponse (Example 29) returns one content binding for the first placement opportunity. The duration is not specified but is assumed to be compliant. Two placements are returned for the second placement opportunity.

```
<PlacementRequest messageId="5B8ECCED-979B-421E-8A6E-E2192B115160"
version="1.4" identity="ADMLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F4" system="adm" updatesAllowed=”true”>
  <Entertainment>
    <core:Content>
      <core:AssetRef providerID="entertainment.com"
assetID="ABCD0123456789012345"/>
    </core:Content>
  </Entertainment>
  <Client>
    <TargetCode key="ZipPlus4">12345-6789</TargetCode>
    <TargetCode key="Gender">Male</TargetCode>
  </Client>
  <ADMData><!--Hidden secret sauce--></ADMData>
  <PlacementOpportunity id="5B8ECCED-979B-421E-8A6E-E2192B115162"
serviceRegistrationRef="ServiceRef1">
    <OpportunityBinding opportunityType="preRoll"/>
    <OpportunityConstraints>
      <Scope>Local</Scope>
    </OpportunityConstraints>
    <PlacementControl id="5B8ECCED-979B-421E-8A6E-E2192B115163" action="fill">
      <OpportunityConstraints>
        <core:AdType>Telescoping</core:AdType>
        <MaxDuration>PT60S</MaxDuration>
        <MinDuration>PT15S</MinDuration>
        <DesiredDuration>PT30S</DesiredDuration>
      </OpportunityConstraints>
    </PlacementControl>
  </PlacementOpportunity>
  <PlacementOpportunity id="5B8ECCED-979B-421E-8A6E-E2192B115164"
serviceRegistrationRef="ServiceRef1">
    <OpportunityBinding opportunityType="postRoll"/>
    <PlacementControl id="5B8ECCED-979B-421E-8A6E-E2192B115165" action="fill">
      <OpportunityConstraints>
        <MaxPlacementCount>3</MaxPlacementCount>
        <MinPlacementCount>1</MinPlacementCount>
      </OpportunityConstraints>
    </PlacementControl>
  </PlacementOpportunity>
</PlacementRequest>
```

Example 28

```
<PlacementResponse messageId="5B8ECCED-979B-421E-8A6E-E2192B11516A"
version="1.4" identity="ADSLogicalService_1170FEF4-C19B-450E-B624-
```

```
421B23F525F5" system="ads" messageRef="5B8ECCED-979B-421E-8A6E-
E2192B115160">
  <core:StatusCode class="0"/>
  <ADMData><!--Hidden secret sauce--></ADMData>
  <PlacementDecision id="PD1" placementOpportunityRef="5B8ECCED-979B-421E-
8A6E-E2192B115162">
    <Placement id="PD1_P1" action="fill" placementControlRef="5B8ECCED-979B-
421E-8A6E-E2192B115163">
      <core:Content>
        <core:AssetRef providerID="adsrus.com" assetID="ZXYW0123456789012345"/>
        <core:Tracking>5B8ECCED-979B-421E-8A6E-E2192B11516C</core:Tracking>
      </core:Content>
    </Placement>
  </PlacementDecision>
  <PlacementDecision id="PD2" placementOpportunityRef="5B8ECCED-979B-421E-
8A6E-E2192B115164">
    <Placement id="PD2_P1" action="fill" placementControlRef="5B8ECCED-979B-
421E-8A6E-E2192B115165">
      <core:Content>
        <core:AssetRef providerID="adsrus.com" assetID="ZXYW0123456789012346"/>
        <core:Tracking>5B8ECCED-979B-421E-8A6E-E2192B11516D</core:Tracking>
      </core:Content>
    </Placement>
    <Placement id="PD2_P2" action="fill" placementControlRef="5B8ECCED-979B-
421E-8A6E-E2192B115165">
      <core:Content>
        <core:AssetRef providerID="adsrus.com" assetID="ZXYW0123456789012347"/>
        <core:Tracking>5B8ECCED-979B-421E-8A6E-E2192B11516E"</core:Tracking>
      </core:Content>
    </Placement>
  </PlacementDecision>
</PlacementResponse>
```

Example 29

12.2 Placement Update Messaging

Placement update messaging refers to an ADS PlacementUpdateNotification message communicating a revision to the active placement decisions delivered previously by either a PlacementResponse or PlacementUpdateNotification message. The ADM ***shall*** respond with a PlacementUpdateAcknowledgement message. The PlacementUpdateNotification message ***shall*** be sent to the address for placement update notification messages as indicated during registration. A PlacementUpdateNotification message ***shall*** only be sent if the ADM allowed it via the PlacementRequest message’s @updatesAllowed attribute being set to the value true.

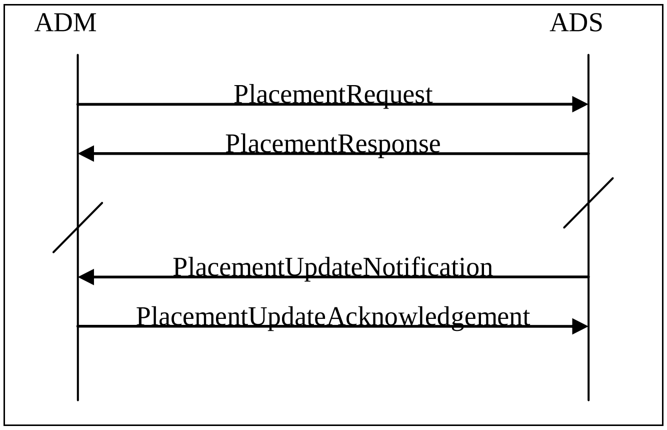


Figure 32: Placement Update Message Exchange Sequence

Placement update messaging ***shall*** be initiated by an ADS. The PlacementUpdateNotification message's placement decision set ***shall*** be based on the placement opportunity set specified in the accepted ADM PlacementRequest message (just like the PlacementResponse message). The accepted PlacementRequest message *may* be either the original message or the retransmitted message. Identification of the accepted PlacementRequest message ***shall*** be via the original message's @messageId attribute or via the retransmitted PlacementRequest message's @resend attribute value. The PlacementUpdateNotification message is the structural equivalent of the PlacementResponse message with similar syntax and semantics.

A PlacementUpdateNotification message ***shall no***t be sent if an ADM indicates it does not support placement update messaging for the placement message exchange (i.e., the PlacementRequest message's @updatesAllowed attribute is set to the value false). Provided the ADM allows updates, indicated by the PlacementRequest message's @updatesAllowed attribute set to the value true, the placement update message exchange *may* occur at any time after the corresponding PlacementResponse message is sent.

Multiple placement update message exchanges *may* occur at the discretion of the ADS.

The PlacementUpdateNotification message uses a design pattern similar to the PlacementResponse message. This allows the ADS to update any or all aspects of the prior active placement decisions. The elements of the PlacementUpdateNotification message ***shall*** conform to the rules outlined for the corresponding elements in the PlacementResponse message.

Each PlacementUpdateNotification message ***shall*** replace the entire original PlacementResponse message and all previous, successfully acknowledged PlacementUpdateNotification messages when the @updateAction is set to the value "replace". The net effect at the ADM should be as if the PlacementUpdateNotification message was the original response to the PlacementRequest message. The implementation and transition to the update placement decision set is up to the ADM and is outside the scope of this specification (e.g., the ADM *may* have already cued for execution elements of the prior placement decision set and *may* not be able to recall them at the time of the update).

Effectively then, the ADS generates a new placement decision set based on the accepted PlacementRequest message and the current state of the campaign. The ADS *may* change any aspect of the original PlacementResponse message, or any intervening PlacementUpdateNotification message, including the structure and decision set.

The placement update message exchange *may* also be used by the ADS to cancel a placement decision set. The ADS ***shall*** cancel a placement decision set by setting the @updateAction attribute to the value “cancel” in a PlacementUpdateNotification message.

Once a placement update message exchange occurs with the @updateAction attribute set to the value ”cancel”, the ADS ***shall*** cease to initiate placement update message exchanges for the cancelled placement decision set and the ADM ***shall*** cease to use the cancelled placement decision set. However, the exact timing of when the ADM ceases to use the cancelled placement decision set is up to the ADM and is outside the scope of this specification (e.g., the ADM *may* have already cued for execution elements of the cancelled placement decision set and *may* not be able to recall them at the time of the update).

Placement update message exchange determination needs along with implementation of the updated placement decision sets are outside the scope this specification. This specification does not preclude or prefer any method or combination thereof.

An example usage is a VOD session placement decision set update. Upon startup of the VOD session, the ADM issues a PlacementRequest message for the specific entertainment content, client, and addressability parameters. The ADS responds with a PlacementResponse message which includes an airline ad for flights to Las Vegas. Prior to the VOD session completing, all of the near term Las Vegas seats are sold out and the airline decides to change the ad asset to a Florida destination commercial. The ADS issues a PlacementUpdateNotification message which changes the airline ad while leaving the other previous placement decisions intact. The ADM responds with a PlacementUpdateAcknowledgement indicating successful change acceptance.

Another example usage is the updating of a preload or default placement decision set. For example, a VOD system *may* request a default placement decision for a pre-roll placement opportunity associated with a specific entertainment content asset. The ADS returns a placement utilizing a rotation list populated with ten content elements. Sometime later, the campaign changes and a specific content asset should no longer be included. The ADS issues a PlacementUpdateNotification message to update the placement decision set specifying a new content rotation list containing only nine assets. The ADM responds with a PlacementUpdateAcknowledgement message indicating update acceptance. Thus, the eliminated content asset is subsequently removed.

The following sections detail the placement update messaging schemas.

12.2.1 PlacementUpdateNotification Message Schema

Figure 33 illustrates the PlacementUpdateNotification message schema.

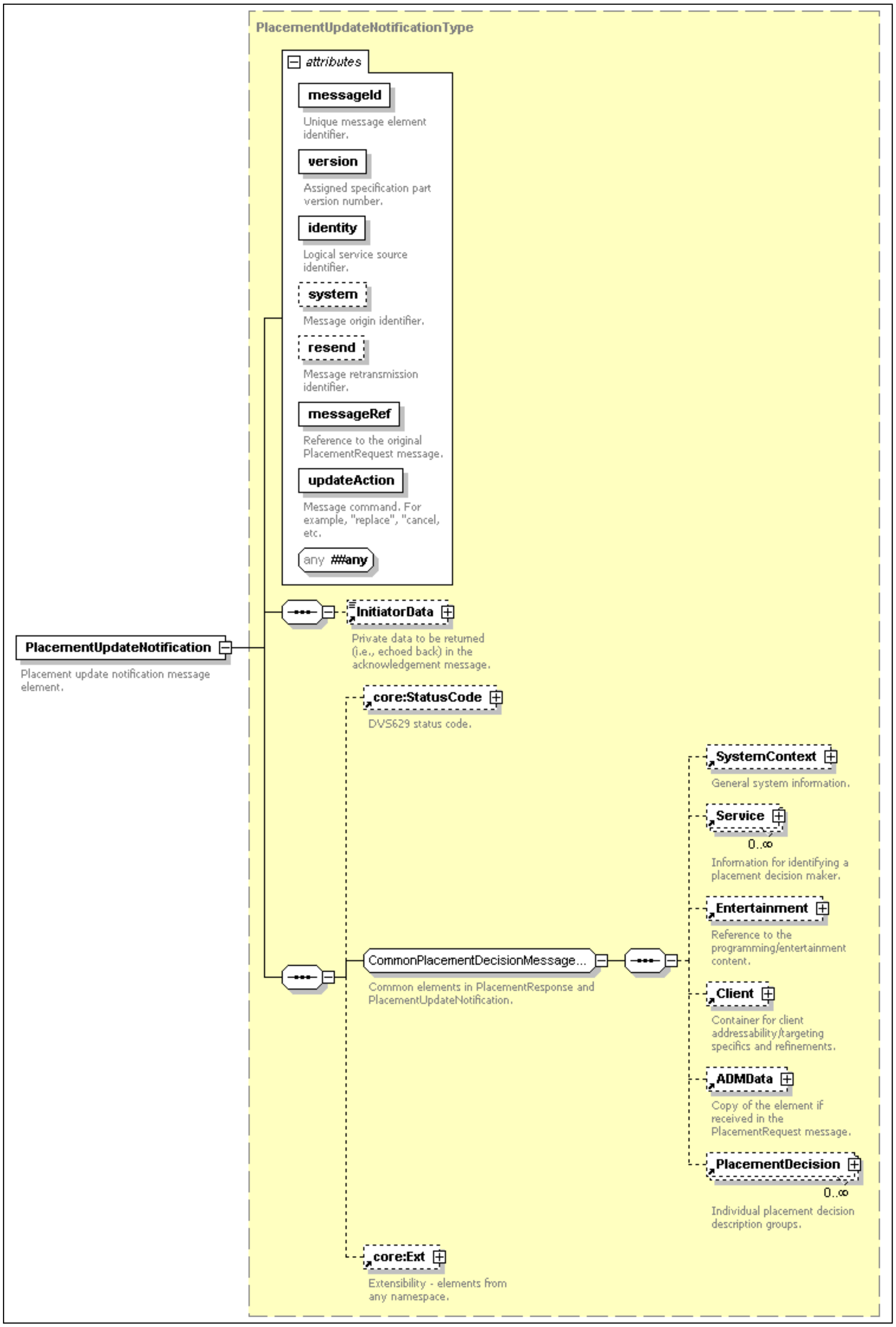

PlacementUpdateNotificationType
attributes
messageId
Unique message element identifier.
version
Assigned specification part version number.
identity
Logical service source identifier.
system
Message origin identifier.
resend
Message retransmission identifier.
messageRef
Reference to the original PlacementRequest message.
updateAction
Message command. For example, "replace", "cancel, etc.
any ##any
PlacementUpdateNotification
Placement update notification message element.
InitiatorData
Private data to be returned (i.e., echoed back) in the acknowledgement message.
core:StatusCode
DVS629 status code.
CommonPlacementDecisionMessage...
Common elements in PlacementResponse and PlacementUpdateNotification.
SystemContext
General system information.
Service
0..∞
Information for identifying a placement decision maker.
Entertainment
Reference to the programming/entertainment content.
Client
Container for client addressability/targeting specifics and refinements.
ADMData
Copy of the element if received in the PlacementRequest message.
PlacementDecision
0..∞
Individual placement decision description groups.
core:Ext
Extensibility - elements from any namespace.

Figure 33: PlacementUpdateNotification Message Schema

12.2.1.1 Semantic Definitions for the PlacementUpdateNotification Message

@messageId [Required, core:messageIdAttrType]—The message identifier. See [SCTE130-2] for additional information.

@version [Required, core:versionAttrType]—The ADM message interface version. See Section 9.1 for additional information.

@identity [Required, core:identityAttrType]—The origin logical service identifier. See [SCTE130-2] for additional information.

@system [Optional, core:systemAttrType]—The message origin identifier. See [SCTE130-2] for additional information.

@resend [Optional, core:resendAttrType]—The message is a retransmit of a previous PlacementUpdateNotification message. The attribute's value is the original PlacementUpdateNotification message's @messageId value. See [SCTE130-2] for additional information.

@messageRef [Required, core:messageRefAttrType]—A reference to the original PlacementRequest message element for which this update notification is being provided. See [SCTE130-2] for additional information. The value ***shall*** be the original PlacementRequest message's @messageId attribute value or the retransmitted PlacementRequest message's @resend attribute's value.

@updateAction [Required, core:nonEmptyStringType]—The message command verb which is a value from Table 8. The attribute's value ***shall*** appear exactly as in Table 8 and ***shall no***t be empty.

Value	Description
cancel	The referenced placement management sequence is terminated.
replace	The most recent PlacementResponse or PlacementUpdateNotification message ***shall*** be replaced in its entirety by this message.
…	User defined and outside the scope of this specification. The string ***shall*** be prefixed with the text "private:".

Table 8: PlacementUpdateNotification/@updateAction Attribute Values

@##any [Optional]—Additional attributes from any namespace.

core:InitiatorData [Optional]—The core:InitiatorData element contains implementation specific private data which ***shall*** be returned in the PlacementUpdateAcknowledgement message. See [SCTE130-2] for additional information.

core:StatusCode [Optional]—A status code value along with zero or more descriptive text elements. See [SCTE130-2] for additional information.

SystemContext [Optional]—System environment information that typically is a copy of the original information. See Section 14.52 for additional information.

Service [Optional]—Zero or more elements containing message local information. Typically, this is a copy of the original information. See Section 14.44 for additional information.

Entertainment [Optional]—The entertainment (i.e., programming) content for which the placement opportunities are associated. See Section 14.11 for additional information. Typically, this is a copy of the original information. If this element is present, the Entertainment element contained within any child PlacementDecision element ***shall no***t be used (i.e., the entertainment asset identification applies to all of this message's PlacementDecision elements and its scope ***shall no***t be overridden). See Section 14.31 for details regarding a PlacementDecision element referencing an individual entertainment content asset which enables a PlacementUpdateNotification message to reference multiple entertainment assets via the usage of multiple PlacementDecision elements.

Client [Optional]—A container for client specific placement focus refinements (i.e., targeting criteria). Typically, this is a copy of the original information. See Section 14.4 for additional information.

ADMData [Optional]—An identical copy of the ADMData element as it appeared in the PlacementRequest message. This element ***shall*** only be present if the ADMData element was present in the corresponding PlacementRequest message. See Section 14.2 for additional information.

PlacementDecision [Optional]—Zero or more PlacementDecision elements with each element containing one or more Placement elements. See Section 14.31 for additional information. The Placement elements define the individual positional placement decisions. Each PlacementDecision element typically should correspond to a PlacementOpportunity element. If a processing error occurred then no elements *may* be present. When the @updateAction attribute is set to the value ”cancel”, no PlacementDecision elements ***shall*** be included.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

The PlacementUpdateNotification element's value *may* be empty (e.g., when @updateAction is set to the value ”cancel”).

12.2.2 PlacementUpdateAcknowledgement Message Schema

Figure 34 illustrates the PlacementUpdateAcknowledgement message schema.

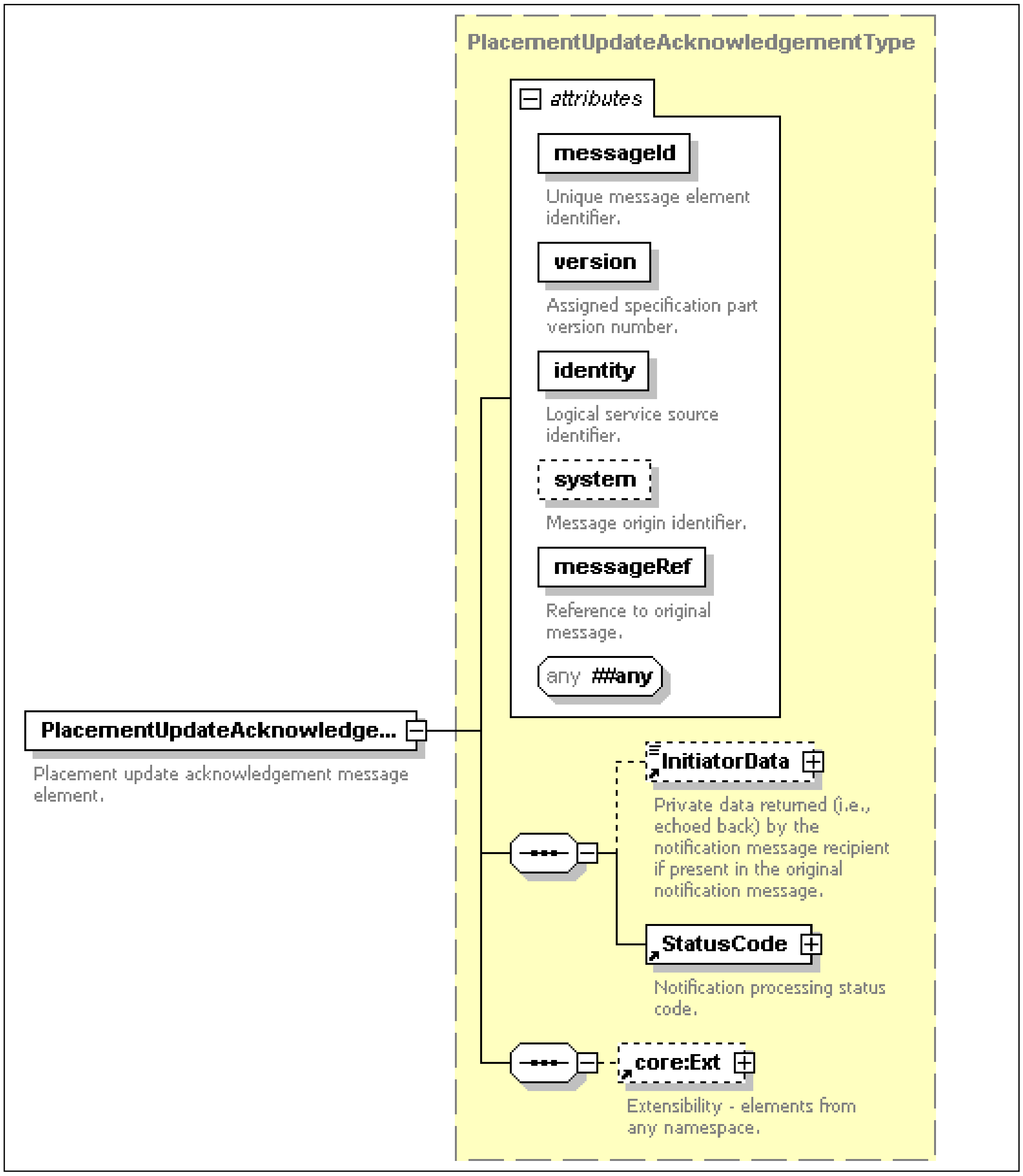

PlacementUpdateAcknowledgementType
attributes
messageId
Unique message element identifier.
version
Assigned specification part version number.
identity
Logical service source identifier.
system
Message origin identifier.
messageRef
Reference to original message.
any ##any
PlacementUpdateAcknowledge...
Placement update acknowledgement message element.
InitiatorData
Private data returned (i.e., echoed back) by the notification message recipient if present in the original notification message.
StatusCode
Notification processing status code.
core:Ext
Extensibility - elements from any namespace.

Figure 34: PlacementUpdateAcknowledgement Message Schema

12.2.2.1 Semantic Definitions for the PlacementUpdateAcknowledgement Message

@messageId [Required, core:messageIdAttrType]—The message identifier. See [SCTE130-2] for additional information.

@version [Required, core:versionAttrType]—The ADM message interface version. See Section 9.1 for additional information.

@identity [Required, core:identityAttrType]—The origin logical service identifier. See [SCTE130-2] for additional information.

@system [Optional, core:systemAttrType]—The message origin identifier. See [SCTE130-2] for additional information.

@messageRef [Required, core:messageRefAttrType]—A reference to the paired PlacementUpdateNotification message element initiating this message exchange. The value ***shall*** be the PlacementUpdateNotification message's @messageId attribute value. See [SCTE130-2] for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:InitiatorData [Optional]—The core:InitiatorData element ***shall*** be an exact copy of the PlacementUpdateNotification message's core:InitiatorData element and ***shall*** only be present if found in the paired request message. See [SCTE130-2] for additional information.

core:StatusCode [Required]—An applicable status code specific to the PlacementUpdateNotification message processing. See [SCTE130-2] and Appendix A for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

12.2.3 Placement Update Examples

Example: The Section 12.1.3 placement decision set (the Example 28 PlacementRequest message and the Example 29 PlacementResponse message) is replaced by this update message exchange (Example 30 notification and Example 31 successful acknowledgement).

```
<PlacementUpdateNotification messageId="5B8ECCED-979B-421E-8A6E-
E2192B11516B" version="1.4" identity="ADSLogicalService_1170FEF4-C19B-450E-
```

```
B624-421B23F525F5" system="ads" messageRef="5B8ECCED-979B-421E-8A6E-
E2192B115160" updateAction="replace"
  <core:StatusCode class="0"/>
  <ADMData><!--Hidden secret sauce--></ADMData>
  <PlacementDecision id="PD1" placementOpportunityRef="5B8ECCED-979B-421E-
8A6E-E2192B115162">
    <Placement id="PD1_P1" action="fill" placementControlRef="5B8ECCED-979B-
421E-8A6E-E2192B115163">
      <core:Content>
        <core:AssetRef providerID="adsrus.com" assetID="ZXYW0123456789012345"/>
        <core:Tracking>5B8ECCED-979B-421E-8A6E-E2192B11516C</core:Tracking>
      </core:Content>
    </Placement>
  </PlacementDecision>
  <PlacementDecision id="PD2" placementOpportunityRef="5B8ECCED-979B-421E-
8A6E-E2192B115164">
    <Placement id="PD2_P1" action="fill" placementControlRef="5B8ECCED-979B-
421E-8A6E-E2192B115165">
      <core:Content>
        <core:AssetRef providerID="adCreativeWorld.com"
assetID="ABCD0123456789876543"/>
        <core:Tracking>5B8ECCED-979B-421E-8A6E-E2192B11634F</core:Tracking>
      </core:Content>
    </Placement>
    <Placement id="PD2_P2" action="fill" placementControlRef="5B8ECCED-979B-
421E-8A6E-E2192B115165">
      <core:Content>
        <core:AssetRef providerID="adsrus.com" assetID="ZXYW0123456789012347"/>
        <core:Tracking>5B8ECCED-979B-421E-8A6E-E2192B11516E"</core:Tracking>
      </core:Content>
    </Placement>
  </PlacementDecision>
</PlacementUpdateNotification>
```

Example 30

```
<PlacementUpdateAcknowledgement messageId="5B8ECCED-979B-421E-8A6E-
E2192B115164" version="1.4" identity="ADMLogicalService_1170FEF4-C19B-450E-
B624-421B23F525F4" system="adm" messageRef="5B8ECCED-979B-421E-8A6E-
E2192B11516B">
  <core:StatusCode class="0"/>
</PlacementUpdateAcknowledgement>
```

Example 31

12.3 Placement Status Messaging

The placement status message exchange ***shall*** be initiated by an ADM to facilitate ADM to ADS placement event reporting. The events, referred to as named event elements from hereon, ***shall*** provide placement decision fulfillment data and *may* include other events an ADM considers of interest to an ADS.

The PlacementStatusNotification message ***shall*** be sent to the ADS callback address supplied in the ADSRegistrationRequest message. The ADS ***shall*** confirm message receipt using the PlacementStatusAcknowledgement message. Figure 35 illustrates the placement status message exchange.

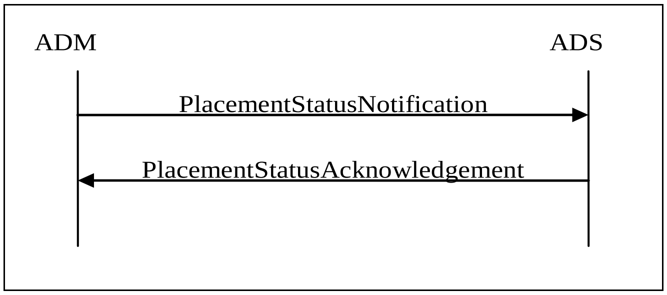


Figure 35: Placement Status Message Exchange

Named event elements are nested within PlayData elements reported to the ADS using the PlacementStatusNotification message. The PlacementStatusNotification message *may* contain any number of PlayData elements which wrap mixed sequences of named event elements. The Events element, which is a child element located any number of times within the PlayData element, contains the actual sequence of named event elements. This specification’s defined named event elements include the following:

- PlacementStatusEvent—Events related to placements, placement execution, and placement operations. For example, the start and end of ad content insertion, or the end of a PlacementRequest message being updatable.
- SessionEvent—Events related to a session. For example, start/end session and start/end entertainment content segment.
- SystemEvent—Global events, unrelated events and events not fitting into the other event elements.
- ViewerEvent—Events altering the viewer presentation (including click events). For example, a press (click) of the fast-forward, rewind, play, stop, or pause buttons, etc.

All specification defined named event elements share a common base definition described in Section 13.2.1. Each named event element ***shall*** have:

- An @type attribute communicating a specific event type within the named element classification.

- An @time attribute identifying the event occurrence time.
- An @identityADS attribute specifying the target ADS @identity attribute. (Note: This value *may* be specified in the PlayData element and consequently indirectly implies the value in the named event element).

The named event elements *may* optionally include any of the following:

- An @identityADM attribute specifying the originating ADM logical service identification.
- An @systemADM attribute providing the originating ADM system identification.
- An @messageRef attribute providing the @messageId attribute's value for a message associated with the event.
- A core:StatusCode element for status and descriptive text carriage.
- A SystemContext element for system parameters of interest.
- A Client element for targeting and refinement criteria.
  - Entertainment element for entertainment asset identification and play time reference or;
  - A Spot element for ad content identification and play time reference or;
- Specific Event Elements, Entertainment or Spot scoped, for additional play position information that provides an overall play time reference typically the clock the viewer associates with a program or entertainment asset including ads.
- A core:Ext element for extensibility.

The placement status message exchange minimally provides status events specific to the ADS placement decisions (i.e., the Placement elements contained within the PlacementDecision elements residing in either a PlacementResponse or PlacementUpdateNotification message). For each Placement element containing a content binding (i.e., the Placement element includes a core:Content element), the ADM should output the following event pairs as often as appropriate during placement delivery:

- A PlacementStatusEvent with the @type attribute set to the value startPlacement (referred to as simply the startPlacement event herein) when entering the content binding from any direction or position.
- A PlacementStatusEvent with the @type attribute set to the value endPlacement (referred to as simply the endPlacement event herein) when exiting an announced entered content binding (i.e., the matched startPlacement event).

If a startPlacement event is output, the ADM should output an endPlacement event. Thus, the two PlacementStatusEvent elements enumerated above should be output in pairs. Additionally, these events ***shall no***t be required to be collocated in the same Events element nor the same PlacementStatusNotification message (i.e., the startPlacement event *may* be contained in one message and the endPlacement event *may* be present in a separate message).

An ADM should generate a startPlacement event every time a Placement element's content binding is entered and a startPlacement event for that Placement element is not currently outstanding. Entry into the Placement element's content binding *may* mean crossing the core:Content element's starting content boundary when streaming in the forward direction (i.e. forward play out such as normal play or fast-forward), crossing the content binding's end boundary when streaming in the reverse direction (i.e., rewind), or entry into the middle of the content binding when seeking directly to an initial position within the content (i.e., jumping into the content). It is the NPT element in the Spot element which enables the entry position to be identified.

An endPlacement event should be generated once each time a Placement element's content binding is exited following the output of the startPlacement event (i.e., the startPlacement event should generally always have an endPlacement event and the startPlacement ***shall*** always be output before an endPlacement event is output). An endPlacement event ***shall no***t be output without a corresponding startPlacement event. Thus, for each entry and exit of a Placement element's content binding, a startPlacement and an endPlacement event should be generated assuming a startPlacement event is generated. While an "endEvents" placement operations event (see below) implies an endPlacement event, the ADM should issue explicit endPlacement events for each outstanding startPlacement event to avoid information loss inherent to an impliend endPlacement. An implementation *may* output the endPlacement event before an "endEvents" notification.

For the two above enumerated PlacementStatusEvent events, the placement context ***shall*** be established via the use of the Spot element and any Tracking element(s) supplied in the corresponding Placement element. The Spot element ***shall*** be specified either indirectly within the SpotScopedEvents element or directly within the PlacementStatusEvent element. The Tracking element(s) ***shall*** be specified in the same scope(s) as in the corresponding Placement element. Additionally and when appropriate, the NPT element in a Spot element should be included in the PlacementStatusEvent element for the startPlacement and endPlacement events. Once a Spot element is specified for a given placement context, PlacementStatusEvent elements with NPT elements can be used as an alternative to a NPT element in a Spot element.

The following examples clarify the startPlacement and endPlacement event requirements should the ADM output the startPlacement event. References to NPT ***shall*** be implemented using the NPT element in a Spot element or just an NPT if the Spot element has been previously passed in a PlacementStatusEvent for this placement context.

- If the ADM is streaming in the forward direction (either normal play time (NPT) or a fast-forward mode), the startPlacement event is output when the ad content is entered which has an ad content NPT value of zero and a positive @scale attribute value. When the ad content end boundary is crossed (i.e., the end of the ad), the endPlacement event is output having a non-zero NPT value and a positive @scale attribute value (i.e., the NPT is the ad's ending NPT value). If a new ad immediately follows this exited ad (i.e., back-to-back ads where a different Placement element's content binding is started), a new startPlacement event is output having an NPT of zero and a positive @scale attribute value.
- If the ADM is streaming in the forward direction, the startPlacement event is output when the ad content is entered (just like the previous example). If the user reverses direction (i.e., the content streaming enters rewind mode or reverse play out) and the stream crosses the same ad start boundary in the reverse direction, an endPlacement event is output when the ad's start boundary is crossed (i.e., the placement has been exited). The NPT value is again zero and the @scale attribute has a negative value.
- If the ADM is streaming in the reverse direction (i.e., a negative @scale attribute value) and the ad content's end boundary is crossed (i.e., entering the ad from its end boundary), a startPlacement event is output with the NPT value of the ad content's end time. If streaming continues, when the ad content's start boundary is encountered, an endPlacement event is generated. The NPT value is zero with a negative @scale attribute value. If the end of another ad is immediately entered (i.e., back-to-back ads) then a new startPlacement event is output having a non-zero NPT and negative @scale attribute value.
- If the ADM is streaming in the reverse direction, the startPlacement event is output when the ad content's end boundary is crossed (just like the previous example). If the user reverses direction (i.e., the content streaming enters normal play or fast-forward mode) and the streaming crosses the same ad end boundary, an endPlacement event is output when the placement is exited. The NPT value is again non-zero and the @scale attribute has a positive value.
- If the ADM jumps into the middle of the Placement element's content binding, a startPlacement event is output with a non-zero NPT. The @scale attribute reflects the play out direction. Exiting the Placement element's content binding results in an endPlacement event. It does not matter whether the content's start or end boundary is crossed. The NPT value indicates the exit point.

When core:Tracking elements are contained in the Placement element, they ***shall*** occupy the same relative position when reflected in the Spot element. That is to say, a core:Tracking element appearing at the Placement element scope ***shall*** be echoed back in the same relative position within the Spot element and a core:Tracking element appearing within the Placement element's core:Content element ***shall*** appear in the core:Content element within the Spot element. Figure 36 and Figure 37 illustrates these relationships.

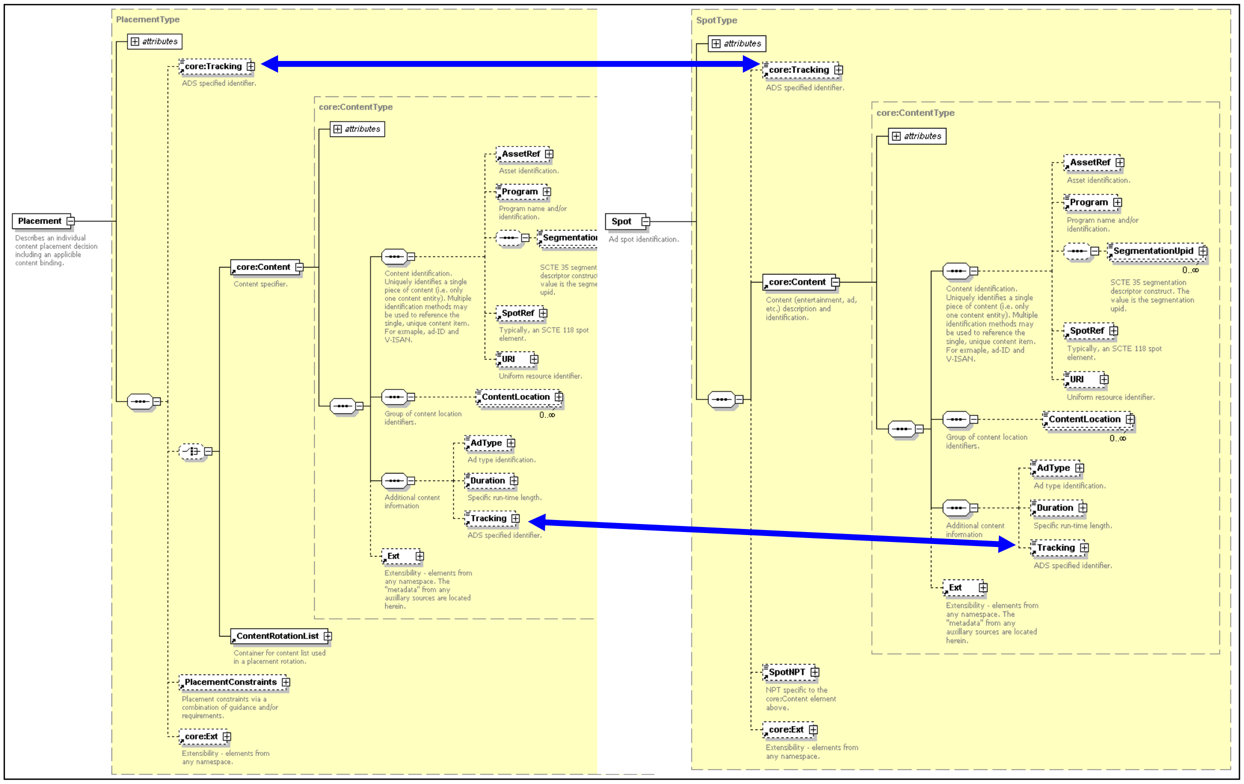

PlacementType
attributes
core:Tracking
ADS specified identifier.
core:ContentType
attributes
AssetRef
Asset identification.
Program
Program name and/or identification.
Segmentatio
SCTE 35 segment descriptor construc value is the segme upid.
Placement
Describes an individual content placement decision including an applicible content binding.
core:Content
Content specifier.
Content identification. Uniquely identifies a single piece of content (i.e. only one content entity). Multiple identification methods may be used to reference the single, unique content item. For exmaple, ad-ID and V-ISAN.
SpotRef
Typically, an SCTE 118 spot element.
URI
Uniform resource identifier.
ContentLocation
0..∞
Group of content location identifiers.
AdType
Ad type identification.
Duration
Specific run-time length.
Additional content information
Tracking
ADS specified identifier.
Ext
Extensibility - elements from any namespace. The "metadata" from any auxillary sources are located herein.
ContentRotationList
Container for content list used in a placement rotation.
PlacementConstraints
Placement constraints via a combination of guidance and/or requirements.
core:Ext
Extensibility - elements from any namespace.
SpotType
attributes
core:Tracking
ADS specified identifier.
core:ContentType
attributes
Spot
Ad spot identification.
AssetRef
Asset identification.
Program
Program name and/or identification.
SegmentationUpid
0..∞
SCTE 35 segmentation descriptor construct. The value is the segmentation upid.
core:Content
Content (entertainment, ad, etc.) description and identification.
Content identification. Uniquely identifies a single piece of content (i.e. only one content entity). Multiple identification methods may be used to reference the single, unique content item. For exmaple, ad-ID and V-ISAN.
SpotRef
Typically, an SCTE 118 spot element.
URI
Uniform resource identifier.
ContentLocation
0..∞
Group of content location identifiers.
AdType
Ad type identification.
Duration
Specific run-time length.
Additional content information
Tracking
ADS specified identifier.
Ext
Extensibility - elements from any namespace. The "metadata" from any auxillary sources are located herein.
SpotNPT
NPT specific to the core:Content element above.
core:Ext
Extensibility - elements from any namespace.

Figure 36: core:Tracking Element Relationships

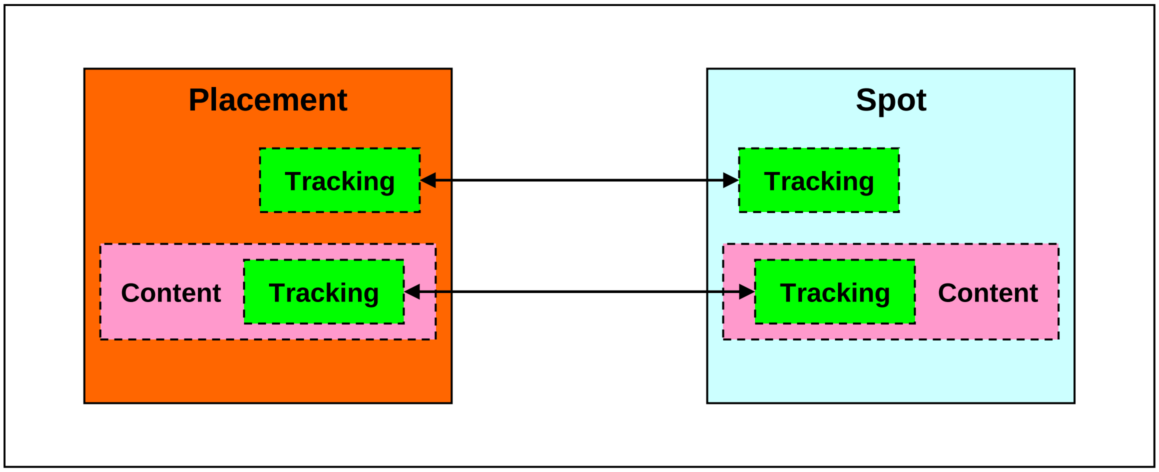


Figure 37: core:Tracking Element Replication in Spot Elements

Anytime an ADS supplies a core:Tracking element, it ***shall*** be contextually referenced by every event as related and appropriate. However, the exact event sequence an ADM ***shall*** output is left to the discretion of the ADM with the exception of the previously stated placement related message minimum.

Beyond startPlacement and endPlacement event reporting, a PlacementStatusEvent element should be used to indicate placement operations completion. Placement operations completion facilitates ADS resource conclusion management. For each PlacementRequest message, the ADM should output at least one placement status event. More than one event *may* be needed when completion signaling occurs in atomic units as signaled by the @type attribute values described below. The number of atomic events required ***shall*** equivalently sum to the value "endAll" described below.

When communicating placement operations completion, a PlacementStatusEvent element ***shall*** include the following attributes:

- An @type attribute having one of the following three values, which *may* be privately extended:
    - "endUpdates"— The PlacementRequest message's placement decision set associated with the @messageRef attribute *may* no longer be altered even if the fulfillment process for the active placement decision set has not completed. Subsequent placement update messaging pertaining to the referenced PlacementRequest message ***shall not*** occur and the ADS is free to release resources associated with the placement decision set. If the PlacementRequest message's @updatesAllowed attribute has the value false, this event signal is not required.
    - "endEvents"— The ADS should not expect to receive any additional named event elements with respect to the PlacementRequest message's referenced placement decision. Any named event elements received following the receipt of this PlacementStatusEvent element *may* be ignored as the ADS is free to release resources associated with the identified PlacementRequest message.
    - "endAll"—Simultaneously indicates both "endUpdates" and "endEvents."
- The @messageRef attribute ***shall*** reference the original PlacementRequest message which initiated the placement management exchange (i.e., the PlacementRequest message's @messageId in an original transmission or the @resend attribute value in a retransmitted message). Consequently, each PlacementRequest message minimally requires a unique placement operations completion event.

The registration-established placement service ADS ***shall*** receive all required placement status events and an ADS ***shall*** be prepared to receive events other than the recommended minimum event element set. An ADM *may* send a PlacementStatusNotification message containing events for more than one ADS. The

ADM *may* group the events together in a per ADS targeted PlayData element or individual named event element *may* contain an ADS destination and this mixture *may* appear in an Events element.

In the case where the PlayData element contains all the events for a single ADS, the PlayData element's @identityADS attribute ***shall*** be set to the destination ADS logical service and all event elements contained with the Events element ***shall*** only be for the specifically identified ADS. If an ADM alternatively mixes the named event elements within the Events element, then an ADM ***shall*** include the @identityADS attribute in each named event element (i.e., the @identityADS attribute ***shall*** be set in each PlacementStatusEvent, SessionEvent, SystemEvent and ViewerEvent element) and the PlayData element's @identityADS attribute ***shall not*** be used. No matter which ADS identity method is used, the ADM ***shall*** include the @identityADS attribute value either in the PlayData element or in each individual named event element.

If the placement decision or placement expires before the PlacementResponse (or PlacementUpdateNotification) message's lifecycle ends, the ADM should (is highly recommended to) perform a placement status message exchange for this and/or other similar conditions.

More information and additional named event elements *may* be included as part of placement status reporting beyond the above recommended minimum event reporting. This inclusion is outside of the scope of this specification.

The interpretation of the placement status event data is outside the scope of this specification.

Figure 38 provides a high-level representation of the major elements comprising the PlacementRequest, PlacementResponse and PlacementStatusNotification messages and it illustrates the event element relationships. Note that Figure 38 does not illustrate all the message elements (i.e., it is not comprehensive or all inclusive). See the individual schemas for additional information.

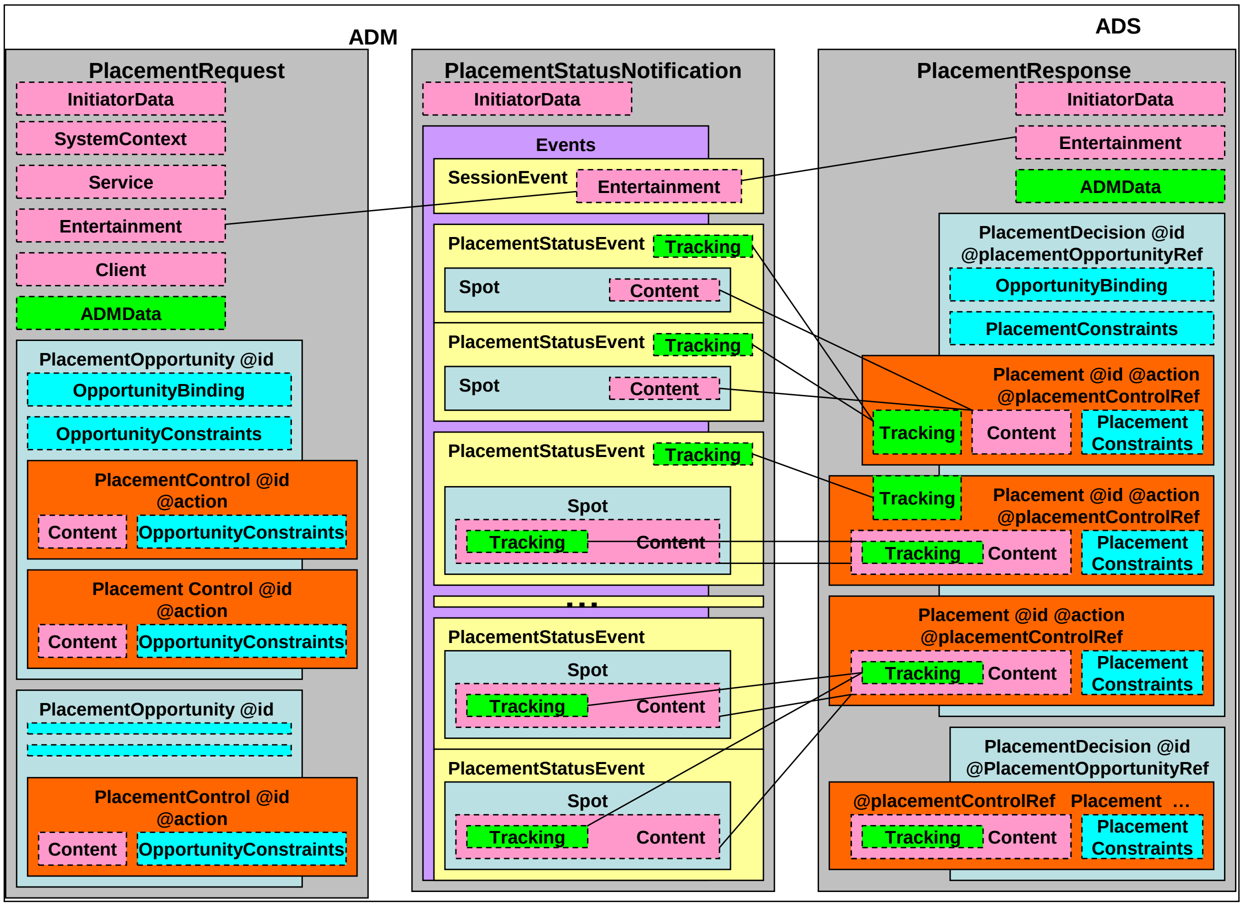


Figure 38: Placement Status High-Level View (not comprehensive)

In Figure 38, the first event (working top down in the diagram) references the Entertainment element. Within the Entertainment element, the core:Content element values should be the same as the PlacementRequest, PlacementResponse, and PlacementUpdateNotification message’s Entertainment element.

The next two event elements reference the first Placement element in the PlacementResponse message. The event is specifically associated with the placement content binding and therefore it ***shall*** contain the Tracking element contained within the Placement element. Since the event references the ad content, it also references the ad’s core:Content element included via the Spot element.

The fourth event is required to contain both Tracking elements as it is referencing a specific content binding in a Placement element where the ADS supplied both Tracking elements. The fifth and sixth events serve to reinforce this relationship.

The following sections detail the placement status messaging schemas.

12.3.1 PlacementStatusNotification Message Schema

Figure 39 illustrates the PlacementStatusNotification message schema used to communicate placement status information.

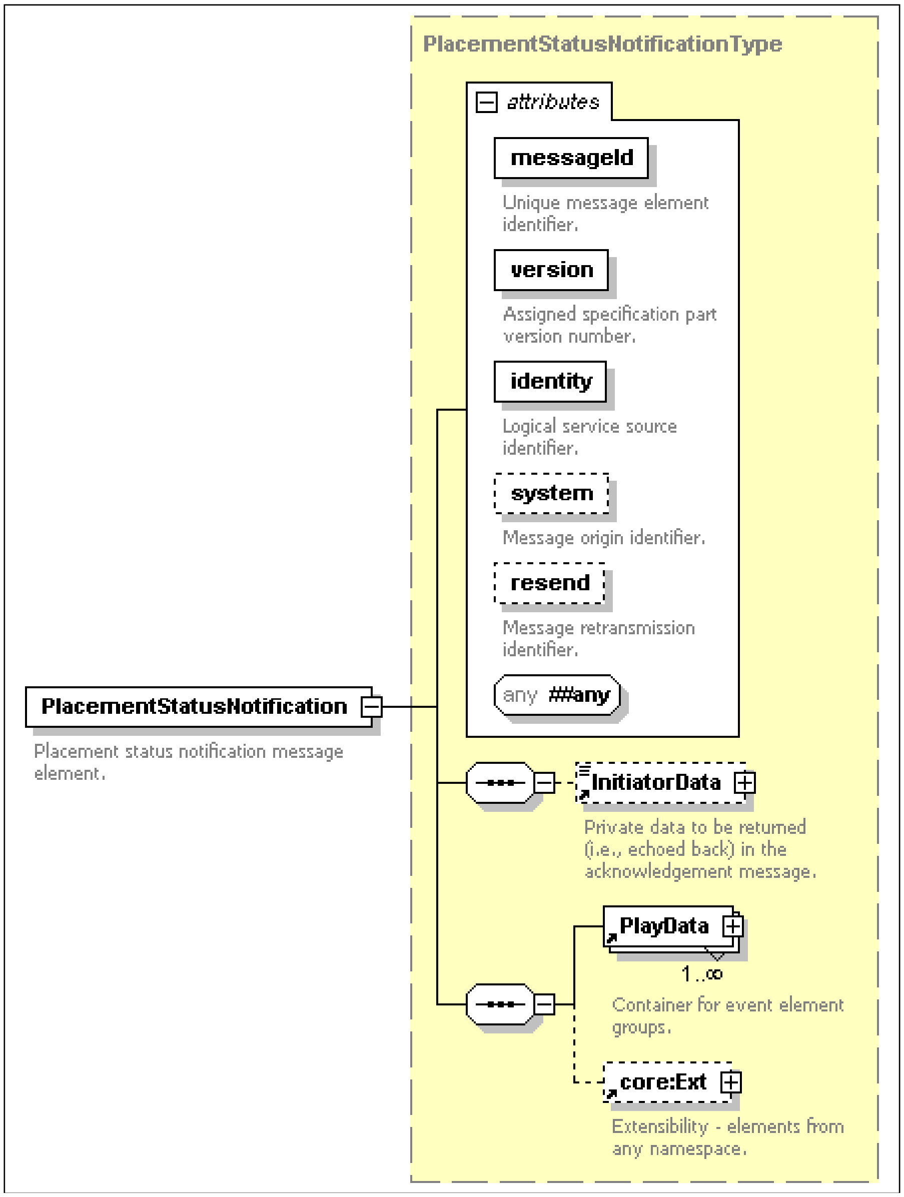

PlacementStatusNotificationType
attributes
messageId
Unique message element identifier.
version
Assigned specification part version number.
identity
Logical service source identifier.
system
Message origin identifier.
resend
Message retransmission identifier.
any ##any
PlacementStatusNotification
Placement status notification message element.
InitiatorData
Private data to be returned (i.e., echoed back) in the acknowledgement message.
PlayData
1..∞
Container for event element groups.
core:Ext
Extensibility - elements from any namespace.

Figure 39: PlacementStatusNotification Message Schema

12.3.1.1 Semantic Definitions for the PlacementStatusNotification Message

@messageId [Required, core:messageIdAttrType]—The message identifier. See [SCTE130-2] for additional information.

@version [Required, core:versionAttrType]—The ADM message interface version. See Section 9.1 for additional information.

@identity [Required, core:identityAttrType]—The origin logical service identifier. See [SCTE130-2] for additional information.

@system [Optional, core:systemAttrType]—The message origin identifier. See [SCTE130-2] for additional information.

@resend [Optional, core:resendAttrType]—The message is a retransmit of a previous PlacementStatusNotification message. The attribute's value is the original message's @messageId value. See [SCTE130-2] for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:InitiatorData [Optional]—The core:InitiatorData element contains implementation specific private data which ***shall*** be returned in the PlacementStatusAcknowledgement message. See [SCTE130-2] for additional information.

PlayData [Optional]—One or more containers of event reporting sequences. See Section 14.35 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

12.3.2 PlacementStatusAcknowledgement Message Schema

Figure 40 illustrates the PlacementStatusAcknowledgement Message Schema.

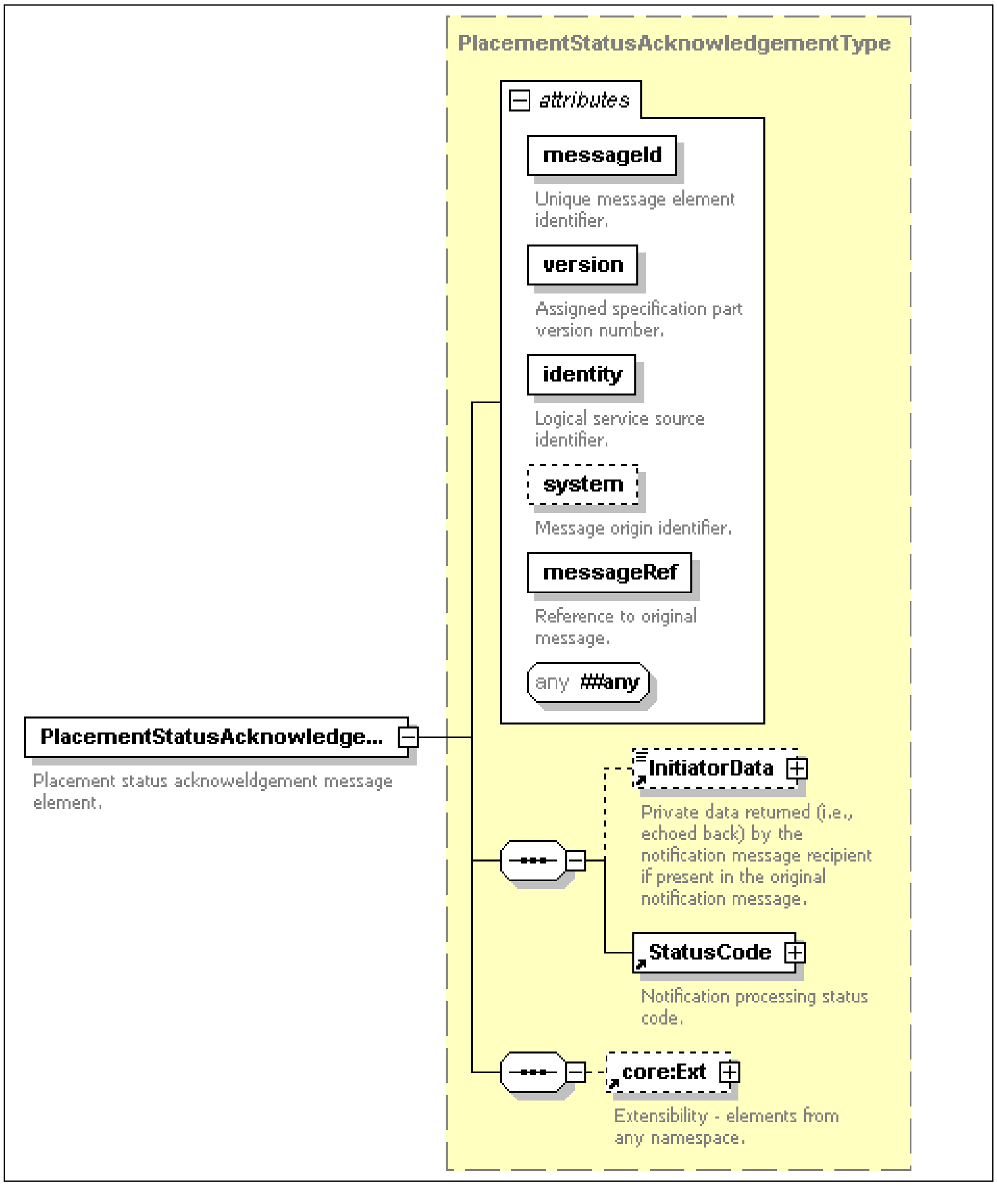

PlacementStatusAcknowledgementType
attributes
messageId
Unique message element identifier.
version
Assigned specification part version number.
identity
Logical service source identifier.
system
Message origin identifier.
messageRef
Reference to original message.
any ##any
PlacementStatusAcknowledge...
Placement status acknoweldgement message element.
InitiatorData
Private data returned (i.e., echoed back) by the notification message recipient if present in the original notification message.
StatusCode
Notification processing status code.
core:Ext
Extensibility - elements from any namespace.

Figure 40: PlacementStatusAcknowledgement Message Schema

12.3.2.1 Semantic Definitions for the PlacementStatusAcknowledgement Message

@messageId [Required, core:messageIdAttrType]—The message identifier. See [SCTE130-2] for additional information.

@version [Required, core:versionAttrType]—The ADM message interface version. See Section 9.1 for additional information.

@identity [Required, core:identityAttrType]—The origin logical service identifier. See [SCTE130-2] for additional information.

@system [Optional, core:systemAttrType]—The message origin identifier. See [SCTE130-2] for additional information.

@messageRef [Required, core:messageRefAttrType]—A reference to the paired PlacementStatusNotification message element initiating this message exchange. The value ***shall*** be the PlacementStatusNotification message's @messageId attribute value. See [SCTE130-2] for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:InitiatorData [Optional]—The core:InitiatorData element ***shall*** be an exact copy of the PlacementStatusNotification message's core:InitiatorData element and ***shall*** only be present if found in the paired notification message. See [SCTE130-2] for additional information.

core:StatusCode [Required]—An applicable status code specific to the PlacementStatusNotification message processing. See [SCTE130-2] and Appendix A for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

12.3.3 Placement Status Examples

Example 32 illustrates a PlacementStatusNotification message reporting the start and end of the first placement and the start of the second placement using self-describing individual events. Note that the example illustrates one way to report events and is not an exhaustive or normative composition. Example 33 illustrates the positive (i.e., success) acknowledgement message.

```
<PlacementStatusNotification messageId="738BE483-4234-4878-8A21-F6B62E1B56D0"
                              version="1.4"
                              identity="ADMLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F4"
                              system="adm"
>
  <PlayData
      identityADS="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5">
    <Events>
      <PlacementStatusEvent type="startPlacement"
                             time="2007-01-09T14:20:00.0Z">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="adsrus.com"
                            assetID="ZXYW0123456789012345"/>
            <core:Tracking>5B8ECCED-979B-421E-8A6E-E2192B11516C</core:Tracking>
          </core:Content>
          <PlayPositions>
            <PlayPositionStart scale="1">
              <NPTOffset>0</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Spot>
      </PlacementStatusEvent>
      <PlacementStatusEvent type="endPlacement"
                             time="2007-01-09T14:20:30.0Z">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="adsrus.com"
                            assetID="ZXYW0123456789012345"/>
            <core:Tracking>5B8ECCED-979B-421E-8A6E-E2192B11516C</core:Tracking>
          </core:Content>
          <PlayPositions>
            <PlayPositionEnd scale="1">
              <NPTOffset>30</NPTOffset>
            </PlayPositionEnd>
          </PlayPositions>
        </Spot>
      </PlacementStatusEvent>
      <PlacementStatusEvent type="startPlacement"
                             time="2007-01-09T14:22:30.0Z">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="adsrus.com"
                            assetID="ZXYW0123456789012346"/>
            <core:Tracking>5B8ECCED-979B-421E-8A6E-E2192B11516D</core:Tracking>
```

```
     </core:Content>
     <PlayPositions>
      <PlayPositionStart scale="1">
       <NPTOffset>0</NPTOffset>
      </PlayPositionStart>
     </PlayPositions>
    </Spot>
   </PlacementStatusEvent>
  </Events>
 </PlayData>
</PlacementStatusNotification>
```

Example 32

```
<PlacementStatusAcknowledgement messageId="87648E84-9D5E-11DB-96CA-
005056C0000a" version="1.4" identity="ADSLogicalService_1170FEF4-C19B-450E-
B624-421B23F525F5" system="ads" messageRef="738BE483-4234-4878-8A21-
F6B62E1B56D0" >
 <core:StatusCode class="0"/>
</PlacementStatusAcknowledgement>
```

Example 33

13.0 SCTE 130 PART 3 ATTRIBUTE TYPES AND COMMON COMPLEX TYPES

The following sections detail the common ADM attributes and complex types utilized by the ADM interface. The common attribute types are detailed first followed by the individual common complex type definitions. These common attributes and complex types are incorporated by the ADM interface elements in subsequent document sections.

13.1 ADM Core Attribute Types

13.1.1 eventRangeEndDateTimeAttrType Attribute Type

eventRangeEndDateTimeAttrType [core:dateTimeTimezoneType]—This attribute type, typically referred to as the @eventRangeEndDateTime attribute, specifies a date and time value. The value ***shall*** be greater than or equal to the latest @time attribute value for any named event element within the container element. By using the @eventRangeStartDateTime and @eventRangeEndDateTime attributes, events *may* be grouped into specific, well-defined time intervals. See [SCTE130-2] for additional information.

13.1.2 eventRangeStartDateTimeAttrType Attribute Type

eventRangeStartDateTimeAttrType [core:dateTimeTimezoneType]—This attribute type, typically referred to as the @eventRangeStartDateTime attribute, specifies a date and time value. The value ***shall*** be less than or equal to the earliest @time attribute value for any named event element within the container element. By using the @eventRangeStartDateTime and @eventRangeEndDateTime attributes, events *may* be grouped into specific, well-defined time intervals. See [SCTE130-2] for additional information.

13.1.3 eventTypeAttrType Attribute Type

eventTypeAttrType [core:nonEmptyStringType]—This attribute type, typically referred to as the @type attribute in an event element, contains a non-empty string identifying the specific event being reported using a named event element. See the individual event elements for additional details.

13.1.4 localServiceRefAttrType Attribute Type

localServiceRefAttrType [core:idAttrType]—This attribute type, typically referred to as the @localServiceRef attribute, contains a core:idAttrType identifying a message local Service element. The attribute provides a cross-reference or linkage identifying the specific Service element within the same message element (i.e., local to the message). The attributes value is the Service element's @id attribute. See the individual elements including this attribute for additional details.

13.1.5 placementActionAttrType Attribute Type

placementActionAttrType [core:nonEmptyStringType]—This attribute type, typically referred to as the @action attribute, conveys the placement execution policy. See the individual including elements for additional details.

13.1.6 placementOpportunitiesExpectedAttrType Attribute Type

placementOpportunitiesExpectedAttrType [xsd:nonNegativeInteger]—This attribute type, typically referred to as the @opportunitiesExpected attribute, communicates the scheduled or planned placement opportunities count relative to the entertainment content. The value zero indicates the attribute's non-usage.

13.1.7 placementOpportunityNumberAttrType Attribute Type

placementOpportunityNumberAttrType [xsd:nonNegativeInteger]—This attribute type, typically referred to as the @opportunityNumber attribute, supplies the placement opportunity identifier or index relative to the entertainment content. Typically, the first placement opportunity is associated with the value one and subsequent placement opportunities increment the index by one. The value zero indicates the attribute's non-usage.

13.1.8 placementOpportunityTypeAttrType Attribute Type

placementOpportunityTypeAttrType [core:nonEmptyStringType]—This attribute type, typically referred to as the @opportunityType attribute, identifies the placement opportunity type. Typically, this is a location relative to the entertainment content for scheduled or planned placement opportunities, or the trigger type for spontaneous opportunities. For the later, pause is an example. The Table 9 values are defined and they ***shall*** appear in an XML document exactly as they appear in the table. Additional values *may* be extended by private agreement outside the scope of this specification.

Value	Description
preRoll	Prior to the start of the entertainment content.
midRoll	Middle of the entertainment content.
postRoll	Post the end of the entertainment content.
interstitial	Between entertainment content.
pause	An opportunity occurring when a display endpoint interaction pauses the entertainment content viewing (e.g., the user pushes the pause button).
fastForward	An opportunity occurring when a display endpoint interaction fast-forwards the entertainment content viewing (e.g., the user pushes the fast-forward button).
rewind	An opportunity occurring when a display endpoint interaction rewinds the entertainment content viewing (e.g., the user pushes the rewind button).
startSession	At the commencement of the implementation defined session initiation boundary.
endSession	At the termination of the implementation defined session completion boundary.
…	User defined and outside the scope of this specification. The string ***shall*** be prefixed with the text “private:”.

Table 9: placementOpportunityTypeAttrType Attribute Values

13.1.9 positionAttrType Attribute Type

positionAttrType [xsd:nonNegativeInteger]—This attribute type, typically referred to as the @position attribute, identifies the temporal location of the element when the reference unit is subdivided. For example, the value *may* be either the positional placement opportunity’s location within a subdivided placement opportunity or a placement’s location within a placement decision. The value zero is reserved and ***shall not*** be used.

13.2 ADM Core Common Complex Types

This section defines the common XML complex type definitions used by multiple elements contained within this specification. These common definitions function as the base element definitions for individual elements and are located here in order to facilitate a single normative definition location. Elements inheriting from this these definitions ***shall*** reference back to these subsections as appropriate.

13.2.1 EventBaseType Complex Type

The EventBaseType complex type is the base definition for all named event elements.

13.2.1.1 EventBaseType Schema

Figure 41 illustrates the EventBaseType complex type's schema.

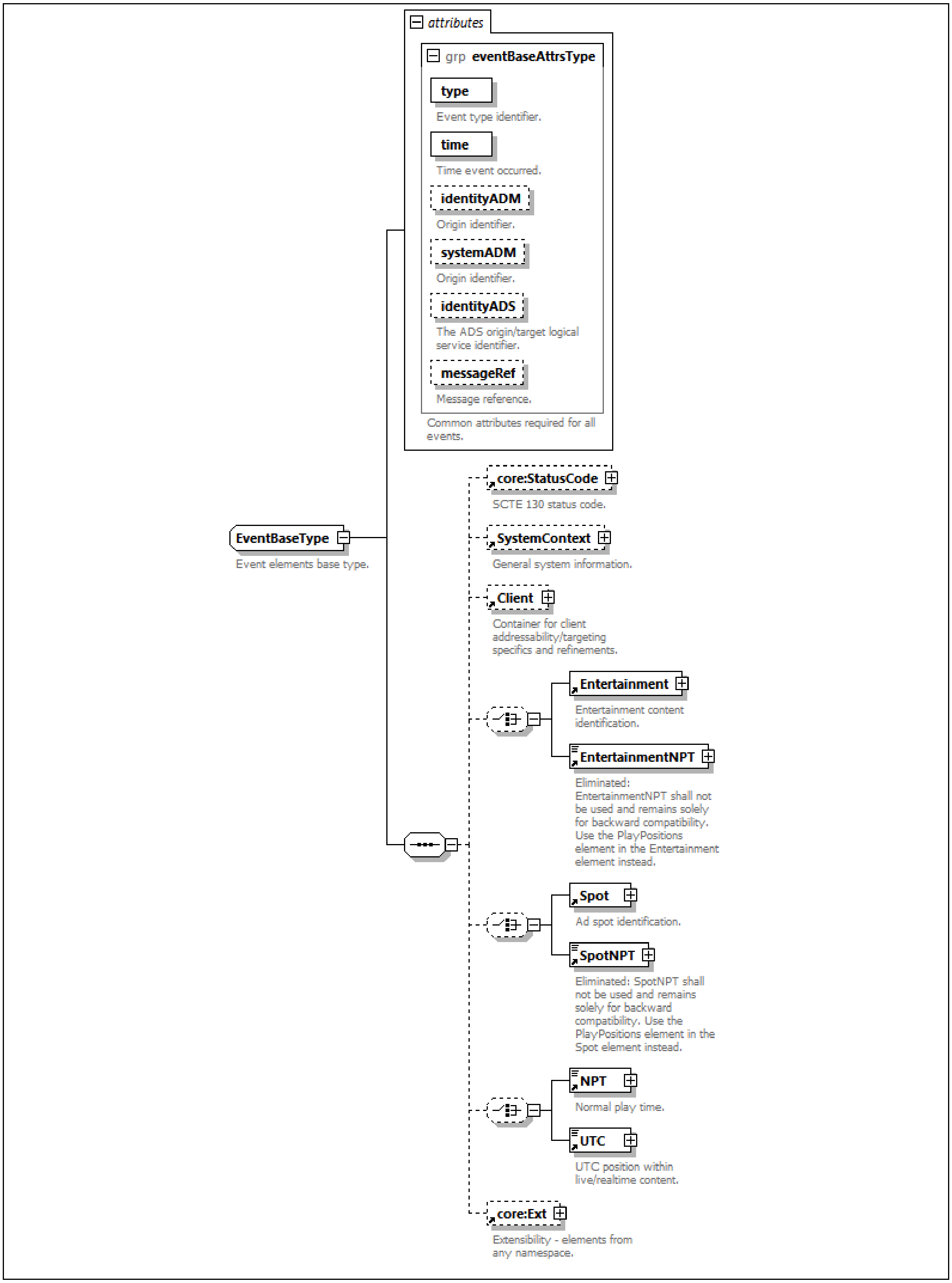

attributes
grp eventBaseAttrsType
type
Event type identifier.
time
Time event occurred.
identityADM
Origin identifier.
systemADM
Origin identifier.
identityADS
The ADS origin/target logical service identifier.
messageRef
Message reference.
Common attributes required for all events.
EventBaseType
Event elements base type.
core:StatusCode
SCTE 130 status code.
SystemContext
General system information.
Client
Container for client addressability/targeting specifics and refinements.
Entertainment
Entertainment content identification.
EntertainmentNPT
Eliminated: EntertainmentNPT shall not be used and remains solely for backward compatibility. Use the PlayPositions element in the Entertainment element instead.
Spot
Ad spot identification.
SpotNPT
Eliminated: SpotNPT shall not be used and remains solely for backward compatibility. Use the PlayPositions element in the Spot element instead.
NPT
Normal play time.
UTC
UTC position within live/realtime content.
core:Ext
Extensibility - elements from any namespace.

Figure 41: EventBaseType Complex Type Schema

13.2.1.2 Semantic Definitions for EventBaseType Complex Type

@type [Required, eventTypeAttrType]—Event identification. See Section 13.1.3 for additional information.

@time [Required, core:dateTimeTimezoneType]—The event occurrence date and time. See [SCTE130-2] for additional information.

@identityADM [Optional, core:identityAttrType]—The originating ADM logical service identifier. See [SCTE130-2] for additional information. This attribute is typically used when an ADS or other event receiver is aggregating events from multiple ADM sources and the value is needed in order to clearly identify the ADM event source. Generally, the ADM does not need to supply this value as the ADM PlacementStatusNotification message's @identity attribute is adequate.

@systemADM [Optional, core:systemAttrType]—The originating ADM system identification. See [SCTE130-2] for additional information.

@identityADS [Optional, core:identityAttrType]—The ADS origin or target logical service identifier. See [SCTE130-2] for additional information. The target ADS identity ***shall*** be present in either the PlayData element or in each named event element and this attribute's usage is mutually exclusive (i.e., either the PlayData element's attribute or this attribute is used).

@messageRef [Optional, core:messageRefAttrType]—The @messageId attribute's value for a message associated with this event. See [SCTE130-2] for additional information.

core:StatusCode [Optional]—An applicable status code and zero or more core:Note elements. The applicable status codes are defined in Appendix A. See [SCTE130-2] for additional information.

SystemContext [Optional]—Parameters describing the event's system context. See Section 14.52 for additional information. If this element is present in a higher level scope element, it ***shall not*** be used in the named event element (i.e., it ***shall not*** appear in the named event element).

Client [Optional]—Addressability or targeting details. See Section 14.4 for additional information. If this element is present in a higher level scope element, it ***shall not*** be used in the named event element (i.e., it ***shall not*** appear in the named event element).

Entertainment [Optional Choice]—Entertainment content asset identification. See Section 14.11 for additional information. If this element is present in a higher level scope element, it ***shall not*** be used in the named event element (i.e., it ***shall not*** appear in the named event element). If the Entertainment element is present in a higher level scope element and a time context relative to the higher level scope Entertainment element is desired, the NPT or UTC element ***shall*** be used instead.

EntertainmentNPT [Optional Choice]—This element ***shall not*** be used and remains for backward compatibility only. In messages having a @version attribute of "1.3" or prior, this element *may* appear. It ***shall not*** be used in messages having an @version attribute of "1.4" or later. An entertainment specific NPT value is provided using the NPTOffset element contained with the Entertainment element.

Spot [Optional Choice]—Ad content asset identification. See Section 14.49 for additional information. If this element is present in a higher level scope element, it ***shall not*** be used in the named event element (i.e., it ***shall not*** appear in the named event element). If the Spot element is present in a higher level scope element and an NPT relative to the higher level scope Spot element is desired, the NPT element ***shall*** be used instead.

SpotNPT [Optional Choice]—This element ***shall not*** be used and remains for backward compatibility only. In messages having a @version attribute of "1.3" or prior, this element *may* appear. It ***shall not*** be used in messages having an @version attribute of "1.4" or later. A spot specific NPT value is provided using the NPTOffset element contained with the Spot element.

NPT [Optional Choice]—A normal play time (NPT) value referenced to a stream absolute position relative to the beginning of a timeline. See Section 14.22 for additional information. The timeline is typically the clock the viewer associates with a program or entertainment asset including ads.

UTC [Optional Choice]—A UTC value referenced to a stream absolute position within live/realtime content. See Section 14.56 for additional information. The UTC is typically the clock the viewer associates with a live/realtime program or entertainment asset including ads that are delivered as part of live/realtime content.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

The named event element’s value *may* be empty.

13.2.2 NormalPlayTimeType Complex Type

The normal play time (NPT) complex type indicates the stream absolute position relative to the beginning of the referenced content with the basis being adopted from RFC2326. See [RFC2326] for additional information. Typically, this is the clock the viewer associates with the referenced content stream.

13.2.2.1 NormalPlayTimeType Schema

Figure 42 illustrates the NormalPlayTimeType complex type's schema.

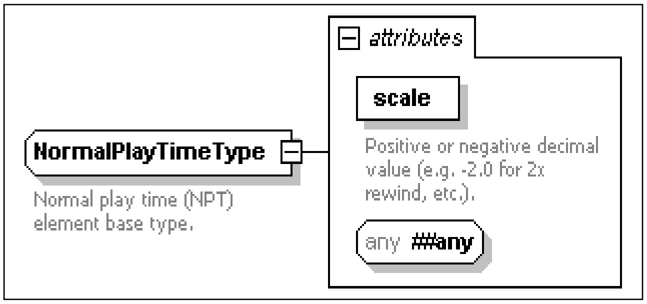

attributes
scale
Positive or negative decimal value (e.g. -2.0 for 2x rewind, etc.).
any ##any
NormalPlayTimeType
Normal play time (NPT) element base type.

Figure 42: NormalPlayTimeType Complex Type Schema

13.2.2.2 Semantic Definitions for the NormalPlayTimeType Complex Type

@scale [Required, xsd:decimal]—The scale value of 1 indicates normal play or record at the natural forward viewing rate. If the attribute is not the value of 1, the value corresponds to the rate with respect to the normal viewing rate. For example, a ratio of 2 indicates twice the natural viewing rate (i.e., "fast forward") and a ratio of 0.5 indicates half the normal viewing rate (i.e., forward at ½ or “slow forward”). In other words, a ratio of 2 has normal playtime increase at twice the wallclock rate. For every second of elapsed (wallclock) time, 2 seconds of content is enumerated. A negative value indicates reverse direction (i.e., “rewind”). For example, the value “-5.0” indicates fives times the normal viewing rate in reverse.

@##any [Optional]—Additional attributes from any namespace.

The NormalPlayTimeType complex type's value is of type core:nonEmptyStringType and it ***shall*** contain either a single NPT time value or a pair of NPT time values separated by a single hyphen with the later describing a range. A NPT time value ***shall*** consist of a decimal fraction with the part left of the decimal expressed in seconds and the part to the right of the decimal measuring fractions of a second. The beginning of the content corresponds to the value BOS (Beginning Of Stream) or 0.0 seconds. The end of the content corresponds to the value EOS (End Of Stream) or the specific number of seconds in the content. Negative values ***shall not*** be permitted and the NPT element's value ***shall not*** be empty.

13.2.3 PlacementCountType Complex Type

The PlacementCountType complex type specifies a placement count.

13.2.3.1 PlacementCountType Complex Type Schema

Figure 43 illustrates the PlacementCountType complex type's schema.

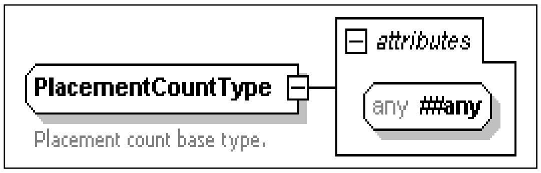


Figure 43: PlacementCountType Complex Type Schema

13.2.3.2 Semantic Definitions for the PlacementCountType Complex Type

@##any [Optional]—Additional attributes from any namespace.

The PlacementCountType complex type's value is of type xsd:nonNegativeInteger. The value zero indicates no placements are desired.

14.0 SCTE 130 PART 3 ELEMENTS

The following sections detail the ADM elements utilized by the previously described ADM interface messages. The individual ADM elements *may* incorporate the previously defined common attribute and complex types in their individual element definitions. Elements utilizing those definitions ***shall*** reference back to those subsections as appropriate.

14.1 ADMCapabilities Element

The ADMCapabilities element allows an ADM to communicate any unique abilities or limitations.

14.1.1 ADMCapabilities Element Schema

Figure 44 illustrates the ADMCapabilities element's schema.

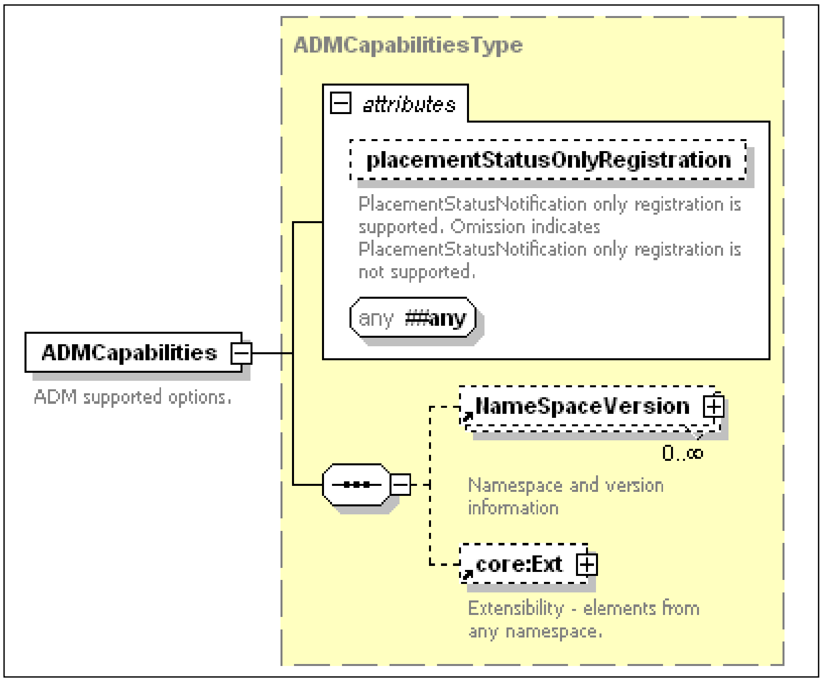

ADMCapabilitiesType
attributes
placementStatusOnlyRegistration
PlacementStatusNotification only registration is supported. Omission indicates PlacementStatusNotification only registration is not supported.
any ##any
ADMCapabilities
ADM supported options.
NameSpaceVersion
0..∞
Namespace and version information
core:Ext
Extensibility - elements from any namespace.

Figure 44: ADMCapabilities Element Schema

14.1.1.1 Semantic Definitions for the ADMCapabilities Element

placementStatusOnlyRegistration [Optional, xsd:boolean]— The value true indicates the ADM supports PlacementStatusNotification message registration only. The value false indicates the ADM does not support PlacementStatusNotification only registration. The attribute's default value is false and if the attribute is omitted, the default value ***shall*** be in effect. Thus, the ADM does not by default support the PlacementStatusNotification only registration.

@##any [Optional]—Additional attributes from any namespace.

NameSpaceVersion [Optional] – The NameSpaceVersion elements contain the list of namespace versions actually supported based on the namespaces included in the ListADMServicesResponse. See Section 14.20 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.1.2 ADMCapabilities Element Examples

Additional examples can be found under ListADMRegistrationResponse (See Section 11.2.3).

```
<ADMCapabilities placementStatusOnlyRegistration="false"/>
<ADMCapabilities placementStatusOnlyRegistration="true">
  < NameSpaceVersion
     namespace="http://www.scte.org/schemas/130-3/2013/adm/podm"
     version="1.1"/>
</ADMCapabilities>
```

14.2 ADMData Element

The ADMData element provides carriage for privately defined attributes and elements that ***shall*** be returned in a response or notification message exactly as received (i.e., echoed back). For example, this element *may* be used to provide high resiliency services or other value added features.

14.2.1 ADMData Element Schema

Figure 45 illustrates the ADMData element’s schema.

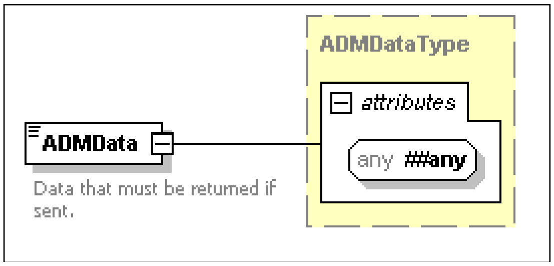


Figure 45: ADMData Element Schema

14.2.1.1 Semantic Definitions for the ADMData Element

@##any [Optional]—Additional attributes from any namespace.

The ADMData element’s value is of type xsd:string. The ADMData element’s content is private. If the ADMData element is present, it ***shall*** be provided exactly as supplied. The ADMData element’s value *may* be empty when all information is communicated using attributes.

14.2.2 ADMData Element Examples

```
<ADMData acme:secretSauce="true">Hidden secret sauce...</ADMData>
```

14.3 Channel Element

The Channel element provides channel identification. For example, the Channel element *may* identify a specific combination of network and zone in a traditional linear model, or it *may* be a synonym for a DVB service or an MPEG-2 Transport Stream program.

14.3.1 Channel Element Schema

Figure 46 illustrates the Channel element's schema.

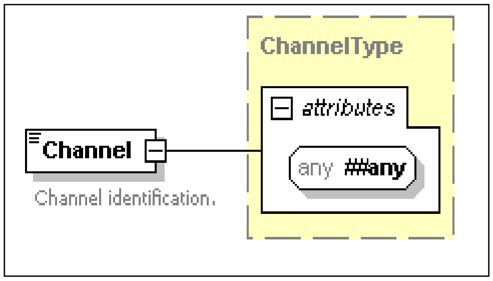


Figure 46: Channel Element Schema

14.3.1.1 Semantic Definitions for the Channel Element

@##any [Optional]—Additional attributes from any namespace.

The Channel element's value is of type core:nonEmptyStringType and contains a channel identifier. The element's value ***shall not*** be empty.

14.3.2 Channel Element Examples

```
<Channel>ESPN</Channel>
<Channel>7</Channel>
<Channel>FoxSportsWest</Channel>
```

14.4 Client Element

The Client element contains addressing or targeting criteria.

14.4.1 Client Element Schema

Figure 47 illustrates the Client element's schema.

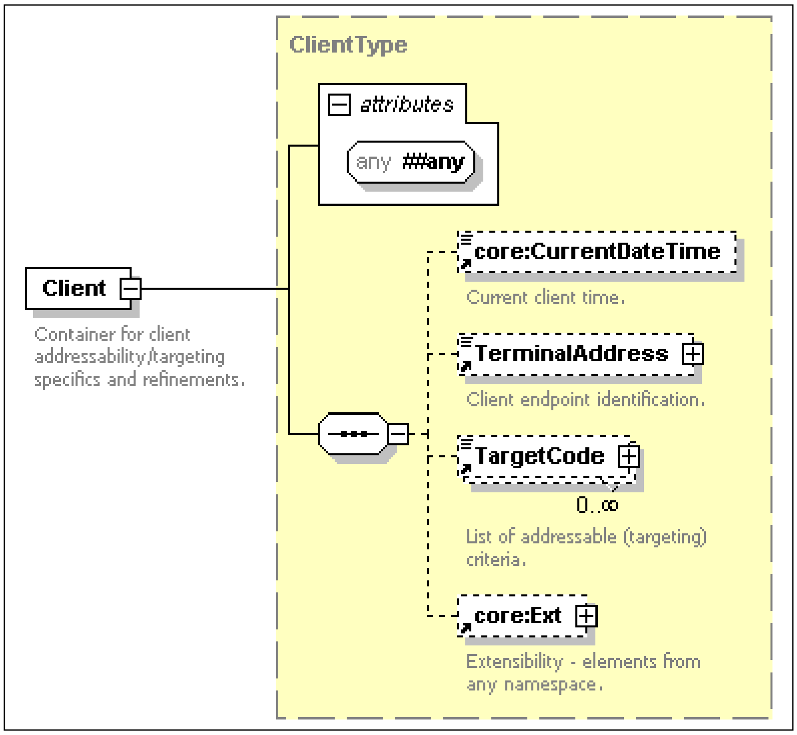

ClientType
attributes
any ##any
Client
Container for client addressability/targeting specifics and refinements.
core:CurrentDateTime
Current client time.
TerminalAddress
Client endpoint identification.
TargetCode
0..∞
List of addressable (targeting) criteria.
core:Ext
Extensibility - elements from any namespace.

Figure 47: Client Element Schema

14.4.1.1 Semantic Definitions for the Client Element

@##any [Optional]—Additional attributes from any namespace.

core:CurrentDateTime [Optional]—The current client or end device date and time. The value ***shall*** be expressed in the client's local time and it ***shall*** include the timezone indicator. See [SCTE130-2] for additional information.

TerminalAddress [Optional]—Client endpoint identification such as the set top box MAC address. See Section 14.55 for additional information.

TargetCode [Optional]—Zero or more TargetCode elements with each element containing addressability information (i.e., targeting criteria). See Section 14.54 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.4.2 Client Element Examples

```
<Client>
  <core:CurrentDateTime>2007-01-05T12:30:36.0-8:00</core:CurrentDateTime>
  <TerminalAddress type="MAC">03-55-78-90-03-02</TerminalAddress>
  <TargetCode key="ZipPlus4">97363-8993</TargetCode>
  <TargetCode key="Gender">Female</TargetCode>
</Client>
```

14.5 ContentProvider Element

The ContentProvider element identifies a content source and optionally an associated content data model.

14.5.1 ContentProvider Element Schema

Figure 48 illustrates the ContentProvider element's schema.

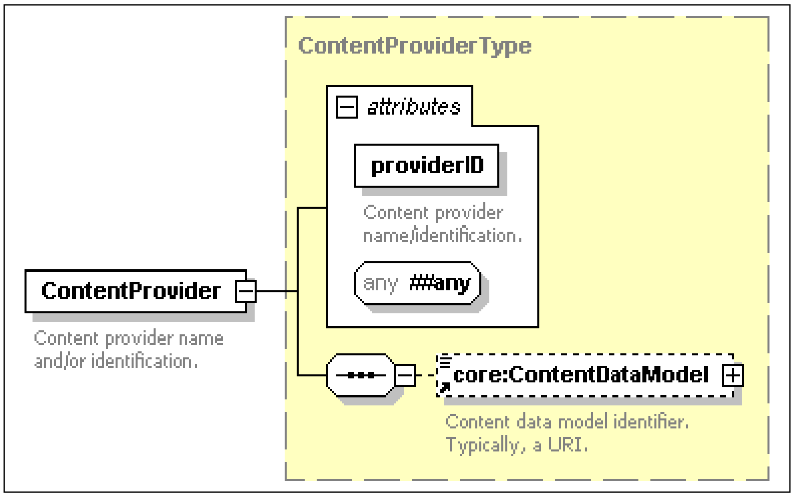

ContentProviderType
attributes
providerID
Content provider name/identification.
any ##any
ContentProvider
Content provider name and/or identification.
core:ContentDataModel
Content data model identifier. Typically, a URI.

Figure 48: ContentProvider Element Schema

14.5.1.1 Semantic Definitions for the ContentProvider Element

@providerID [Required, core:NonEmptyStringType]—Content source identification. Typically, this is a CableLabs ADI providerID value.

@##any [Optional]—Additional attributes from any namespace.

core:ContentDataModel [Optional]—Content data model identification for the content source. See [SCTE130-2] for additional information.

The ContentProvider element's value *may* be empty.

14.5.2 ContentProvider Element Examples

```
<ContentProvider providerID="jumpthesharktv"/>

<ContentProvider providerID="mybettercontent.com">
  <core:ContentDataModel type="SCTE236">valid URI</core:ContentDataModel>
</ContentProvider>
```

14.6 ContentRotationList Element

The ContentRotationList element contains two or more core:Content elements used to fulfill a single placement decision. How the ADM utilizes the list is outside the scope of this specification.

14.6.1 ContentRotationList Element Schema

Figure 49 illustrates the ContentRotationList element's schema.

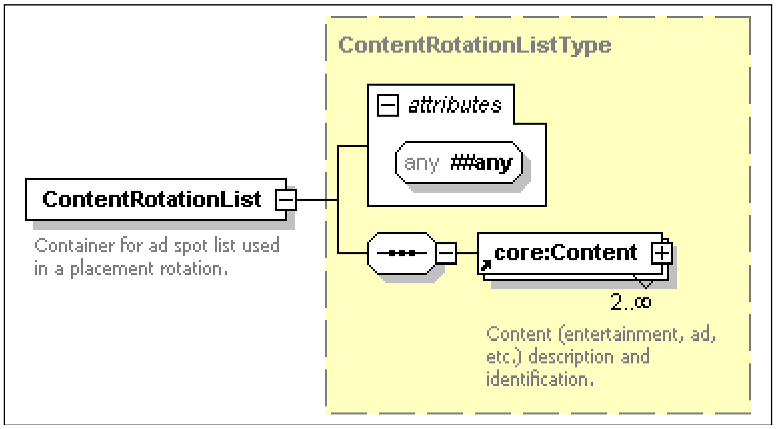

ContentRotationListType
attributes
any ##any
ContentRotationList
Container for ad spot list used in a placement rotation.
core:Content
2..∞
Content (entertainment, ad, etc.) description and identification.

Figure 49: ContentRotationList Element Schema

14.6.1.1 Semantic Definitions for the ContentRotationList Element

@##any [Optional]—Additional attributes from any namespace.

core:Content [Required]—Two or more elements describing a set of content assets applicable to a single placement decision binding. See [SCTE130-2] for additional information.

14.6.2 ContentRotationList Element Examples

```
<ContentRotationList>
  <core:Content id="Ad 1">
    <core:AssetRef providerID="adco.com" assetID="ITEM1012345678901234"/>
    <core:Tracking>ContentTrackingID1</core:Tracking>
  </core:Content>
  <!--N more Content element entries.-->
  <core:Content id="Ad N">
    <core:AssetRef providerID="adco.com" assetID="ITEM2012345678901239"/>
    <core:Tracking>ContentTrackingIDN</core:Tracking>
  </core:Content>
</ContentRotationList>
```

14.7 DesiredDuration Element

The DesiredDuration element describes an ideal or targeted run-time in years, months, days, hours, minutes, seconds, and milliseconds. This element is typically used in conjunction with the MaxDuration and MinDuration elements as it allows the interface to express the ideal duration for the context. See Section 14.16 and Section 14.18 for additional information regarding the MaxDuration and MinDuration elements respectively.

An example usage is for a linear placement opportunity which ideally is two minutes in duration (i.e., four 30 second ads for example). However, the placement execution implementation *may* allow small variances since a perfect two minutes *may* be difficult to achieve. The two minute value is the target value for the ad decision service but the minimum and maximum time, specified using the MinDuration and MaxDuration elements, formally places constraints on the range bounds. Precise duration values are supplied using the Duration element. See [SCTE130-2] for additional information regarding the Duration element.

14.7.1 DesiredDuration Element Schema

Figure 50 illustrates the DesiredDuration element’s schema.

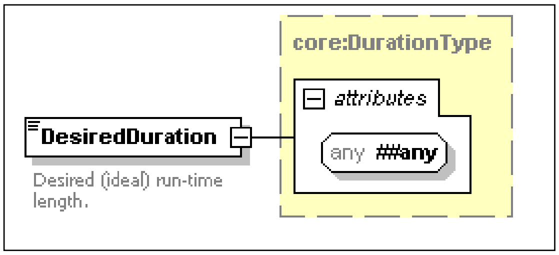


Figure 50: DesiredDuration Element Schema

14.7.1.1 Semantic Definitions for the DesiredDuration Element

@##any [Optional]—Additional attributes from any namespace.

The DesiredDuration element's value is of type xsd:duration where the format is PnYnMnDTnHnMn.nnnS.

14.7.2 DesiredDuration Element Examples

```
<DesiredDuration>PT30S</DesiredDuration> <!--30 seconds-->
<DesiredDuration>PT2M</DesiredDuration> <!--2 minutes-->
```

14.8 DesiredPlacementCount Element

The DesiredPlacementCount element specifies an ideal or targeted placement count. This element is typically used in conjunction with the MaxPlacementCount and MinPlacementCount elements as it allows the interface to express the ideal placement count for the context. See Section 14.17 and Section 14.19 for additional information regarding the MaxPlacementCount and MinPlacementCount elements respectively.

14.8.1 DesiredPlacementCount Element Schema

Figure 51 illustrates the DesiredPlacementCount element's schema.

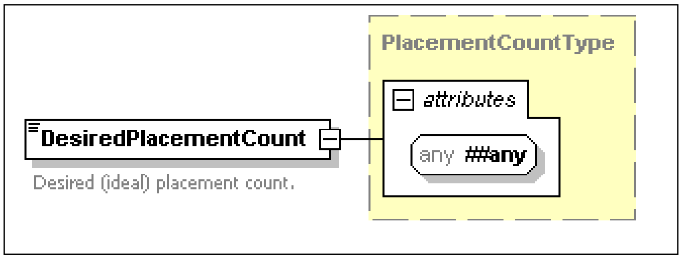


Figure 51: DesiredPlacementCount Element Schema

14.8.1.1 Semantic Definitions for the DesiredPlacementCount Element

The DesiredPlacementCount element inherits from the PlacementCountType complex type. See Section 13.2.3 for semantic definitions applicable to this element.

14.8.2 DesiredPlacementCount Element Examples

```
<DesiredPlacementCount>1</DesiredPlacementCount>
<DesiredPlacementCount>3</DesiredPlacementCount>
```

14.9 EffectiveEndDateTime Element

The EffectiveEndDateTime element is a date and time when the container element becomes invalid or cancelled. For example, the date and time when a placement decision expires. For a container element whose schema includes this element, if the element is omitted, the container never expires. An implementer should focus careful attention when implementing this element as it *may* be used to enforce business constraints.

14.9.1 EffectiveEndDateTime Element Schema

Figure 52 illustrates the EffectiveEndDateTime element's schema.

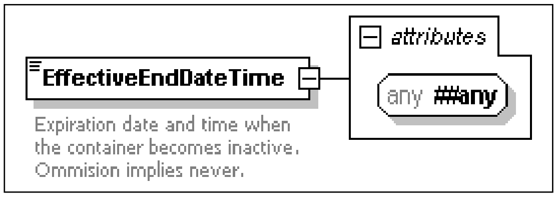

attributes
EffectiveEndDateTime
any ##any
Expiration date and time when the container becomes inactive. Ommision implies never.

Figure 52: EffectiveEndDateTime Element Schema

14.9.1.1 Semantic Definitions for the EffectiveEndDateTime Element

@##any [Optional]—Additional attributes from any namespace.

The EffectiveEndDateTime element's value is of type core:dateTimeTimezoneType. See [SCTE130-2] for additional information.

14.9.2 EffectiveEndDateTime Element Examples

```
<EffectiveEndDateTime>2009-04-03T12:00:00.0Z</EffectiveEndDateTime>
```

14.10 EffectiveStartDateTime Element

The EffectiveStartDateTime element is a date and time when the container element becomes valid or active. For example, this element *may* accompany preloaded placement decisions to indicate when the decisions become active. For a container element whose schema includes this element, if the element is omitted, the container is active immediately. An implementer should focus careful attention when implementing this element as it *may* be used to enforce business constraints.

14.10.1 EffectiveStartDateTime Element Schema

Figure 53 illustrates the EffectiveStartDateTime element's schema.

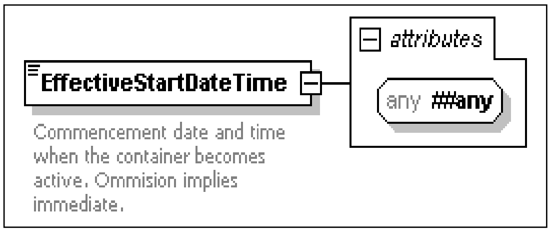

EffectiveStartDateTime
attributes
any ##any
Commencement date and time when the container becomes active. Ommision implies immediate.

Figure 53: EffectiveStartDateTime Element Schema

14.10.1.1 Semantic Definitions for the EffectiveStartDateTime Element

@##any [Optional]—Additional attributes from any namespace.

The EffectiveStartDateTime element's value is of type core:dateTimeTimezoneType. See [SCTE130-2] for additional information.

14.10.2 EffectiveStartDateTime Element Examples

```
<EffectiveStartDateTime>2009-04-03T12:00:00.0Z</EffectiveStartDateTime>
```

14.11 Entertainment Element

The Entertainment element identifies the entertainment (programming) content using the core:Content element and it optionally contains a PlayPositions element. The PlayPositions element provides the ability to specify a start and/or end marker via a signal identifier, UTC time or NPT offset.

14.11.1 Entertainment Element Schema

Figure 54 illustrates the Entertainment element's schema.

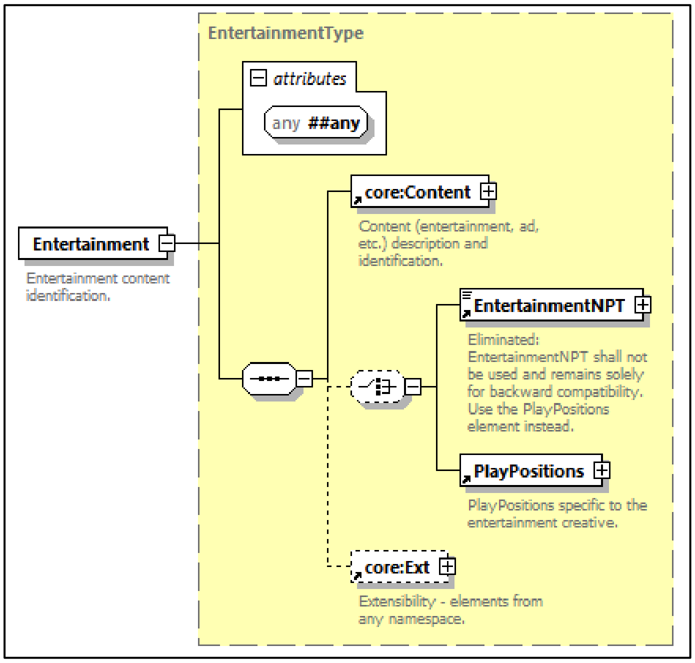

EntertainmentType
attributes
any ##any
core:Content
Content (entertainment, ad, etc.) description and identification.
Entertainment
Entertainment content identification.
EntertainmentNPT
Eliminated: EntertainmentNPT shall not be used and remains solely for backward compatibility. Use the PlayPositions element instead.
PlayPositions
PlayPositions specific to the entertainment creative.
core:Ext
Extensibility - elements from any namespace.

Figure 54: Entertainment Element Schema

14.11.1.1 Semantic Definitions for the Entertainment Element

@##any [Optional]—Additional attributes from any namespace.

core:Content [Required]—Entertainment content identification. See [SCTE130-2] for additional information.

EntertainmentNPT [Optional]—This element ***shall not*** be used and remains for backward compatibility only. In messages having a @version attribute of "1.3" or prior, this element *may* appear. It ***shall not*** be used in messages having an @version attribute of "1.4" or later. An entertainment specific NPT value is provided using the NPTOffset element contained with the PlayPositions element.

PlayPositions [Optional]—PlayPositions relative to the paired core:Content element. See Section 0 for additional information.

The Entertainment element *may* contain a PlayPositions element. This usage facilitates describing descrete points or a range within the entertainment content using a heterogenous mix of positional values.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.11.2 Entertainment Element Examples

```
<Entertainment>
  <core:Content>
    <core:AssetRef providerID="provider.com"
                 assetID="TNTA0000000000000000"/>
  </core:Content>
  <PlayPositions>
    <PlayPositionStart>
      <!-- start play at 11am GMT. -->
      <UTC scale="1">2010-11-09T11:00:00-00:00</UTC>
    </PlayPositionStart>
    <PlayPositionEnd>
      <!-- Signal ID of the out point. -->
      <SignalId>SIGNAL:ISKQWoVaEeGB5Z8NCkgkAZs=</SignalId>
      <!-- UTC time of the out point. -->
      <UTC scale="1.0">2010-11-09T11:12:10.123Z</UTC>
    </PlayPositionEnd>
  </PlayPositions>
</Entertainment>
```

In the above Entertainment element example, the AssetRef element identifies the entertainment content asset. The PlayPostitionStart element provides a wall clock date/time marker within the asset. The PlayPositionEnd element provides both a wall clock date/time marker and a signal ID within the asset.

This example shows one way to utilize the PlayPositionStart and PlayPositionEnd elements within live content that was recorded. While the signal IDs provide frame accurate positioning, the provided UTC time codes provide wall clock information to the ADM so an accurate progress bar can be calculated for the region of interest identified by the PlayPositions element.

The following example illustrates how the NPT range is represented usingNPTOffset and PlayPositions.

```
<Entertainment>
  <core:Content>
    <core:AssetRef providerID="provider.com"
                assetID="TNTA0000000000000000"/>
  </core:Content>
  <PlayPositions>
    <PlayPositionStart scale="1.0">
      <NPTOffset>15.539</NPTOffset>
    </PlayPositionStart>
    <PlayPositionEnd>
      <NPTOffset>150.714</NPTOffset>
    </PlayPositionEnd>
  </PlayPositions>
</Entertainment>
```

14.12 EntertainmentNPT Element (Eliminated)

The EntertainmentNPT element ***shall not*** be used and remains for backward compatibility only. In messages having a @version attribute of "1.3" or prior, this element *may* appear. It ***shall not*** be used in messages having an @version attribute of "1.4" or later. See the PlayPositions element in Section 0 for additional information on how to carry an Entertainment specific NPT value.

14.13 EntertainmentScopedEvents Element

The EntertainmentScopedEvents element reduces named event element size and information replication. This information reduction occurs by allowing the Entertainment element to be declared once and the subsequent named event elements contained within the Events and the SpotScopedEvents elements to indirectly reference the Entertainment element through this scoping.

14.13.1 EntertainmentScopedEvents Element Schema

Figure 55 illustrates the EntertainmentScopedEvents element’s schema.

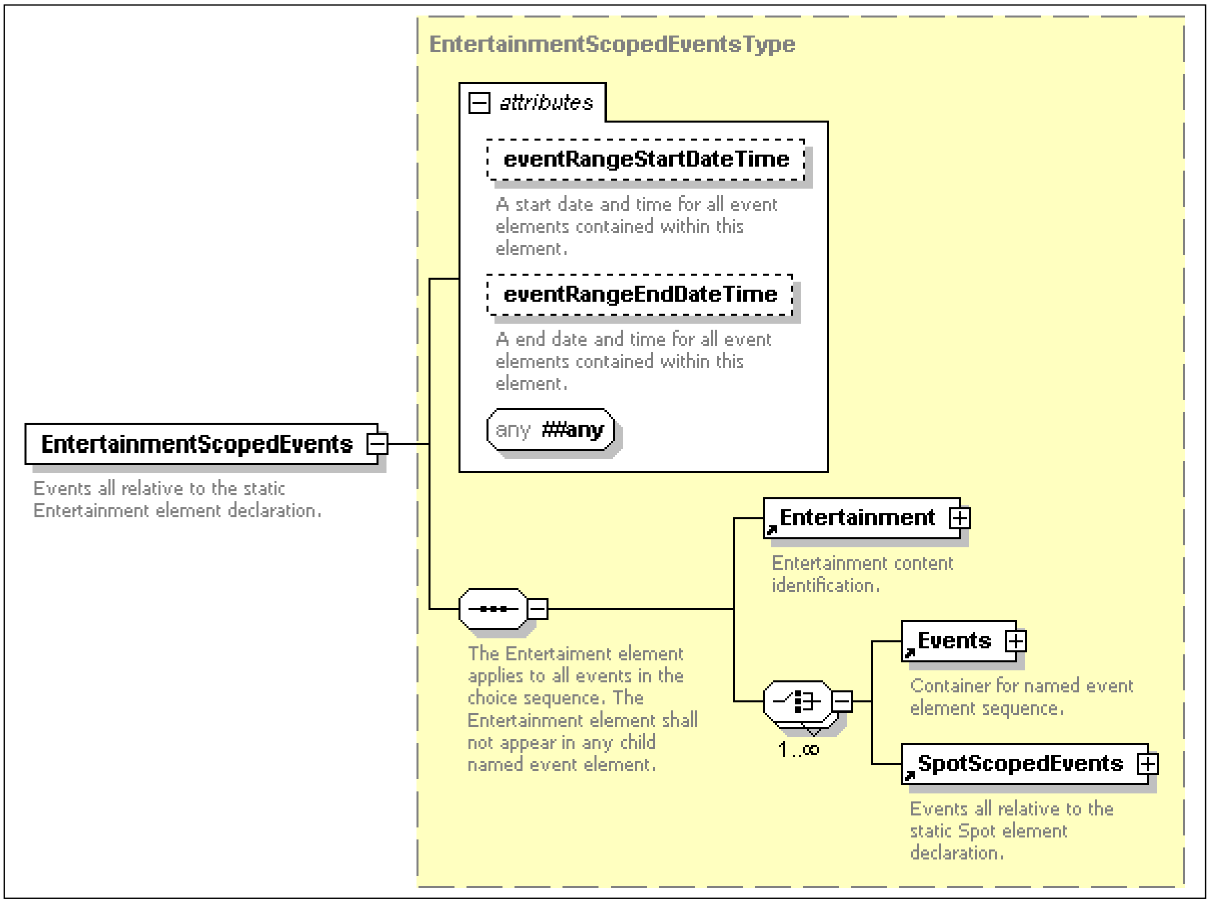

EntertainmentScopedEventsType
attributes
eventRangeStartDateTime
A start date and time for all event elements contained within this element.
eventRangeEndDateTime
A end date and time for all event elements contained within this element.
any ##any
EntertainmentScopedEvents
Events all relative to the static Entertainment element declaration.
Entertainment
Entertainment content identification.
The Entertaiment element applies to all events in the choice sequence. The Entertainment element shall not appear in any child named event element.
Events
Container for named event element sequence.
1..∞
SpotScopedEvents
Events all relative to the static Spot element declaration.

Figure 55: EntertainmentScopedEvents Element Schema

14.13.1.1 Semantic Definitions for the EntertainmentScopedEvents Element

@eventRangeStartDateTime [Optional, eventRangeStartDateTimeAttrType]—An initial date and time relative to the named event elements contained within the Events and the SpotScopedEvents elements. See Section 13.1.2 for additional details.

@eventRangeEndDateTime [Optional, eventRangeEndDateTimeAttrType]—An ending date and time relative to the named event elements contained within the Events and the SpotScopedEvents elements. See Section 13.1.1 for additional details.

@##any [Optional]—Additional attributes from any namespace.

Entertainment [Required]—Entertainment content identification. See Section 14.11 for additional details. The Entertainment element *may* contain a PlayPositions element.

The choice sequence allows for one or more Events or SpotScopedEvents elements to be present and combined in any quantity and any order (i.e., the element count and mix is unspecified but at least one element ***shall*** be present). Any named event element contained within the Events or the SpotScopedEvents elements ***shall not*** contain an Entertainment element as the Entertainment element specified herein takes precedence.

Each named event element contained within the Events element *may* contain an NPT or UTC element.

Each named event element contained within the SpotScopedEvents element *may* contain an NPT or UTC element. See EventBaseType in Section 13.2.1for additional information.

This usage facilitates describing a range of the entertainment or advertising content, for example a chapter within entertainment, while each named event references a specific timepoint within the range.

Events [Choice]—A container for named event elements. See Section 0 for additional information.

SpotScopedEvents [Choice]—A container for named event elements scoped by a Spot element. See Section 14.51 for additional information.

14.13.2 EntertainmentScopedEvents Element Examples

In this example, the entertainment content is static and two Events elements and one SpotScopedEvents element provide grouped sequences of named event elements which reference and are relative to the declared static Entertainment element.

```
<EntertainmentScopedEvents>
 <Entertainment>
  <core:Content>
   <core:AssetRef providerID="videobus.com"
          assetID="BAAD9876543210987654"/>
  </core:Content>
 </Entertainment>
 <Events>
  <SessionEvent time="2007-01-31T01:59:59Z" type="startSession">
   <core:StatusCode class="3">
    <core:Note>Session start.</core:Note>
   </core:StatusCode>
  </SessionEvent>
 </Events>
 <SpotScopedEvents>
  <Spot>
   <core:Tracking>FFFFFDC7-6C60-0030-B2B7-B68DAF952C95</core:Tracking>
   <core:Content>
    <core:AssetRef providerID="advertiser1.com"
           assetID="ADVR1000123456789001"/>
    <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C92</core:Tracking>
   </core:Content>
  </Spot>
  <Events>
   <PlacementStatusEvent time="2007-01-31T02:00:00Z"
              type="startPlacement">
    <Spot>
     <core:Content/>
     <PlayPositions scale="1">
      <PlayPositionStart>
       <NPTOffset>0</NPTOffset>
      </PlayPositionStart>
     </PlayPositions>
    </Spot>
   </PlacementStatusEvent>
   <PlacementStatusEvent time="2007-01-31T02:00:20Z"
              type="endPlacement">
    <Spot>
     <core:Content/>
     <PlayPositions scale="1">
      <PlayPositionEnd>
       <NPTOffset>15</NPTOffset>
      </PlayPositionEnd>
     </PlayPositions>
    </Spot>
   </PlacementStatusEvent>
```

```
    </Events>
  </SpotScopedEvents>
  <Events>
    <PlacementStatusEvent time="2007-01-31T02:00:20Z" type="status">
      <core:StatusCode class="1" detail="11">
        <core:Note>Resource not found.</core:Note>
      </core:StatusCode>
      <Spot>
        <core:Content>
          <core:AssetRef providerID="advertiser2.com"
                  assetID="ADVR2000123456789001"/>
          <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C93</core:Tracking>
        </core:Content>
      </Spot>
    </PlacementStatusEvent>
  </Events>
</EntertainmentScopedEvents>
```

14.14 Events Element

The Events element is a sequence container for one or more different, individually defined named event elements with any element appearing as often as desired. There are no limitations to the number of events or the order of the individual events. Events typically should be listed in the order completed (i.e., when the event’s @time attribute becomes known.)

14.14.1 Events Element Schema

Figure 56 illustrates the Events element’s schema.

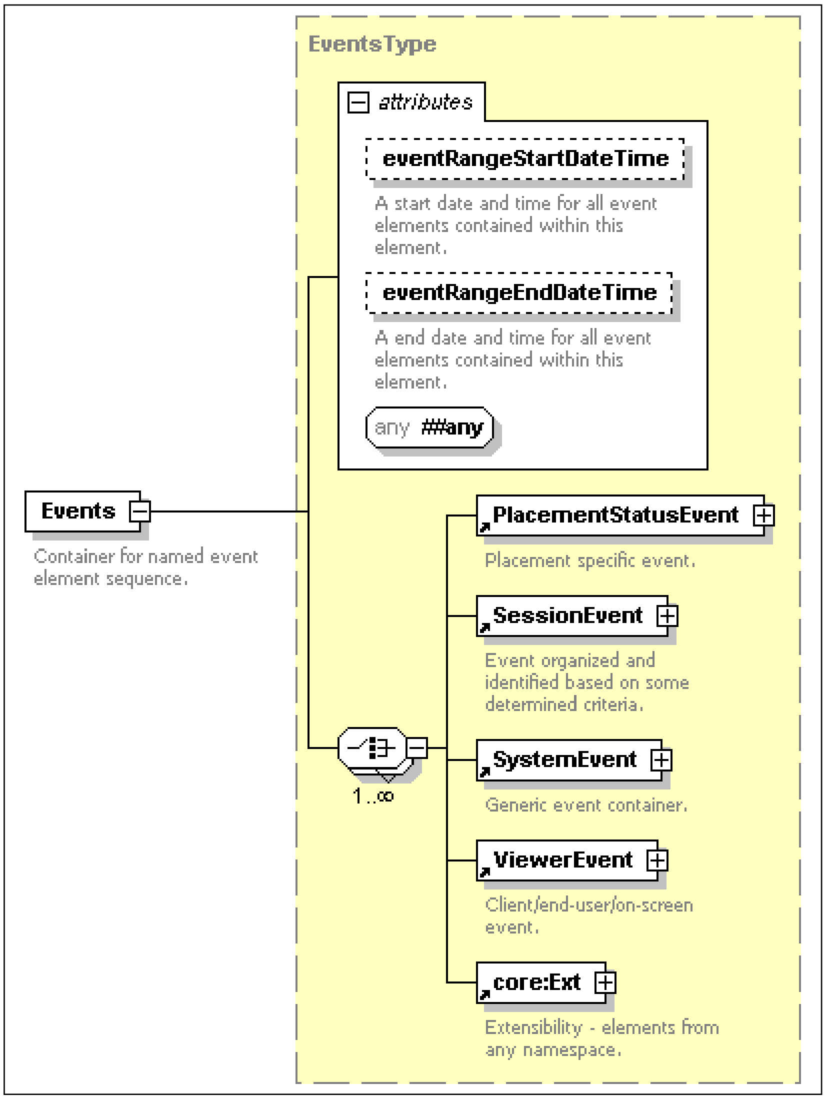

EventsType
attributes
eventRangeStartDateTime
A start date and time for all event elements contained within this element.
eventRangeEndDateTime
A end date and time for all event elements contained within this element.
any ##any
Events
Container for named event element sequence.
1..∞
PlacementStatusEvent
Placement specific event.
SessionEvent
Event organized and identified based on some determined criteria.
SystemEvent
Generic event container.
ViewerEvent
Client/end-user/on-screen event.
core:Ext
Extensibility - elements from any namespace.

Figure 56: Events Element Schema

14.14.1.1 Semantic Definitions for the Events Element

@eventRangeStartDateTime [Optional, eventRangeStartDateTimeAttrType]—An initial date and time for the named event elements contained within the choice sequence. See Section 13.1.2 for additional details.

@eventRangeEndDateTime [Optional, eventRangeEndDateTimeAttrType]—An ending date and time for the named event elements contained within the choice sequence. See Section 13.1.1 for additional details.

@##any [Optional]—Additional attributes from any namespace.

PlacementStatusEvent [Optional]—A placement status specific report. See Section 14.34 for additional information.

SessionEvent [Optional]—A session specific report. See Section 14.48 for additional information.

SystemEvent [Optional]—A system specific report. See Section 14.53 for additional information.

ViewerEvent [Optional]—A viewer action report. See Section 14.57 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.14.2 Events Element Examples

```
 <Events>
  <!--Individual events are defined hereafter. There may be any number of events within.-
->
  <SessionEvent type="endSession" time="2008-02-22T09:30:47Z"/>
 </Events>
 <Events>
  <SessionEvent time="2007-01-31T01:59:59Z" type="startSession">
   <core:StatusCode class="3">
    <core:Note>Session start.</core:Note>
   </core:StatusCode>
  </SessionEvent>
  <PlacementStatusEvent time="2007-01-31T02:00:00Z" type="startPlacement">
   <Spot>
    <core:Content>
     <core:AssetRef providerID="advertiser1.com"
            assetID="ADVR1000123456789001"/>
     <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C92</core:Tracking>
    </core:Content>
    <PlayPositions scale="1">
     <PlayPositionStart>
      <NPTOffset>0</NPTOffset>
     </PlayPositionStart>
    </PlayPositions>
   </Spot>
  </PlacementStatusEvent>
  <ViewerEvent time="2007-01-31T02:00:05Z" type="pause"
        eventSource="remote">
   <Spot>
    <core:Content>
     <core:AssetRef providerID="advertiser1.com"
            assetID="ADVR1000123456789001"/>
     <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C92</core:Tracking>
    </core:Content>
    <PlayPositions scale="0">
     <PlayPosition>
      <NPTOffset>5</NPTOffset>
     </PlayPosition>
    </PlayPositions>
   </Spot>
  </ViewerEvent>
 </Events>
```

14.15 LinearAvailBinding Element

The LinearAvailBinding element functions as a bridge for linear placement opportunity information (i.e., the traditionally termed avail or break). The element contains information specific to defining, refining, or labeling the placement opportunity. The PlacementRequest message operates in the define mode while the other placement management messages operate in the refinement manner. The element's attributes and sub-elements are designed to directly map to an SCTE 35 splice_insert() command and its related descriptors. For example, the SCTE 35 avail_num bitfield maps to @opportunityNumber, the avails_expected bitfield maps to @opportunitiesExpected, and so on. See [SCTE35] for additional information.

14.15.1 LinearAvailBinding Element Schema

Figure 57 illustrates the LinearAvailBinding element's schema.

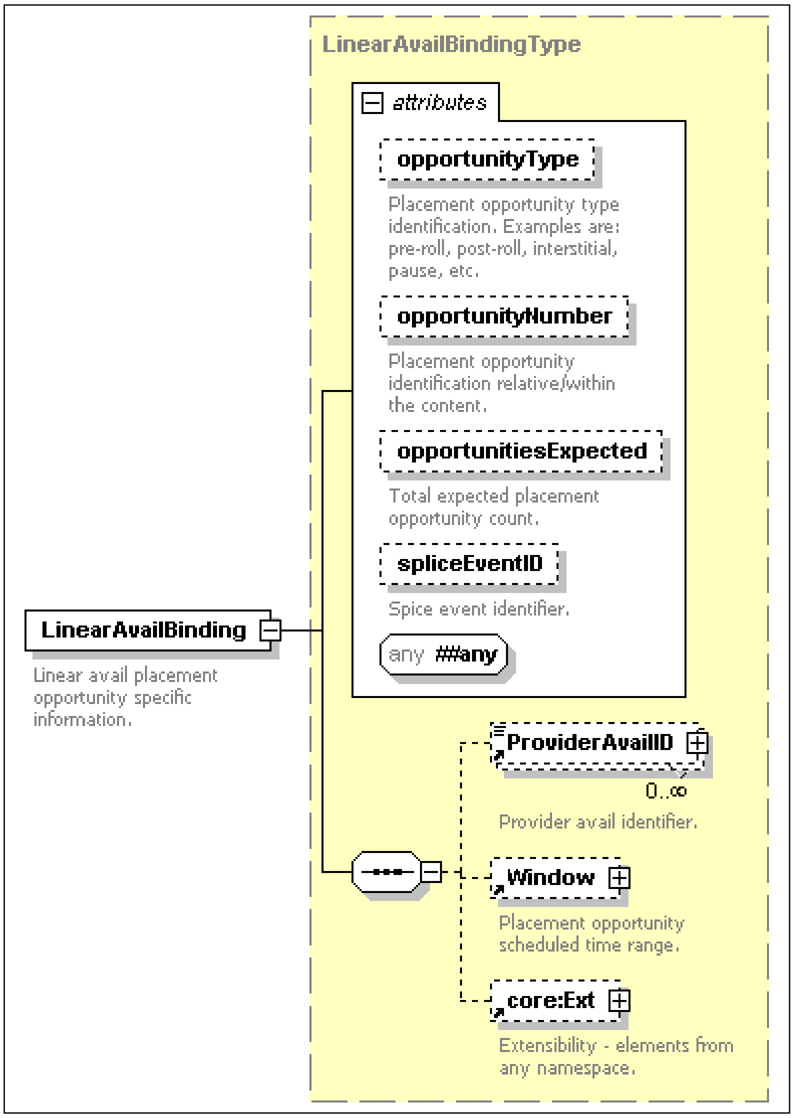

LinearAvailBindingType
attributes
opportunityType
Placement opportunity type identification. Examples are: pre-roll, post-roll, interstitial, pause, etc.
opportunityNumber
Placement opportunity identification relative/within the content.
opportunitiesExpected
Total expected placement opportunity count.
spliceEventID
Spice event identifier.
any ##any
LinearAvailBinding
Linear avail placement opportunity specific information.
ProviderAvailID
0..∞
Provider avail identifier.
Window
Placement opportunity scheduled time range.
core:Ext
Extensibility - elements from any namespace.

Figure 57: LinearAvailBinding Element Schema

14.15.1.1 Semantic Definitions for the LinearAvailBinding Element

@opportunityType [Optional, placementOpportunityTypeAttrType]—The placement opportunity type. See Section 13.1.8 for additional information. For example, in classic linear the typical value for this attribute is ‘interstitial.’

@opportunityNumber [Optional, placementOpportunityNumberAttrType]—The placement opportunity identification value relative to the entertainment content. See Section 13.1.7 for additional information.

@opportunitiesExpected [Optional, placementOpportunitiesExpectedAttrType]—The total number of planned placement opportunities relative to the entertainment content. See Section 13.1.6 for additional information.

@spliceEventID [Optional, xsd:nonNegativeInteger]—The splice event identifier. Typically, this value is obtained from the SCTE 35 splice_insert() command's field of the same approximate name. See [SCTE35] for additional information.

@##any [Optional]—Additional attributes from any namespace.

ProviderAvailID [Optional]—Zero or more elements that a receiving device *may* utilize to alter its behavior. See Section 14.42 for additional information.

Window [Optional]—The placement opportunity's scheduled time range. See Section 14.58 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

The LinearAvailBinding element's value *may* be empty.

14.15.2 LinearAvailBinding Element Examples

```
  <LinearAvailBinding opportunityType="interstitial" opportunityNumber="1"
opportunitiesExpected="4"/>

  <LinearAvailBinding opportunityType="interstitial" opportunityNumber="2"
spliceEventID="83983">
     <ProviderAvailID>938</ProviderAvailID>
     <ProviderAvailID>2373</ProviderAvailID>
  </LinearAvailBinding>

  <LinearAvailBinding>
   <Window start="2007-06-03T09:30:47.0Z" end="2007-06-03T09:40:47.0Z"/>
  </LinearAvailBinding>
```

The initial example LinearAvailBinding element is the first of four expected placement opportunities. The next example is placement opportunity number two in the series and it has an SCTE 35 spice event ID of 83938. The binding also has two associated provider identifiers. The third example binds the placement window to the placement opportunity.

14.16 MaxDuration Element

The MaxDuration element describes a maximum run-time in years, months, days, hours, minutes, seconds, and milliseconds.

14.16.1 MaxDuration Element Schema

Figure 58 illustrates the MaxDuration element's schema.

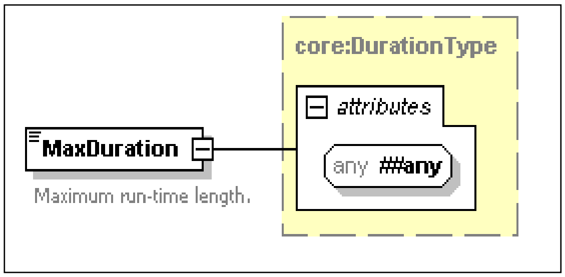


Figure 58: MaxDuration Element Schema

14.16.1.1 Semantic Definitions for the MaxDuration Element

@##any [Optional]—Additional attributes from any namespace.

The MaxDuration element's value is of type xsd:duration where the format is PnYnMnDTnHnMn.nnnS.

14.16.2 MaxDuration Element Examples

```
<!--One minute-->
<MaxDuration>PT1M</MaxDuration>
<MaxDuration>PT60S</MaxDuration>

<!--Exactly 30 seconds-->
<MaxDuration>PT30.000S</MaxDuration>

<!--One hour, 5 minutes, 23 seconds and 666 milliseconds-->
<MaxDuration>PT1H5M23.666S</MaxDuration>

<!--One year, 2 months, 3 days, 10 hours, 30 minutes and zero seconds.-->
<MaxDuration>P1Y2M3DT10H30M</MaxDuration>
```

14.17 MaxPlacementCount Element

The MaxPlacementCount element specifies a maximum placement count.

14.17.1 MaxPlacementCount Element Schema

Figure 59 illustrates the MaxPlacementCount element's schema.

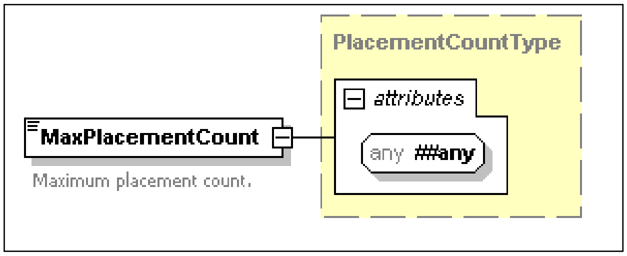


Figure 59: MaxPlacementCount Element Schema

14.17.1.1 Semantic Definitions for the MaxPlacementCount Element

The MaxPlacementCount element inherits from the PlacmentCountType complex type. See Section 13.2.3 for semantic definitions applicable to this element.

14.17.2 MaxPlacementCount Element Examples

```
<MaxPlacementCount>5</MaxPlacementCount>
```

14.18 MinDuration Element

The MinDuration element describes a minimum run-time in years, months, days, hours, minutes, seconds, and milliseconds

14.18.1 MinDuration Element Schema

Figure 60 illustrates the MinDuration element's schema.

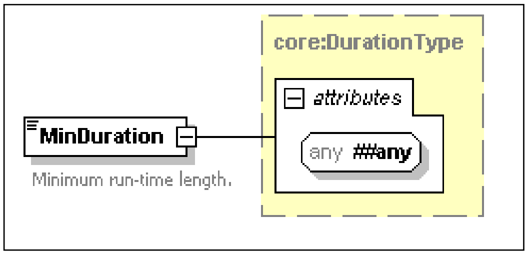


Figure 60: MinDuration Element Schema

14.18.1.1 Semantic Definitions for the MinDuration Element

@##any [Optional]—Additional attributes from any namespace.

The MinDuration element's value is of type xsd:duration where the format is PnYnMnDTnHnMn.nnnS.

14.18.2 MinDuration Element Examples

```
<!--15 seconds-->
<MinDuration>PT15S</MinDuration>

<!--Exactly 30 seconds-->
<MinDuration>PT30.000S</MinDuration>

<!--One year, 2 months, 3 days, 10 hours, 30 minutes and zero seconds.-->
<MinDuration>P1Y2M3DT10H30M</MinDuration>
```

14.19 MinPlacementCount Element

The MinPlacementCount element specifies a minimum placement count.

14.19.1 MinPlacementCount Element Schema

Figure 61 illustrates the MinPlacementCount element's schema.

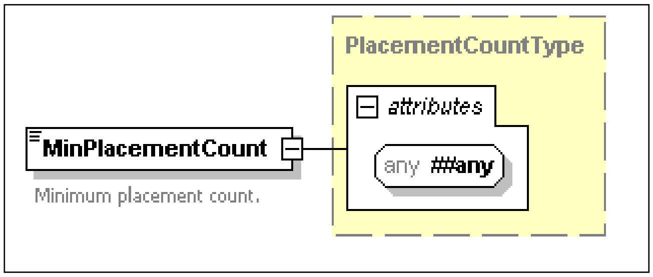


Figure 61: MinPlacementCount Element Schema

14.19.1.1 Semantic Definitions for the MinPlacementCount Element

The MinPlacementCount element inherits from the PlacmentCountType complex type. See Section 13.2.3 for semantic definitions applicable to this element.

14.19.2 MinPlacementCount Element Examples

```
<MinPlacementCount>0</MinPlacementCount>
<MinPlacementCount>2</MinPlacementCount>
```

14.20 NameSpaceVersion Element

The NameSpaceVersion element identifies namespaces and their versions. For example, in a ListADMServicesResponse one or more NameSpaceVersion elements *may* be supplied to enumerate namespaces and versions supported by the ADM. (e.g., SCTE 130 PODM).

14.20.1 NameSpaceVersion Element Schema

Figure 62 illustrates the NameSpaceVersion element's schema.

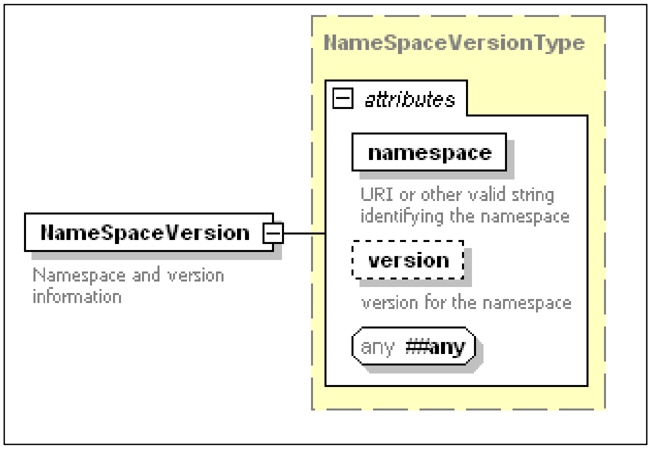

NameSpaceVersionType
attributes
namespace
URI or other valid string identifying the namespace
NameSpaceVersion
Namespace and version information
version
version for the namespace
any ##any

Figure 62: NameSpaceVersion Element Schema

14.20.1.1 Semantic Definitions for the NameSpaceVersion Element

@namespace [Required core:NonEmptyStringType] – Namespace identification. This can be a URI or other valid string.

@version [Optional core:NonEmptyStringType] – Specify a version for the namespace.

@##any [Optional]—Additional attributes from any namespace.

14.20.2 NameSpaceVersion Element Examples

```
<NameSpaceVersion namespace="http://www.scte.org/schemas/130-3/2013/adm/podm"
version="1.1"/>
<NameSpaceVersion namespace="http://www.scte.org/schemas/130-3/2013/adm/podm"
version="1.2"/>
<NameSpaceVersion namespace="urn:cablelabs:md:xsd:placementopportunity:3.0"/>
```

In above NameSpaceVersion Element examples, two versions of the *podm* data model are illustrated along with the CableLabs metadata Placement Opportunity schema.

14.21 Network Element

The Network element facilitates network identification. For example, the Network element *may* identify the primary channel programming network in a traditional linear advertising context, or it *may* be the name of the original content programming source in the context of a delayed playback of a primary network feed.

14.21.1 Network Element Schema

Figure 63 illustrates the Network element’s schema.

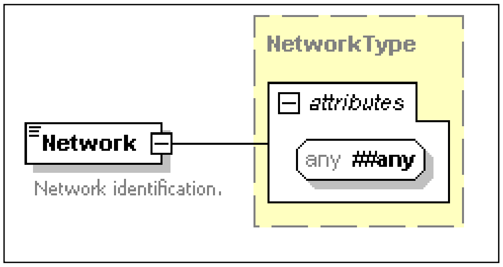


Figure 63: Network Element Schema

14.21.1.1 Semantic Definitions for the Network Element

@##any [Optional]—Additional attributes from any namespace.

The Network element’s value is of type core:nonEmptyStringType and ***shall not*** be empty.

14.21.2 Network Element Examples

```
<Network>ABC</Network>
<Network>154</Network>
```

14.22 NPT Element

The normal play time (NPT) element indicates the stream absolute position relative to the beginning of the referenced content with the basis being adopted from RFC2326. See [RFC2326] for additional information.

14.22.1 NPT Element Schema

Figure 64 illustrates the NPT element’s schema.

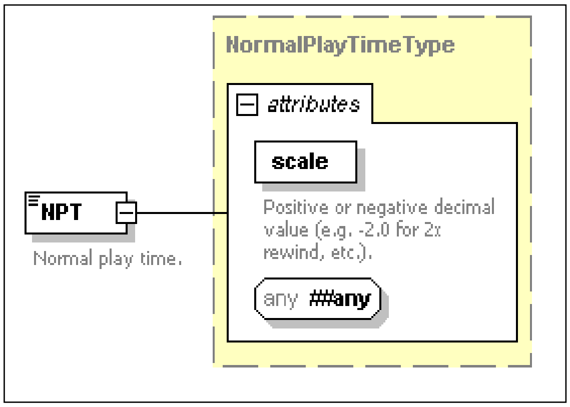

NormalPlayTimeType
attributes
scale
Positive or negative decimal value (e.g. -2.0 for 2x rewind, etc.).
any ##any
NPT
Normal play time.

Figure 64: NPT Element Schema

14.22.1.1 Semantic Definitions for the NPT Element

The NPT element inherits from the NormalPlayTimeType complex type. See Section 13.2.2 for semantic definitions applicable to this element.

14.22.2 NPT Element Examples

```
<!-- "Normal" play from content start.-->
<NPT scale="1.0">0.0</NPT>
<NPT scale="1">0</NPT>

<!-- Pause/stop at content start. -->
<NPT scale="0">0</NPT>

<!--Pause at 5 seconds-->
<NPT scale="0.0">5</NPT>

<!-- Pause at point in content.-->
<NPT scale="0">333.3</NPT>

<!--"Normal" play speed at 123.45 seconds into content.-->
<NPT scale="1.0">123.45</NPT>

<!--Fast-forward at 2x starting 10.333 seconds into content.-->
<NPT scale="2.0">10.333</NPT>
<NPT scale="2">10.333</NPT>

<!--Fast-rewind at 6x starting 30 seconds into content.-->
<NPT scale="-6.0">30.0</NPT>

<!--Range-->
<NPT scale="1.0">123.45-125</NPT>

<!--Rewind at 2x for the range.-->
<NPT scale="-2">130.0-87.5</NPT>

<!--Fast forward at 1/2 speed starting at one second-->
<NPT scale="0.5">1</NPT>
```

14.23 NPTOffset Element

The normal play time (NPT) offset element indicates the stream absolute position relative to the beginning of the referenced (entertainement/spot) content with the basis being adopted from RFC2326. See [RFC2326] for additional information.

14.23.1 NPTOffset Element Schema

Figure 64 illustrates the NPTOffset element’s schema.

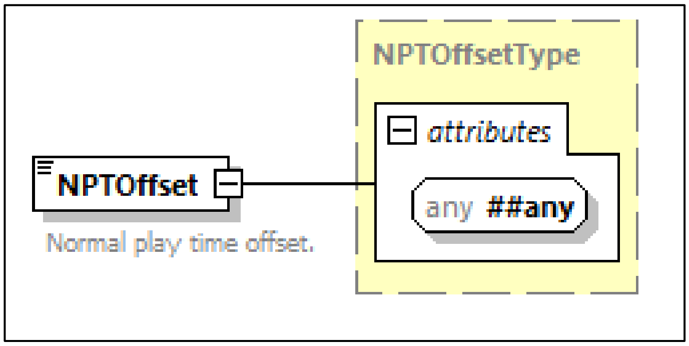


Figure 65: NPTOffset Element Schema

14.23.1.1 Semantic Definitions for the NPTOffset Element

The NPTOffset element inherits from the NormalPlayTimeType complex type. See Section 13.2.2 for semantic definitions applicable to this element.

14.23.2 NPTOffset Element Examples

```
<NPTOffset>5</NPTOffset>
<NPTOffset>55.9</NPTOffset>
```

14.24 OpportunityBinding Element

The OpportunityBinding element contains information specific to defining, refining, or labeling the placement opportunity. The PlacementRequest message operates in the define mode while the other placement management messages operate in the refinement manner.

14.24.1 OpportunityBinding Element Schema

Figure 66 illustrates the OpportunityBinding element's schema.

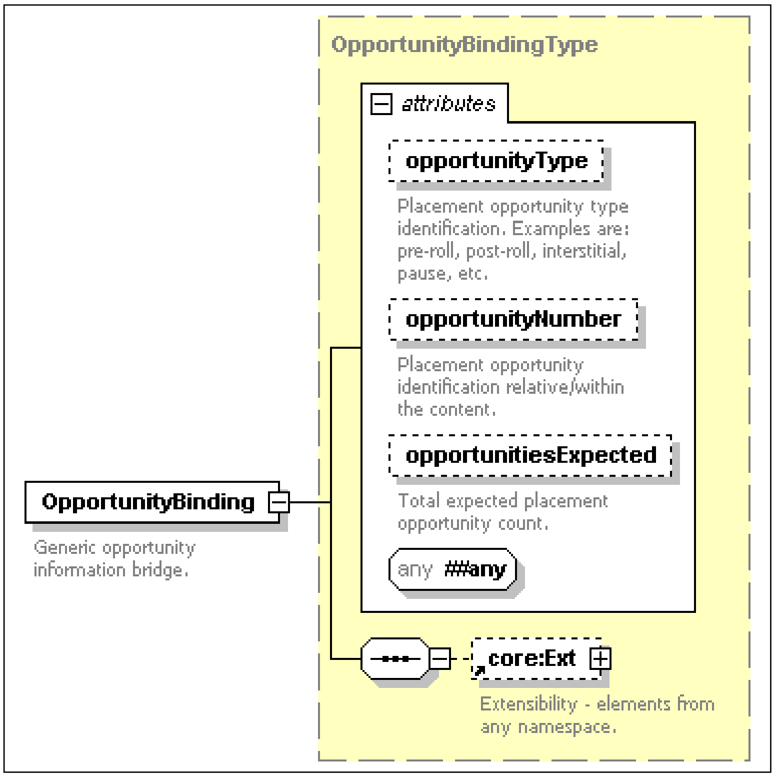

OpportunityBindingType
attributes
opportunityType
Placement opportunity type identification. Examples are: pre-roll, post-roll, interstitial, pause, etc.
opportunityNumber
Placement opportunity identification relative/within the content.
opportunitiesExpected
Total expected placement opportunity count.
any ##any
OpportunityBinding
Generic opportunity information bridge.
core:Ext
Extensibility - elements from any namespace.

Figure 66: OpportunityBinding Element Schema

14.24.1.1 Semantic Definitions for the OpportunityBinding Element

@opportunityType [Optional, placementOpportunityTypeAttrType]—The placement opportunity type. See Section 13.1.8 for additional information.

@opportunityNumber [Optional, placementOpportunityNumberAttrType]—The placement opportunity identification value relative to the entertainment content. See Section 13.1.7 for additional information.

@opportunitiesExpected [Optional, placementOpportunitiesExpectedAttrType]—The total number of planned placement opportunities relative to the entertainment content. See Section 13.1.6 for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

The OpportunityBinding element's value *may* be empty.

14.24.2 OpportunityBinding Element Examples

```
  <OpportunityBinding opportunityType="preRoll"/>

  <OpportunityBinding opportunityNumber="2" opportunitiesExpected="5"/>

  <OpportunityBinding opportunityType="postRoll" opportunityNumber="3"
opportunitiesExpected="3"/>
```

The first example OpportunityBinding element indicates the placement opportunity type as a pre-roll meaning the ad content comes before the entertainment content.

The second example does not describe the opportunity type only that it is opportunity number two of five expected. It is likely that the registration-established service details implied the placement opportunity type.

The final example signals a post-roll opportunity identified as opportunity number three relative to the entertainment content and this opportunity is the last of the expected opportunities.

14.25 OpportunityConstraints Element

The OpportunityConstraints element defines placement opportunity restrictions and limitations applicable to the container element’s scope (i.e., the higher level scope element that “wraps” this element). The OpportunityConstraints element *may* be included in a PlacementRequest message as part of the ADM communicating details regarding the placement opportunities. If the OpportunityConstraints element is located within the PlacementOpportunity element, the constraints apply to the entire opportunity. If the OpportunityConstraints element is contained within the PlacementControl element, the limits apply to the PlacementControl element’s positional scope. The OpportunityConstraints element *may* be included in both scopes simultaneously and in the event of a conflict, the PlacementOpportunity element scoped values take precedence.

The ADS placement decisions should meet the supplied constraints and the ADM should verify response conformance. Verification and error handling are outside the scope of this specification.

An ADS is allowed to alter the opportunity structures within the ADM presented opportunity constraints. The ADM is ultimately responsible for placement decision fulfillment and therefore, validating the placement decision set correctly adheres to the constraints. How the ADM accomplishes this validation is outside the scope of this specification.

14.25.1 OpportunityConstraints Element Schema

Figure 67 illustrates the OpportunityConstraints element’s schema.

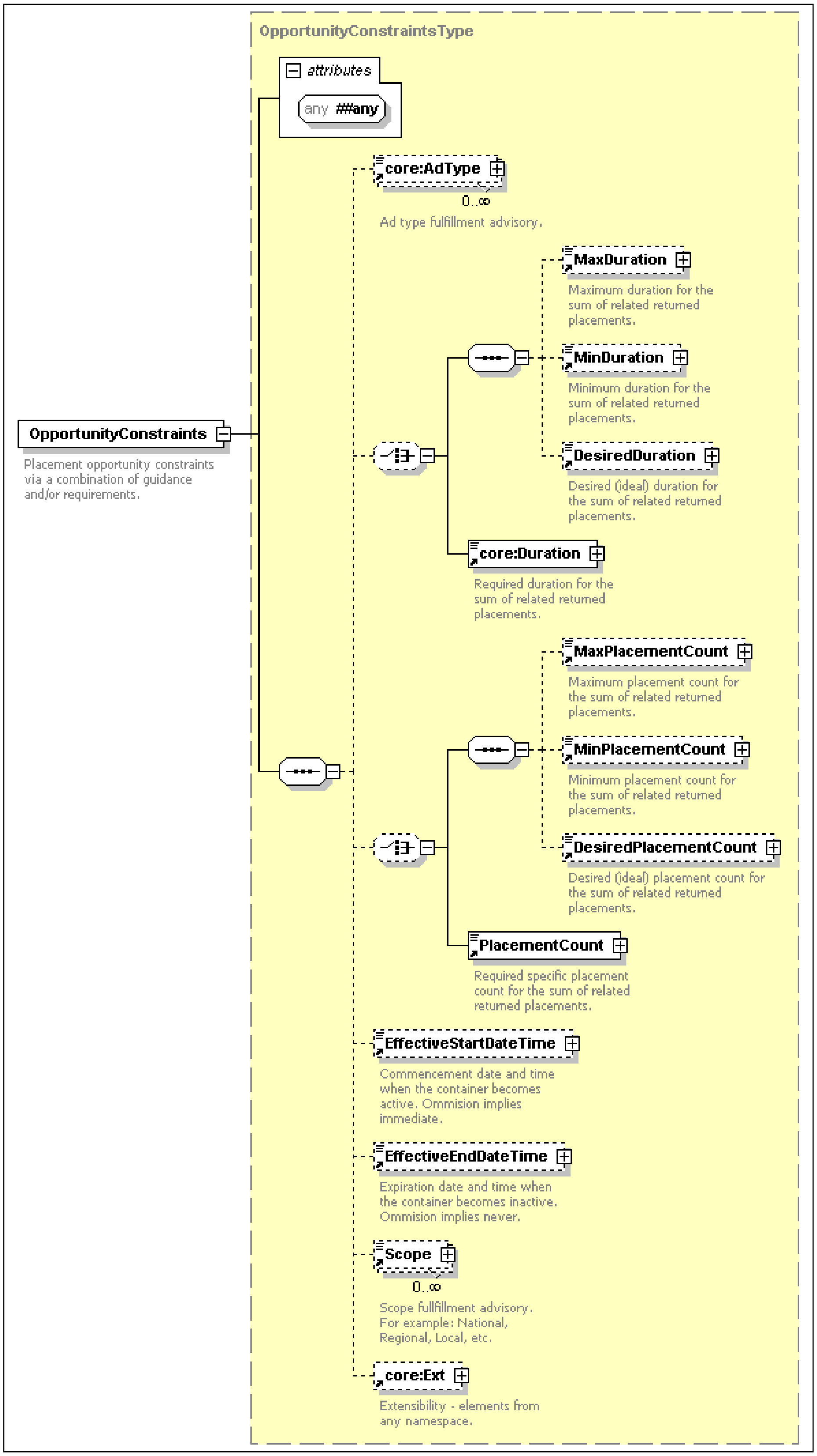

OpportunityConstraintsType
attributes
any ##any
core:AdType
0..∞
Ad type fulfillment advisory.
MaxDuration
Maximum duration for the sum of related returned placements.
MinDuration
Minimum duration for the sum of related returned placements.
DesiredDuration
Desired (ideal) duration for the sum of related returned placements.
core:Duration
Required duration for the sum of related returned placements.
OpportunityConstraints
Placement opportunity constraints via a combination of guidance and/or requirements.
MaxPlacementCount
Maximum placement count for the sum of related returned placements.
MinPlacementCount
Minimum placement count for the sum of related returned placements.
DesiredPlacementCount
Desired (ideal) placement count for the sum of related returned placements.
PlacementCount
Required specific placement count for the sum of related returned placements.
EffectiveStartDateTime
Commencement date and time when the container becomes active. Ommision implies immediate.
EffectiveEndDateTime
Expiration date and time when the container becomes inactive. Ommision implies never.
Scope
0..∞
Scope fullfillment advisory. For example: National, Regional, Local, etc.
core:Ext
Extensibility - elements from any namespace.

Figure 67: OpportunityConstraints Element Schema

14.25.1.1 Semantic Definitions for the OpportunityConstraints Element

In this semantic definition section, a reference to a placement opportunity *may* also apply to a positional placement opportunity dependent on the OpportunityConstraints element's location in the XML document. The text uses placement opportunity to cover both the placement opportunity and the positional placement opportunity cases.

@##any [Optional]—Additional attributes from any namespace.

core:AdType [Optional]—Zero or more elements specifying specific ad types that ***shall*** satisfy the placement opportunity. See [SCTE130-2] for additional information.

MaxDuration [Optional Choice]—An element specifying the maximum total placement opportunity duration. The total placement set duration ***shall not*** exceed this value but *may* be less than this value. If an exact duration is required, the Duration element ***shall*** be used. See Section 14.16 for additional information.

MinDuration [Optional Choice]—An element providing the minimum total placement opportunity duration. The total placement set duration ***shall not*** be less than this value. If an exact duration requirement is desired, the Duration element ***shall*** be used. See Section 14.18 for additional information.

DesiredDuration [Optional Choice]—An element specifying the ideal (or desired) total placement opportunity duration. The total placement set duration should match this value but *may* range within the tolerance specified by the MaxDuration and the MinDuration elements. If an exact duration is required, the Duration element ***shall*** be used. See Section 14.7 for additional information.

core:Duration [Optional Choice]—An element indicating a required total placement opportunity duration for which the placement decision set duration ***shall*** match. See [SCTE130-2] for additional information.

MaxPlacementCount [Optional Choice]—An element specifying the maximum Placement element count that *may* be defined to fulfill the placement opportunity. If an exact requirement is necessary, the PlacementCount element ***shall*** be used. See Section 14.17 for additional information.

MinPlacementCount [Optional Choice]—An element specifying the minimum Placement element count that *may* be defined to fulfill the placement opportunity. If an exact requirement is necessary, the PlacementCount element ***shall*** be used. See Section 14.19 for additional information.

DesiredPlacementCount [Optional Choice]—An element specifying the ideal (or desired) Placement element count that *may* be defined to fulfill the placement opportunity. The total placement set duration should match this value but *may* range within the tolerance specified by the MaxPlacementCount and the MinPlacementCount elements. If an exact requirement is necessary, the PlacementCount element ***shall*** be used. See Section 14.8 for additional information.

PlacementCount [Optional Choice]—An element indicating the exact number of Placement elements that ***shall*** be defined in the placement decision set. See Section 14.29 for additional information.

EffectiveStartDateTime [Optional]—An element identifying when the placement opportunity information becomes active. If the element is omitted, active immediately is implied. See Section 14.10 for additional information.

EffectiveEndDateTime [Optional]—An element identifying when the placement opportunity information becomes inactive or cancelled. If the element is omitted, active infinitely is implied. See Section 14.9 for additional information.

Scope [Optional]—Zero or more elements specifying the fulfillment scope. See Section 14.43 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

The OpportunityConstraints element's value *may* be empty.

14.25.2 OpportunityConstraints Element Examples

```
<OpportunityConstraints>
  <MaxDuration>PT30S</MaxDuration>
</OpportunityConstraints>

<OpportunityConstraints>
  <core:AdType>Graphic Overlay</core:AdType>
  <MaxPlacementCount>3</MaxPlacementCount>
  <EffectiveStartDateTime>2007-12-10T12:00:00Z</EffectiveStartDateTime>
  <EffectiveEndDateTime>2008-01-10T12:00:00Z</EffectiveEndDateTime>
  <Scope>Local</Scope>
  <Scope>Regional</Scope>
</OpportunityConstraints>
```

The first example OpportunityConstraints element specifies the placement opportunity has a maximum duration of thirty seconds. Thus, the ADS *may* provide any combination of ads that have a total play out time not exceeding thirty seconds. The ADS *may* respond with an empty decision (i.e., no ad content).

The second example OpportunityConstraints element specifies that up to three graphic overlay ads *may* be returned. The effective start and end date define a time period of 31 days and the ads *may* be of either local or regional scope.

14.26 Placement Element

The Placement element contains a specific ADS placement decision in response to a placement decision request (i.e., a PlacementRequest message). Within the Placement element, the ADS behavior decision or action is specified using the @action attribute. The Placement element *may* include a content binding which pairs a content asset with the placement, an ADS assigned tracking identifier, and a set of placement constraints.

14.26.1 Placement Element Schema

Figure 68 illustrates the Placement element’s schema.

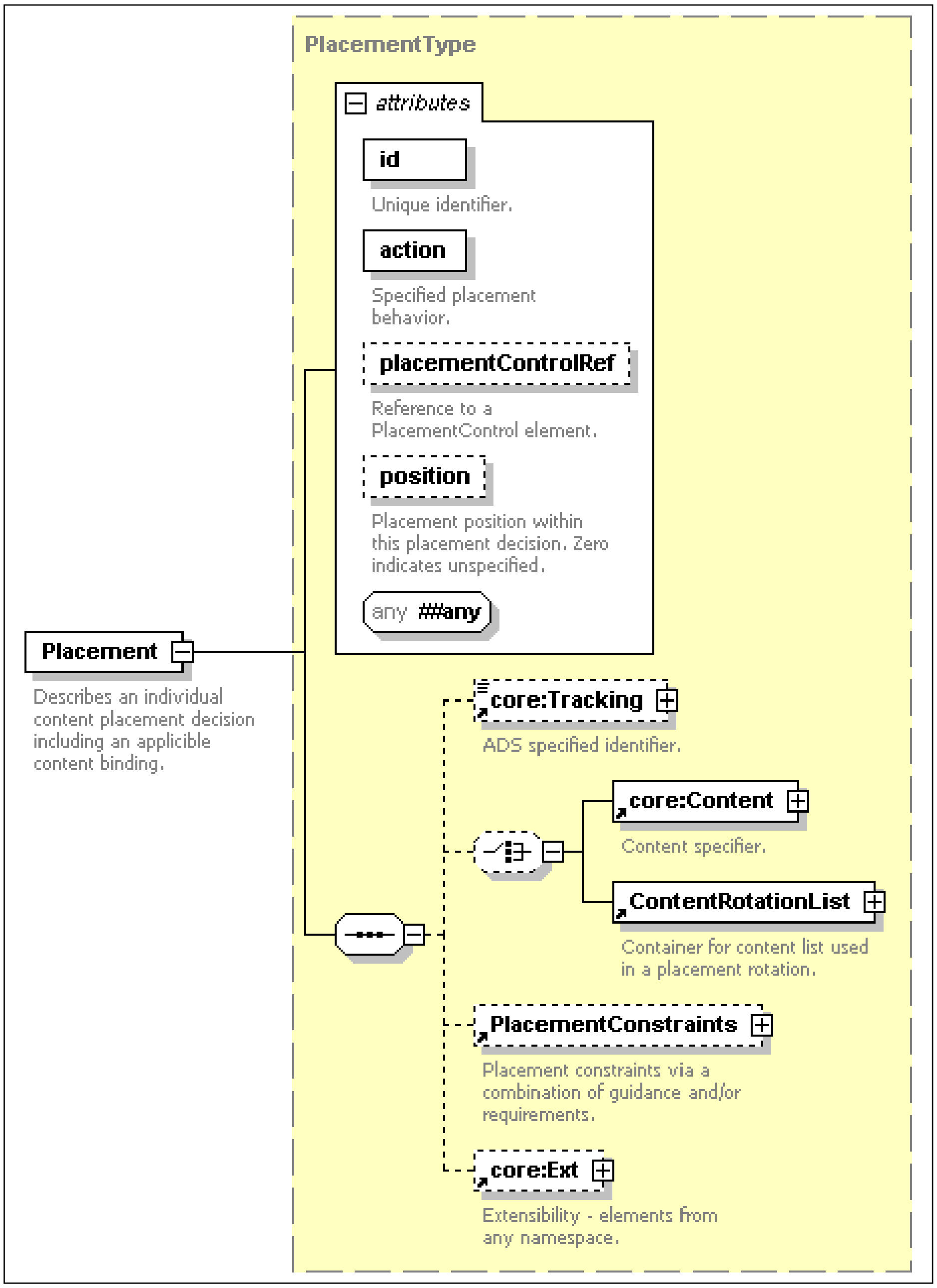

PlacementType
attributes
id
Unique identifier.
action
Specified placement behavior.
placementControlRef
Reference to a PlacementControl element.
position
Placement position within this placement decision. Zero indicates unspecified.
any ##any
Placement
Describes an individual content placement decision including an applicible content binding.
core:Tracking
ADS specified identifier.
core:Content
Content specifier.
ContentRotationList
Container for content list used in a placement rotation.
PlacementConstraints
Placement constraints via a combination of guidance and/or requirements.
core:Ext
Extensibility - elements from any namespace.

Figure 68: Placement Element Schema

14.26.1.1 Semantic Definitions for the Placement Element

@id [Required, core:idAttrType]—An element unique identifier attribute which ***shall*** be exclusive within the scope of the encompassing container element. See [SCTE130-2] for additional information.

@action [Required, placementActionAttrType]—The ADS specified behavior that the ADM ***shall*** carry out for this placement. See Section 13.1.5 for additional information. Table 10 defines attribute values and behaviors which *may* be extended by private agreement outside the scope of this specification. The attribute's value ***shall*** appear exactly as shown in Table 10.

Value	Description
delete	The associated placement opportunity or positional placement opportunity ***shall*** be removed. This value alters the placement structure removing the placement opportunity or positional placement opportunity. No ad content ***shall*** be output and any existing ad content occupying this location ***shall*** be removed. No core:Content or ContentRotationList element ***shall*** appear in the Placement element.
fill	The placement opportunity or positional placement opportunity ***shall*** be added to or inserted at this location. This value *may* alter the placement opportunity structure. Either a core:Content or ContentRotationList element ***shall*** be present in the Placement element.
fixed	The placement opportunity or positional placement opportunity ***shall not*** be changed or altered. No core:Content or ContentRotationList element ***shall*** appear in the Placement element.
replace	The currently occupying placement opportunity or positional placement opportunity content ***shall*** be replaced by the supplied content bindings. Either a core:Content or ContentRotationList element ***shall*** be present in the Placement element.
…	User defined and outside the scope of this specification. The string ***shall*** be prefixed with the text "private:".

Table 10: Placement Element's @action Attribute Values

@placementControlRef [Optional, core:idAttrType]—A reference to a PlacementRequest message's PlacementControl element. This attribute ***shall*** be included if the original PlacementRequest contained a PlacementControl element being fulfilled by this Placement element. This attribute ***shall*** be omitted if there is no PlacementControl element available to reference. The reference facilitates ADM verification of the ADS decision and it enables positional references for non-consecutive placements.

@position [Optional, positionTypeAttr]—Placement location within the placement decision set. See Section 13.1.9 for additional information. Typically, this attribute should be included when the placement is not consecutive.

@##any [Optional]—Additional attributes from any namespace.

core:Tracking [Optional]—A placement identification element typically supplied by an ADS. See [SCTE130-2] for additional information. The Tracking element, if present, ***shall*** be placed into all reporting events associated with this placement. See Section 12.3 for references to the appropriate events where this element ***shall*** be echoed.

core:Content[Optional Choice]—A content binding occupying this placement position. See [SCTE130-2] for additional information.

ContentRotationList [Optional Choice]—A sequence of content bindings used to satisfy the placement and its usage is outside the scope this specification. See Section 14.6 for additional information.

PlacementConstraints [Optional]—The specific limitations that ***shall*** be applied to the placement. See Section 14.27 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.26.2 Placement Element Examples

Example 34 deletes the corresponding placement opportunity.

```
  <Placement id="DeletePosition1Decision" action="delete" position="1"/>
```

Example 34

Example 35 replaces the single positional placement opportunity using the supplied content binding.

```
  <Placement id="ReplacePostion1Decision" action="replace"
placementControlRef="PCtrlSlot1">
    <!--Placement decision tracking.-->
    <core:Tracking>Placement tracking tag1</core:Tracking>
    <!--Content binding.-->
    <core:Content>
      <core:AssetRef providerID="contentco.com" assetID="CNCO8899663683678632"/>
    </core:Content>
  </Placement>
```

Example 35

Example 36 replaces the single positional placement opportunity using the supplied content rotation list. The PlacementConstraints element makes the placement exactly 30 seconds in duration. The placement is assigned a cancellation date and time and is marked as a locally scoped placement.

```
  <Placement id="ReplacePosition1Decision" action="replace"
placementControlRef="PCtrlSlot1">
    <ContentRotationList>
      <core:Content>
        <!--Content tracking and binding.-->
        <core:AssetRef providerID="contentco.com" assetID="CNCO8899663683678632"/>
        <core:Tracking>Placement content tracking tag1</core:Tracking>
      </core:Content>
      <core:Content>
        <!--Content tracking and binding.-->
        <core:AssetRef providerID="contentco.com" assetID="CNCO8899663683678633"/>
        <core:Tracking>Placement content tracking tag2</core:Tracking>
      </core:Content>
    </ContentRotationList>
    <!--Constraints on the placement-->
    <PlacementConstraints>
      <!--Make exactly 30 seconds.-->
      <core:Duration>PT30.000S</core:Duration>
      <!--Cancellation date and time.-->
      <EffectiveEndDateTime>2009-04-03T12:00:00.0Z</EffectiveEndDateTime>
      <Scope>Local</Scope>
    </PlacementConstraints>
  </Placement>
```

Example 36

14.27 PlacementConstraints Element

The PlacementConstraints element contains ADS supplied placement restrictions and limitations applicable to a placement decision set or to an individual placement. The PlacementConstraints element's applicability is relative to the containing element's scope. For example, if the PlacementConstraints element is located within the PlacementDecision element, the constraints apply to the entire placement decision including all its positional placements. If the PlacementConstraints element is contained within the Placement element (i.e., the positional placement decision), the limits apply only to the subunit's positional scope.

If PlacementConstraints are located at both XML element levels and there is a conflict, then the PlacementConstraints scoped to the outer level, (i.e., the PlacementDecision element) ***shall*** take precedence.

14.27.1 PlacementConstraints Element Schema

Figure 69 illustrates the PlacementConstraints element's schema.

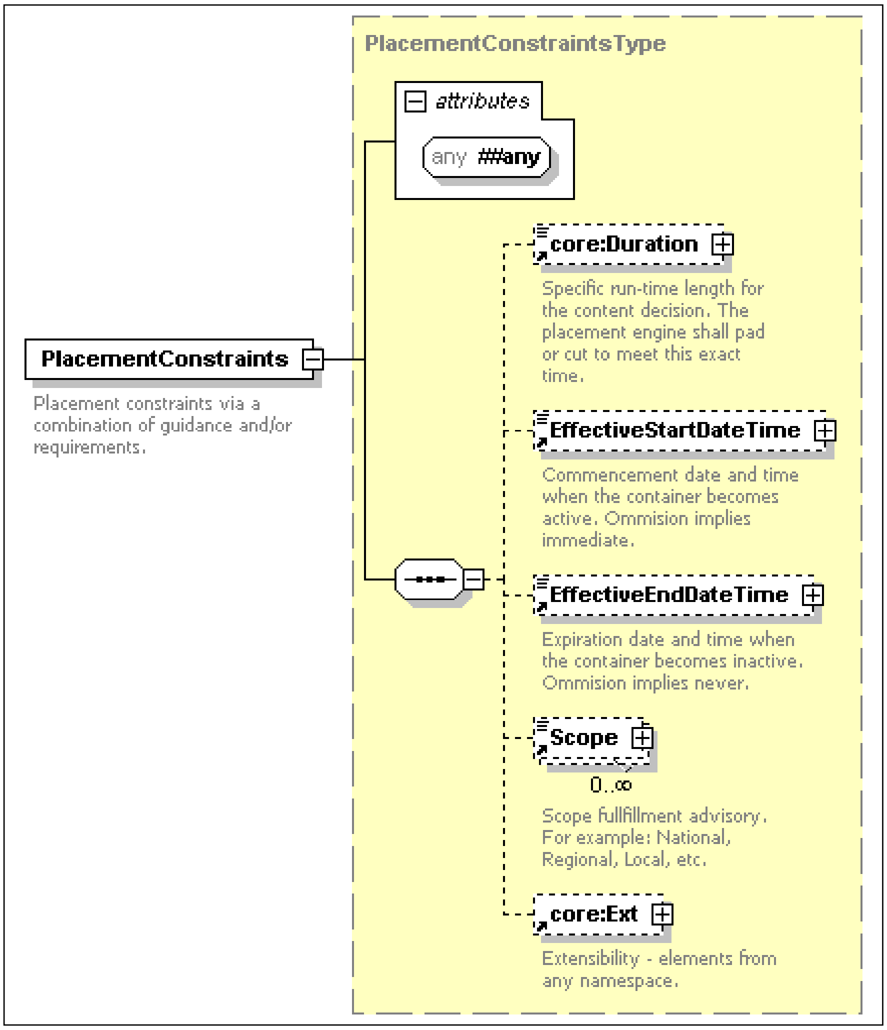

PlacementConstraintsType
attributes
any ##any
core:Duration
Specific run-time length for the content decision. The placement engine shall pad or cut to meet this exact time.
EffectiveStartDateTime
Commencement date and time when the container becomes active. Ommision implies immediate.
EffectiveEndDateTime
Expiration date and time when the container becomes inactive. Ommision implies never.
Scope
0..∞
Scope fullfillment advisory. For example: National, Regional, Local, etc.
core:Ext
Extensibility - elements from any namespace.
PlacementConstraints
Placement constraints via a combination of guidance and/or requirements.

Figure 69: PlacementConstraints Element Schema

14.27.1.1 Semantic Definitions for the PlacementConstraints Element

@##any [Optional]—Additional attributes from any namespace.

core:Duration [Optional]—The total placement duration. The placement implementation ***shall*** pad or cut the content asset as necessary and appropriate to meet this duration. See [SCTE130-2] for additional information.

EffectiveStartDateTime [Optional]—A date and time when the placement information becomes active. If the element is omitted, active immediately is implied. See Section 14.10 for additional information.

EffectiveEndDateTime [Optional]—A date and time when the placement information becomes inactive. If the element is omitted then active infinitely is implied. See Section 14.9 for additional information.

Scope [Optional]—A sequence of fulfillment scopes. See Section 14.43 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

The PlacementConstraints element's value *may* be empty if the constraints are supplied using only attributes from other namespaces.

14.27.2 PlacementConstraints Element Examples

```
<PlacementConstraints>
  <!--Make the placement last exactly 30 seconds.-->
  <core:Duration>PT30.000S</core:Duration>
  <!--Commencement and cancellation date and time.-->
  <EffectiveStartDateTime>2009-04-02T12:00:00Z</EffectiveStartDateTime>
  <EffectiveEndDateTime>2009-04-03T12:00:00.0Z</EffectiveEndDateTime>
  <!--Placement scope-->
  <Scope>Local</Scope>
</PlacementConstraints>
```

14.28 PlacementControl Element

The PlacementControl element describes an individual positional placement opportunity subunit within a placement opportunity description. The PlacementControl element *may* be included in a PlacementRequest message as part of the ADM communicating details regarding the positional placement opportunity. See Section 12.1 for additional information.

14.28.1 PlacementControl Element Schema

Figure 70 illustrates the PlacementControl element's schema.

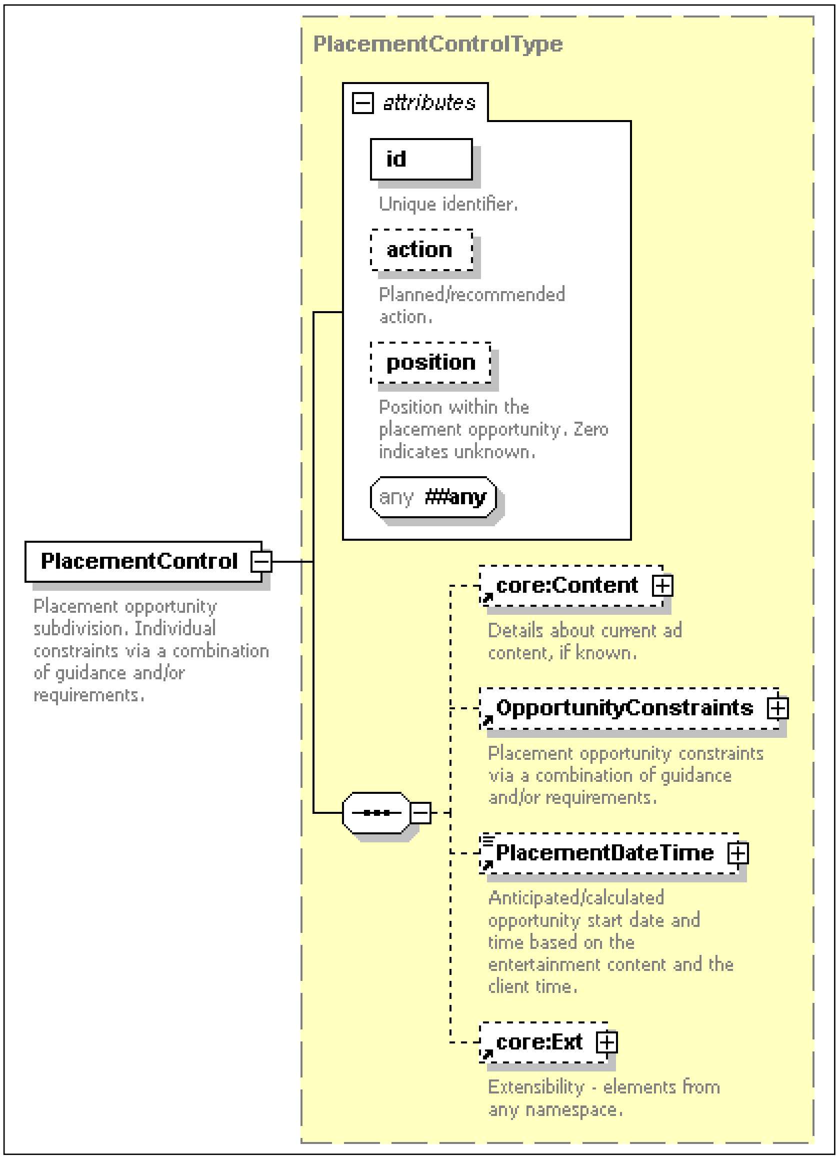

PlacementControlType
attributes
id
Unique identifier.
action
Planned/recommended action.
position
Position within the placement opportunity. Zero indicates unknown.
any ##any
PlacementControl
Placement opportunity subdivision. Individual constraints via a combination of guidance and/or requirements.
core:Content
Details about current ad content, if known.
OpportunityConstraints
Placement opportunity constraints via a combination of guidance and/or requirements.
PlacementDateTime
Anticipated/calculated opportunity start date and time based on the entertainment content and the client time.
core:Ext
Extensibility - elements from any namespace.

Figure 70: PlacementControl Element Schema

14.28.1.1 Semantic Definitions for the PlacementControl Element

@id [Required, core:idAttrType]—An element unique identifier. See [SCTE130-2] for additional information.

@action [Optional, placementActionAttrType]—The behavior planned to be executed by the ADM. See Section 13.1.5 for additional information. Table 11 lists the defined values, which *may* be extended by private agreement outside the scope of this specification. The attribute's value ***shall*** appear exactly as shown in Table 11.

Value	Description
delete	The positional placement opportunity and its currently associated ad content are planned to be removed. Confirmation of this value by the ADS ***shall*** alter the placement structure removing the positional placement opportunity. No currently occupying ad content associated with this positional placement opportunity ***shall*** be output and any existing ad ***`shall`*** be removed.
fill	The positional placement opportunity is currently unoccupied and the placement decision's content bindings are planned to be added or inserted into this location.
fixed	The positional placement opportunity and its associated content ***shall not*** be altered or changed. The element is providing information only.
replace	The currently occupying ad content is planned to be replaced by the supplied content bindings.
…	User defined and outside the scope of this specification. The string ***shall*** be prefixed with the text "private:".

Table 11: PlacementControl Element's @action Attribute Values

@position [Optional, positionTypeAttr]—Positional placement opportunity subunit within the placement opportunity subdivision. See Section 13.1.9 for additional information. Typically, this attribute should be included when the positional placement opportunity is not the first or consecutive element within the positional placement opportunity sequence.

@##any [Optional]—Additional attributes from any namespace.

core:Content [Optional]—Identification and details regarding the current content asset occupying this placement opportunity position. See [SCTE130-2] for additional information.

OpportunityConstraints [Optional]—Information detailing specific limitations or requirements applicable to the placement opportunity. See Section 14.25 for additional information.

PlacementDateTime [Optional]—The expected or calculated date and time when this positional placement opportunity subunit ***shall*** be viewed. See Section 14.30 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.28.2 PlacementControl Element Examples

Example 37 illustrates a minimum PlacementControl element which identifies by name the positional placement opportunity. This creates one positional placement opportunity within the placement opportunity context.

```
<PlacementControl id="MinimumDescription"/>
```

Example 37

Example 38 is a positional placement opportunity with a planned action of “replace” and located in the first position (i.e., the start of the placement opportunity). The ad content is to be 30 seconds in duration.

```
<!--Example positional placement opportunity.-->
<PlacementControl id="Slot1" action="replace" position="1">
  <core:Content>
    <!--Current ad content occupying this position.-->
    <core:SpotRef trafficId="8989" spotId="Spot1"/>
  </core:Content>
  <OpportunityConstraints>
    <!--A 30 second ad is required.-->
    <core:Duration>PT30S</core:Duration>
  </OpportunityConstraints>
</PlacementControl>
```

Example 38

Example 39 is a placement opportunity requiring a specific ad type to “replace” the content in the second ad position.

```
<!--Placement opportunity for a specific ad type.-->
<PlacementControl id="Slot2" action="replace" position="2">
  <OpportunityConstraints>
    <!--Want an ad of this specific type.-->
    <core:AdType>Telescoping</core:AdType>
  </OpportunityConstraints>
</PlacementControl>
```

Example 39

Example 40 is a positional placement opportunity requiring at least one ad and up to eight ads to be added. The total ad duration for this position ***shall not*** exceed two minutes.

```
<!--Placement opportunity providing guidance and requirements.-->
<!--One ad required and up to 8 may be provided.-->
<!--Total ad time shall not exceed 2 minutes.-->
<PlacementControl id="Slot3" action="fill">
  <OpportunityConstraints>
    <MaxDuration>PT2M</MaxDuration>
    <MaxPlacementCount>8</MaxPlacementCount>
    <MinPlacementCount>1</MinPlacementCount>
  </OpportunityConstraints>
</PlacementControl>
```

Example 40

14.29 PlacementCount Element

The PlacementCount element specifies a specific Placement element count that ***shall*** be contained within an associated PlacementDecision element.

14.29.1 PlacementCount Element Schema

Figure 71 illustrates the PlacementCount element's schema.

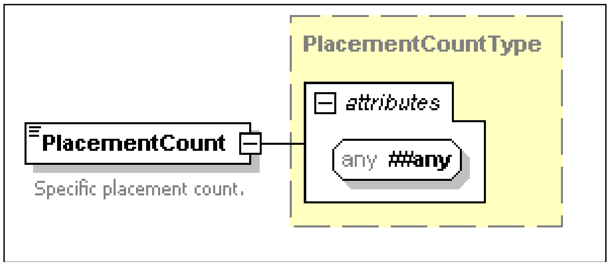


Figure 71: PlacementCount Element Schema

14.29.1.1 Semantic Definitions for the PlacementCount Element

The PlacementCount element inherits from the PlacementCountType complex type. See Section 13.2.3 for semantic definitions applicable to this element.

14.29.2 PlacementCount Element Examples

```
<PlacementCount>0</PlacementCount>
<PlacementCount>2</PlacementCount>
```

14.30 PlacementDateTime Element

The PlacementDateTime element supplies the anticipated or calculated start date and time for a placement opportunity or a placement opportunity subdivision depending on the PlacementDateTime element's usage. The Placement date and time value are based upon the entertainment content asset and the client time.

14.30.1 PlacementDateTime Element Schema

Figure 72 illustrates the PlacementDateTime element's schema.

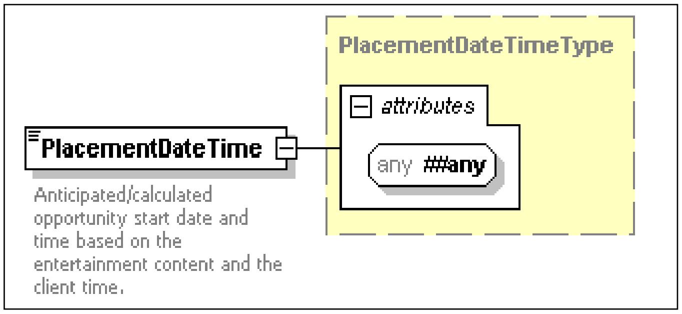

PlacementDateTimeType
attributes
any ##any
PlacementDateTime
Anticipated/calculated opportunity start date and time based on the entertainment content and the client time.

Figure 72: PlacementDateTime Element Schema

14.30.1.1 Semantic Definitions for the PlacementDateTime Element

@##any [Optional]—Additional attributes from any namespace.

The PlacementDateTime element's value is of type core:dateTimeTimezoneType. See [SCTE130-2] for additional information.

14.30.2 PlacementDateTime Element Examples

```
<PlacementDateTime>2007-09-07T05:35:32.0Z</PlacementDateTime>
```

14.31 PlacementDecision Element

The PlacementDecision element conveys the ADS placement decisions. The element is the PlacementResponse or PlacementUpdateNotification message's parallel structure to the PlacementRequest message's PlacementOpportunity element. The element functions as a container for the individual placement decisions supplied as Placement elements. See Section 12.1 for additional information.

The PlacementDecision element ***shall*** identify its associated placement opportunity via the @placementOpportunityRef attribute. The PlacementDecision element *may* assign content bindings to each placement subdivision along with constraints applicable to the placement decision set.

The PlacementDecision element *may* identify the Entertainment asset for which this decision is applicable. If the Entertainment element contained within this element is used, the Entertainment element located at the message root level ***shall not*** be used (i.e., the elements are mutually exclusive).

14.31.1 PlacementDecision Element Schema

Figure 73 illustrates the PlacementDecision element's schema.

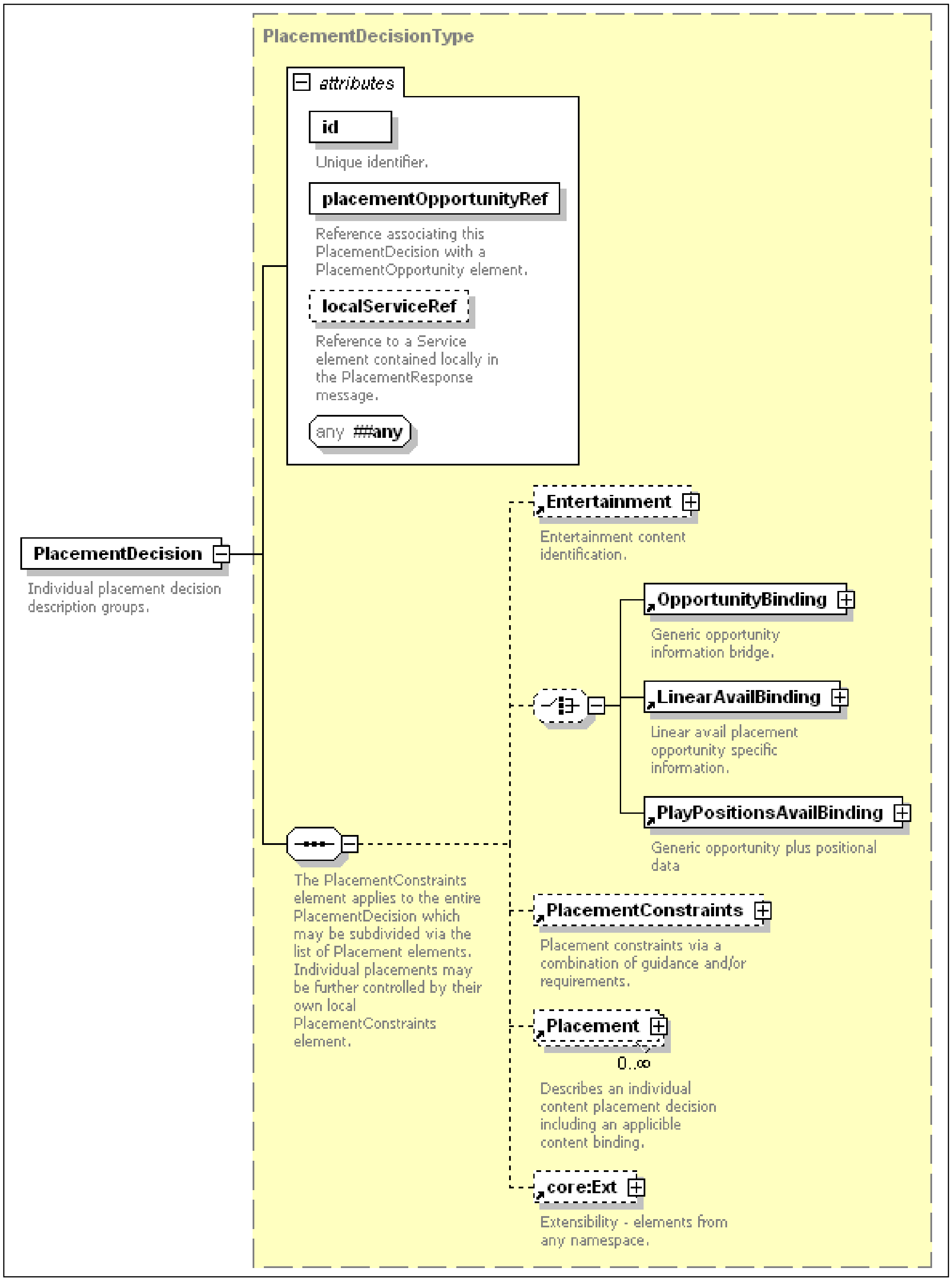

PlacementDecisionType
attributes
id
Unique identifier.
placementOpportunityRef
Reference associating this PlacementDecision with a PlacementOpportunity element.
localServiceRef
Reference to a Service element contained locally in the PlacementResponse message.
any ##any
PlacementDecision
Individual placement decision description groups.
Entertainment
Entertainment content identification.
OpportunityBinding
Generic opportunity information bridge.
LinearAvailBinding
Linear avail placement opportunity specific information.
PlayPositionsAvailBinding
Generic opportunity plus positional data
The PlacementConstraints element applies to the entire PlacementDecision which may be subdivided via the list of Placement elements. Individual placements may be further controlled by their own local PlacementConstraints element.
PlacementConstraints
Placement constraints via a combination of guidance and/or requirements.
Placement
0..∞
Describes an individual content placement decision including an applicible content binding.
core:Ext
Extensibility - elements from any namespace.

Figure 73: PlacementDecision Element Schema

14.31.1.1 Semantic Definitions for the PlacementDecision Element

@id [Required, core:idAttrType]—A element unique attribute identifying this PlacementDecision element. See [SCTE130-2] for additional information.

@placementOpportunityRef [Required, core:idAttrType]—A cross-reference to a PlacementOpportunity element. The @placementOpportunityRef attribute contains the value for the corresponding PlacementOpportunity element's @id attribute.

@localServiceRef [Optional, localServiceRefAttrType]—A cross-reference to a message local Service element providing message specific registration refinement. The value is the message local Service element's @id attribute. See Section 12.1 and Section 13.1.4 for additional information.

@##any [Optional]—Additional attributes from any namespace.

Entertainment [Optional]—The entertainment (i.e., programming) content for which this placement decision is associated. If this element is present, the Entertainment element specified within the root message element ***shall not*** be used (i.e., the entertainment asset identification applies to only this PlacementDecision element). For details regarding a PlacementResponse or a PlacementUpdateNotification message's usage of a single Entertainment element for all message contained PlacementDecision elements, see Section 12.1.2 and Section 12.2.1 respectively. See Section 14.11 for additional information on the Entertainment element.

OpportunityBinding [Optional Choice]—Information specific to defining the placement opportunity relative to the entertainment content. See Section 14.24 for additional information.

LinearAvailBinding [Optional Choice] —Information specific to a linear placement opportunity. See Section 14.15 for additional information.

PlayPositionsAvailBinding [Optional Choice] —Information specific to defining the placement opportunity relative to the entertainment content using a mix of positional information. See Section 14.40 for additional information.

PlacementConstraints [Optional]—The specific limitations that ***shall*** be applied to all placements. See Section 14.27 for additional information.

Placement [Optional]—Zero or more Placement elements describing the individual ADS placement decisions which *may* include a content binding. Each Placement element represents a positional subpart of the placement decision set. See Section 14.26 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.31.2 PlacementDecision Element Examples

In Example 41, the ADS is not altering the current placement arrangement. This fulfills the requirement to reference every placement opportunity.

```
<!--No changes or no applicable decision.-->
<PlacementDecision id="Break1" placementOpportunityRef="Op1"/>
```

Example 41

Example 42 illustrates the placement decision correlated to an existing placement opportunity where two placements are supplied. The first placement is active immediately and the second one replaces the first at the effective start date and time.

```
<PlacementDecision id="Break2" placementOpportunityRef="AC1">
  <Placement id="Slot1 of Break2 active immediately" action="replace">
    <!--Placement decision details...-->
  </Placement>
</PlacementDecision>
<PlacementDecision id="Break2b" placementOpportunityRef="AC1">
  <PlacementConstraints>
    <EffectiveStartDateTime>2007-05-31T12:00:00.0Z</EffectiveStartDateTime>
  </PlacementConstraints>
  <Placement id="Slot1 of Break2 superseding the previous placement decision at the
supplied time" action="replace">
    <!--Placement decision details...-->
  </Placement>
</PlacementDecision>
```

Example 42

Example 43 is a new placement opportunity created by the ADS.

```
  <PlacementDecision id="Break3 - New avail added by ADS"
placementOpportunityRef="AC1”>
    <OpportunityBinding opportunityNumber="3"/>
    <PlacementConstraints>
      <Scope>Local</Scope>
    </PlacementConstraints>
    <Placement id="New spot 1" action="fill">
      <core:Content>
        <core:SpotRef trafficId="897" spotId="spot1"/>
        <core:Tracking>ADSAssignedTrackingID1</core:Tracking>
      </core:Content>
    </Placement>
    <Placement id="New spot 2" action="fill">
      <core:Content>
        <core:SpotRef trafficId="6533" spotId="spot2"/>
        <core:Tracking>ADSAssignedTrackingID2</core:Tracking>
      </core:Content>
    </Placement>
  </PlacementDecision>
```

Example 43

Example 44 is another new placement opportunity created by the ADS. In this case a post roll is defined starting at NPT 120.829 and ending at EOS. Specific Placement’s are included in the PlacementDecision.

```
<PlacementDecision id="Post-Roll - New avail added by ADS"
placementOpportunityRef="AC1" >
  <PlayPositionsAvailBinding>
            <PlayPositionStart scale="1.0">
                      <NPTOffset>120.829</NPTOffset>
            </PlayPositionStart>
            <PlayPositionEnd scale="1.0">
                      <NPTOffset>EOS</NPTOffset>
            </PlayPositionEnd>
  </PlayPositionsAvailBinding>
  <PlacementConstraints>
            <Scope>Local</Scope>
  </PlacementConstraints>
  <Placement id="New spot 1" action="fill">
            <core:Content>
                      <core:SpotRef trafficId="897" spotId="spot1"/>
                      <core:Tracking>ADSAssignedTrackingID1</core:Tracking>
            </core:Content>
  </Placement>
  <Placement id="New spot 2" action="fill">
            <core:Content>
                      <core:SpotRef trafficId="6533" spotId="spot2"/>
                      <core:Tracking>ADSAssignedTrackingID2</core:Tracking>
            </core:Content>
  </Placement>
</PlacementDecision>
```

Example 44

14.32 PlacementDecisionOwner Element

The PlacementDecisionOwner identifies the placement opportunity decision ownership.

14.32.1 PlacementDecisionOwner Element Schema

Figure 74 illustrates the PlacementDecisionOwner element’s schema.

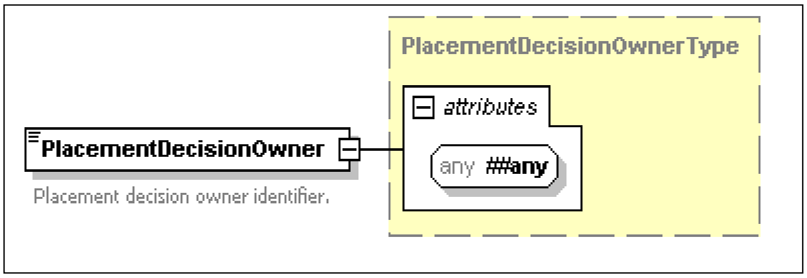

PlacementDecisionOwnerType
attributes
PlacementDecisionOwner
any ##any
Placement decision owner identifier.

Figure 74: PlacementDecisionOwner Element Schema

14.32.1.1 Semantic Definitions for the PlacementDecisionOwner Element

@##any [Optional]—Additional attributes from any namespace.

The PlacementDecisionOwner element's value is of type core:nonEmptyStringType and ***shall not*** be empty. The value identifies ownership of the placement decision.

14.32.2 PlacementDecisionOwner Element Examples

```
<PlacementDecisionOwner>CableCo</PlacementDecisionOwner>
<PlacementDecisionOwner>ESPN</PlacementDecisionOwner>
<PlacementDecisionOwner>Walt</PlacementDecisionOwner>
```

14.33 PlacementOpportunity Element

The ADM uses the PlacementOpportunity element in a PlacementRequest message to communicate details regarding a placement opportunity. The PlacementOpportunity element describes the individual placement opportunities associated with an entertainment content asset. The PlacementOpportunity element cross-references and *may* optionally refine the original ADS registration. This refinement is useful for providing additional details regarding the placement opportunity for broad placement service registrations. Additionally, the element *may* contain opportunity specific information including constraints applicable to the entire placement opportunity and it *may* include a calculated placement start time estimate.

The placement opportunity *may* be subdivided and each subunit (referred to as a positional placement opportunity) is described using the PlacementControl element. See Section 12.1 for additional information. PlacementControl elements *may* describe the current content binding (i.e., the ad currently located in the position) and any applicable limitations (i.e., constraints).

The PlacementOpportunity element *may* identify the Entertainment asset for which this opportunity is applicable. If the Entertainment element contained within this element is used, the Entertainment element located at the message root level ***shall not*** be used (i.e., the elements are mutually exclusive).

14.33.1 PlacementOpportunity Element Schema

Figure 75 illustrates the PlacementOpportunity element’s schema.

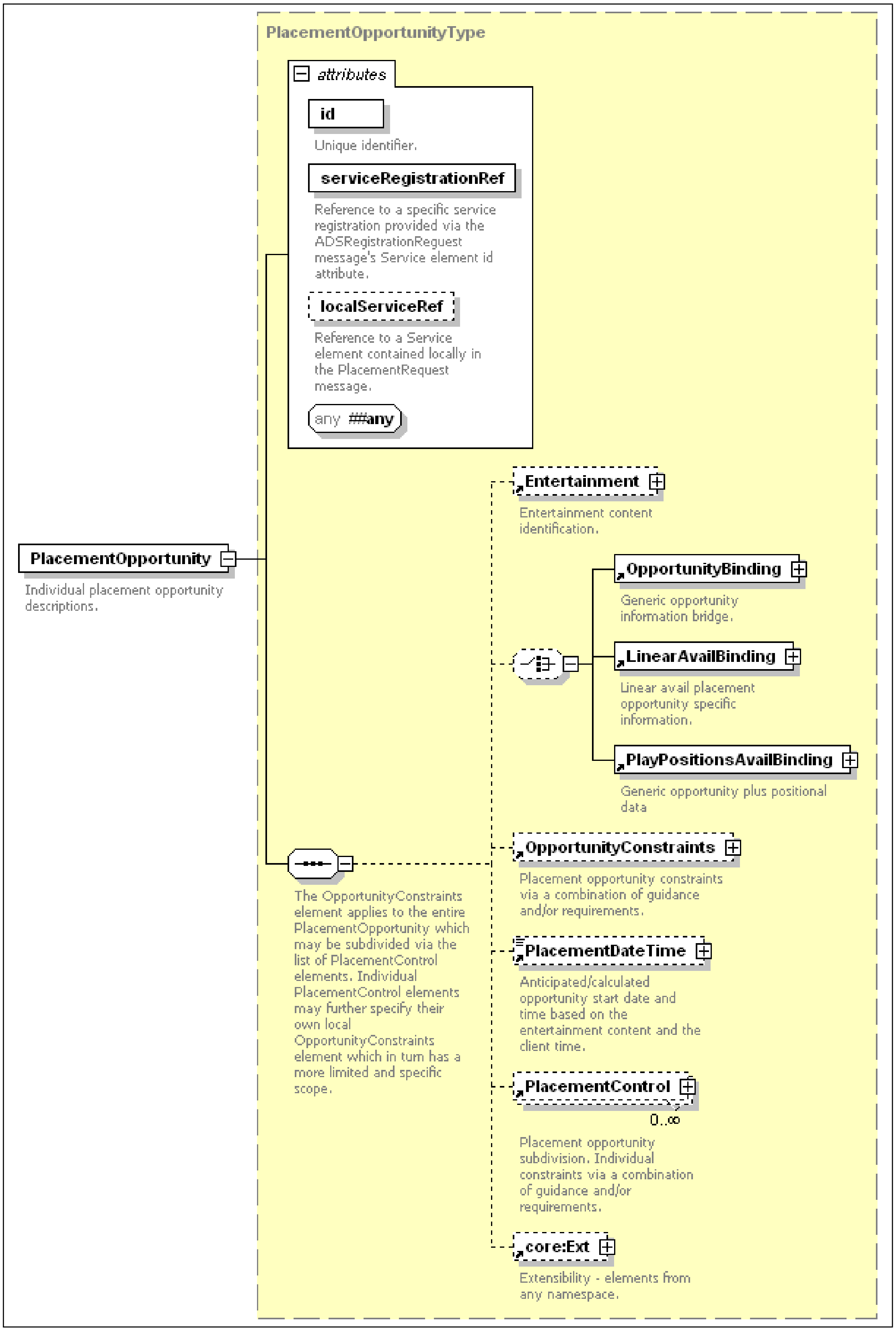

PlacementOpportunityType
attributes
id
Unique identifier.
serviceRegistrationRef
Reference to a specific service registration provided via the ADSRegistrationRequest message's Service element id attribute.
localServiceRef
Reference to a Service element contained locally in the PlacementRequest message.
any ##any
PlacementOpportunity
Individual placement opportunity descriptions.
Entertainment
Entertainment content identification.
OpportunityBinding
Generic opportunity information bridge.
LinearAvailBinding
Linear avail placement opportunity specific information.
PlayPositionsAvailBinding
Generic opportunity plus positional data
OpportunityConstraints
Placement opportunity constraints via a combination of guidance and/or requirements.
The OpportunityConstraints element applies to the entire PlacementOpportunity which may be subdivided via the list of PlacementControl elements. Individual PlacementControl elements may further specify their own local OpportunityConstraints element which in turn has a more limited and specific scope.
PlacementDateTime
Anticipated/calculated opportunity start date and time based on the entertainment content and the client time.
PlacementControl
0..∞
Placement opportunity subdivision. Individual constraints via a combination of guidance and/or requirements.
core:Ext
Extensibility - elements from any namespace.

Figure 75: PlacementOpportunity Element Schema

14.33.1.1 Semantic Definitions for the PlacementOpportunity Element

@id [Required, core:idAttrType]—A element unique placement opportunity identifier. See [SCTE130-2] for additional information.

@serviceRegistrationRef [Optional, core:idAttrType]—A cross-reference to the registration-established placement service that was "matched" facilitating the message delivery to a specific ADS. When present the value ***shall*** be the Service element's @id attribute's value of the matched placement service (i.e., the ServiceDescription element's Service element @id attribute). The registration reference *may* be refined using other elements contained in the PlacementRequest message. See Section 12.1 and [SCTE130-2] for additional information.

@localServiceRef [Optional, localServiceRefAttrType]—A cross-reference to a message local Service element providing message specific registration refinement (i.e., a cross-reference to a Service element local to the message which provides a message specific registration refinement). The value is the message local Service element's @id attribute. See Section 12.1 and Section 13.1.4 for additional information.

@##any [Optional]—Additional attributes from any namespace.

Entertainment [Optional]—The entertainment (i.e., programming) content for which this placement opportunity is associated. If this element is present, the Entertainment element specified within the root message element ***shall not*** be used (i.e., the entertainment asset identification applies to only this PlacementOpportunity element). For details regarding a PlacementRequest message's usage of a single Entertainment element for all message contained PlacementOpportunity elements, see Section 12.1.1. See Section 14.11 for additional information on the Entertainment element.

OpportunityBinding [Optional Choice]—Information specific to defining the placement opportunity relative to the entertainment content. See Section 14.24 for additional information.

LinearAvailBinding [Optional Choice] —Information specific to a linear placement opportunity. See Section 14.15 for additional information.

PlayPositionsAvailBinding [Optional Choice] —Information specific to defining the placement opportunity relative to the entertainment content using a mix of positional information. See Section 14.40 for additional information.

OpportunityConstraints [Optional]—Information detailing specific limitations or requirements applicable to the placement opportunity. See Section 14.25 for additional information.

PlacementDateTime [Optional]—Estimated placement date and time. See Section 14.30 for additional information.

PlacementControl [Optional]—Zero or more positional placement opportunity descriptions which further refine the placement opportunity. See Section 14.28 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.33.2 PlacementOpportunity Element Examples

Example 45 is a minimum placement opportunity description. The description contains the mandatory identifier and optional correlation to a matching registration-established placement service.

```
<PlacementOpportunity id="AC0" serviceRegistrationRef="MatchingRegRef1"/>
```

Example 45

Example 46 illustrates two placement opportunities one for a pre-roll and one for a post-roll. The pre-roll opportunity is unconstrained while the post-roll opportunity requires locally scoped placements which are being added (via the “fill” @action attribute value). The post-roll opportunity further constrains the first positional placement being the specified ad type.

```
<PlacementOpportunity id="Break1" serviceRegistrationRef="ServiceRef1">
  <OpportunityBinding opportunityType="preRoll"/>
</PlacementOpportunity>
<PlacementOpportunity id="Break2" serviceRegistrationRef="ServiceRef1">
  <OpportunityBinding opportunityType="postRoll" opportunityNumber="1"/>
  <OpportunityConstraints>
    <Scope>Local</Scope>
  </OpportunityConstraints>
  <PlacementControl id="PC1" action="fill">
    <OpportunityConstraints>
      <core:AdType>Telescoping</core:AdType>
    </OpportunityConstraints>
  </PlacementControl>
</PlacementOpportunity>
```

Example 46

Example 47 describes a linear replacement placement opportunity.

```
<PlacementOpportunity id="ADifferenBreak1" serviceRegistrationRef="serviceRef3">
  <LinearAvailBinding opportunityType="interstitial">
    <ProviderAvailID>893</ProviderAvailID>
  </LinearAvailBinding>
</PlacementOpportunity>
```

Example 47

Example 48 describes a linear replacement placement opportunity with associated signalId's at the beginning and end of the PO. The provided UTC time codes provide wall clock information to the ADM so an accurate progress bar can be calculated in the case where replacement ad durations do not match the duration of the underlying advertisements.

```
<PlacementOpportunity id="LinearBreak1" serviceRegistrationRef="serviceRef1">
  <PlayPositionsAvailBinding>
            <PlayPositionStart>
                      <SignalId>SIGNAL:DH1h4oVbEeG7r3ULSCQBmw==</SignalId>
                      <!-- UTC time of the in point -->
                      <UTC scale="1.0">2010-11-09T11:13:10.123Z</UTC>
            </PlayPositionStart>
            <!-- Signal ID associated with the next out point -->
            <PlayPositionEnd>
                      <SignalId>SIGNAL:N6t2mBpHTO6ZsZmzpsDVMg==</SignalId>
                      <!-- UTC time of the out point -->
                      <UTC scale="1.0">2010-11-09T11:41:19.456Z</UTC>
            </PlayPositionEnd>
  </PlayPositionsAvailBinding>
</PlacementOpportunity>
```

Example 48

14.34 PlacementStatusEvent Element

The PlacementStatusEvent element provides reporting details regarding an individual placement related happening which the ADM system considers of interest. Each event should only specify a single occurrence of the fact of interest.

14.34.1 PlacementStatusEvent Element Schema

Figure 76 illustrates the PlacementStatusEvent element's schema.

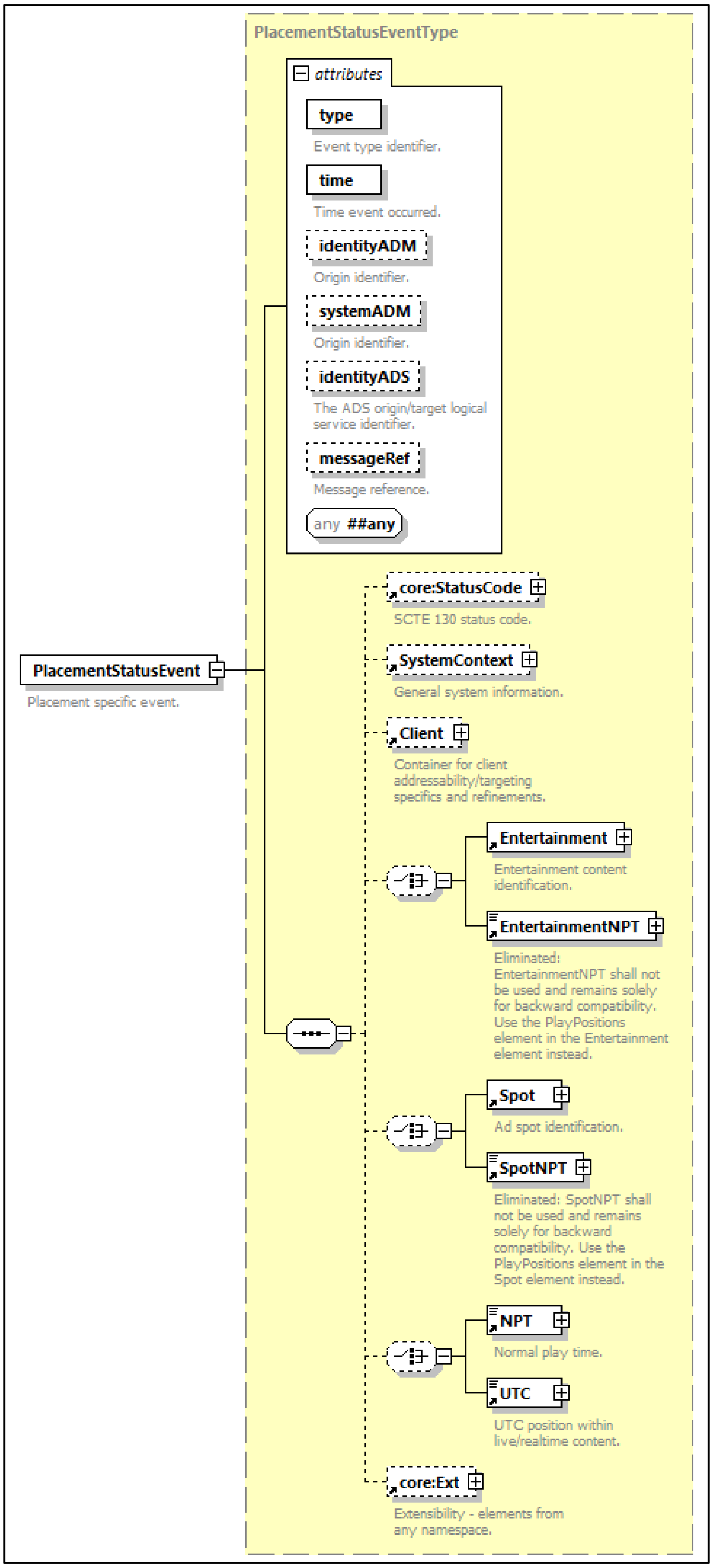

PlacementStatusEventType
attributes
type
Event type identifier.
time
Time event occurred.
identityADM
Origin identifier.
systemADM
Origin identifier.
identityADS
The ADS origin/target logical service identifier.
messageRef
Message reference.
any ##any
core:StatusCode
SCTE 130 status code.
SystemContext
General system information.
Client
Container for client addressability/targeting specifics and refinements.
Entertainment
Entertainment content identification.
EntertainmentNPT
Eliminated: EntertainmentNPT shall not be used and remains solely for backward compatibility. Use the PlayPositions element in the Entertainment element instead.
Spot
Ad spot identification.
SpotNPT
Eliminated: SpotNPT shall not be used and remains solely for backward compatibility. Use the PlayPositions element in the Spot element instead.
NPT
Normal play time.
UTC
UTC position within live/realtime content.
core:Ext
Extensibility - elements from any namespace.
PlacementStatusEvent
Placement specific event.

Figure 76: PlacementStatusEvent Element Schema

14.34.1.1 Semantic Definitions for the PlacementStatusEvent Element

@type [Required, eventTypeAttrType]—Placement status event type identifier as defined in Section 13.2.1. Table 12 lists the defined values for this element that *may* be extended by private agreement outside the scope of this specification. The attribute's value ***shall*** appear exactly as shown in Table 12.

Value	Description
status	The event contains placement status related status information. The event should contain a core:StatusCode element and zero or more Note elements.
startPlacement	The event indicates an entrance to Placement element's content binding. Either this event element or a higher-level scope element ***shall*** contain a Spot element.
endPlacement	The event indicates an exit from a Placement element's content binding. Either this event element or a higher-level scope element ***shall*** contain a Spot element.
endPlacementAll	Simultaneously indicates both "endPlacementUpdates" and "endPlacementsEvents" for a Placement element's content binding. Either this event element or a higher-level scope element ***shall*** contain a Spot element.
endPlacementEvents	The ADS should not expect to receive any additional named event elements with respect to the Placement element's content binding with the @tracking attribute. Any named event elements received following the receipt of this PlacementStatusEvent element *may* be ignored as the ADS is free to release resources associated with the identified Placement element. Either this event element or a higher-level scope element ***shall*** contain a Spot element.
endPlacementUpdates	The Placement element's content binding with the @tracking attribute no longer *may* be altered even if the fulfillment process for the active placement decision set containing the Placement element has not completed. Subsequent placement update messaging pertaining to the referenced Placement element ***shall not*** occur and the ADS is free to release resources associated with the Placement element. If the PlacementRequest message's @updatesAllowed attribute has the value false, this event signal is not required. Either this event element or a higher level scope element ***shall*** contain a Spot element.
endUpdates	The PlacementRequest message's placement decision set associated with the @messageRef attribute no longer *may* be altered even if the fulfillment process for the active placement decision set has not completed. Subsequent placement update messaging pertaining to the referenced PlacementRequest message ***shall not*** occur and the ADS is free to release resources associated with the placement decision set. If the PlacementRequest message's @updatesAllowed attribute has the value false, this event signal is not required. The PlacementStatusEvent element ***shall*** include the @messageRef attribute and ***shall*** include the @identityADS attribute either in the higher-level scope PlayData element or in this element.

endEvents	The ADS should not expect to receive any additional named event elements with respect to the PlacementRequest message's referenced placement decision. Any named event elements received following the receipt of this PlacementStatusEvent element *may* be ignored as the ADS is free to release resources associated with the identified PlacementRequest message. The PlacementStatusEvent element ***shall*** include the @messageRef attribute and ***shall*** include the @identityADS attribute either in the higher-level scope PlayData element or in this element.
endAll	Simultaneously indicates both "endUpdates" and "endEvents". The PlacementStatusEvent element ***shall*** include the @messageRef attribute and ***shall*** include the @identityADS attribute either in the higher-level scope PlayData element or in this element.
…	User defined and outside the scope of this specification. The value ***shall*** be preceded by the string "private:".

Table 12: PlacementStatusEvent @type Attribute Values

Table 13 identifies which PlayPosition elements *may* be used with each of the @type attribute values.

	Element		
Value	**PlayPosition**	**PlayPositionStart**	**PlayPositionEnd**
status	Yes	No	No
startPlacement	No	Yes	No
endPlacement	No	No	Yes
endPlacementAll	No	No	Yes
endPlacementEvents	No	No	Yes
endPlacementUpdates	No	No	Yes
endUpdate	No	No	Yes
endEvents	No	No	Yes
endAll	No	No	Yes

Table 13: PlacementStatusEvent @type Attribute Position Element Usage

@time [Required, core:dateTimeTimezoneType]—The event occurrence date and time. See Section 13.2.1 for additional information.

@identityADM [Optional, core:identityAttrType]—The origin ADM logical service identifier. See Section 13.2.1 for additional information.

@systemADM [Optional, core:systemAttrType]—The originating ADM system identification. See Section 13.2.1 for additional information.

@identityADS [Optional, core:identityAttrType]—The ADS origin or target logical service identifier. See Section 13.2.1 for additional information.

@messageRef [Optional, core:messageRefAttrType]—The message associated with this event. See Section 13.2.1 for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:StatusCode [Optional]—An applicable status code as defined in Appendix A. See Section 13.2.1 for additional information.

SystemContext [Optional]—Parameters describing the event's system context. See Section 13.2.1 for additional information.

Client [Optional]—Addressability or targeting details. See Section 13.2.1 for additional information.

Entertainment [Optional Choice]—Entertainment content asset identification. See Section 13.2.1 for additional information.

EntertainmentNPT [Optional Choice]—This element ***shall not*** be used and remains for backward compatibility only. In messages having a @version attribute of "1.3" or prior, this element *may* appear. It ***shall not*** be used in messages having an @version attribute of "1.4" or later. An entertainment specific NPT value is provided using the NPTOffset element contained with the Entertainment element.

Spot [Optional Choice]—Ad content asset identification. See Section 13.2.1 for additional information.

SpotNPT [Optional Choice]—This element ***shall not*** be used and remains for backward compatibility only. In messages having a @version attribute of "1.3" or prior, this element *may* appear. It ***shall not*** be used in messages having an @version attribute of "1.4" or later. A spot specific NPT value is provided using the NPTOffset element contained with the Spot element.

NPT [Optional Choice]—Viewer perceived NPT. See Section 13.2.1 for additional information.

UTC [Optional Choice]—A UTC value referenced to a stream absolute position within live/realtime content. See Section 14.56 for additional information. The UTC is typically the clock the viewer associates with a live/realtime program or entertainment asset including ads that are delivered as part of a live/realtime asset.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.34.2 PlacementStatusEvent Element Examples

Example 49 illustrates a PlacementStatusEvent for a startPlacement event. The event is self describing and the ADS included a core:Tracking element in the original Placement element core:Content element (not shown). Thus, the core:Tracking element is echoed in the same relative position.

```
<PlacementStatusEvent time="2007-01-31T02:00:00Z" type="startPlacement">
  <Spot>
    <core:Content>
      <core:AssetRef providerID="advertiser1.com"
                assetID="ADVR1000123456789001"/>
      <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C92</core:Tracking>
    </core:Content>
    <PlayPositions scale="1">
      <PlayPositionStart>
        <NPTOffset>0</NPTOffset>
      </PlayPositionStart>
    </PlayPositions>
  </Spot>
</PlacementStatusEvent>
```

Example 49

Example 50 illustrates a PlacementStatusEvent for an endPlacement event. The event is self describing and the ADS included a core:Tracking element in the original Placement element’s scope (not shown). Thus, the core:Tracking element is echoed in the Spot element scope.

```
<PlacementStatusEvent time="2007-01-31T02:30:00Z" type="endPlacement">
  <Spot>
    <core:Tracking>FFFFDC7-6C60-0030-B2B7-B68DAF952C83</core:Tracking>
    <core:Content>
      <core:AssetRef providerID="advertiser1.com"
                assetID="ADVR1000123456789002"/>
    </core:Content>
    <PlayPositions scale="1">
      <PlayPositionEnd>
        <NPTOffset>30</NPTOffset>
      </PlayPositionEnd>
    </PlayPositions>
  </Spot>
</PlacementStatusEvent>
```

Example 50

Example 51 illustrates the startPlacement and endPlacement events which *may* be contained in a SpotScopedEvents element (not shown) and contain a minimum information set.

```
<PlacementStatusEvent time="2007-01-31T02:00:00Z" type="startPlacement">
  <Spot>
    <core:Content/>
    <PlayPositions>
      <PlayPositionStart>
        <NPTOffset>0</NPTOffset>
      </PlayPositionStart>
    </PlayPositions>
  </Spot>
</PlacementStatusEvent>
<PlacementStatusEvent time="2007-01-31T02:30:00Z" type="endPlacement">
  <Spot>
    <core:Content/>
    <PlayPositions>
      <PlayPositionEnd>
        <NPTOffset>30</NPTOffset>
      </PlayPositionEnd>
    </PlayPositions>
  </Spot>
</PlacementStatusEvent>
```

Example 51

Example 52 illustrates a PlacementStatusEvent for a startPlacement event. The event is self describing and it references the container entertainment asset. The event also includes an NPT element indicating the event occurred at 5 minutes and 10.3 seconds into the viewer experience timeline.

```
<PlacementStatusEvent time="2007-01-31T02:00:00Z" type="startPlacement">
  <Entertainment>
    <core:Content>
      <core:AssetRef providerID="videobus.com"
                assetID="BAAD9876543210987654"/>
    </core:Content>
  </Entertainment>
  <Spot>
    <core:Content>
      <core:AssetRef providerID="advertiser1.com"
                assetID="ADVR1000123456789001"/>
      <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C92</core:Tracking>
    </core:Content>
    <PlayPositions>
      <PlayPositionStart>
        <NPTOffset>0</NPTOffset>
      </PlayPositionStart>
    </PlayPositions>
  </Spot>
  <NPT scale="1">310.3</NPT>
</PlacementStatusEvent>
```

Example 52

Example 53 illustrates a PlacementStatusEvent for an endPlacementAll event. The event is self describing and it references the container Spot asset. The event also includes an NPT element indicating the event occurred at the end of the referenced Spot asset.

```
<PlacementStatusEvent time="2007-01-31T02:00:00Z" type="endPlacementAll">
  <Spot>
    <core:Content>
      <core:AssetRef providerID="advertiser1.com"
                assetID="ADVR1000123456789001"/>
      <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C92</core:Tracking>
    </core:Content>
    <PlayPositions>
      <PlayPositionEnd>
        <NPTOffset>EOS</NPTOffset>
      </PlayPositionEnd>
    </PlayPositions>
  </Spot>
</PlacementStatusEvent>
```

Example 53

14.35 PlayData Element

The PlayData element functions as a top level container for collections of named event elements. The named event elements *may* be organized in several different ways. For example, each named event *may* be self describing and all the events *may* be in a single Events element. (See Section 0 for additional information on the Events element.)

Alternatively, an ADM implementation *may* desire to minimize the named event data size and consequently, *may* choose to use the EntertainmentScopedEvents element and SpotScopedEvents element to reduce the data replication. See Section 14.13 and Section 14.51 for additional information regarding the EntertainmentScopedEvents and SpotScopedEvents elements respectively. In this approach, static data is output in a higher level scoped element and the named events are then interpreted in the context of the higher level scoped element’s applicable data.

14.35.1 PlayData Element Schema

Figure 77 illustrates the PlayData element’s schema.

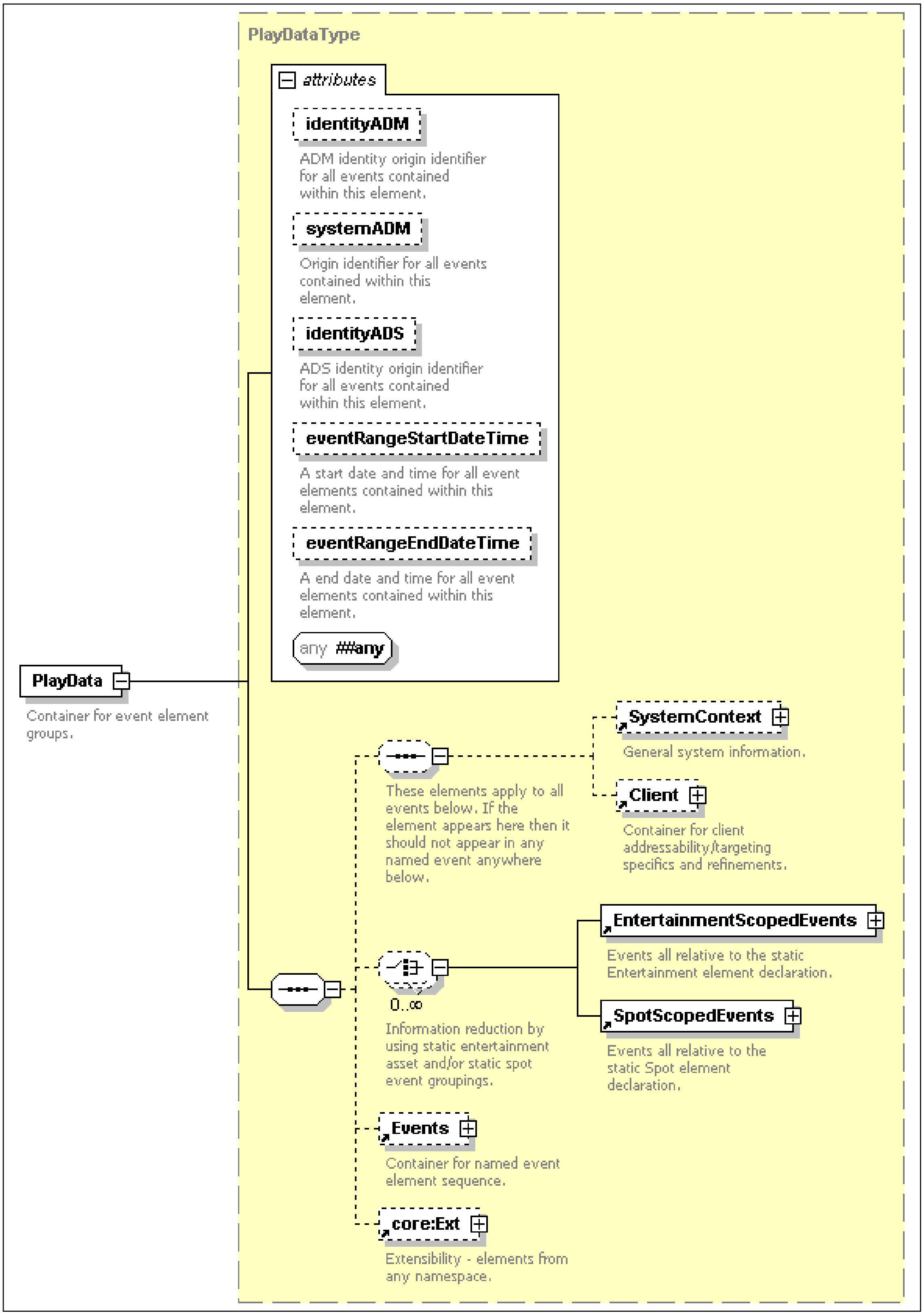

PlayDataType
attributes
identityADM
ADM identity origin identifier for all events contained within this element.
systemADM
Origin identifier for all events contained within this element.
identityADS
ADS identity origin identifier for all events contained within this element.
eventRangeStartDateTime
A start date and time for all event elements contained within this element.
eventRangeEndDateTime
A end date and time for all event elements contained within this element.
any ##any
PlayData
Container for event element groups.
These elements apply to all events below. If the element appears here then it should not appear in any named event anywhere below.
SystemContext
General system information.
Client
Container for client addressability/targeting specifics and refinements.
EntertainmentScopedEvents
Events all relative to the static Entertainment element declaration.
SpotScopedEvents
Events all relative to the static Spot element declaration.
0..∞
Information reduction by using static entertainment asset and/or static spot event groupings.
Events
Container for named event element sequence.
core:Ext
Extensibility - elements from any namespace.

Figure 77: PlayData Element Schema

When the @identityADM, the @systemADM, the @identityADS, the SystemContext element and the Client element are included in the PlayData element, the information ***shall*** apply to all elements contained in any child event sequences. Additionally, if the SystemContext or Client elements appear at this scope, the elements ***shall not*** appear in any child element.

14.35.1.1 Semantic Definitions for the PlayData Element

@identityADM [Optional, core:identityAttrType]—The ADM origin logical service identifier. All events contained within this PlayData element originated from this specific ADM. See [SCTE130-2] for additional information.

@systemADM [Optional, core:systemAttrType]—The originating ADM system identification applicable to all events contained within this PlayData element. See [SCTE130-2] for additional information.

@identityADS [Optional, core:identityAttrType]—The ADS origin or target logical service identifier. All events contained within this PlayData element are applicable to the listed ADS. See [SCTE130-2] for additional information.

@eventRangeStartDateTime [Optional, eventRangeStartDateTimeAttrType]—An initial date and time for the named event elements contained within this PlayData element. See Section 13.1.2 for additional details.

@eventRangeEndDateTime [Optional, eventRangeEndDateTimeAttrType]—An ending date and time for the named event elements contained within this Playdata element. See Section 13.1.1 for additional details.

@##any [Optional]—Additional attributes from any namespace.

SystemContext [Optional]—The system environment applicable to all listed events. See Section 14.52 for additional information. If present, this element ***shall not*** appear in any child element.

Client [Optional]—A container element describing the client context applicable to all listed events. See Section 14.4 for additional information. If present, this element ***shall not*** appear in any child element.

The choice sequence allows for zero or more EntertainmentScopedEvents or SpotScopedEvents elements to be present and combined in any quantity and any order (i.e., the element count and mix is unspecified).

EntertainmentScopedEvents [Optional Choice]—A container for named event elements scoped by an Entertainment element. See Section 14.13 for additional information.

SpotScopedEvents [Optional Choice]—A container for named event elements scoped by a Spot element. See Section 14.51 for additional information.

Events [Optional]—A container for named event elements describing individual events. See Section 0 for additional information. Named events contained within this element are all self describing.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.35.2 PlayData Element Examples

The PlayData element in Example 54 shows how a sequence of self-describing named event elements is carried. The PlayData element's @identityADS values highlights that all the events are for the same ADS.

```
<PlayData
  identityADS="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5"
  eventRangeStartDateTime="2007-01-31T02:00:00Z"
  eventRangeEndDateTime="2007-01-31T02:00:30Z">
 <Events>
  <PlacementStatusEvent time="2007-01-31T02:00:00Z"
             type="startPlacement">
   <Spot>
    <core:Content>
     <core:Tracking>5B8ECCED-979B-421E-8A6E-E2192B11516C</core:Tracking>
    </core:Content>
    <PlayPositions>
     <PlayPositionStart>
      <NPTOffset>0</NPTOffset>
     </PlayPositionStart>
    </PlayPositions>
   </Spot>
  </PlacementStatusEvent>
  <!--N more events.-->
  <PlacementStatusEvent time="2007-01-31T02:00:30Z"
             type="endPlacement">
   <Spot>
    <core:Content>
     <core:Tracking>5B8ECCED-979B-421E-8A6E-E2192B11516C</core:Tracking>
    </core:Content>
    <PlayPositions>
     <PlayPositionEnd>
      <NPTOffset>30</NPTOffset>
     </PlayPositionEnd>
    </PlayPositions>
   </Spot>
  </PlacementStatusEvent>
 </Events>
</PlayData>
```

Example 54

Example 55 illustrates a PlayData element containing named event elements in the Events element which each identify a unique destination ADS. Consequently, the PlayData element’s @identityADS attribute is not used.

```
 <PlayData>
  <Events>
   <SessionEvent identityADS="ADSLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F5" type="startSession" time="2008-02-22T09:30:47Z">
    <SystemContext>
     <Session>SessionID98398938</Session>
    </SystemContext>
   </SessionEvent>
   <SessionEvent identityADS="ADS_FFFFFEF4-C19B-450E-B624-421B23F52FFF"
type="endSession" time="2008-02-22T09:30:48Z">
    <SystemContext>
     <Session>ID9837464</Session>
    </SystemContext>
   </SessionEvent>
  </Events>
 </PlayData>
```

Example 55

Example 56 shows how information *may* be reduced using the EntertainmentScopedEvents and SpotScopedEvents elements.

```
<PlayData
  identityADS="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5">
 <SystemContext>
  <Session>ExampleSession1</Session>
 </SystemContext>
 <EntertainmentScopedEvents>
  <Entertainment>
   <core:Content>
    <core:AssetRef providerID="videobus.com"
            assetID="BAAD9876543210987654"/>
   </core:Content>
  </Entertainment>
  <Events>
   <!--Session creation. Informational note only.-->
   <SessionEvent time="2007-01-31T01:59:59Z" type="startSession">
    <core:StatusCode class="3">
     <core:Note>Session start.</core:Note>
    </core:StatusCode>
   </SessionEvent>
  </Events>
  <SpotScopedEvents>
   <Spot>
    <core:Content>
     <core:AssetRef providerID="advertiser1.com"
             assetID="ADVR1000123456789001"/>
     <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C92</core:Tracking>
    </core:Content>
   </Spot>
   <Events>
    <PlacementStatusEvent time="2007-01-31T02:00:00Z"
               type="startPlacement">
     <Spot>
      <core:Content/>
      <PlayPositions>
       <PlayPositionStart>
        <NPTOffset>0</NPTOffset>
       </PlayPositionStart>
      </PlayPositions>
     </Spot>
    </PlacementStatusEvent>
    <ViewerEvent time="2007-01-31T02:00:05Z" type="pause"
          eventSource="remote">
     <Spot>
      <core:Content/>
      <PlayPositions scale="0">
       <PlayPosition>
```

```
            <NPTOffset>5</NPTOffset>
          </PlayPosition>
        </PlayPositions>
      </Spot>
    </ViewerEvent>
    <ViewerEvent time="2007-01-31T02:00:10Z" type="play"
                 eventSource="remote">
      <Spot>
        <core:Content/>
        <PlayPositions scale="1">
          <PlayPositionStart>
            <NPTOffset>5</NPTOffset>
          </PlayPositionStart>
        </PlayPositions>
      </Spot>
    </ViewerEvent>
    <PlacementStatusEvent time="2007-01-31T02:00:20Z"
                          type="endPlacement">
      <Spot>
        <core:Content/>
        <PlayPositions scale="1.0">
          <PlayPositionEnd>
            <NPTOffset>15</NPTOffset>
          </PlayPositionEnd>
        </PlayPositions>
      </Spot>
    </PlacementStatusEvent>
  </Events>
</SpotScopedEvents>
<Events>
  <PlacementStatusEvent time="2007-01-31T02:00:20Z" type="status">
    <core:StatusCode class="1" detail="11">
      <core:Note>Resource not found.</core:Note>
    </core:StatusCode>
    <Spot>
      <core:Content>
        <core:AssetRef providerID="advertiser2.com"
                       assetID="ADVR2000123456789001"/>
        <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C93</core:Tracking>
      </core:Content>
    </Spot>
  </PlacementStatusEvent>
</Events>
<SpotScopedEvents>
  <Spot>
    <core:Content>
```

```
        <core:AssetRef providerID="advertiser3.com"
                       assetID="ADVR3000123456789001"/>
        <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C94</core:Tracking>
      </core:Content>
    </Spot>
    <Events>
      <PlacementStatusEvent time="2007-01-31T02:00:20Z"
                            type="startPlacement">
        <Spot>
          <core:Content/>
          <PlayPositions scale="1">
            <PlayPositionStart>
              <NPTOffset>0</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Spot>
      </PlacementStatusEvent>
      <ViewerEvent time="2007-01-31T02:00:35Z" type="fastForward"
                   eventSource="remote">
        <Spot>
          <core:Content/>
          <PlayPositions scale="1">
            <PlayPositionEnd>
              <NPTOffset>15</NPTOffset>
            </PlayPositionEnd>
          </PlayPositions>
        </Spot>
      </ViewerEvent>
      <ViewerEvent time="2007-01-31T02:00:36Z" type="fastForward"
                   eventSource="remote">
        <Spot>
          <core:Content/>
          <PlayPositions scale="2">
            <PlayPositionStart>
              <NPTOffset>16</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Spot>
      </ViewerEvent>
      <ViewerEvent time="2007-01-31T02:00:37Z" type="fastForward"
                   eventSource="remote">
        <Spot>
          <core:Content/>
          <PlayPositions scale="4">
            <PlayPositionStart>
              <NPTOffset>18</NPTOffset>
```

```
                </PlayPositionStart>
              </PlayPositions>
            </Spot>
          </ViewerEvent>
          <PlacementStatusEvent time="2007-01-31T02:00:40Z"
                                type="endPlacement">
            <Spot>
              <core:Content/>
              <PlayPositions scale="4">
                <PlayPositionEnd>
                  <NPTOffset>30</NPTOffset>
                </PlayPositionEnd>
              </PlayPositions>
            </Spot>
          </PlacementStatusEvent>
        </Events>
      </SpotScopedEvents>
      <Events>
        <!--Start of the entertainment asset with the subscriber still being in 4x fast forward
mode.-->
        <SessionEvent time="2007-01-31T02:00:40Z" type="startSegment">
          <Entertainment>
            <core:Content/>
            <PlayPositions>
              <PlayPositionStart scale="4">
                <NPTOffset>0</NPTOffset>
              </PlayPositionStart>
            </PlayPositions>
          </Entertainment>
        </SessionEvent>
      </Events>
    </EntertainmentScopedEvents>
  </PlayData>
```

Example 56

14.36 PlayPosition Element

The PlayPosition element provides the ability to specify a position based on a signal identifier, a UTC time or an NPTOffset element. While more than one position element can be supplied in a PlayPosition element, the interpretation by an implementation is out of scope of this standard.

14.36.1 PlayPosition Element Schema

Figure 78 illustrates the PlayPositionType element's schema.

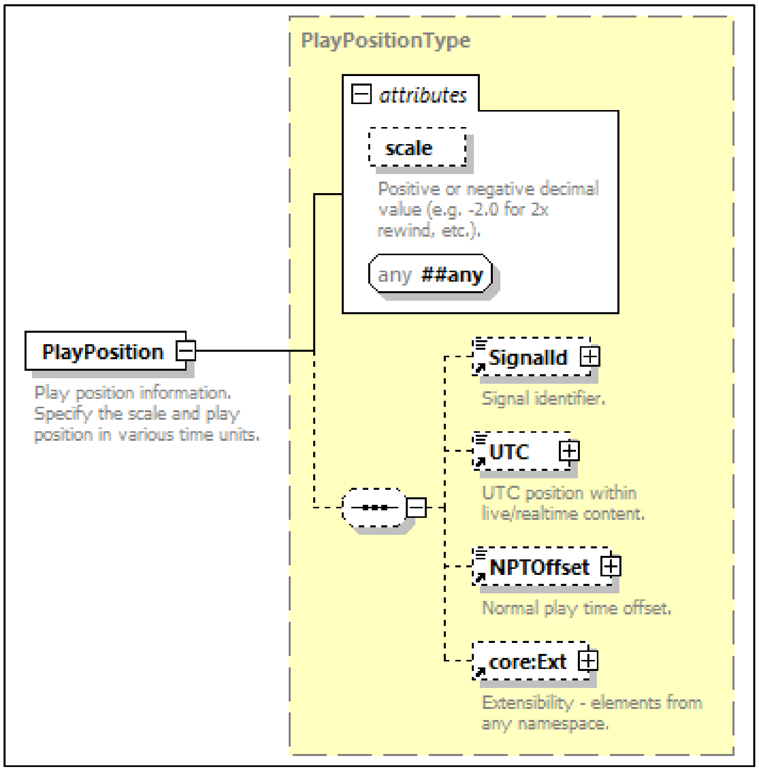

PlayPositionType
attributes
scale
Positive or negative decimal value (e.g. -2.0 for 2x rewind, etc.).
any ##any
PlayPosition
Play position information. Specify the scale and play position in various time units.
SignalId
Signal identifier.
UTC
UTC position within live/realtime content.
NPTOffset
Normal play time offset.
core:Ext
Extensibility - elements from any namespace.

Figure 78: PlayPosition Element Schema

14.36.1.1 Semantic Definitions for the PlayPosition Element

@scale [Optional, xsd:decimal]—The scale value of 1 indicates normal play or record at the natural forward viewing rate. If the attribute is not the value of 1, the value corresponds to the rate with respect to the normal viewing rate. For example, a ratio of 2 indicates twice the natural viewing rate (i.e., "fast forward") and a ratio of 0.5 indicates half the normal viewing rate (i.e., forward at ½ or "slow forward"). In other words, a ratio of 2 has normal playtime increase at twice the wallclock rate. For every second of elapsed (wallclock) time, 2 seconds of content is enumerated. A negative value indicates reverse direction (i.e., "rewind"). For example, the value "-5" indicates fives times the normal viewing rate in reverse.

@##any [Optional]—Additional attributes from any namespace.

SignalId [Optional] – This element provides a signal identifier value reference to a stream within content. The content *may* be live/realtime or a stored asset. Signal identifiers *may* be carried within the stream in an SCTE 35 segmentation descriptor (See [SCTE35]) or described via external metadata (See [SCTE236]).

UTC [Optional] – This element provides a UTC value reference to a stream absolute position within live/realtime content. See Section 14.56 for additional information. The UTC is typically the clock the viewer associates with a live/realtime program or entertainment asset including ads that are delivered as part of a live/realtime asset.

NPTOffset [Optional] – This element provides an NPT relative to a previously specified Entertainment element that was declared within a higher-level scope element's context. See Section 0 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.36.2 PlayPosition Element Examples

The following are usage examples for the PlayPosition element. The first and third examples each contain a single positional element. The second example illustrates 2 positional elements, signal ID and UTC. As stated above, the precise behavior of an implementation when more than one element is supplied is out of scope of this standard (See 0).

```
<PlayPosition>
  <!-- Play at normal forward speed at 11am GMT. -->
  <UTC scale="1">2010-11-09T11:00:00-00:00</UTC>
</PlayPosition>
<PlayPosition>
  <!-- Signal ID of a point of interest. -->
  <SignalId>SIGNAL:ISKQWoVaEeGB5Z8NCkgkAZs=</SignalId>
  <!-- UTC time of the point of interest. -->
  <UTC scale="1.0">2010-11-09T11:12:10.123Z</UTC>
</PlayPosition>
<PlayPosition>
  <!-- 2X play at NPT position 120.383. -->
  <NPTOffset>120.383</NPTOffset>
</PlayPosition>
```

14.37 PlayPositionEnd Element

The PlayPositionEnd element provides the ability to specify an end position based on a signal identifier, a UTC time or an NPTOffset element. While more than one position element *may* be supplied in a PlayPosition element, the interpretation by an implementation is out of scope of this standard.

14.37.1 PlayPositionEnd Element Schema

Figure 78 illustrates the PlayPositionEnd element's schema.

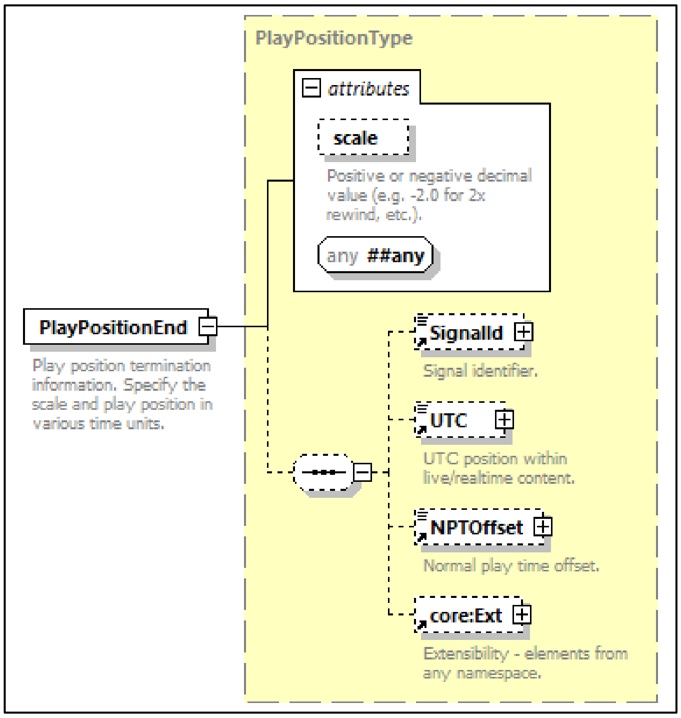

PlayPositionType
attributes
scale
Positive or negative decimal value (e.g. -2.0 for 2x rewind, etc.).
any ##any
PlayPositionEnd
Play position termination information. Specify the scale and play position in various time units.
SignalId
Signal identifier.
UTC
UTC position within live/realtime content.
NPTOffset
Normal play time offset.
core:Ext
Extensibility - elements from any namespace.

Figure 79: PlayPositionEnd Element Schema

14.37.1.1 Semantic Definitions for the PlayPositionEnd Element

@scale [Optional, xsd:decimal]—The scale value of 1 indicates normal play or record at the natural forward viewing rate. If the attribute is not the value of 1, the value corresponds to the rate with respect to the normal viewing rate. For example, a ratio of 2 indicates twice the natural viewing rate (i.e., "fast forward") and a ratio of 0.5 indicates half the normal viewing rate (i.e., forward at ½ or “slow forward”). In other words, a ratio of 2 has normal playtime increase at twice the wallclock rate. For every second of elapsed (wallclock) time, 2 seconds of content is enumerated. A negative value indicates the reverse direction (i.e., “rewind”). For example, the value “-5” indicates fives times the normal viewing rate in reverse.

@##any [Optional]—Additional attributes from any namespace.

SignalId [Optional]—This element provides a signal identifier value reference to a stream within content. The content *may* be live/realtime or a stored asset. Signal identifiers *may* be carried within the stream using an SCTE 35 segmentation descriptor (see [SCTE35]) or described via external metadata (see [SCTE236]).

UTC [Optional]—This element provides a UTC value reference to a stream absolute position within live/realtime content. See Section 14.56 for additional information. The UTC is typically the clock the viewer associates with a live/realtime program or entertainment asset including ads that are delivered as part of a live/realtime asset.

NPTOffset [Optional]—This element provides an NPT relative to a previously specified content element declared within a higher-level scope element's context. See Section 0 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.37.2 PlayPositionEnd Element Examples

The following are usage examples for the PlayPositionEnd element. As stated above, the precise behavior of an implementation when more than one element is supplied is out of scope of this standard (See 0).

```
<PlayPositionEnd>
  <UTC>2010-11-09T11:12:10.123Z</UTC>
  <NPTOffset>22.333</NPTOffset>
</PlayPositionEnd>
```

14.38 PlayPositionStart Element

The PlayPositionStart element provides the ability to specify a start position based on a signal identifier, a UTC time or an NPTOffset element. While more than one position element *may* be supplied in a PlayPosition element, the interpretation by an implementation is out of scope of this standard.

14.38.1 PlayPositionStart Element Schema

Figure 78 illustrates the PlayPositionStart element's schema.

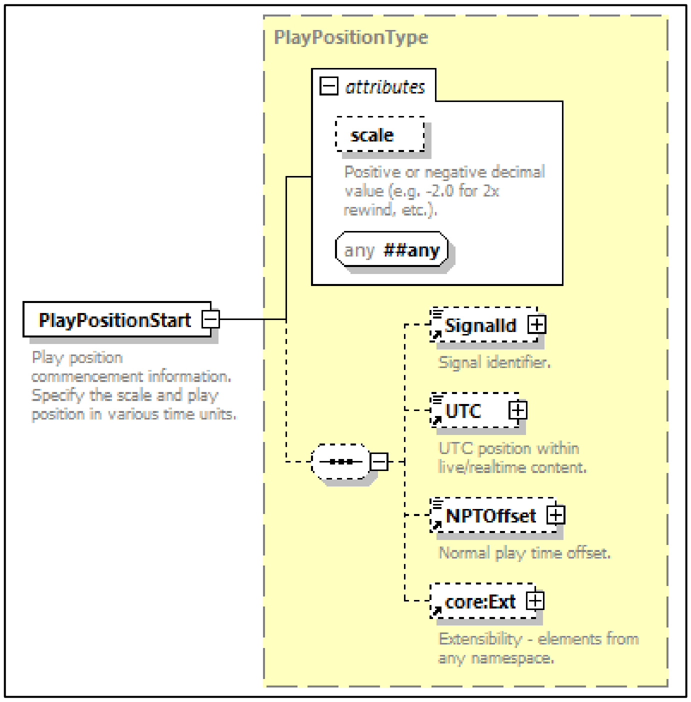

PlayPositionType
attributes
scale
Positive or negative decimal value (e.g. -2.0 for 2x rewind, etc.).
any ##any
PlayPositionStart
Play position commencement information. Specify the scale and play position in various time units.
SignalId
Signal identifier.
UTC
UTC position within live/realtime content.
NPTOffset
Normal play time offset.
core:Ext
Extensibility - elements from any namespace.

Figure 80: PlayPositionStart Element Schema

14.38.1.1 Semantic Definitions for the PlayPositionStart Element

@scale [Optional, xsd:decimal]—The scale value of 1 indicates normal play or record at the natural forward viewing rate. If the attribute is not the value of 1, the value corresponds to the rate with respect to the normal viewing rate. For example, a ratio of 2 indicates twice the natural viewing rate (i.e., "fast forward") and a ratio of 0.5 indicates half the normal viewing rate (i.e., forward at ½ or “slow forward”). In other words, a ratio of 2 has normal playtime increase at twice the wallclock rate. For every second of elapsed (wallclock) time, 2 seconds of content is enumerated. A negative value indicates the reverse direction (i.e., “rewind”). For example, the value “-5” indicates fives times the normal viewing rate in reverse.

@##any [Optional]—Additional attributes from any namespace.

SignalId [Optional]—This element provides a signal identifier value reference to a stream within content. The content *may* be live/realtime or a stored asset. Signal identifiers *may* be carried within the stream using an SCTE 35 segmentation descriptor (see [SCTE35]) or described via external metadata (see [SCTE236]).

UTC [Optional]—This element provides a UTC value reference to a stream absolute position within live/realtime content. See Section 14.56 for additional information. The UTC is typically the clock the viewer associates with a live/realtime program or entertainment asset including ads that are delivered as part of a live/realtime asset.

NPTOffset [Optional]—This element provides an NPT relative to a previously specified content element declared within a higher-level scope element’s context. See Section 0 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.38.2 PlayPositionStart Element Examples

The following are usage examples for the PlayPositionStart element. As stated above, the precise behavior of an implementation when more than one element is supplied is out of scope of this standard (See 0).

```
<PlayPositionStart scale="1">
  <NPTOffset>0</NPTOffset>
</PlayPositionStart>
```

14.39 PlayPositions Element

The PlayPositions element provides the capability to specify a position, a start and/or an end position within an asset.

14.39.1 PlayPositions Element Schema

Figure 81 illustrates the PlayPositions element’s schema.

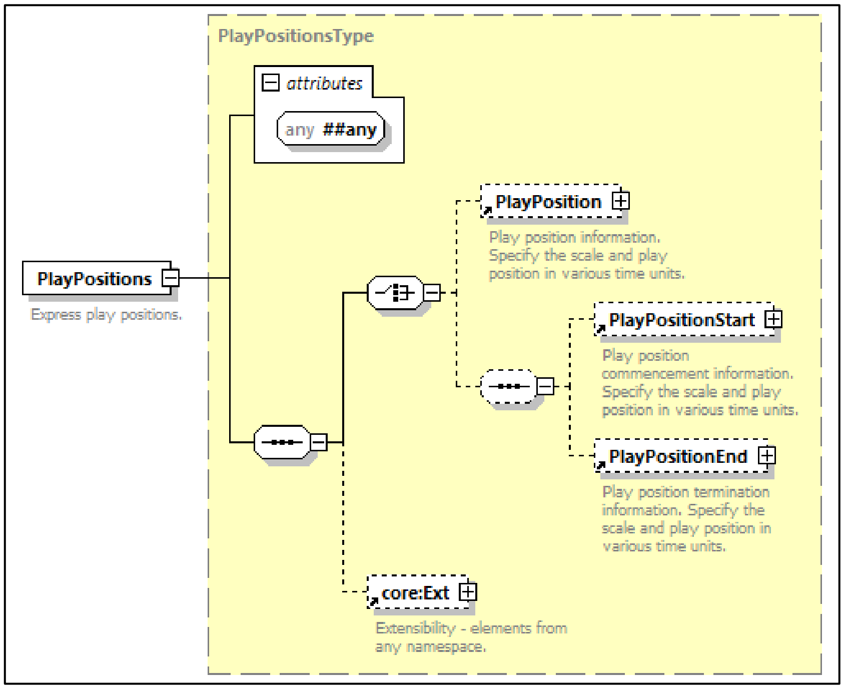

PlayPositionsType
attributes
any ##any
PlayPositions
Express play positions.
PlayPosition
Play position information. Specify the scale and play position in various time units.
PlayPositionStart
Play position commencement information. Specify the scale and play position in various time units.
PlayPositionEnd
Play position termination information. Specify the scale and play position in various time units.
core:Ext
Extensibility - elements from any namespace.

Figure 81: PlayPositions Element Schema

14.39.1.1 Semantic Definitions for the PlayPositions Element

@##any [Optional]—Additional attributes from any namespace.

PlayPosition [Optional Choice]—This element identifies a position within content that typically is not a start or end location. See 0 for additional information.

PlayPositionStart [Optional Choice]—This element identifies a start position within content. See 0 for additional information.

PlayPositionEnd [Optional Choice]—This element identifies an end position within content. See 0 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.39.2 PlayPositions Element Examples

Example 57 illustates the basic usage of the PlayPositions Element.

```
<PlayPositions>
  <PlayPositionStart>
    <!-- Start play at 11am GMT. -->
    <UTC scale="1">2010-11-09T11:00:00-00:00</UTC>
  </PlayPositionStart>
  <PlayPositionEnd>
    <!-- Signal ID of the out point. -->
    <SignalId>SIGNAL:ISKQWoVaEeGB5Z8NCkgkAZs=</SignalId>
    <!-- UTC time of the out point. -->
    <UTC scale="1.0">2010-11-09T11:12:10.123Z</UTC>
  </PlayPositionEnd>
</PlayPositions>
```

Example 57

Example 58 illustrates a more complete usage of the PlayPositions Element. This use case illustrates a DVR based PlacementResponse based on a PlacementRequest for content recorded on 11/9/2010 at 11 GMT for 1 hour. This example uses a combination of PlayPosition elements to fully express the playback sequence of the recording.

The first PlayPosition element contains a PlayPostionStart with a UTC value of 11/9/2010 at 11 GMT. This is the start of the recorded content. The PlayPositionEnd Element has both a Signal ID and UTC time for the first Out Point during the recorded interval.

The second PlayPostion element contains a PlayPositionStart and PlayPostitionEnd that includes both a Signal ID and UTC time. The Signal ID in the PlayPositionStart references the In Point. The Signal ID in the PlayPositionEnd refereces the next Out Point in the recorded interval.

The last PlayPosition element contains a PlayPostionStart with both a Signal ID and UTC time for the second In Point during the recorded interval. The PlayPositionEnd Element has a UTC value of 11/9/2010 at 12 GMT. This is the end of the recorded content.

The UTC times allows an implementation to properly construct a progess bar based on the UTC time deltas for each PlayPosition element plus the core:Duration values for each Placement.

```
<PlacementResponse messageId="5B8ECCED-979B-421E-8A6E-E2192B11516A"
version="1.4" identity="ADSLogicalService_1170FEF4-C19B-450E-B624-
```

```
421B23F525F5" system="ads" messageRef="5B8ECCED-979B-421E-8A6E-
E2192B115160">
    <core:StatusCode class="0"/>
    <SystemContext>
        <Session>b782ce94-b3a7-4d3c-9487-6d9367975c2c</Session>
    </SystemContext>
    <PlacementDecision placementOpportunityRef="528573CD-1485-13A6-
56B778222E2B2936" id="1f87463c-9f60-4d60-ab00-f18daf50a786">
        <Entertainment>
            <core:Content>
                <core:AssetRef providerID="provider.com"
assetID="TNTA000000000000000"/>
            </core:Content>
            <PlayPositions>
                <PlayPositionStart>
                    <!-- Start play at 11am GMT -->
                    <UTC scale="1">2010-11-09T11:00:00-00:00</UTC>
                </PlayPositionStart>
                <PlayPositionEnd>
                    <!-- Signal ID of the out point -->
                    <SignalId>SIGNAL:ISKQWoVaEeGB5Z8NCkgkAZs=</SignalId>
                    <!-- UTC time of the out point -->
                    <UTC scale="1.0">2010-11-09T11:12:10.123Z</UTC>
                </PlayPositionEnd>
            </PlayPositions>
        </Entertainment>
    </PlacementDecision>
    <PlacementDecision placementOpportunityRef="528573CD-1485-13A6-
56B778222E2B2936" id="d61038af-a2d9-4270-bded-2bc2f2caf6bd">
        <Placement action="fill" id="d457c56c-143b-425f-a718-96e27cb88f10">
            <core:Content>
                <core:AssetRef assetID="000CNFF4630" providerID="spotlight.com"/>
                <core:Duration>PT30S</core:Duration>
                <core:Tracking>b9f3dcbd-d37f-4d45-a53a-
39423957e2a5</core:Tracking>
            </core:Content>
        </Placement>
        <Placement action="fill" id="e3fb3eb8-c3d8-4c5d-9678-a34a59d955a4">
            <core:Content>
                <core:AssetRef assetID="0000C183427" providerID="spotlight.com"/>
                <core:Duration>PT30S</core:Duration>
                <core:Tracking>6856a857-bc8c-403a-a13e-
3d0ff7839eb8</core:Tracking>
            </core:Content>
        </Placement>
    </PlacementDecision>
    <PlacementDecision placementOpportunityRef="528573CD-1485-13A6-
56B778222E2B2936" id="a5cd4554-46cc-4167-a7d3-322115367769">
        <Entertainment>
            <core:Content>
                <core:AssetRef providerID="provider.com"
assetID="TNTA0000000000000000"/>
            </core:Content>
            <PlayPositions>
                <!-- Signal ID associated with the in point -->
                <PlayPositionStart>
                    <SignalId>SIGNAL:DH1h4oVbEeG7r3ULSCQBmw==</SignalId>
                    <!-- UTC time of the in point -->
                    <UTC scale="1.0">2010-11-09T11:13:10.123Z</UTC>
                </PlayPositionStart>
                <!-- Signal ID associated with the next out point -->
                <PlayPositionEnd>
                    <SignalId>SIGNAL:N6t2mBpHTO6ZsZmzpsDVMg==</SignalId>
                    <!-- UTC time of the out point -->
                    <UTC scale="1.0">2010-11-09T11:41:19.456Z</UTC>
                </PlayPositionEnd>
            </PlayPositions>
        </Entertainment>
    </PlacementDecision>
    <PlacementDecision placementOpportunityRef="528573CD-1485-13A6-
56B778222E2B2936" id="443778a4-dab9-4849-ac2b-cad678651671">
        <Placement action="fill" id="fc914c63-9f29-4fc7-8096-6e85113fe90a">
            <core:Content>
                <core:AssetRef assetID="000CNAF2530" providerID="spotlight.com"/>
                <core:Duration>PT30S</core:Duration>
                <core:Tracking>09e6c50d-58d9-48cc-98d9-
1a6378c0db83</core:Tracking>
            </core:Content>
        </Placement>
        <Placement action="fill" id="6545d81a-2e46-4269-baa7-efcc04e5b60b">
            <core:Content>
                <core:AssetRef assetID="000CNAF2230" providerID="spotlight.com"/>
                <core:Duration>PT30S</core:Duration>
                <core:Tracking>13a51d1a-fd13-4a9e-8579-
1e4b5af13309</core:Tracking>
            </core:Content>
        </Placement>
        <Placement action="fill" id="2963d905-95a3-424c-8b5b-eb38e6fb2565">
            <core:Content>
                <core:AssetRef assetID="0000C182608" providerID="spotlight.com"/>
                <core:Duration>PT30S</core:Duration>
```

```
                    <core:Tracking>1f8bb476-d0c5-49b2-8c8c-
3e031d85cb15</core:Tracking>
                  </core:Content>
              </Placement>
          </PlacementDecision>
          <PlacementDecision placementOpportunityRef="528573CD-1485-13A6-
56B778222E2B2936" id="c011b500-8f2b-4d4d-91f8-1087b16599db">
              <Entertainment>
                  <core:Content>
                      <core:AssetRef providerID="provider.com"
assetID="TNTA0000000000000000"/>
                  </core:Content>
                  <PlayPositions>
                      <PlayPositionStart>
                          <!-- Signal ID associated with the in point -->
                          <SignalId>SIGNAL:E03WjIVbEeGajYILSCQBmw==</SignalId>
                          <!-- UTC time of the in point -->
                          <UTC scale="1.0">2010-11-09T11:43:19.456Z</UTC>
                      </PlayPositionStart>
                      <PlayPositionEnd>
                          <!-- End play at NOON GMT -->
                          <UTC scale="1">2010-07-20T12:00:00-00:00</UTC>
                      </PlayPositionEnd>
                  </PlayPositions>
              </Entertainment>
          </PlacementDecision>
  </PlacementResponse>
```

Example 58

14.40 PlayPositionsAvailBinding Element

The PlayPositionsAvailBinding element provides a capability to define a placement opportunity based on a heterogeous mix of start and/or end points and positional designators. The element contains information specific to defining, refining, or labeling a placement opportunity.

14.40.1 PlayPositionsAvailBinding Element Schema

Figure 82 illustrates the PlayPositionsAvailBinding element's schema.

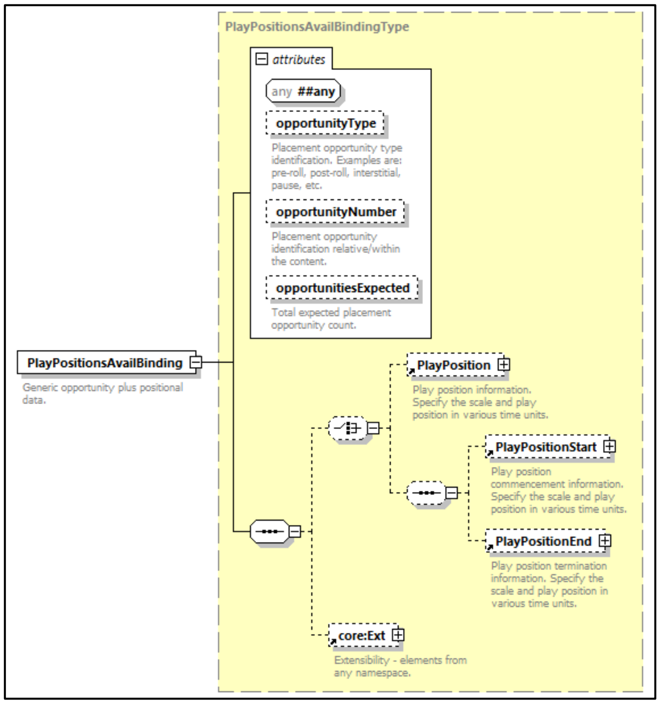

PlayPositionsAvailBindingType
attributes
any ##any
opportunityType
Placement opportunity type identification. Examples are: pre-roll, post-roll, interstitial, pause, etc.
opportunityNumber
Placement opportunity identification relative/within the content.
opportunitiesExpected
Total expected placement opportunity count.
PlayPositionsAvailBinding
Generic opportunity plus positional data.
PlayPosition
Play position information. Specify the scale and play position in various time units.
PlayPositionStart
Play position commencement information. Specify the scale and play position in various time units.
PlayPositionEnd
Play position termination information. Specify the scale and play position in various time units.
core:Ext
Extensibility - elements from any namespace.

Figure 82: PlayPositionsAvailBinding Element Schema

14.40.1.1 Semantic Definitions for the PlayPositionsAvailBinding Element

@opportunityType [Optional, placementOpportunityTypeAttrType]—The placement opportunity type. See Section 13.1.8 for additional information. For example, in classic linear the typical value for this attribute is ‘interstitial.’

@opportunityNumber [Optional, placementOpportunityNumberAttrType]—The placement opportunity identification value relative to the entertainment content. See Section 13.1.7 for additional information.

@opportunitiesExpected [Optional, placementOpportunitiesExpectedAttrType]—The total number of planned placement opportunities relative to the entertainment content. See Section 13.1.6 for additional information.

@##any [Optional]—Additional attributes from any namespace.

PlayPosition [Optional Choice]—This element identifies a position within content that typically is not a start or end location. See 0 for additional information.

PlayPositionStart [Optional]—Optional element to identify the start position. See Section 0 for additional information.

PlayPositionEnd [Optional]—Optional element to identify the end position. See Section 0 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.40.2 PlayPositionsAvailBinding Element Examples

```
<PlayPositionsAvailBinding>
   <PlayPositionStart>
     <!-- Signal ID associated with the start point. -->
     <SignalId>SIGNAL:E03WjIVbEeGajYILSCQBmw==</SignalId>
     <!-- UTC time of the start point. -->
     <UTC scale="1.0">2010-11-09T11:43:19.456Z</UTC>
   </PlayPositionStart>
   <PlayPositionEnd>
     <!-- End position at NOON GMT. -->
     <UTC scale="1">2010-07-20T12:00:00-00:00</UTC>
   </PlayPositionEnd>
</PlayPositionsAvailBinding>

<PlayPositionsAvailBinding>
   <PlayPositionStart scale="1">
     <!-- NPT of BOS for the start point -->
     <NPTOffset>BOS</NPTOffset>
   </PlayPositionStart>
</PlayPositionsAvailBinding>
```

14.41 ProductType Element

The ProductType element provides product identification.

14.41.1 ProductType Element Schema

Figure 83 illustrates the ProductType element’s schema.

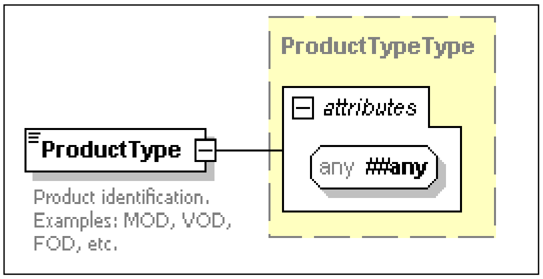


Figure 83: ProductType Element Schema

14.41.1.1 Semantic Definitions for the ProductType Element

@##any [Optional]—Additional attributes from any namespace.

The ProductType element's value is of type core:nonEmptyStringType and ***shall not*** be empty.

14.41.2 ProductType Element Examples

```
<ProductType>MOD</ProductType>
<ProductType>FOD</ProductType>
```

14.42 ProviderAvailID Element

The ProviderAvailID element allows an authorization identifier to be associated with the placement opportunity. Typically, this value is supplied via the SCTE 35 avail_descriptor() provider_avail_id field. See [SCTE35] for additional information.

14.42.1 ProviderAvailID Element Schema

Figure 84 illustrates the ProviderAvailID element's schema.

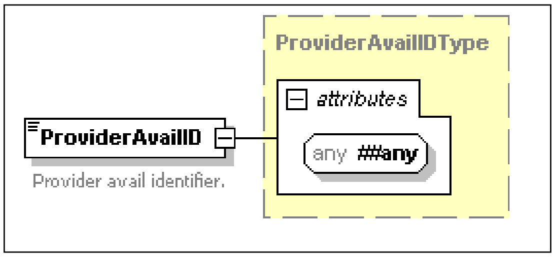


Figure 84: ProviderAvailID Element Schema

14.42.1.1 Semantic Definitions for the ProviderAvailID Element

@##any [Optional]—Additional attributes from any namespace.

The ProviderAvailID element's value is of type xsd:nonNegativeInteger.

14.42.2 ProviderAvailID Element Examples

```
<ProviderAvailID>893</ProviderAvailID>
```

14.43 Scope Element

The Scope element supplies a range fulfillment advisory.

14.43.1 Scope Element Schema

Figure 85 illustrates the Scope element's schema.

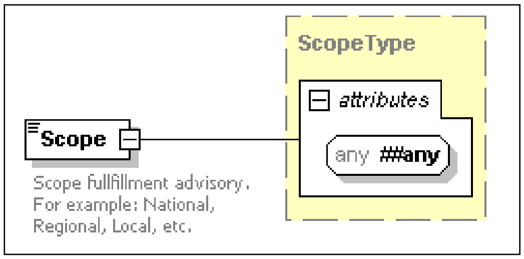

ScopeType
attributes
Scope
any ##any
Scope fullfillment advisory. For example: National, Regional, Local, etc.

Figure 85: Scope Element Schema

14.43.1.1 Semantic Definitions for the Scope Element

@##any [Optional]—Additional attributes from any namespace.

The Scope element's value is of type core:nonEmptyStringType and ***shall not*** be empty. Table 14 lists the defined values which *may* be extended by private agreement outside the scope of this specification. The value ***shall*** exactly match the appearance in Table 14.

Value	Description
National	Not regional or local.
Regional	Not national or local.
Local	Not national or regional.
Provider	Not distributor. See [SCTE35].
Distributor	Not provider. See [SCTE35].
…	Defined outside the scope of this specification and the value ***shall*** start with the string "private:".

Table 14: Scope Element Values

14.43.2 Scope Element Examples

```
<Scope>National</Scope>
<Scope>Regional</Scope>
<Scope>Local</Scope>
<Scope>Provider</Scope>
<Scope>Distributor</Scope>
<Scope>private:Zoned</Scope>
```

14.44 Service Element

The Service element is a placement service identification component included by an ADS when identifying itself as a placement decision owner. The element is a named information collection and one or more Service elements are included in the ServiceDescription element as part of the ADSRegistrationRequest message. Inclusion facilitates placement service matching for determining the correct ADS in order to send the placement operations messages. The element *may* also be used to provide additional information when refining a service description match (via inclusion in the PlacementRequest message).

14.44.1 Service Element Schema

Figure 86 illustrates the Service element’s schema.

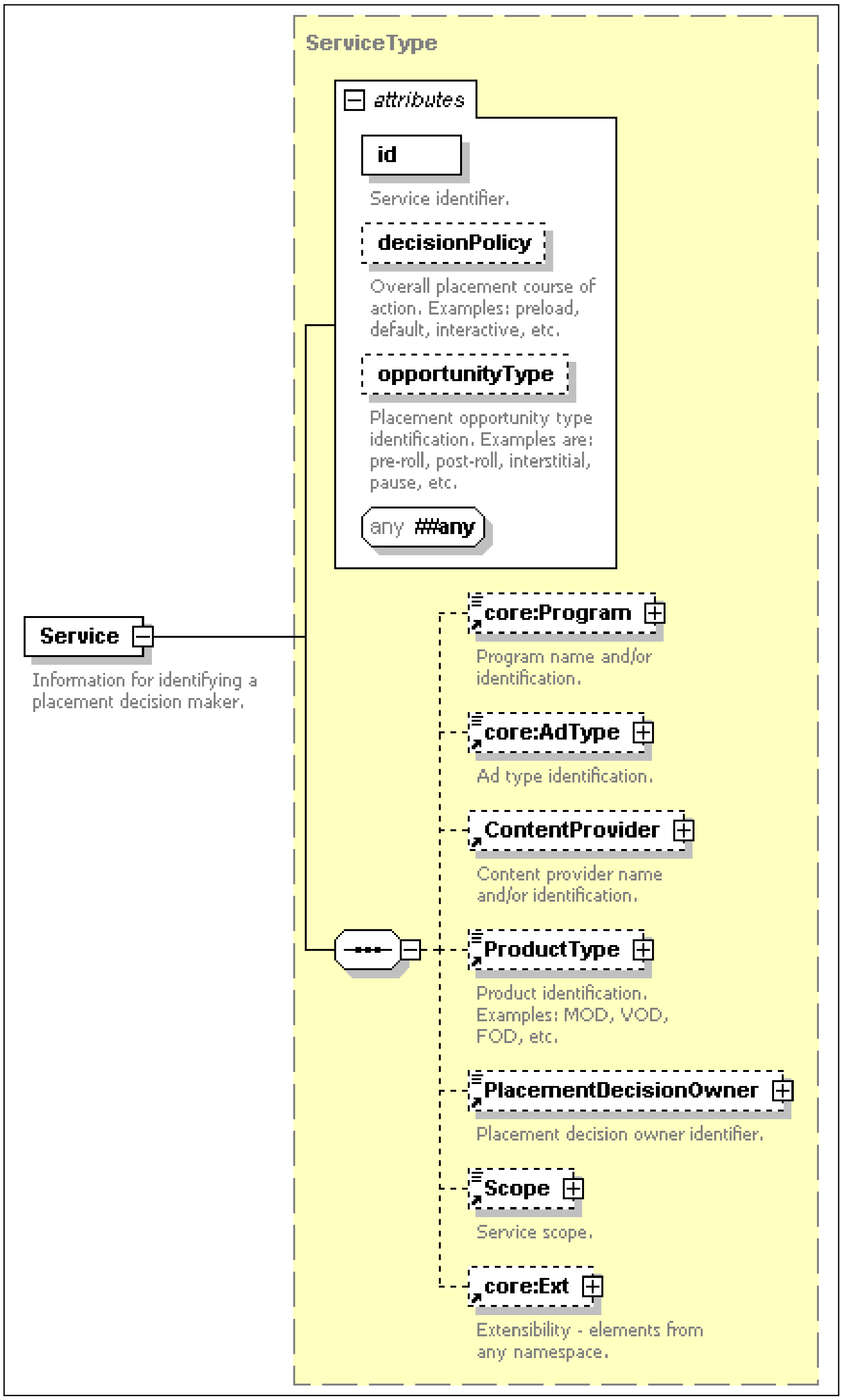

ServiceType
attributes
id
Service identifier.
decisionPolicy
Overall placement course of action. Examples: preload, default, interactive, etc.
opportunityType
Placement opportunity type identification. Examples are: pre-roll, post-roll, interstitial, pause, etc.
any ##any
Service
Information for identifying a placement decision maker.
core:Program
Program name and/or identification.
core:AdType
Ad type identification.
ContentProvider
Content provider name and/or identification.
ProductType
Product identification. Examples: MOD, VOD, FOD, etc.
PlacementDecisionOwner
Placement decision owner identifier.
Scope
Service scope.
core:Ext
Extensibility - elements from any namespace.

Figure 86: Service Element Schema

14.44.1.1 Semantic Definitions for the Service Element

@id [Required, core:idAttrType]—Service identifier specified by the ADS. The identifier is service channel unique (i.e., scoped by the message's @identity attribute) when included in an

ADSRegistrationRequest message and element unique when included in a PlacementRequest or ListADMServicesResponse message. See [SCTE130-2] for additional information.

@decisionPolicy [Optional, core:nonEmptyStringType]—Table 15 lists the defined values, which *may* be extended by private agreement outside the scope of this specification. The value ***shall*** exactly match the appearance in Table 15 and the value ***shall not*** be empty.

Value	Description
default	The ADS placement decisions ***shall*** be used only under extraordinary circumstances and these decisions are not expected to be part of the normal ADM placement execution. During normal operation, an ADS should not expect these decisions to be used. The ADM ***shall*** signal if updates are allowed as part of the PlacementRequest message requesting the default placement decision set.
interactive	The ADM ***shall*** perform a placement message exchange requesting the placement decision set. The response decision set ***shall*** only be applied a single time. The ADM ***shall*** signal if updates are allowed as part of the PlacementRequest message.
preload	Placement decisions ***shall*** be requested in advance, typically using a single PlacementRequest message, and these decisions ***shall*** be applied as often as applicable thereafter. The ADM ***shall not*** perform a placement message exchange before applying these decisions. The ADM ***shall*** signal if updates are allowed as part of the PlacementRequest message when acquiring the preload placement decision set.
…	User defined and outside the scope of this specification. The string ***shall*** be prefixed with the text "private:".

Table 15: Service/@decisionPolicy Attribute Values

@opportunityType [Optional, placementOpportunityTypeAttrType]—Placement opportunity type identifier to match. See Section 13.1.8 for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:Program [Optional]—The Program identifier value to match or that was matched. See [SCTE130-2] for additional information.

core:AdType [Optional]—The ad type identifier value to match or that was matched. See [SCTE130-2] for additional information.

ContentProvider [Optional]—The content source identifier value to match or that was matched. See Section 14.5 or additional information.

ProductType [Optional]—The product type identifier value to match or that was matched. See Section 0 for additional information.

PlacementDecisionOwner [Optional]—The placement decision owner value to match or that was matched. See Section 14.32 for additional information.

Scope [Optional]—The service scope value to match or that was matched. See Section 14.43 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.44.2 Service Element Examples

Example 59 is a minimal Service element that *may* be interpreted as "all placement services". Note that the string "All Services" is an arbitrary name which is not a standard mandate or standard preferred identifier.

```
<Service id="All Services"/>
```

Example 59

Example 60 illustrates a placement service being identified in part using the @uniqueProgramID value.

```
<Service id="Service1">
  <core:Program uniqueProgramID="345"/>
</Service>
<Service id="Service1" decisionPolicy="interactive">
  <core:Program uniqueProgramID="345"/>
</Service>
```

Example 60

Example 61 illustrates a placement service being identified in part as ad type matching the string "Telescoping" and being owned by identifier "Walt."

```
<Service id="Service2" decisionPolicy="default">
  <core:AdType>Telescoping</core:AdType>
  <PlacementDecisionOwner>Walt</PlacementDecisionOwner>
</Service>
```

Example 61

Example 62 is a placement service for the pause opportunity type.

```
<Service id="Service5" opportunityType="pause" decisionPolicy="interactive"/>
```

Example 62

Example 63 is placement service with a fine grained placement service matching criteria.

```
<Service id="Very Precise" opportunityType="interstitial" decisionPolicy="interactive">
  <core:Program>Ugly Betty</core:Program>
  <core:AdType>Overlay</core:AdType>
  <ContentProvider providerID="ABC"/>
  <ProductType>nPVR</ProductType>
  <PlacementDecisionOwner>Corey's Ad Shop</PlacementDecisionOwner>
  <Scope>Local</Scope>
</Service>
```

Example 63

14.45 ServiceDescription Element

The ServiceDescription element is a placement service identification component included by an ADS when identifying itself as a placement decision producer. The element is a container for describing a named placement service via the SystemContext and Service elements. Inclusion facilitates placement service matching for determining the correct ADS in order to send the placement operations message exchanges. The element is a cornerstone of the registration-established placement service.

14.45.1 ServiceDescription Element Schema

Figure 87 illustrates the ServiceDescription element’s schema.

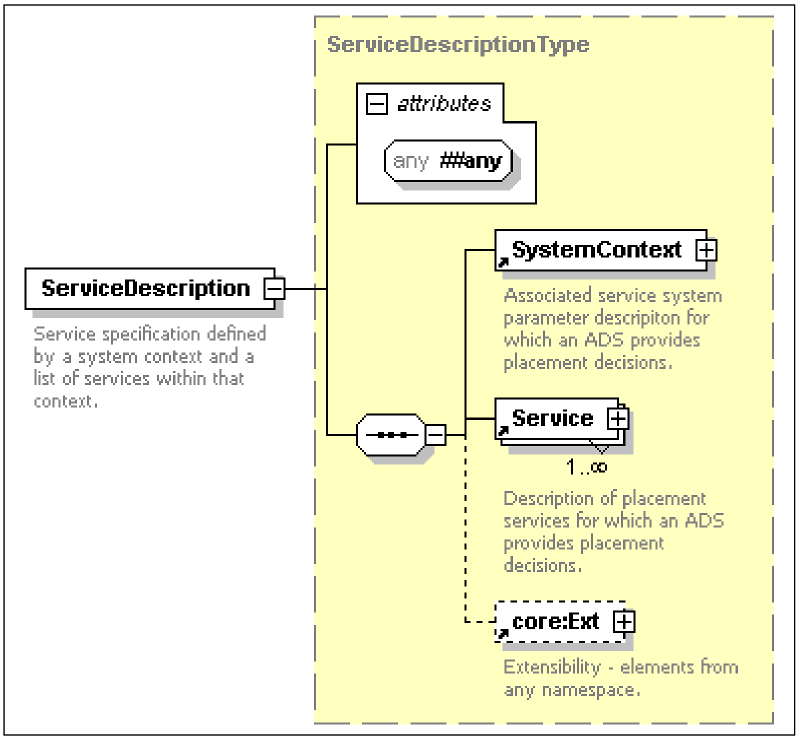

ServiceDescriptionType
attributes
any ##any
ServiceDescription
Service specification defined by a system context and a list of services within that context.
SystemContext
Associated service system parameter descripiton for which an ADS provides placement decisions.
Service
1..∞
Description of placement services for which an ADS provides placement decisions.
core:Ext
Extensibility - elements from any namespace.

Figure 87: ServiceDescription Element Schema

14.45.1.1 Semantic Definitions for the ServiceDescription Element

@##any [Optional]—Additional attributes from any namespace.

SystemContext [Required]—System parameters identifying the placement decision service. The parameters apply to all Service elements included in the ServiceDescription. See Section 14.52 for additional information.

Service [Required]—One or more elements each identifying a service relative to the SystemContext parameters. See Section 14.44 for additional information. Services requiring a unique SystemContext ***shall*** be described using multiple ServiceDescription elements.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.45.2 ServiceDescription Element Examples

Example 64 illustrates the placement service definition which *may* be interpreted as all placement services. Note that the string “All Services” is an arbitrary name which is not a standard mandate or standard preferred identifier.

```
<ServiceDescription>
  <SystemContext/>
  <Service id="All Services"/>
</ServiceDescription>
```

Example 64

Example 65 illustrates a placement service identified primarily using the SystemContext element.

```
<ServiceDescription>
  <SystemContext>
    <Zone>ESPNWest</Zone>
  </SystemContext>
  <Service id="EspnWestZone"/>
</ServiceDescription>
```

Example 65

Example 66 illustrates a placement service identified primarily using the Service element.

```
<ServiceDescription>
  <SystemContext/>
  <Service id="AvailTypeOnly" opportunityType="pause"/>
</ServiceDescription>
```

Example 66

Example 67 illustrates a placement service identified using a combination of the SystemContext and Service elements.

```
<ServiceDescription>
  <SystemContext>
    <Network>ESPN</Network>
    <Channel>ESPN 8</Channel>
  </SystemContext>
  <Service id="The Ocho">
    <core:AdType>Telescoping</core:AdType>
    <Scope>Provider</Scope>
  </Service>
</ServiceDescription>
```

Example 67

14.46 ServiceGroup Element

The ServiceGroup element provides collection association identification. Traditionally, this organization identifies physical grouping aspects based on the fulfillment delivery systems. However, the group organization is not limited to this dimension and *may* be significantly more arbitrary. Group organization is outside of the scope of this specification.

14.46.1 ServiceGroup Element Schema

Figure 88 illustrates the ServiceGroup element's schema.

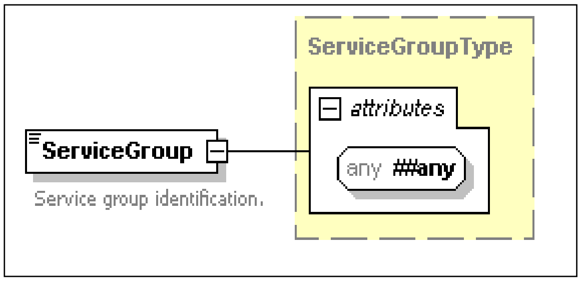


Figure 88: ServiceGroup Element Schema

14.46.1.1 Semantic Definitions for the ServiceGroup Element

@##any [Optional]—Additional attributes from any namespace.

The ServiceGroup element's value is of type core:nonEmptyStringType and ***shall not*** be empty.

14.46.2 ServiceGroup Element Examples

```
<ServiceGroup>IceHockeyLovers</ServiceGroup>
<ServiceGroup>83983</ServiceGroup>
```

14.47 Session Element

The Session element provides session identification typically associated, but not limited to, a resource set in the fulfillment system.

14.47.1 Session Element Schema

Figure 89 illustrates the Session element's schema.

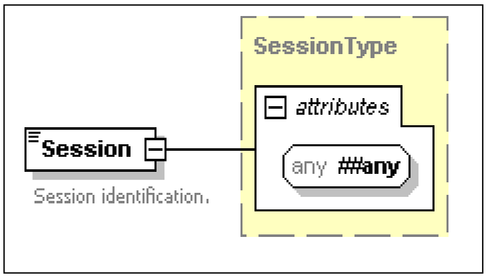


Figure 89: Session Element Schema

14.47.1.1 Semantic Definitions for the Session Element

@##any [Optional]—Additional attributes from any namespace.

The Session element's value is a core:nonEmptyStringType and ***shall not*** be empty.

14.47.2 Session Element Examples

```
<Session>SharksWinIn2008</Session>
<Session>GuysSession</Session>
<Session>38793893</Session>
<Session>8FA2C7BD-2751-4D18-A2C6-372D8CE5F100</Session>
```

14.48 SessionEvent Element

The SessionEvent element provides reporting details regarding a single session event which the ADM system considers of interest. Each event should only specify a single occurrence of the fact of interest.

14.48.1 SessionEvent Element Schema

Figure 90 illustrates the SessionEvent element's schema.

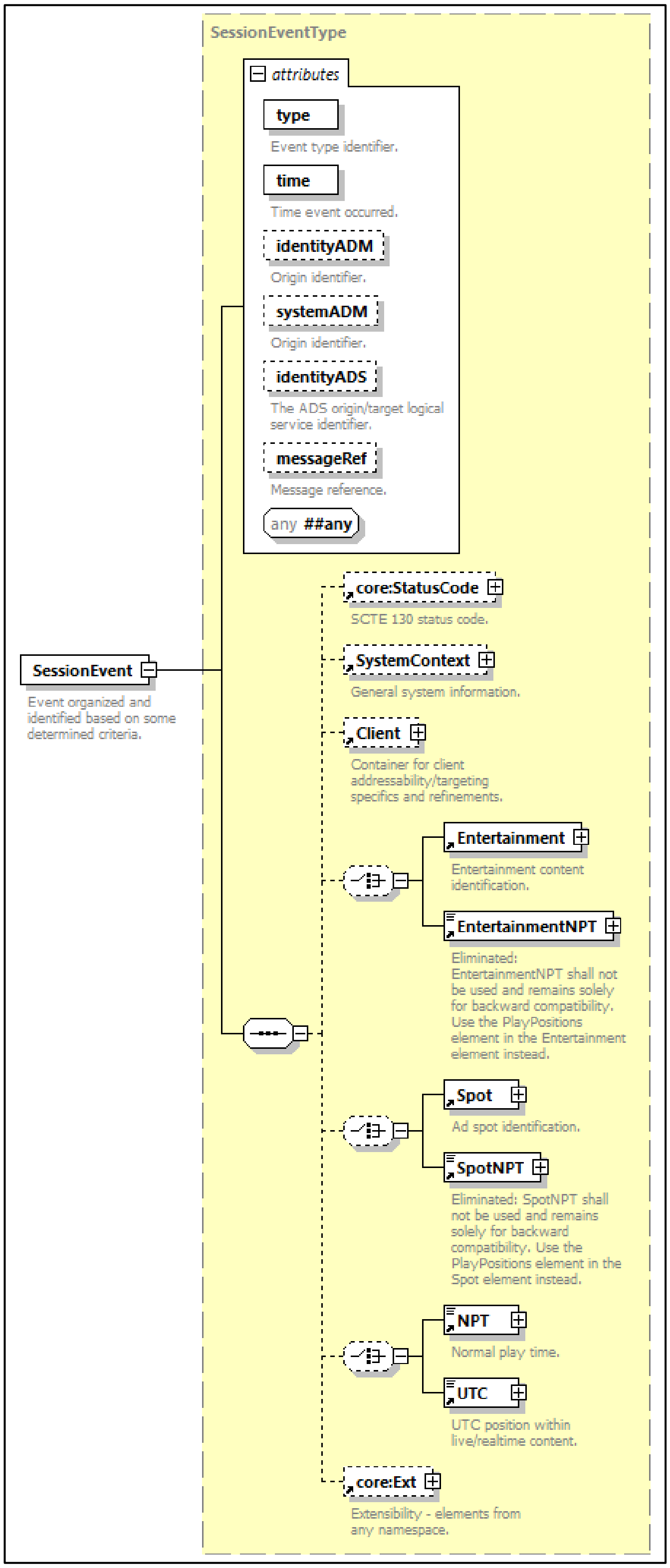

SessionEventType
attributes
type
Event type identifier.
time
Time event occurred.
identityADM
Origin identifier.
systemADM
Origin identifier.
identityADS
The ADS origin/target logical service identifier.
messageRef
Message reference.
any ##any
SessionEvent
Event organized and identified based on some determined criteria.
core:StatusCode
SCTE 130 status code.
SystemContext
General system information.
Client
Container for client addressability/targeting specifics and refinements.
Entertainment
Entertainment content identification.
EntertainmentNPT
Eliminated: EntertainmentNPT shall not be used and remains solely for backward compatibility. Use the PlayPositions element in the Entertainment element instead.
Spot
Ad spot identification.
SpotNPT
Eliminated: SpotNPT shall not be used and remains solely for backward compatibility. Use the PlayPositions element in the Spot element instead.
NPT
Normal play time.
UTC
UTC position within live/realtime content.
core:Ext
Extensibility - elements from any namespace.

Figure 90: SessionEvent Element Schema

14.48.1.1 Semantic Definitions for the SessionEvent Element

@type [Required, eventTypeAttrType]—Session event type identifier as defined in Section 13.2.1. Table 16 lists the defined values for this element that *may* be extended by private agreement outside the scope of this specification. The attribute's value ***shall*** appear exactly as shown in Table 16.

Value	Description
status	The event contains session related status information. The event should contain a core:StatusCode element and zero or more Note elements.
startSegment	The event indicates an entry into an entertainment content binding which is not specifically referenced by a Placement element. Placement element content indications ***shall*** use a PlacementStatusEvent element. Either this event element or a higher-level scoped element should contain an Entertainment element.
endSegment	The event indicates an exit from an entertainment content binding which is not specifically referenced by a Placement element. Placement element content indications ***shall*** use a PlacementStatusEvent element. Either this event element or a higher-level scoped element should contain an Entertainment element.
startSession	The event indicates the session start. Either this event element or a higher-level scoped element should contain a Session element.
endSession	The event indicates the session end. Either this event element or a higher-level scoped element should contain a Session element.
resumeSession	The event indicates a session resume. Either this event element or a higher-level scoped element should contain a Session element. The Session element should match a Session element from a previous startSession SessionEvent.
…	User defined and outside the scope of this specification. The value ***shall*** be preceded by the string "private:".

Table 16: SessionEvent @type Attribute Values

startSegment and endSegment ***shall*** be used in the same manner as startPlacement and endPlacement as described in Section 12.3 with the difference being StartSegment and endSegment refer to Entertainment element content bindings not Placement element content bindings.

Table 17 identifies which PlayPosition elements *may* be used with each of the @type attribute values.

	Element		
Value	**PlayPosition**	**PlayPositionStart**	**PlayPositionEnd**
status	Yes	No	No
startSegment	No	Yes	No
endSegment	No	No	Yes
startSession	No	Yes	No
endSession	No	No	Yes
resumeSession	Yes	No	No

Table 17: SessionEvent @type Attribute Position Element Usage

@time [Required, core:dateTimeTimezoneType]—The event occurrence date and time. See Section 13.2.1 for additional information.

@identityADM [Optional, core:identityAttrType]—The origin ADM logical service identifier. See Section 13.2.1 for additional information.

@systemADM [Optional, core:systemAttrType]—The originating ADM system identification. See Section 13.2.1 for additional information.

@identityADS [Optional, core:identityAttrType]—The ADS origin or target logical service identifier. See Section 13.2.1 for additional information.

@messageRef [Optional, core:messageRefAttrType]—The message associated with this event. See Section 13.2.1 for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:StatusCode [Optional]—An applicable status code as defined in Appendix A. See Section 13.2.1 for additional information.

SystemContext [Optional]—Parameters describing the event’s system context. See Section 13.2.1 for additional information.

Client [Optional]—Addressability or targeting details. See Section 13.2.1 for additional information.

Entertainment [Optional Choice]—Entertainment content asset identification. See Section 13.2.1 for additional information.

EntertainmentNPT [Optional Choice]—This element ***shall not*** be used and remains for backward compatibility only. In messages having a @version attribute of "1.3" or prior, this element *may* appear. It ***shall not*** be used in messages having an @version attribute of "1.4" or later. An entertainment specific NPT value is provided using the NPTOffset element contained with the Entertainment element.

Spot [Optional Choice]—Ad content asset identification. See Section 13.2.1 for additional information.

SpotNPT [Optional Choice]—This element ***shall not*** be used and remains for backward compatibility only. In messages having a @version attribute of "1.3" or prior, this element *may* appear. It ***shall not*** be used in messages having an @version attribute of "1.4" or later. A spot specific NPT value is provided using the NPTOffset element contained with the Spot element.

NPT [Optional Choice]—User perceived NPT. See Section 13.2.1 for additional information.

UTC [Optional Choice]—A UTC value referenced to a stream absolute position within live/realtime content. See Section 14.56 for additional information. The UTC is typically the clock the viewer associates with a live/realtime program or entertainment asset including ads that are delivered as part of a live/realtime asset.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.48.2 SessionEvent Element Examples

```
<SessionEvent type="startSession" time="2007-01-05T12:01:01.0Z">
  <SystemContext>
    <Session>ExampleSessionID 399383</Session>
  </SystemContext>
</SessionEvent>

<SessionEvent type="startSession" time="2007-01-05T12:01:01.0Z"/>

<SessionEvent type="endSegment" time="2007-01-05T12:03:33.0Z">
  <Entertainment>
    <core:Content>
      <core:SegmentationUpid type="2">8583</core:SegmentationUpid>
    </core:Content>
  </Entertainment>
</SessionEvent>
```

14.49 Spot Element

The Spot element identifies the ad content referenced in a named event element. The element contains a core:Content element identifying the ad content asset and it *may* optionally contain a correlated NPT element.

The element ***shall*** also include any core:Tracking elements provided in a Placement element. See Section 12.3 for additional information.

14.49.1 Spot Element Schema

Figure 91 illustrates the Spot element's schema.

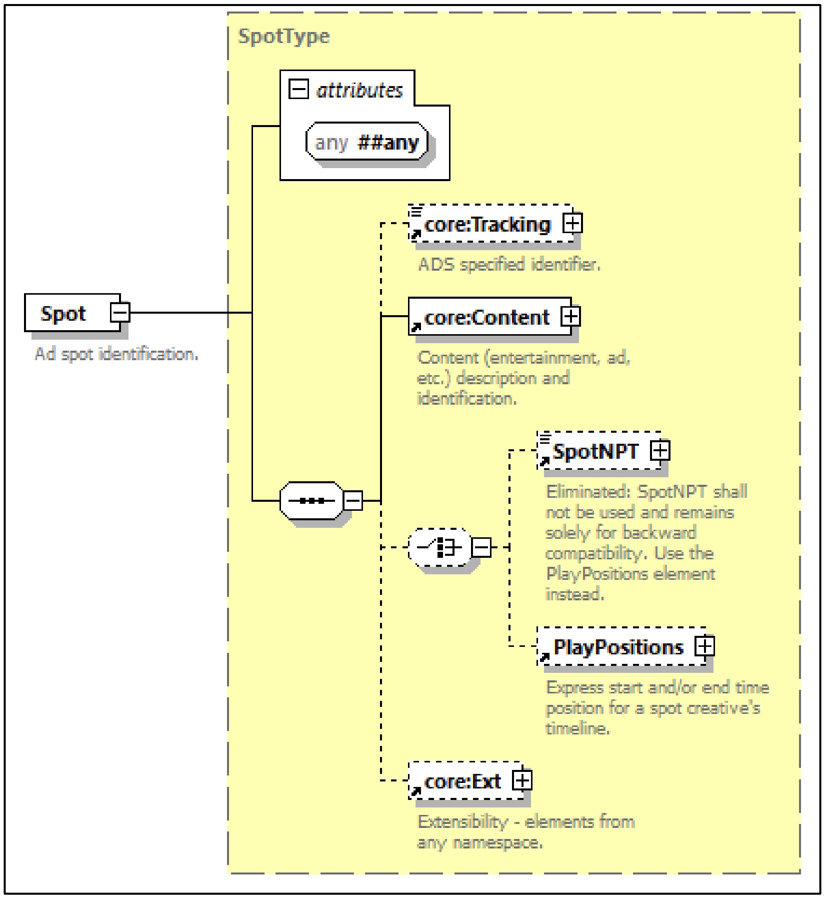

SpotType
attributes
any ##any
core:Tracking
ADS specified identifier.
Spot
Ad spot identification.
core:Content
Content (entertainment, ad, etc.) description and identification.
SpotNPT
Eliminated: SpotNPT shall not be used and remains solely for backward compatibility. Use the PlayPositions element instead.
PlayPositions
Express start and/or end time position for a spot creative's timeline.
core:Ext
Extensibility - elements from any namespace.

Figure 91: Spot Element Schema

14.49.1.1 Semantic Definitions for the Spot Element

@##any [Optional]—Additional attributes from any namespace.

core:Tracking [Optional]—An ADS assigned core:Tracking element. See [SCTE130-2] for additional information. This element corresponds to a PlacementResponse or PlacementUpdateNotification message’s Placement element containing a core:Tracking element.

core:Content [Required]—Placement content identification. See [SCTE130-2] for additional information. If the ADS returned a Tracking element in the Placement element’s corresponding core:Content element, the Tracking element ***shall*** be repeated within this element.

SpotNPT [Optional Choice]—This element ***shall not*** be used and remains for backward compatibility only. In messages having a @version attribute of "1.3" or prior, this element *may* appear. It ***shall not*** be used in messages having an @version attribute of "1.4" or later. A spot specific NPT value is provided using the NPTOffset element contained with the Spot element.

PlayPositions [Optional Choice]—PlayPositions relative to the paired core:Content element. See Section 0 for additional information.

The Spot element *may* contain a PlayPositions element. This usage facilitates describing descrete points or a range within the ad creative content using a heterogenous mix of positional values.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.49.2 Spot Element Examples

```
<Spot>
  <core:Tracking>999ECCED-979B-421E-8A6E-E2192B11515D</core:Tracking>
  <core:Content>
    <core:AssetRef providerID="adco.com"
                   assetID="ABCD0123456789012345"/>
    <core:Tracking>5B8ECCED-979B-421E-8A6E-E2192B11516C</core:Tracking>
  </core:Content>
  <PlayPositions>
    <PlayPosition>
      <NPTOffset>26</NPTOffset>
    </PlayPosition>
  </PlayPositions>
</Spot>
```

14.50 SpotNPT Element (Eliminated)

The SpotNPT element ***shall not*** be used and remains for backward compatibility only. In messages having a @version attribute of "1.3" or prior, this element *may* appear. It ***shall not*** be used in messages having an @version attribute of "1.4" or later. See the PlayPositions element in Section 0 for additional information on how to carry a Spot specific NPT value.

14.51 SpotScopedEvents Element

The SpotScopedEvents element reduces named event element size and data replication. The Spot element ***shall*** be declared once and subsequent named event elements contained within the Events elements imply reference to this Spot element through this scoping.

14.51.1 SpotScopedEvents Element Schema

Figure 92 illustrates the SpotScopedEvents element's schema

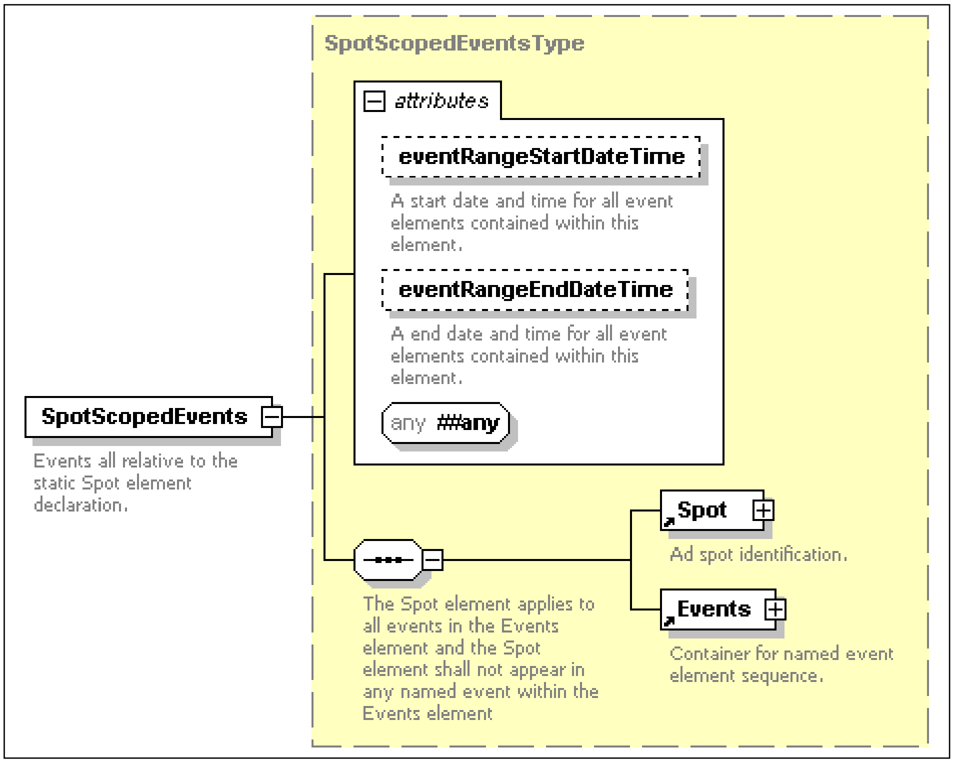

SpotScopedEventsType
attributes
eventRangeStartDateTime
A start date and time for all event elements contained within this element.
eventRangeEndDateTime
A end date and time for all event elements contained within this element.
any ##any
SpotScopedEvents
Events all relative to the static Spot element declaration.
Spot
Ad spot identification.
Events
Container for named event element sequence.
The Spot element applies to all events in the Events element and the Spot element shall not appear in any named event within the Events element

Figure 92: SpotScopedEvents Element Schema

14.51.1.1 Semantic Definitions for the SpotScopedEvents Element

@eventRangeStartDateTime [Optional, eventRangeStartDateTimeAttrType]—An initial date and time relative to the named event elements contained within the Events element. See Section 13.1.2 for additional details.

@eventRangeEndDateTime [Optional, eventRangeEndDateTimeAttrType]—An ending date and time relative to the named event elements contained within the Events element. See Section 13.1.1 for additional details.

@##any [Optional]—Additional attributes from any namespace.

Spot [Required]—Spot content identification. See Section 14.49 for additional details.

Events [Required]—A container for named event elements. See Section 0 for additional information.

The Spot element *may* contain a NPT element and each named event element contained within the Events element *may* also contain an NPT element but no named event element ***shall*** contain a Spot element. This usage facilitates describing a range of the spot content while each named event references a specific timepoint within the range.

14.51.2 SpotScopedEvents Element Examples

```
<SpotScopedEvents>
 <Spot>
  <core:Content>
   <core:Tracking>5B8ECCED-979B-421E-8A6E-E2192B11516C</core:Tracking>
  </core:Content>
 </Spot>
 <Events>
  <PlacementStatusEvent time="2007-01-31T02:00:00Z"
             type="startPlacement">
   <Spot>
    <core:Content/>
    <PlayPositions scale="1.0">
     <PlayPositionStart>
      <NPTOffset>0</NPTOffset>
     </PlayPositionStart>
    </PlayPositions>
   </Spot>
  </PlacementStatusEvent>
  <PlacementStatusEvent time="2007-01-31T02:00:30Z"
             type="endPlacement">
   <Spot>
    <core:Content/>
    <PlayPositions scale="1.0">
     <PlayPositionEnd>
      <NPTOffset>30</NPTOffset>
     </PlayPositionEnd>
    </PlayPositions>
   </Spot>
  </PlacementStatusEvent>
 </Events>
</SpotScopedEvents>
```

14.52 SystemContext Element

The SystemContext element is a container for the system and environment specific elements.

14.52.1 SystemContext Element Schema

Figure 93 illustrates the SystemContext element’s schema.

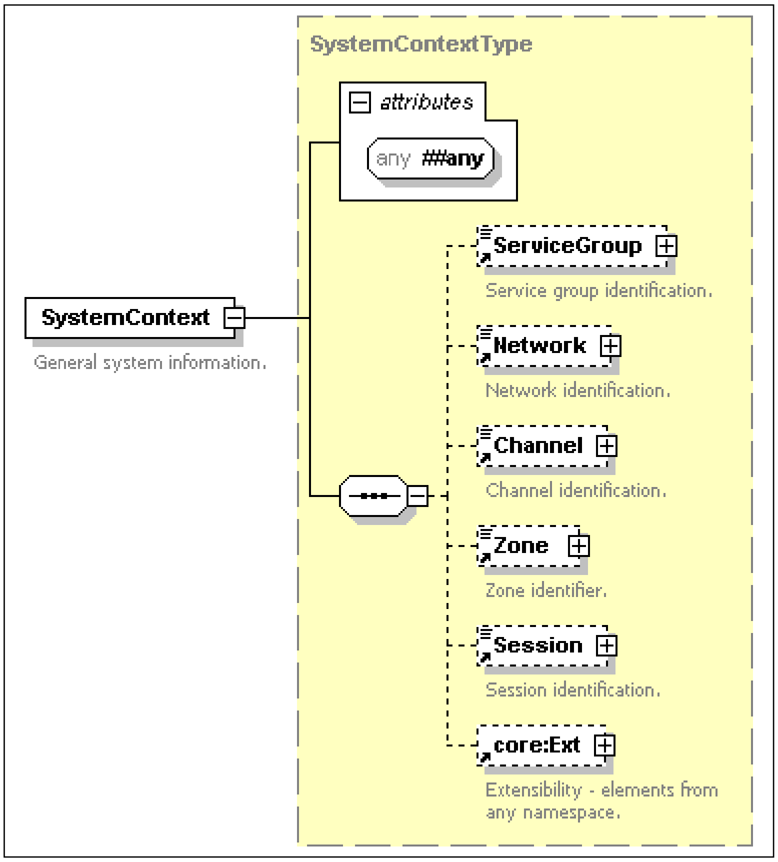


Figure 93: SystemContext Element Schema

14.52.1.1 Semantic Definitions for the SystemContext Element

@##any [Optional]—Additional attributes from any namespace.

ServiceGroup [Optional]—Service group identification. See Section 14.46 for additional information.

Network [Optional]—Network identification. See Section 14.20 for additional information.

Channel [Optional]—Channel identification. See Section 14.3 for additional information.

Zone [Optional]—Zone identification. See Section 14.59 for additional information.

Session [Optional]—Session identification. See Section 14.47 for additional information.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

The SystemContext element *may* be empty (typically as part of a ServiceDescription element).

14.52.2 SystemContext Element Examples

```
<SystemContext/>

<SystemContext>
  <ServiceGroup>Den</ServiceGroup>
  <Network>SharksNet</Network>
  <Zone>5</Zone>
  <Session>Session123</Session>
</SystemContext>
```

14.53 SystemEvent Element

The SystemEvent element provides reporting details regarding a single happening which the ADM system considers of interest. Each event should only specify a single occurrence of the fact of interest. The SystemEvent facilitates a general event reporting element which *may* be used when the other defined event elements (PlacementStatusEvent, SessionEvent, or ViewerEvent) *may* be inappropriate.

14.53.1 SystemEvent Element Schema

Figure 94 illustrates the SystemEvent element's schema.

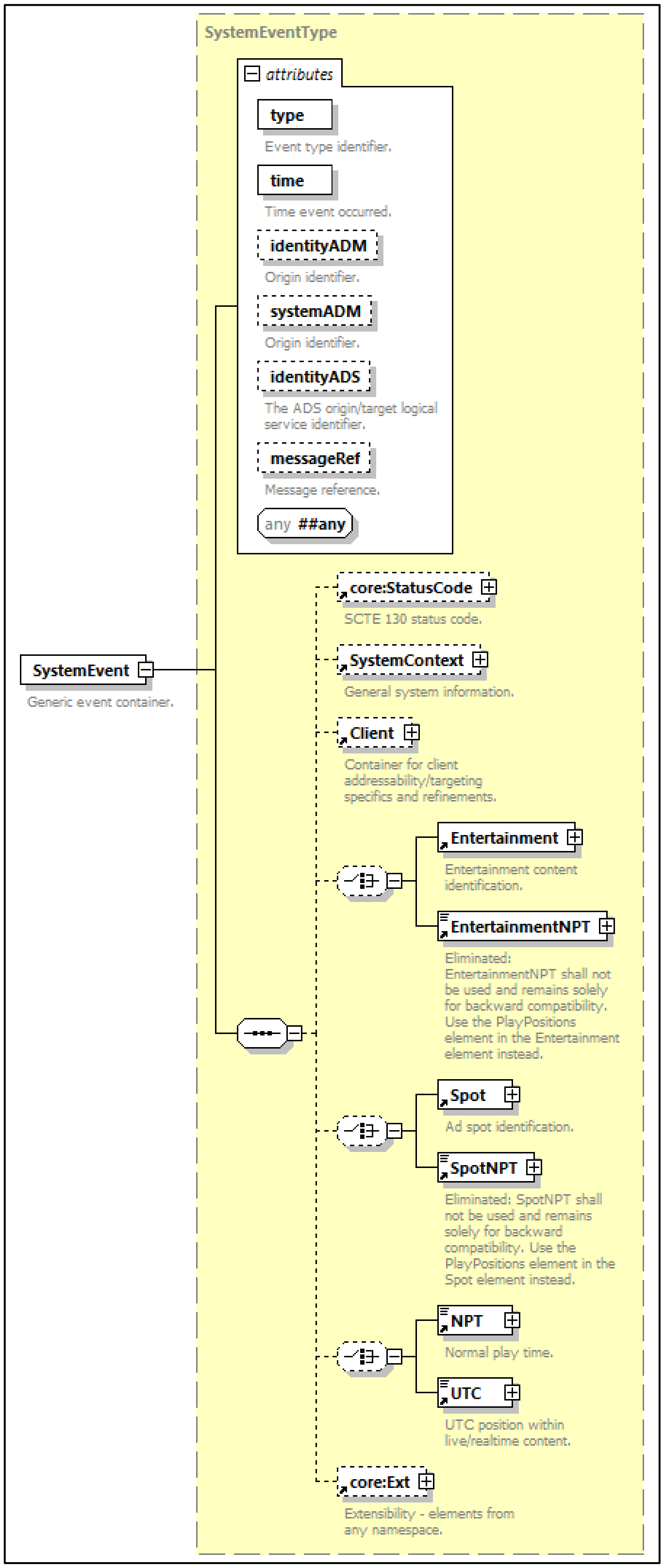

SystemEventType
attributes
type
Event type identifier.
time
Time event occurred.
identityADM
Origin identifier.
systemADM
Origin identifier.
identityADS
The ADS origin/target logical service identifier.
messageRef
Message reference.
any ##any
core:StatusCode
SCTE 130 status code.
SystemContext
General system information.
Client
Container for client addressability/targeting specifics and refinements.
Entertainment
Entertainment content identification.
EntertainmentNPT
Eliminated: EntertainmentNPT shall not be used and remains solely for backward compatibility. Use the PlayPositions element in the Entertainment element instead.
Spot
Ad spot identification.
SpotNPT
Eliminated: SpotNPT shall not be used and remains solely for backward compatibility. Use the PlayPositions element in the Spot element instead.
NPT
Normal play time.
UTC
UTC position within live/realtime content.
core:Ext
Extensibility - elements from any namespace.
SystemEvent
Generic event container.

Figure 94: SystemEvent Element Schema

14.53.1.1 Semantic Definitions for the SystemEvent Element

@type [Required, eventTypeAttrType]—The event identifier. See Section 13.2.1 for additional information. Table 18 lists the defined values for this element which *may* be extended by private agreement outside the scope of this specification. The attribute's value ***shall*** appear exactly as shown in Table 18.

Value	Description
status	The event contains related status information. The event should contain a core:StatusCode element and zero or more Note elements.
…	User defined and outside the scope of this specification. The value ***shall*** be preceded by the string "private:".

Table 18: SystemEvent @type Attribute Values

Table 19 identifies which PlayPosition elements *may* be used with each of the @type attribute values.

	Element		
Value	**PlayPosition**	**PlayPositionStart**	**PlayPositionEnd**
status	Yes	No	No

Table 19: SystemEvent @type Attribute Position Element Usage

@time [Required, core:dateTimeTimezoneType]—The event occurrence date and time. See Section 13.2.1 for additional information.

@identityADM [Optional, core:identityAttrType]—The origin ADM logical service identifier. See Section 13.2.1 for additional information.

@systemADM [Optional, core:systemAttrType]—The originating ADM system identification. See Section 13.2.1 for additional information.

@identityADS [Optional, core:identityAttrType]—The ADS origin or target logical service identifier. See Section 13.2.1 for additional information.

@messageRef [Optional, core:messageRefAttrType]—The message associated with this event. See Section 13.2.1 for additional information.

@##any [Optional]—Additional attributes from any namespace.

core:StatusCode [Optional]—An applicable status code as defined in Appendix A. See Section 13.2.1 for additional information.

SystemContext [Optional]—Parameters describing the event's system context. See Section 13.2.1 for additional information.

Client [Optional]—Addressability or targeting details. See Section 13.2.1 for additional information.

Entertainment [Optional Choice]—Entertainment content asset identification. See Section 13.2.1 for additional information.

EntertainmentNPT [Optional Choice]—This element ***shall not*** be used and remains for backward compatibility only. In messages having a @version attribute of "1.3" or prior, this element *may* appear. It ***shall not*** be used in messages having an @version attribute of "1.4" or later. An entertainment specific NPT value is provided using the NPTOffset element contained with the Entertainment element.

Spot [Optional Choice]—Ad content asset identification. See Section 13.2.1 for additional information.

SpotNPT [Optional Choice]—This element ***shall not*** be used and remains for backward compatibility only. In messages having a @version attribute of "1.3" or prior, this element *may* appear. It ***shall not*** be used in messages having an @version attribute of "1.4" or later. A spot specific NPT value is provided using the NPTOffset element contained with the Spot element.

NPT [Optional Choice]—User perceived NPT. See Section 13.2.1 for additional information.

UTC [Optional Choice]—A UTC value referenced to a stream absolute position within live/realtime content. See Section 14.56 for additional information. The UTC is typically the clock the viewer associates with a live/realtime program or entertainment asset including ads that are delivered as part of a live/realtime asset.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

14.53.2 SystemEvent Element Examples

```
<SystemEvent time="2007-04-01T01:03:03.0Z" type="status">
  <core:StatusCode class="3">
    <core:Note>Go San Jose Sharks!</core:Note>
  </core:StatusCode>
</SystemEvent>

<SystemEvent time="2007-04-02T01:03:55.1Z" type="status">
  <core:StatusCode class="2" detail="11">
    <core:Note>Resource not found.</core:Note>
  </core:StatusCode>
  <core:Tracking>PlacementResponse-Placement tracking tag #1</core:Tracking>
  <Spot>
    <core:Content>
      <core:AssetRef providerID="adco.com" assetID="lostad1"/>
    </core:Content>
  </Spot>
</SystemEvent>
```

14.54 TargetCode Element

The TargetCode element facilitates specific targeting criteria description by the use of a key/value pair.

14.54.1 TargetCode Element Schema

Figure 95 illustrates the TargetCode element’s schema.

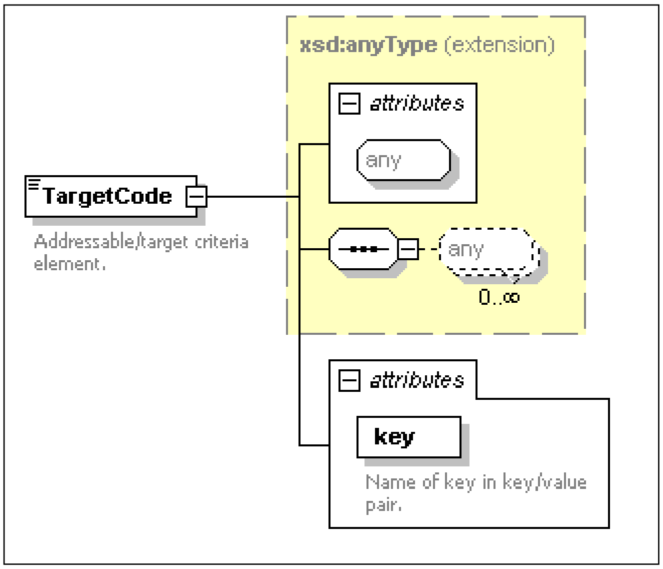

xsd:anyType (extension)
attributes
any
TargetCode
Addressable/target criteria element.
any
0..∞
attributes
key
Name of key in key/value pair.

Figure 95: TargetCode Element Schema

14.54.1.1 Semantic Definitions for the TargetCode Element

@key [Required, core:nonEmptyStringType]—The key portion of a key/value pair. The value portion of a key/value pair is the element's actual value.

@##any [Optional]—Additional attributes from any namespace.

The TargetCode element's value *may* be of any type, i.e., the type is xsd:anyType, and it *may* be empty. Note: The xsd:anyType has the unique characteristic where it *may* function as a complex and a simple type definition. Thus, the TargetCode element's value *may* be mixed content which allows any combination of elements and/or simple types.

14.54.2 TargetCode Element Examples

```
  <TargetCode key="ZipPlus4">94053-8332</TargetCode>
  <TargetCode key="MyKeyName">MyKeyValue</TargetCode>
  <TargetCode key="Gender">Male</TargetCode>
  <TargetCode key="Clowns">Left handed</TargetCode>

  <TargetCode key="Embedded element example">
    <core:ContentLocation>My content location</core:ContentLocation>
    <core:Address>22 Easy street</core:Address>
  </TargetCode>

  <TargetCode key="Mixed type example">
    <core:AdType>My ad type</core:AdType>
    Mixed content string.
    <core:Note>See the string following the element above? Any mix of elements and basic
types is allowed here.</core:Note>
  </TargetCode>
```

14.55 TerminalAddress Element

The TerminalAddress element identifies a client endpoint.

14.55.1 TerminalAddress Element Schema

Figure 96 illustrates the TerminalAddress element's schema.

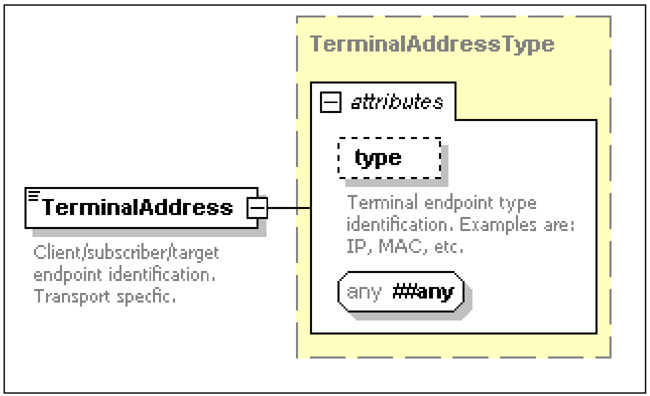

TerminalAddressType
attributes
type
Terminal endpoint type identification. Examples are: IP, MAC, etc.
any ##any
TerminalAddress
Client/subscriber/target endpoint identification. Transport specfic.

Figure 96: TerminalAddress Element Schema

14.55.1.1 Semantic Definitions for the TerminalAddress Element

@type [Optional, core:nonEmptyStringType]—Interpretation identification for the element. Table 20 lists the defined values which *may* be extended by private agreement outside the scope of this specification. The attribute’s value ***shall*** appear exactly as shown in Table 20.

Value	Description
IP	An IP version 4 (IPv4) address or an IP version 6 (IPv6) address and the address *may* optionally include a port identifier. The TerminalAddress element’s value ***shall*** be an IPv4 or IPv6 address conformant with the addressing formats defined in SCTE 130 Part 2. See [SCTE130-2] for additional information.
MAC	A 6 byte IEEE MAC address. The TerminalAddress element’s value ***shall*** be a MAC address conformant with the addressing formats defined in SCTE 130 Part 2. See [SCTE130-2] for additional information.
DEVICEID	A nonemptystring identifying the device. The Device Identifier element’s value should be a globally unique value.
…	User defined and outside the scope of this specification. The value ***shall*** be preceded by the string “private:”.

Table 20: TerminalAddress Element’s @type Attribute Values

@##any [Optional]—Additional attributes from any namespace.

The TerminalAddress element’s value is of type core:nonEmptyStringType and is the endpoint identifier. The element’s value ***shall not*** be empty.

14.55.2 TerminalAddress Element Examples

```
 <TerminalAddress type="IP">53.192.83.105</TerminalAddress>

 <TerminalAddress
type="IP">2001:0DB8:0000:0000:0000:0000:1428:57AB</TerminalAddress>

 <TerminalAddress type="MAC">02-90-95-43-55-73</TerminalAddress>
```

14.56 UTC Element

The UTC Element indicates the stream absolute position relative to a date/time and an optional viewing rate.

14.56.1 UTC Element Schema

Figure 97 illustrates the UTC element’s schema.

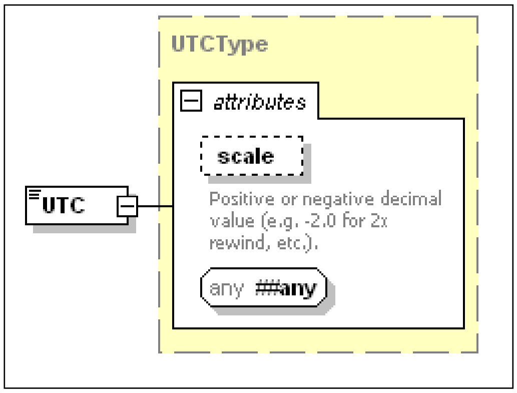

UTCType
attributes
scale
Positive or negative decimal value (e.g. -2.0 for 2x rewind, etc.).
any ##any
UTC

Figure 97: UTC Element Schema

14.56.1.1 Semantic Definitions for the UTC Element

@scale [Optional, xsd:decimal]—The scale value of 1 indicates normal play or record at the natural forward viewing rate. If the attribute is not the value of 1, the value corresponds to the rate with respect to the normal viewing rate. For example, a ratio of 2 indicates twice the natural viewing rate (i.e., "fast forward") and a ratio of 0.5 indicates half the normal viewing rate (i.e., forward at ½ or “slow forward”). In other words, a ratio of 2 has normal playtime increase at twice the wallclock rate. For every second of elapsed (wallclock) time, 2 seconds of content is enumerated. A negative value indicates reverse direction (i.e., “rewind”). For example, the value “-5.0” indicates fives times the normal viewing rate in reverse.

@##any [Optional]—Additional attributes from any namespace.

The UTC element’s value is of type core:dateTimeTimezoneType. The element’s value ***shall not*** be empty.

14.56.2 UTC Element Examples

```
<UTC scale="1">2010-11-09T11:00:00-00:00</UTC>
<UTC scale="1.0">2010-11-09T11:12:10.123Z</UTC>
<UTC>2010-11-09T11:12:10.123Z</UTC>
```

14.57 ViewerEvent Element

The ViewerEvent element provides user presentation reporting details regarding a single happening which the ADM system considers of interest. Each change in a presentation event should only specify a single occurrence of the event of interest.

14.57.1 ViewerEvent Element Schema

Figure 98 illustrates the ViewerEvent element’s schema.

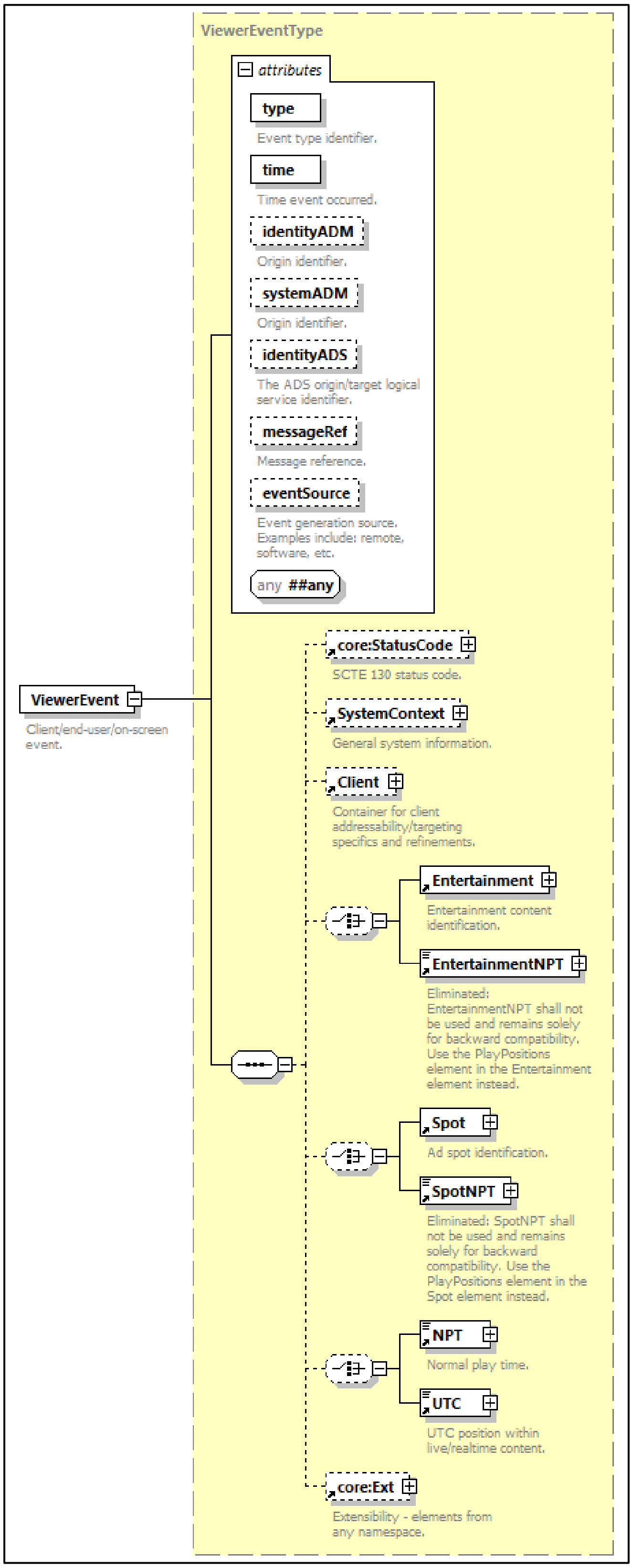

ViewerEventType
attributes
type
Event type identifier.
time
Time event occurred.
identityADM
Origin identifier.
systemADM
Origin identifier.
identityADS
The ADS origin/target logical service identifier.
messageRef
Message reference.
eventSource
Event generation source. Examples include: remote, software, etc.
any ##any
ViewerEvent
Client/end-user/on-screen event.
core:StatusCode
SCTE 130 status code.
SystemContext
General system information.
Client
Container for client addressability/targeting specifics and refinements.
Entertainment
Entertainment content identification.
EntertainmentNPT
Eliminated: EntertainmentNPT shall not be used and remains solely for backward compatibility. Use the PlayPositions element in the Entertainment element instead.
Spot
Ad spot identification.
SpotNPT
Eliminated: SpotNPT shall not be used and remains solely for backward compatibility. Use the PlayPositions element in the Spot element instead.
NPT
Normal play time.
UTC
UTC position within live/realtime content.
core:Ext
Extensibility - elements from any namespace.

Figure 98: ViewerEvent Element Schema

14.57.1.1 Semantic Definitions for the ViewerEvent Element

@type [Required, eventTypeAttrType]—Event identification. See Section 13.2.1 for additional information. Table 21 lists the defined values for this element which *may* be extended by private agreement outside the scope of this specification. The attribute's value ***shall*** appear exactly as shown in Table 21.

Value	Description
status	The event contains viewer event related status information. The event should contain a core:StatusCode element and zero or more Note elements.
fastForward	The event indicates a fast-forward (FF) event.
play	The event indicates a play event.
pause	The event indicates a pause event.
rewind	The event indicates a rewind (RW) event.
stop	The event indicates a stop event.
overlayPresented	The event indicates an overlay or other image (banner ad, etc.) was made visible.
overlayRemoved	The event indicates an overlay or other image (banner ad, etc.) is no longer visible.
…	User defined and outside the scope of this specification. The value ***shall*** be preceded by the string "private:".

Table 21: ViewerEvent @type Attribute Values

Table 23 identifies which PlayPosition elements *may* be used with each of the @type attribute values.

	Element		
Value	**PlayPosition**	**PlayPositionStart**	**PlayPositionEnd**
status	Yes	No	No
fastForward	No	Yes	Yes
play	No	Yes	Yes
pause	Yes	No	No
rewind	No	Yes	Yes
stop	Yes	No	No
overlayPresented	No	Yes	No
overlayRemoved	No	No	Yes

Table 22: ViewerEvent @type Attribute Position Element Usage

@time [Required, core:dateTimeTimezoneType]—The event occurrence date and time. See Section 13.2.1 for additional information.

@identityADM [Optional, core:identityAttrType]—The origin ADM logical service identifier. See Section 13.2.1 for additional information.

@systemADM [Optional, core:systemAttrType]—The originating ADM system identification. See Section 13.2.1 for additional information.

@identityADS [Optional, core:identityAttrType]—The ADS origin or target logical service identifier. See Section 13.2.1 for additional information.

@messageRef [Optional, core:messageRefAttrType]—The message associated with this event. See Section 13.2.1 for additional information.

@eventSource [Optional, core:nonEmptyStringType]—Event generation source identifier. Table 23 lists the defined values for this element which *may* be extended by private agreement outside the scope of this specification.

Value	Description
remote	A button on the (subscriber) remote control was pressed.
software	A software application initiated event (for example, a software timer expiration).
…	User defined and outside the scope of this specification. The value ***shall*** be preceded by the string “private:”.

Table 23: ViewerEvent Element @eventSource Attribute Values

@##any [Optional]—Additional attributes from any namespace.

core:StatusCode [Optional]—An applicable status code as defined in Appendix A. See Section 13.2.1 for additional information.

SystemContext [Optional]—Parameters describing the event’s system context. See Section 13.2.1 for additional information.

Client [Optional]—Addressability or targeting details. See Section 13.2.1 for additional information.

Entertainment [Optional Choice]—Entertainment content asset identification. See Section 13.2.1 for additional information.

EntertainmentNPT [Optional Choice]—This element ***shall not*** be used and remains for backward compatibility only. In messages having a @version attribute of "1.3" or prior, this element *may* appear. It ***shall not*** be used in messages having an @version attribute of "1.4" or later. An entertainment specific NPT value is provided using the NPTOffset element contained with the Entertainment element.

Spot [Optional Choice]—Ad content asset identification. See Section 13.2.1 for additional information.

SpotNPT [Optional Choice]—This element ***shall not*** be used and remains for backward compatibility only. In messages having a @version attribute of "1.3" or prior, this element *may* appear. It ***shall not*** be used in messages having an @version attribute of "1.4" or later. A spot specific NPT value is provided using the NPTOffset element contained with the Spot element.

NPT [Optional Choice]—User perceived NPT. See Section 13.2.1 for additional information.

UTC [Optional Choice]—A UTC value referenced to a stream absolute position within live/realtime content. See Section 14.56 for additional information. The UTC is typically the clock the viewer associates with a live/realtime program or entertainment asset including ads that are delivered as part of a live/realtime asset.

core:Ext [Optional]—A container holding additional attributes and elements from any namespace. See [SCTE130-2] for additional information.

The ViewerEvent element's value *may* be empty.

14.57.2 ViewerEvent Element Examples

The following examples are all self describing ViewerEvent elements.

```
<ViewerEvent time="2007-03-19T02:30:47.0Z" type="pause">
  <SystemContext>
    <Session>Session0068</Session>
  </SystemContext>
  <Spot>
    <core:Content>
      <core:AssetRef providerID="adco.com" assetID="ADCO123456789012345"/>
      <core:AdType>Telescoping</core:AdType>
      <core:Duration>PT30S</core:Duration>
      <core:Tracking>AssignedTrackingId1</core:Tracking>
    </core:Content>
  </Spot>

</ViewerEvent>
<ViewerEvent time="2007-03-19T02:30:57.0Z" type="fastForward">
  <SystemContext>
    <Session>Session0068</Session>
  </SystemContext>
  <Spot>
    <core:Content>
      <core:AssetRef providerID="adco.com" assetID="ADCO123456789012345"/>
      <core:AdType>Telescoping</core:AdType>
      <core:Duration>PT30S</core:Duration>
      <core:Tracking>AssignedTrackingId1</core:Tracking>
    </core:Content>
  </Spot>
</ViewerEvent>

<ViewerEvent time="2007-03-19T02:31:10.0Z" type="status">
  <core:StatusCode class="1" detail="5">
    <core:Note>Ambiguous details. Unrecognized button press.</core:Note>
  </core:StatusCode>
  <SystemContext>
    <Session>Session0068</Session>
  </SystemContext>
  <Spot>
    <core:Content>
      <core:AssetRef providerID="adco.com" assetID="ADCO123456789012345"/>
      <core:AdType>Telescoping</core:AdType>
      <core:Duration>PT30S</core:Duration>
      <core:Tracking>AssignedTrackingId1</core:Tracking>
    </core:Content>
  </Spot>
</ViewerEvent>
```

The first example ViewerEvent element identifies a pause button event for a specific session and asset. The Content element provides as much information as is known about the ad asset including the ADS assigned Tracking element. The Tracking element contains the ADS assigned tracking value which was returned in the Placement element.

The second example is a fast forward button event. The final example ViewerEvent element supplies information about an unrecognized button event relative to the same ad asset.

14.58 Window Element

The Window element contains a start and end date and time typically from a schedule. The values provide a range for when the placement opportunity is scheduled.

14.58.1 Window Element Schema

Figure 99 illustrates the Window element's schema.

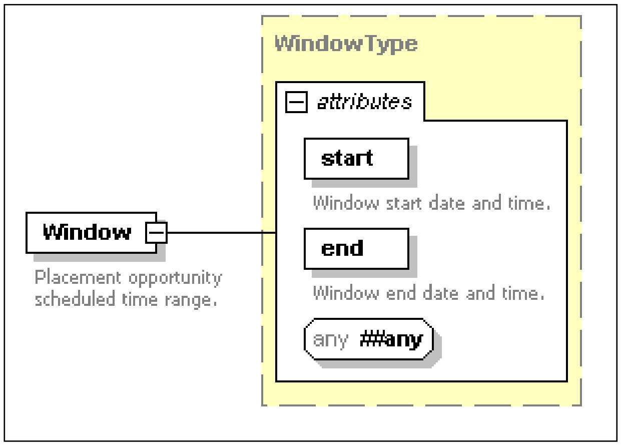

WindowType
attributes
start
Window start date and time.
end
Window end date and time.
any ##any
Window
Placement opportunity scheduled time range.

Figure 99: Window Element Schema

14.58.1.1 Semantic Definitions for the Window Element

@start [Required, core:dateTimeTimezoneType]—Range commencement date and time. See [SCTE130-2] for additional information.

@end [Required, core:dateTimeTimezoneType]—Range termination date and time. See [[SCTE130-2] for additional information.

@##any [Optional]—Additional attributes from any namespace.

The Window element's value ***shall*** be empty.

14.58.2 Window Element Examples

```
<Window start="2007-06-06T03:33:43.0Z" end="2007-06-06T03:43:43.0Z"/>
```

14.59 Zone Element

The Zone element facilitates zone identification which traditionally has been a geographical area in a fulfillment delivery system. This specification does not restrict, limit or curb the definition in any way.

14.59.1 Zone Element Schema

Figure 100 illustrates the Zone element’s schema.

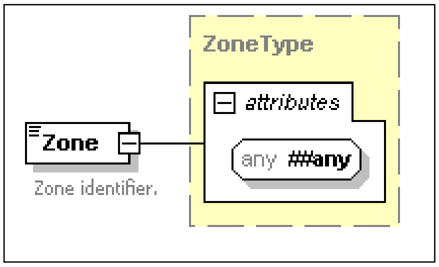


Figure 100: Zone Element Schema

14.59.1.1 Semantic Definitions for the Zone Element

@##any [Optional]—Additional attributes from any namespace.

The Zone element’s value is of type core:nonEmptyStringType and ***shall not*** be empty.

14.59.2 Zone Element Examples

```
<Zone>123</Zone>
<Zone>Den</Zone>
```

APPENDIX A. MESSAGE STATUS CODES (NORMATIVE)

In the ensuing sections, the tables contain applicable values for the StatusCode element's @detail attribute where a checkmark (✓) in the table column indicates when the code *may* be used.

A.1 Message Column Acronyms Used In Subsequent Tables

The following acronyms are used as table column headings in the subsequent sections.

SCR=ServiceCheckResponse

SSN=ServiceStatusNotification

SSA=ServiceStatusAcknowledgement

LSR=ListADMServicesResponse

RR=ADSRegistrationResponse

LRR=ListADSRegistrationResponse

DR=ADSDeregisterResponse

DRN=ADSDeregistrationNotification

DRA=ADSDeregistrationAcknowledgement

PRS=PlacementResponse

PUN=PlacementUpdateNotification

PUA=PlacementUpdateAcknowledgement

PSA=PlacementStatusAcknowledgement

E=any named event (i.e., PlacementStatusEvent, SessionEvent, SystemEvent, ViewerEvent, etc.)

A.2 SCTE 130 Part 2 StatusCode Element @detail Attribute Values Applied

Description	S C R	S S N	S S A	L S R	R R	L R R	D R	D R N	D R A	P R S	P U N	P U A	P S A	E
Reserved														
Incomplete message	✓		✓	✓	✓	✓	✓		✓	✓		✓	✓	
Message validation failed	✓		✓	✓	✓	✓	✓		✓	✓		✓	✓	
Registration overlap					✓									
Query failed														
Ambiguous details	✓		✓	✓	✓	✓	✓		✓	✓		✓	✓	✓
Unsupported protocol														
Network address does not exist		✓			✓									
Network address/port in use	✓	✓			✓									
Duplicate message id	✓	✓	✓	✓	✓	✓	✓		✓	✓		✓	✓	
Network connection lost	✓	✓		✓	✓	✓	✓		✓	✓		✓	✓	
Resource not found	✓	✓												✓
Not supported												✓		
Not authorized						✓	✓							
Unknown message reference	✓		✓	✓	✓	✓	✓		✓	✓		✓	✓	
Resend forced abandonment	✓		✓	✓	✓	✓	✓		✓	✓		✓	✓	
Out of resources	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Timeout	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
General error	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

A.3 SCTE 130 Part 3 StatusCode Element @detail Attribute Values Defined and Applied

@detail Value	Description	SCR	SSN	SSA	LSR	RR	LRR	DR	DRN	DRA	PRS	PUN	PUA	PSA	E
3000	Reserved														
3001	Placement duration exceeded request														✓
3002	Placement duration does not match asset length														✓
3003	Content failed analysis												✓		✓
3004	Abnormal termination of play out														✓
3005	Abnormal transition between segments														✓
3006	Resumed from pause														✓
3007	Session terminated on pause														✓
3008	Session terminated by user														✓
3009	Session resource unavailable														✓
3010	Session inadequate resources												✓		✓
3011	Placement not executed														✓

APPENDIX B. VIDEO ON DEMAND EXAMPLE (INFORMATIVE)

The following example is a simple video on demand session that assumes the appropriate registration has already taken place. A single PlacementRequest message is sent on session initiation requesting the pre-roll and post-roll placement decisions. How the ADM determined the pre-roll and post-roll placement opportunities is outside the example scope.

The entertainment content is from videobus.com and is twenty minutes in duration.

The ADS responds with placement decisions for both the pre-roll and post-roll positions as follows:

- Pre-roll ad #1 is from advertiser1.com and is 15 seconds duration
- Pre-roll ad #2 is from advertiser2.com and is 15 seconds duration
- Pre-roll ad #3 is from advertiser3.com and is 30 seconds duration.
- Post-roll ad #1 is from advertiser1.com and is 60 seconds in duration
- Post-roll ad #2 is from advertiser2.com and is 60 seconds in duration
- Post-roll ad #3 is from advertiser3.com and is 60 seconds in duration

Example 68 illustrates a PlacementRequest message.

```
<PlacementRequest messageId="5B8ECCED-979B-421E-8A6E-E2192B115160"
version="1.4" identity="ADMLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F4" system="adm" updatesAllowed="false">
  <SystemContext>
    <Session>ExampleSession1</Session>
  </SystemContext>
  <Entertainment>
    <core:Content>
      <core:AssetRef providerID="videobus.com" assetID="BAAD9876543210987654"/>
      <core:Duration>PT20M</core:Duration>
    </core:Content>
  </Entertainment>
  <Client>
    <core:CurrentDateTime>2007-01-31T01:59:59+05:00</core:CurrentDateTime>
    <TargetCode key="ZipPlus4">12345-6789</TargetCode>
    <TargetCode key="Gender">Male</TargetCode>
  </Client>
  <ADMData>Hidden secret sauce</ADMData>
  <PlacementOpportunity id="1B8ECCED-979B-421E-8A6E-E2192B115161"
serviceRegistrationRef="ServiceRef1">
    <OpportunityBinding opportunityType="preRoll"/>
  </PlacementOpportunity>
  <PlacementOpportunity id="1B8ECCED-979B-421E-8A6E-E2192B115162"
serviceRegistrationRef="ServiceRef1">
    <OpportunityBinding opportunityType="postRoll"/>
  </PlacementOpportunity>
</PlacementRequest>
```

Example 68

Example 69 illustrate a PlacementResponse message.

```
<PlacementResponse messageId="5B8ECCED-979B-421E-8A6E-E2192B11516A"
version="1.4" identity="ADSLogicalService_1170FEF4-C19B-450E-B624-
```

```
421B23F525F5" system="ads" messageRef="5B8ECCED-979B-421E-8A6E-
E2192B115160">
  <core:StatusCode class="0"/>
  <ADMData>Hidden secret sauce</ADMData>
  <PlacementDecision id="5B8ECCED-979B-421E-8A6E-E2192B115162"
placementOpportunityRef="1B8ECCED-979B-421E-8A6E-E2192B115161">
    <OpportunityBinding opportunityType="preRoll"/>
    <Placement id="5B8ECCED-979B-421E-8A6E-E2192B115163" action="fill">
      <core:Content>
        <core:AssetRef providerID="advertiser1.com"
assetID="ADVR1000123456789001"/>
        <core:Duration>PT15S</core:Duration>
        <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C92</core:Tracking>
      </core:Content>
    </Placement>
    <Placement id="5B8ECCED-979B-421E-8A6E-E2192B115163" action="fill">
      <core:Content>
        <core:AssetRef providerID="advertiser2.com"
assetID="ADVR2000123456789001"/>
        <core:Duration>PT15S</core:Duration>
        <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C93</core:Tracking>
      </core:Content>
    </Placement>
    <Placement id="5B8ECCED-979B-421E-8A6E-E2192B115163" action="fill">
      <core:Content>
        <core:AssetRef providerID="advertiser3.com"
assetID="ADVR3000123456789001"/>
        <core:Duration>PT30.000S</core:Duration>
        <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C94</core:Tracking>
      </core:Content>
    </Placement>
  </PlacementDecision>
  <PlacementDecision id="5B8ECCED-979B-421E-8A6E-E2192B115164"
placementOpportunityRef="1B8ECCED-979B-421E-8A6E-E2192B115162">
    <OpportunityBinding opportunityType="postRoll"/>
    <Placement id="5B8ECCED-979B-421E-8A6E-E2192B115165" action="fill">
      <core:Content>
        <core:AssetRef providerID="advertiser1.com"
assetID="ADVR1000123456789002"/>
        <core:Duration>PT60.000S</core:Duration>
        <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C96</core:Tracking>
      </core:Content>
    </Placement>
    <Placement id="5B8ECCED-979B-421E-8A6E-E2192B115165" action="fill">
      <core:Content>
```

```
        <core:AssetRef providerID="advertiser2.com"
assetID="ADVR2000123456789002"/>
        <core:Duration>PT01M00.000S</core:Duration>
        <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C97</core:Tracking>
      </core:Content>
    </Placement>
    <Placement id="5B8ECCED-979B-421E-8A6E-E2192B115165" action="fill">
      <core:Content>
        <core:AssetRef providerID="advertiser3.com"
assetID="ADVR3000123456789002"/>
        <core:Duration>PT60.000S</core:Duration>
        <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C98</core:Tracking>
      </core:Content>
    </Placement>
  </PlacementDecision>
</PlacementResponse>
```

Example 69

Example 70 and Example 71 illustrate the same PlacementStatusNotification messages elements using two different styles. Example 70 uses self-describing events and Example 71 uses data minimization. The two messages are identical in meaning and they illustrate two possible ADM implementation choices. Additionally, there are numerous other element combinations which would equate to the same meaning. These messages are examples only and should not be considered the only correct XML documents.

The event sequence encompassed in the PlacementStatusNotification messages is illustrated as follows:

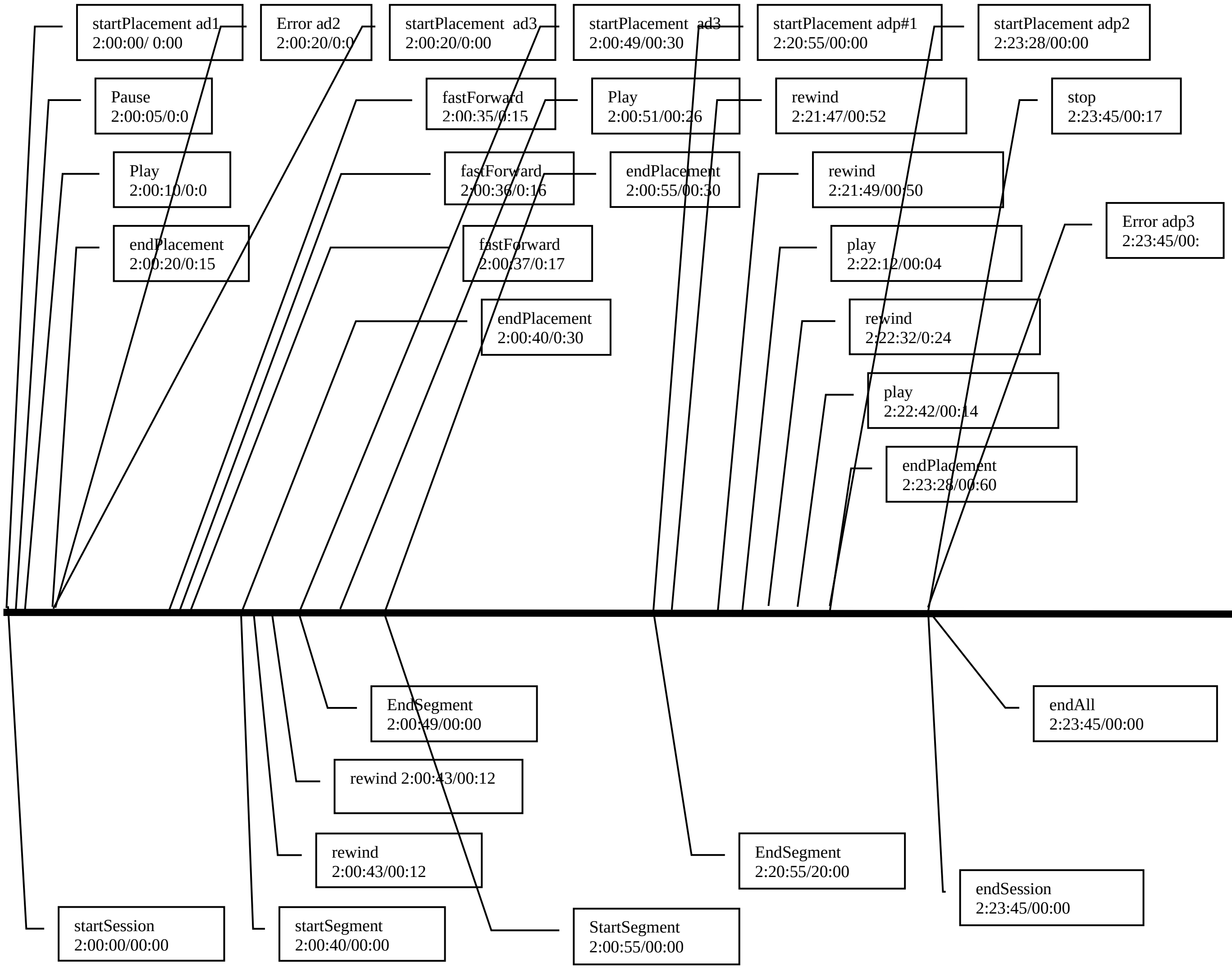


Figure 101: Placement Status Timeline

```
<PlacementStatusNotification messageId="3EE278EC-A61F-D080-2D5C-
62806A8D146F"
                        identity="ADMLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F4"
                        system="ADM01" version="1.4"
>
  <PlayData
     identityADS="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5">
    <SystemContext>
      <Session>ExampleSession1</Session>
    </SystemContext>
    <Events>
      <!--Session creation. Informational note only.-->
      <SessionEvent time="2007-01-31T02:00:00Z" type="startSession">
        <core:StatusCode class="3">
          <core:Note>Session start.</core:Note>
        </core:StatusCode>
      </SessionEvent>
      <!--Pre-roll ad #1: The ad is played from start to finish in normal play time
          with exactly one 5-second pause at the 5-second point. The ad's duration is
          15 seconds.-->
      <PlacementStatusEvent time="2007-01-31T02:00:00Z"
                       type="startPlacement">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="advertiser1.com"
                       assetID="ADVR1000123456789001"/>
            <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C92</core:Tracking>
          </core:Content>
          <PlayPositions>
            <PlayPositionStart scale="1">
              <NPTOffset>0</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Spot>
      </PlacementStatusEvent>
      <ViewerEvent time="2007-01-31T02:00:05Z" type="pause"
              eventSource="remote">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="advertiser1.com"
                       assetID="ADVR1000123456789001"/>
            <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C92</core:Tracking>
          </core:Content>
          <PlayPositions>
            <PlayPosition scale="0">
              <NPTOffset>5</NPTOffset>
            </PlayPosition>
          </PlayPositions>
        </Spot>
      </ViewerEvent>
      <ViewerEvent time="2007-01-31T02:00:10Z" type="play"
              eventSource="remote">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="advertiser1.com"
                       assetID="ADVR1000123456789001"/>
            <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C92</core:Tracking>
          </core:Content>
          <PlayPositions>
            <PlayPositionStart scale="1">
              <NPTOffset>5</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Spot>
      </ViewerEvent>
      <PlacementStatusEvent time="2007-01-31T02:00:20Z"
                       type="endPlacement">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="advertiser1.com"
                       assetID="ADVR1000123456789001"/>
            <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C92</core:Tracking>
          </core:Content>
          <PlayPositions>
            <PlayPositionEnd scale="1">
              <NPTOffset>15</NPTOffset>
            </PlayPositionEnd>
          </PlayPositions>
        </Spot>
      </PlacementStatusEvent>
      <!--Pre-roll ad #2: The 15 second ad did not play due to an error.-->
      <PlacementStatusEvent time="2007-01-31T02:00:20Z" type="status">
        <core:StatusCode class="1" detail="11">
          <core:Note>Resource not found.</core:Note>
        </core:StatusCode>
        <Spot>
          <core:Content>
            <core:AssetRef providerID="advertiser2.com"
                       assetID="ADVR2000123456789001"/>
            <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C93</core:Tracking>
          </core:Content>
        </Spot>
      </PlacementStatusEvent>
      <!--Pre-roll ad #3: A 30-second ad is played half way, then is fast forwarded to the end.
          The FF speed is eventually 4x. However, the user "cranks-up" from 1x FF, to
          2x FF, to 4x FF. This takes three button presses each occuring one second apart.
          Consequently, the the ad's remaining 15 seconds play in 5 seconds.-->
      <PlacementStatusEvent time="2007-01-31T02:00:20Z"
                       type="startPlacement">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="advertiser3.com"
                       assetID="ADVR3000123456789001"/>
            <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C94</core:Tracking>
          </core:Content>
          <PlayPositions>
            <PlayPositionStart scale="1">
              <NPTOffset>0</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Spot>
      </PlacementStatusEvent>
      <ViewerEvent time="2007-01-31T02:00:35Z" type="fastForward"
              eventSource="remote">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="advertiser3.com"
                       assetID="ADVR3000123456789001"/>
            <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C94</core:Tracking>
          </core:Content>
          <PlayPositions>
            <PlayPositionStart scale="1">
              <NPTOffset>15</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Spot>
      </ViewerEvent>
      <ViewerEvent time="2007-01-31T02:00:36Z" type="fastForward"
              eventSource="remote">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="advertiser3.com"
                       assetID="ADVR3000123456789001"/>
            <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C94</core:Tracking>
          </core:Content>
          <PlayPositions>
            <PlayPositionStart scale="2">
```

```
              <NPTOffset>16</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Spot>
      </ViewerEvent>
      <ViewerEvent time="2007-01-31T02:00:37Z" type="fastForward"
                  eventSource="remote">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="advertiser3.com"
                          assetID="ADVR3000123456789001"/>
            <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C94</core:Tracking>
          </core:Content>
          <PlayPositions>
            <PlayPositionStart scale="4">
              <NPTOffset>18</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Spot>
      </ViewerEvent>
      <PlacementStatusEvent time="2007-01-31T02:00:40Z"
                           type="endPlacement">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="advertiser3.com"
                          assetID="ADVR3000123456789001"/>
            <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C94</core:Tracking>
          </core:Content>
          <PlayPositions>
            <PlayPositionEnd scale="4">
              <NPTOffset>30</NPTOffset>
            </PlayPositionEnd>
          </PlayPositions>
        </Spot>
      </PlacementStatusEvent>
      <!--Start of the entertainment asset with the subscriber still being in 4x fast forward
mode.-->
      <SessionEvent time="2007-01-31T02:00:40Z" type="startSegment">
        <Entertainment>
          <core:Content>
            <core:AssetRef providerID="videobus.com"
                          assetID="BAAD9876543210987654"/>
          </core:Content>
          <PlayPositions>
            <PlayPositionStart scale="4">
              <NPTOffset>0</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Entertainment>
      </SessionEvent>
      <!--The user stops the fast forward after 3 seconds (which is 12 seconds into content).
          The user rewinds back by cranking up to 2x. Thus, there are two button pushes that
          are "instantaneous" given the clock granularity in this example. The user
          travels back 8 seconds which lands 4 seconds back into the ad. The user then starts
          playing at normal speed for the duration of the entertainment asset (which
          is 20 minutes in duration).-->
      <ViewerEvent time="2007-01-31T02:00:43Z" type="rewind"
                  eventSource="remote">
        <Entertainment>
          <core:Content>
            <core:AssetRef providerID="videobus.com"
                          assetID="BAAD9876543210987654"/>
          </core:Content>
          <PlayPositions>
            <PlayPositionStart scale="-1">
              <NPTOffset>12</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Entertainment>
      </ViewerEvent>
      <ViewerEvent time="2007-01-31T02:00:43Z" type="rewind"
                  eventSource="remote">
        <Entertainment>
          <core:Content>
            <core:AssetRef providerID="videobus.com"
                          assetID="BAAD9876543210987654"/>
          </core:Content>
          <PlayPositions>
            <PlayPositionStart scale="-2">
              <NPTOffset>12</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Entertainment>
      </ViewerEvent>
      <SessionEvent time="2007-01-31T02:00:49Z" type="endSegment">
        <Entertainment>
          <core:Content>
            <core:AssetRef providerID="videobus.com"
                          assetID="BAAD9876543210987654"/>
          </core:Content>
          <PlayPositions>
            <PlayPositionEnd scale="-2">
          <NPTOffset>0</NPTOffset>
         </PlayPositionEnd>
        </PlayPositions>
       </Entertainment>
      </SessionEvent>
      <PlacementStatusEvent time="2007-01-31T02:00:49Z"
                 type="startPlacement">
       <Spot>
        <core:Content>
         <core:AssetRef providerID="advertiser3.com"
                assetID="ADVR3000123456789001"/>
         <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C94</core:Tracking>
        </core:Content>
        <PlayPositions>
         <PlayPositionStart scale="-2">
          <NPTOffset>30</NPTOffset>
         </PlayPositionStart>
        </PlayPositions>
       </Spot>
      </PlacementStatusEvent>
      <ViewerEvent time="2007-01-31T02:00:51Z" type="play"
            eventSource="remote">
       <Spot>
        <core:Content>
         <core:AssetRef providerID="advertiser3.com"
                assetID="ADVR3000123456789001"/>
         <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C94</core:Tracking>
        </core:Content>
        <PlayPositions>
         <PlayPositionStart scale="1">
          <NPTOffset>26</NPTOffset>
         </PlayPositionStart>
        </PlayPositions>
       </Spot>
      </ViewerEvent>
      <PlacementStatusEvent time="2007-01-31T02:00:55Z"
                 type="endPlacement">
       <Spot>
        <core:Content>
         <core:AssetRef providerID="advertiser3.com"
                assetID="ADVR3000123456789001"/>
         <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C94</core:Tracking>
        </core:Content>
        <PlayPositions>
         <PlayPositionEnd scale="1">
          <NPTOffset>30</NPTOffset>
```

```
      </PlayPositionEnd>
    </PlayPositions>
  </Spot>
</PlacementStatusEvent>
<SessionEvent time="2007-01-31T02:00:55Z" type="startSegment">
  <Entertainment>
    <core:Content>
      <core:AssetRef providerID="videobus.com"
                    assetID="BAAD9876543210987654"/>
    </core:Content>
    <PlayPositions>
      <PlayPositionStart scale="1">
        <NPTOffset>0</NPTOffset>
      </PlayPositionStart>
    </PlayPositions>
  </Entertainment>
</SessionEvent>
<SessionEvent time="2007-01-31T02:20:55Z" type="endSegment">
  <Entertainment>
    <core:Content>
      <core:AssetRef providerID="videobus.com"
                    assetID="BAAD9876543210987654"/>
    </core:Content>
    <PlayPositions>
      <PlayPositionEnd scale="1">
        <NPTOffset>1200</NPTOffset>
      </PlayPositionEnd>
    </PlayPositions>
  </Entertainment>
</SessionEvent>
<!--Post-roll ad #1: The 60 second ad was played from start to finish with 2 rewinds
    inside of the ad. The rewinds never exit the ad. The first rewind occurs
    52 seconds into the ad.-->
<PlacementStatusEvent time="2007-01-31T02:20:55Z"
                      type="startPlacement">
  <Spot>
    <core:Content>
      <core:AssetRef providerID="advertiser1.com"
                    assetID="ADVR1000123456789002"/>
      <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C96</core:Tracking>
    </core:Content>
    <PlayPositions>
      <PlayPositionStart scale="1">
        <NPTOffset>0</NPTOffset>
      </PlayPositionStart>
    </PlayPositions>
  </Spot>
</PlacementStatusEvent>
<!--Begin rewinding at 1x speed for 2 seconds going back 2 seconds in 1x play time.
    The user then goes to the 2x rewind speed.-->
<ViewerEvent time="2007-01-31T02:21:47Z" type="rewind"
            eventSource="remote">
  <Spot>
    <core:Content>
      <core:AssetRef providerID="advertiser1.com"
                    assetID="ADVR1000123456789002"/>
      <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C96</core:Tracking>
    </core:Content>
    <PlayPositions>
      <PlayPositionStart scale="-1">
        <NPTOffset>52</NPTOffset>
      </PlayPositionStart>
    </PlayPositions>
  </Spot>
</ViewerEvent>
<!--Rewind at 2x speed for 23 seconds going back 46 seconds in 1x play time.-->
<ViewerEvent time="2007-01-31T02:21:49Z" type="rewind"
            eventSource="remote">
  <Spot>
    <core:Content>
      <core:AssetRef providerID="advertiser1.com"
                    assetID="ADVR1000123456789002"/>
      <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C96</core:Tracking>
    </core:Content>
    <PlayPositions>
      <PlayPositionStart scale="-2">
        <NPTOffset>50</NPTOffset>
      </PlayPositionStart>
    </PlayPositions>
  </Spot>
</ViewerEvent>
<ViewerEvent time="2007-01-31T02:22:12Z" type="play"
            eventSource="remote">
  <Spot>
    <core:Content>
      <core:AssetRef providerID="advertiser1.com"
                    assetID="ADVR1000123456789002"/>
      <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C96</core:Tracking>
    </core:Content>
    <PlayPositions>
      <PlayPositionStart scale="1">
        <NPTOffset>4</NPTOffset>
```

```
   </PlayPositionStart>
  </PlayPositions>
 </Spot>
</ViewerEvent>
<!--After playing for 20 seconds, rewind at 1x speed for 10 seconds going
  back 10 seconds in 1x play time and then play for the remainder of the ad.-->
<ViewerEvent time="2007-01-31T02:22:32Z" type="rewind"
      eventSource="remote">
 <Spot>
  <core:Content>
   <core:AssetRef providerID="advertiser1.com"
          assetID="ADVR1000123456789002"/>
   <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C96</core:Tracking>
  </core:Content>
  <PlayPositions>
   <PlayPositionStart scale="-1">
    <NPTOffset>24</NPTOffset>
   </PlayPositionStart>
  </PlayPositions>
 </Spot>
</ViewerEvent>
<ViewerEvent time="2007-01-31T02:22:42Z" type="play"
      eventSource="remote">
 <Spot>
  <core:Content>
   <core:AssetRef providerID="advertiser1.com"
          assetID="ADVR1000123456789002"/>
   <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C96</core:Tracking>
  </core:Content>
  <PlayPositions>
   <PlayPositionStart scale="1">
    <NPTOffset>14</NPTOffset>
   </PlayPositionStart>
  </PlayPositions>
 </Spot>
</ViewerEvent>
<PlacementStatusEvent time="2007-01-31T02:23:28Z"
           type="endPlacement">
 <Spot>
  <core:Content>
   <core:AssetRef providerID="advertiser1.com"
          assetID="ADVR1000123456789002"/>
   <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C96</core:Tracking>
  </core:Content>
  <PlayPositions>
   <PlayPositionEnd scale="1">
```

```
              <NPTOffset>60</NPTOffset>
            </PlayPositionEnd>
          </PlayPositions>
        </Spot>
      </PlacementStatusEvent>
      <!--Post-roll ad #2: The ad was was started and then the session was stopped.-->
      <PlacementStatusEvent time="2007-01-31T02:23:28Z"
                           type="startPlacement">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="advertiser2.com"
                          assetID="ADVR2000123456789002"/>
            <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C97</core:Tracking>
          </core:Content>
          <PlayPositions>
            <PlayPositionStart scale="1">
              <NPTOffset>0</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Spot>
      </PlacementStatusEvent>
      <ViewerEvent time="2007-01-31T02:23:45Z" type="stop"
                   eventSource="remote">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="advertiser2.com"
                          assetID="ADVR2000123456789002"/>
            <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C97</core:Tracking>
          </core:Content>
          <PlayPositions>
            <PlayPosition>
              <NPTOffset>17</NPTOffset>
            </PlayPosition>
          </PlayPositions>
        </Spot>
      </ViewerEvent>
      <!-- Post-roll ad #3 did not play since the session is terminated. -->
      <!--End of session notification.-->
      <SessionEvent time="2007-01-31T02:23:45Z" type="endSession">
        <core:StatusCode class="3" detail="3008">
          <core:Note>Session terminated by user.</core:Note>
        </core:StatusCode>
      </SessionEvent>
      <!--Post-roll ad #3 did not play and the ADM in cleaning up the session notifies the
ADS.
          This notification is not required and this is an ADM implementation choice.-->
```

```
   <PlacementStatusEvent time="2007-01-31T02:23:45Z" type="status">
    <core:StatusCode class="1" detail="3011">
     <core:Note>Placement not executed.</core:Note>
     <core:Note>Asset not played. Session ended before playout.
     </core:Note>
    </core:StatusCode>
    <Spot>
     <core:Content>
      <core:AssetRef providerID="advertiser3.com"
             assetID="ADVR3000123456789002"/>
      <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C98
      </core:Tracking>
     </core:Content>
    </Spot>
   </PlacementStatusEvent>
  </Events>
 </PlayData>
 <PlayData
   identityADS="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5">
  <Events>
   <!--The ADM implementation decided to put this event is a seperate PlayData
     element. This not required and it could have immediately followed the
     last PlacementStatusEvent above.-->
   <PlacementStatusEvent time="2007-01-31T02:23:50Z" type="endAll"
              messageRef="5B8ECCED-979B-421E-8A6E-E2192B11516A"/>
  </Events>
 </PlayData>
</PlacementStatusNotification>
```

Example 70

Example 71 illustrates the data minimized PlacementStatusNotification message.

```
<PlacementStatusNotification messageId="3EE278EC-A61F-D080-2D5C-
62806A8D146F"
                            identity="ADMLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F4"
                            system="ADM01" version="1.4"
>
  <PlayData
      identityADS="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5">
    <SystemContext>
      <Session>ExampleSession1</Session>
    </SystemContext>
    <EntertainmentScopedEvents>
      <Entertainment>
        <core:Content>
          <core:AssetRef providerID="videobus.com"
                        assetID="BAAD9876543210987654"/>
        </core:Content>
      </Entertainment>
      <Events>
        <!--Session creation. Informational note only.-->
        <SessionEvent time="2007-01-31T02:00:00Z" type="startSession">
          <core:StatusCode class="3">
            <core:Note>Session start.</core:Note>
          </core:StatusCode>
        </SessionEvent>
      </Events>
      <SpotScopedEvents>
        <!--Pre-roll ad #1: The ad is played from start to finish in normal play time
            with exactly one 5-second pause at the 5-second point. The ad's duration is
            15 seconds.-->
        <Spot>
          <core:Content>
            <core:AssetRef providerID="advertiser1.com"
                          assetID="ADVR1000123456789001"/>
            <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C92</core:Tracking>
          </core:Content>
        </Spot>
        <Events>
          <PlacementStatusEvent time="2007-01-31T02:00:00Z"
                               type="startPlacement">
            <Spot>
              <core:Content/>
              <PlayPositions>
                <PlayPositionStart scale="1">
                  <NPTOffset>0</NPTOffset>
                </PlayPositionStart>
              </PlayPositions>
            </Spot>
          </PlacementStatusEvent>
          <ViewerEvent time="2007-01-31T02:00:05Z" type="pause"
                       eventSource="remote">
            <Spot>
              <core:Content/>
              <PlayPositions>
                <PlayPosition scale="0">
                  <NPTOffset>5</NPTOffset>
                </PlayPosition>
              </PlayPositions>
            </Spot>
          </ViewerEvent>
          <ViewerEvent time="2007-01-31T02:00:10Z" type="play"
                       eventSource="remote">
            <Spot>
              <core:Content/>
              <PlayPositions>
                <PlayPositionStart scale="1">
                  <NPTOffset>5</NPTOffset>
                </PlayPositionStart>
              </PlayPositions>
            </Spot>
          </ViewerEvent>
          <PlacementStatusEvent time="2007-01-31T02:00:20Z"
                               type="endPlacement">
            <Spot>
              <core:Content/>
              <PlayPositions>
                <PlayPositionEnd scale="1">
                  <NPTOffset>15</NPTOffset>
                </PlayPositionEnd>
              </PlayPositions>
            </Spot>
          </PlacementStatusEvent>
        </Events>
      </SpotScopedEvents>
      <Events>
        <!--Pre-roll ad #2: The 15 second ad did not play due to an error.-->
        <PlacementStatusEvent time="2007-01-31T02:00:20Z" type="status">
          <core:StatusCode class="1" detail="11">
            <core:Note>Resource not found.</core:Note>
          </core:StatusCode>
          <Spot>
            <core:Content>
```

```
       <core:AssetRef providerID="advertiser2.com"
              assetID="ADVR2000123456789001"/>
       <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C93</core:Tracking>
      </core:Content>
     </Spot>
    </PlacementStatusEvent>
   </Events>
   <SpotScopedEvents>
    <!--Pre-roll ad #3: A 30-second ad is played half way, then is fast forwarded to the
end.
      The FF speed is eventually 4x. However, the user "cranks-up" from 1x FF, to
      2x FF, to 4x FF. This takes three button presses each occuring one second apart.
      Consequently, the the ad's remaining 15 seconds play in 5 seconds.-->
    <Spot>
     <core:Content>
      <core:AssetRef providerID="advertiser3.com"
             assetID="ADVR3000123456789001"/>
      <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C94</core:Tracking>
     </core:Content>
    </Spot>
    <Events>
     <PlacementStatusEvent time="2007-01-31T02:00:20Z"
                type="startPlacement">
      <Spot>
       <core:Content/>
       <PlayPositions>
        <PlayPositionStart scale="1">
         <NPTOffset>0</NPTOffset>
        </PlayPositionStart>
       </PlayPositions>
      </Spot>
     </PlacementStatusEvent>
     <ViewerEvent time="2007-01-31T02:00:35Z" type="fastForward"
           eventSource="remote">
      <Spot>
       <core:Content/>
       <PlayPositions>
        <PlayPositionStart scale="1">
         <NPTOffset>15</NPTOffset>
        </PlayPositionStart>
       </PlayPositions>
      </Spot>
     </ViewerEvent>
     <ViewerEvent time="2007-01-31T02:00:36Z" type="fastForward"
           eventSource="remote">
      <Spot>
```

```
       <core:Content/>
       <PlayPositions>
        <PlayPositionStart scale="2">
         <NPTOffset>16</NPTOffset>
        </PlayPositionStart>
       </PlayPositions>
      </Spot>
     </ViewerEvent>
     <ViewerEvent time="2007-01-31T02:00:37Z" type="fastForward"
           eventSource="remote">
      <Spot>
       <core:Content/>
       <PlayPositions>
        <PlayPositionStart scale="4">
         <NPTOffset>18</NPTOffset>
        </PlayPositionStart>
       </PlayPositions>
      </Spot>
     </ViewerEvent>
     <PlacementStatusEvent time="2007-01-31T02:00:40Z"
                type="endPlacement">
      <Spot>
       <core:Content/>
       <PlayPositions>
        <PlayPositionEnd scale="4">
         <NPTOffset>30</NPTOffset>
        </PlayPositionEnd>
       </PlayPositions>
      </Spot>
     </PlacementStatusEvent>
    </Events>
   </SpotScopedEvents>
   <Events>
    <!--Start of the entertainment asset with the subscriber still being in 4x fast forward
mode.-->
    <SessionEvent time="2007-01-31T02:00:40Z" type="startSegment">
     <Spot>
      <core:Content/>
      <PlayPositions>
       <PlayPositionStart scale="54">
        <NPTOffset>0</NPTOffset>
       </PlayPositionStart>
      </PlayPositions>
     </Spot>
    </SessionEvent>
    <!--The user stops the fast forward after 3 seconds (which is 12 seconds into content).
```

```
      The user rewinds back by cranking up to 2x. Thus, there are two button pushes that
      are "instantaneous" given the clock granularity in this example. The user
      travels back 8 seconds which lands 4 seconds back into the ad. The user then starts
      playing at normal speed for the duration of the entertainment asset (which
      is 20 minutes in duration).-->
    <ViewerEvent time="2007-01-31T02:00:43Z" type="rewind"
                eventSource="remote">
      <Spot>
        <core:Content/>
        <PlayPositions>
          <PlayPositionStart scale="-1">
            <NPTOffset>12</NPTOffset>
          </PlayPositionStart>
        </PlayPositions>
      </Spot>
    </ViewerEvent>
    <ViewerEvent time="2007-01-31T02:00:43Z" type="rewind"
                eventSource="remote">
      <Spot>
        <core:Content/>
        <PlayPositions>
          <PlayPositionStart scale="-2">
            <NPTOffset>12</NPTOffset>
          </PlayPositionStart>
        </PlayPositions>
      </Spot>
    </ViewerEvent>
    <SessionEvent time="2007-01-31T02:00:49Z" type="endSegment">
      <Spot>
        <core:Content/>
        <PlayPositions>
          <PlayPositionEnd scale="-2">
            <NPTOffset>0</NPTOffset>
          </PlayPositionEnd>
        </PlayPositions>
      </Spot>
    </SessionEvent>
  </Events>
  <SpotScopedEvents>
    <Spot>
      <core:Content>
        <core:AssetRef providerID="advertiser3.com"
                      assetID="ADVR3000123456789001"/>
        <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C94</core:Tracking>
      </core:Content>
    </Spot>
    <Events>
      <PlacementStatusEvent time="2007-01-31T02:00:49Z"
                           type="startPlacement">
        <Spot>
          <core:Content/>
          <PlayPositions>
            <PlayPositionStart scale="-2">
              <NPTOffset>30</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Spot>
      </PlacementStatusEvent>
      <ViewerEvent time="2007-01-31T02:00:51Z" type="play"
                  eventSource="remote">
        <Spot>
          <core:Content/>
          <PlayPositions>
            <PlayPositionStart scale="1">
              <NPTOffset>26</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Spot>
      </ViewerEvent>
      <PlacementStatusEvent time="2007-01-31T02:00:55Z"
                           type="endPlacement">
        <Spot>
          <core:Content/>
          <PlayPositions>
            <PlayPositionEnd scale="1">
              <NPTOffset>30</NPTOffset>
            </PlayPositionEnd>
          </PlayPositions>
        </Spot>
      </PlacementStatusEvent>
    </Events>
  </SpotScopedEvents>
  <Events>
    <SessionEvent time="2007-01-31T02:00:55Z" type="startSegment">
      <Spot>
        <core:Content/>
        <PlayPositions>
          <PlayPositionStart scale="1">
            <NPTOffset>0</NPTOffset>
          </PlayPositionStart>
        </PlayPositions>
      </Spot>
    </SessionEvent>
    <SessionEvent time="2007-01-31T02:20:55Z" type="endSegment">
      <Spot>
        <core:Content/>
        <PlayPositions>
          <PlayPositionEnd scale="1">
            <NPTOffset>1200</NPTOffset>
          </PlayPositionEnd>
        </PlayPositions>
      </Spot>
    </SessionEvent>
  </Events>
  <SpotScopedEvents>
    <Spot>
      <core:Content>
        <core:AssetRef providerID="advertiser1.com"
                       assetID="ADVR1000123456789002"/>
        <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C96</core:Tracking>
      </core:Content>
    </Spot>
    <Events>
      <!--Post-roll ad #1: The 60 second ad was played from start to finish with 2 rewinds
      inside of the ad. The rewinds never exit the ad. The first rewind occurs
      52 seconds into the ad.-->
      <PlacementStatusEvent time="2007-01-31T02:20:55Z"
                           type="startPlacement">
        <Spot>
          <core:Content/>
          <PlayPositions>
            <PlayPositionStart scale="1">
              <NPTOffset>0</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Spot>
      </PlacementStatusEvent>
      <!--Begin rewinding at 1x speed for 2 seconds going back 2 seconds in 1x play time.
      The user then goes to the 2x rewind speed.-->
      <ViewerEvent time="2007-01-31T02:21:47Z" type="rewind"
                  eventSource="remote">
        <Spot>
          <core:Content/>
          <PlayPositions>
            <PlayPositionStart scale="-1">
              <NPTOffset>52</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Spot>
      </ViewerEvent>
      <!--Rewind at 2x speed for 23 seconds going back 46 seconds in 1x play time.-->
      <ViewerEvent time="2007-01-31T02:21:49Z" type="rewind"
                  eventSource="remote">
        <Spot>
          <core:Content/>
          <PlayPositions>
            <PlayPositionStart scale="-2">
              <NPTOffset>50</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Spot>
      </ViewerEvent>
      <ViewerEvent time="2007-01-31T02:22:12Z" type="play"
                  eventSource="remote">
        <Spot>
          <core:Content/>
          <PlayPositions>
            <PlayPositionStart scale="1">
              <NPTOffset>4</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Spot>
      </ViewerEvent>
      <!--After playing for 20 seconds, rewind at 1x speed for 10 seconds going
      back 10 seconds in 1x play time and then play for the remainder of the ad.-->
      <ViewerEvent time="2007-01-31T02:22:32Z" type="rewind"
                  eventSource="remote">
        <Spot>
          <core:Content/>
          <PlayPositions>
            <PlayPositionStart scale="-1">
              <NPTOffset>24</NPTOffset>
            </PlayPositionStart>
          </PlayPositions>
        </Spot>
      </ViewerEvent>
      <ViewerEvent time="2007-01-31T02:22:42Z" type="play"
                  eventSource="remote">
        <Spot>
          <core:Content/>
          <PlayPositions>
            <PlayPositionStart scale="1">
              <NPTOffset>14</NPTOffset>
            </PlayPositionStart>
```

```
        </PlayPositions>
      </Spot>
    </ViewerEvent>
    <PlacementStatusEvent time="2007-01-31T02:23:28Z"
                          type="endPlacement">
      <Spot>
        <core:Content/>
        <PlayPositions>
          <PlayPositionEnd scale="1">
            <NPTOffset>60</NPTOffset>
          </PlayPositionEnd>
        </PlayPositions>
      </Spot>
    </PlacementStatusEvent>
  </Events>
</SpotScopedEvents>
<SpotScopedEvents>
  <Spot>
    <core:Content>
      <core:AssetRef providerID="advertiser2.com"
                     assetID="ADVR2000123456789002"/>
      <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C97</core:Tracking>
    </core:Content>
  </Spot>
  <Events>
    <!--Post-roll ad #2: The ad was was started and then the session was stopped.-->
    <PlacementStatusEvent time="2007-01-31T02:23:28Z"
                          type="startPlacement">
      <Spot>
        <core:Content/>
        <PlayPositions>
          <PlayPositionStart scale="1">
            <NPTOffset>0</NPTOffset>
          </PlayPositionStart>
        </PlayPositions>
      </Spot>
    </PlacementStatusEvent>
    <ViewerEvent time="2007-01-31T02:23:45Z" type="stop"
                 eventSource="remote">
      <Spot>
        <core:Content/>
        <PlayPositions>
          <PlayPosition>
            <NPTOffset>17</NPTOffset>
          </PlayPosition>
        </PlayPositions>
      </Spot>
     </ViewerEvent>
    </Events>
   </SpotScopedEvents>
   <!-- Post-roll ad #3 did not play since the session is terminated. -->
   <Events>
    <!--End of session notification.-->
    <SessionEvent time="2007-01-31T02:23:45Z" type="endSession">
     <core:StatusCode class="3" detail="3008">
      <core:Note>Session terminated by user.</core:Note>
     </core:StatusCode>
    </SessionEvent>
    <!--Post-roll ad #3 did not play and the ADM in cleaning up the session notifies the
ADS.
     This notification is not required and this is an ADM implementation choice.-->
    <PlacementStatusEvent time="2007-01-31T02:23:45Z" type="status">
     <core:StatusCode class="1" detail="3011">
      <core:Note>Placement not executed.</core:Note>
      <core:Note>Asset not played. Session ended before
       playout.
      </core:Note>
     </core:StatusCode>
     <Spot>
      <core:Content>
       <core:AssetRef providerID="advertiser3.com"
               assetID="ADVR3000123456789002"/>
       <core:Tracking>6F05FDC7-6C60-0030-B2B7-B68DAF952C98
       </core:Tracking>
      </core:Content>
     </Spot>
    </PlacementStatusEvent>
   </Events>
  </EntertainmentScopedEvents>
 </PlayData>
 <PlayData
   identityADS="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5">
  <Events>
   <!--The ADM implementation decided to put this event is a separate PlayData
     element. This not required and it could have immediately followed the
     last PlacementStatusEvent above.-->
   <PlacementStatusEvent time="2007-01-31T02:23:50Z" type="endAll"
              messageRef="5B8ECCED-979B-421E-8A6E-E2192B11516A"/>
  </Events>
 </PlayData>
</PlacementStatusNotification>
```

Example 71

APPENDIX C. WSDL (NORMATIVE)

SCTE 130 Part 3 (this document) includes two separate WSDL document for the ADM and the ADS interfaces. See the individual WSDL document for details regarding the wsdl:portType definitions for the service endpoints along with the service definitions, binding types, and input/output parameter mappings. Table 1 specifies the normative WSDL XML namespace using the prefixes ‘admwsdl’ and ‘adswsdl’. SCTE 130 Part 7 provides additional WSDL specification details. See [SCTE130-7] for more information. See the normative Part 3 WSDL documents for all other details.

APPENDIX D. THE PLACEMENT OPPORTUNITY DATA MODEL (NORMATIVE)

D.1 Scope (NORMATIVE)

This appendix defines the Placement Opportunity Data Model (PODM) consistent with other parts of SCTE 130 Digital Program Insertion—Advertising Systems Interfaces specifications. The PODM defines a concrete representation of a Placement Opportunity (typically used in association with advertisements) and expressed as XML elements and attributes exchangeable by collaborating SCTE 130 logical services.

The PODM expresses the features, the attributes and the constraints for each Placement Opportunity, appropriate for the platform, the rights, and the policies including those of the content in which it exists. These Placement Opportunities *may* or *may* not be content specific and, in such cases, attributes and constraints *may* vary by network, geographic region, or other content distribution dimension.

The PODM defines a standardized data model for accessing Placement Opportunity information and this data model *may* be used in conjunction with a messaging interface, such as SCTE 130 Part 5, [SCTE130-5] the Placement Opportunity Information Service (POIS). An implementation *may* choose not to use this data model.

D.2 Overview of the Placement Opportunity Data Model (INFORMATIVE)

D.2.1 What is a Placement Opportunity?

A Placement Opportunity is a potentially constrained location — relative to digital content — where ad placement or content alterations *may* occur. The alterations *may* include insertions, replacements, or deletions of content, and *may* be whole or partial for both media stream duration or presentation characteristics and effects on the media stream such as a graphical overlay. The locations contain the opportunity for content insertion, traditionally referred to as avails [see SCTE 35] for linear video content. The new term "Placement Opportunity" was chosen to capture a more broadly defined concept including classic linear avails but also Video On Demand (VOD) bumper ads, banner images, overlay images, barker ads on user guides and generally any opportunity to advertise. This information *may* be tightly associated to content, or more generally stipulated for particular cable or network systems.

This specification provides a generic and extensible framework for Placement Opportunities and provides details on specific well-known use cases such as classic linear advertising, VOD advertising, interactive and graphical overlays. In these particular use cases and for future extensions, the specification provides for Placement Opportunities to contain the following subcomponents:

- Some description of the entertainment content. In classic linear, this concept is the underlying broadcast channel. In VOD, this notion is the requested movie or content. Other examples *may* contain interactive applications, games.

- Some description of the location for the advertisement. The location *may* be temporal, e.g. classic linear, or spatial, e.g. graphical overlays.

- Some description of other opportunity constraints such as the duration of the advertisement or the number of advertisements that can be grouped together.

A break in entertainment content provides an opportunity for placements that require specification. For non-linear content, the opportunity *may* be to expand or contract (in the case of replacement or removal) the timeline by inserting or removing content between segments. Multiple Placement Opportunities *may* be available between the two segments of non-linear entertainment content. For both linear and non-linear content, the Placement Opportunity *may* be to replace or overlay content already present. Each of these Placement Opportunities *may* allow for certain interactive, addressable, or other types of placement.

The existence of a Placement Opportunity does not imply physical presence of referenced content assets, or that such streams are guaranteed to exist. Placement Opportunities *may* be localized. As such, Placement Opportunities in one geographic area, demographic group, day part, window, or other dimension, *may* be different even for the same content.

A Placement Opportunity *may* have multiple owners, rights and business terms dictating the types of placements that *may* occur, and identifying the systems making such decisions. Some Placement Opportunities *may* be split, that is, multiple placements *may* be made within a single opportunity. Placement Opportunities *may* or *may* not need to be filled. Also, Placement decisions *may* result in additional Placement Opportunities.

A Placement Opportunity contains the required information for placement decisions. The placement decision *may* place content (e.g., advertisement) that complies with the described Placement Opportunity and its restrictions, if any, and should be available for validation at decision time.

How the Placement Opportunity information is obtained is outside the scope of this document.

The upper portion of Figure 102 depicts Placement Opportunities relative to entertainment content. The lower half of the image shows the same capabilities, but this time within a rigid linear entertainment content break.

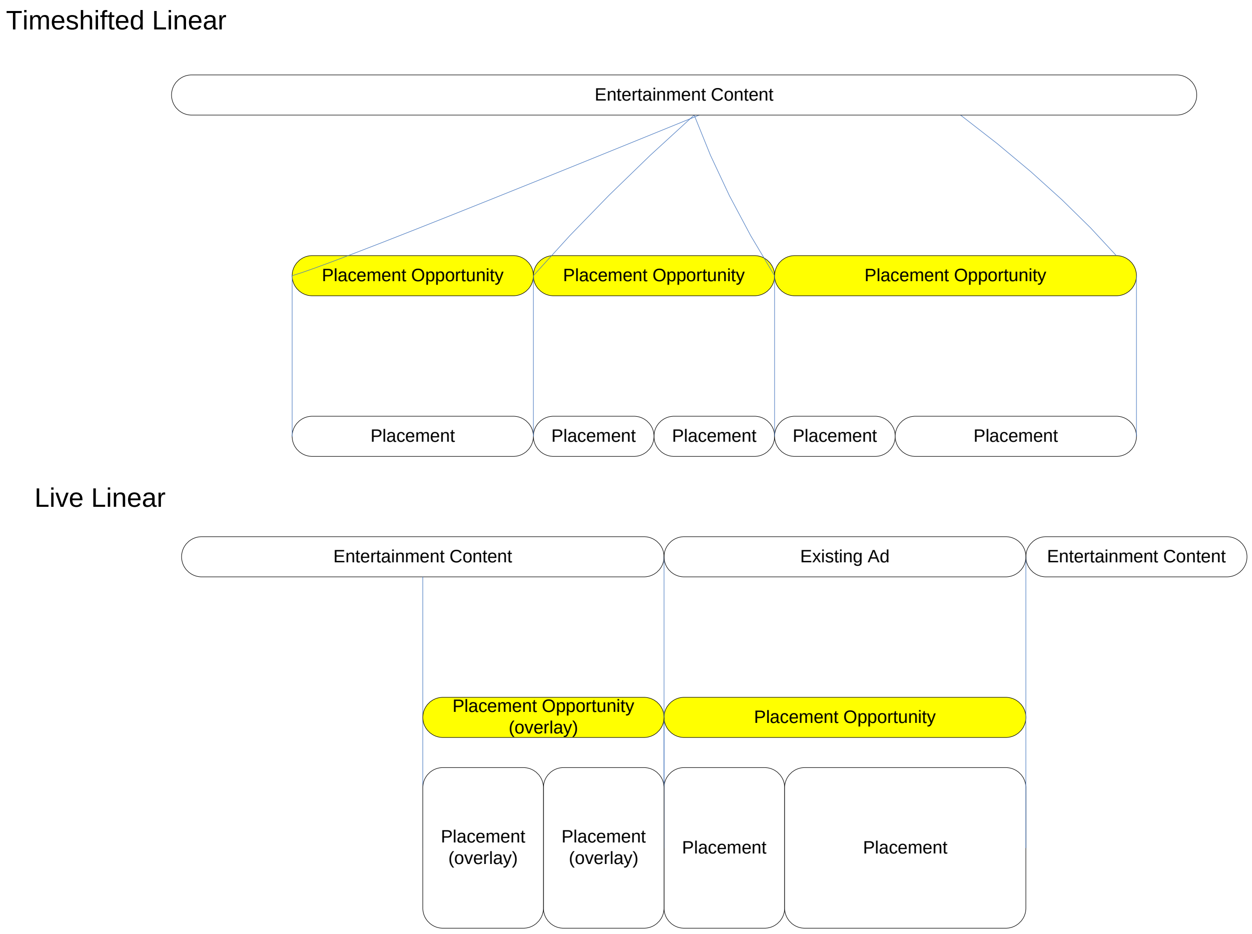


Figure 102: Visual Representation of Placement Opportunities

D.3 XML NAMESPACES (NORMATIVE)

This appendix uses the ‘podm’ prefix, as described in Table 1, for the interface associated with the specific XML namespace URI that ***shall*** be used by all implementations. Table 1 lists the prefix, the corresponding namespace, and a description of the defining specification used herein.

Unless otherwise stated, all references to XML elements illustrated in this document are from the ‘podm’ namespace. Elements from other namespaces will be prefixed with the name of the external namespace, e.g. <core:XXXX>.

The PODM specified herein leverages multiple XML namespaces. Where this document and any other specification differ, the PODM elements ***shall*** take precedence.

D.4 Messages Containing The Placement Opportunity Data Model (NORMATIVE)

Any SCTE 130 message interface *may* utilize the PODM. The message interfaces utilizing the PODM are outside the scope of this specification. Refer to the individual messaging specifications for additional information.

D.4.1 PODM NameSpaceVersion

The Placement Opportunity Data Model ***shall*** be communicated via the NameSpaceVersion Element. The NameSpaceVersion@namespace ***shall*** be set to “http://www.scte.org/schemas/130-3/2013/adm/podm" and the specific PODM version ***shall*** be passed in NameSpaceVersion@version. (See Section 14.20). The PODM version numbers are listed in Table 24: PODM Version Numbers.

Version	Description
1.1	This identifies a pre-PODM release of SCTE 130-3.
1.2	This identifies the PODM data model described in this section.

Table 24: PODM Version Numbers

A ”1.4” capable ADS signals its facilities and determines an ADM endpoint’s operating ability by sending a ListADMServicesRequest message with the @version attribute set to “1.4”. The ADM responds in one of the following ways and the ADS/ADM pair ***shall*** operate as defined:

- A “1.1” capable ADM responds with an error and a core:StatusCode/@detail attribute having the value core:UnsupportedProtocol or the ADM responds with a successful ListADMServicesResponse message having the @version attribute set to the value “1.1”, which indicates the ADM is operating as a “1.1” ADM only. These scenarios indicate the ADM is a “1.1” ADM only and does not support SCTE 130-3 version "1.4", “1.3” or PODM “1.2” or greater message exchanges. All messages exchanges ***shall*** set the @version attribute to “1.1” and ***shall*** comply with the 1.1 version of this specification. The PODM “1.2” (i.e., the PlacementOpportunityV2 element or any other element defined within this appendix) ***shall not*** be included in any message.

- A “1.2” capable ADM responds with an error and a core:StatusCode/@detail attribute having the value core:UnsupportedProtocol or the ADM responds with a successful ListADMServicesResponse message having the @version attribute set to the value “1.2” indicating the ADM is operating as a “1.2” ADM and “1.2” PODM. All message exchanges ***shall*** use the value “1.2” for the @version attribute. (Note: This return value should not be interpreted to mean the ADM is simultaneously operating as both a “1.1” AND “1.2” ADM.

The dual version operating characteristic is optional, implementation specific and outside the scope of this specification. The ADM or ADS supplied endpoints, i.e., the core:Address elements, contain endpoints capable of handling version “1.2” messages and elements.) The “1.2” PODM (i.e., the PlacementOpportunityV2 element or any other element defined within this appendix) *may* be included in any message as allowed by the normative schema and semantics defined herein.

- A “1.3” capable ADM responds with a successful ListADMServicesResponse message having the @version attribute set to the value “1.3” indicating the ADM is operating as a “1.3” ADM. All message exchanges ***shall*** use the value “1.3” for the @version attribute. The ADM ***shall*** also return a NameSpaceVersion Element for PODM with NameSpaceVersion@version attribute set to the value of “1.1” or “1.2” indicating the ADM’s PODM’s capabilities. A ListADMServicesResponse message *may* return a NameSpaceVersion Element for each of version “1.1” and “1.2”. All future messages carrying PODM elements and/or attributes ***shall*** be compliant with the PODM NameSpaceVersion@version value passed in the ADSRegistrationRequest message. See section 11.3 for additional information.

- A "1.4" capable ADM responds similarly to a "1.3" capable ADM regarding PODM. In addition, a "1.4" capable ADM indicates it ***shall not*** use the eliminated EntertainmentNPT element nor shall it use the eliminated SpotNPT element. Instead, it shall use the PlayPositions element and its children.

A “1.1” capable ADS, signaled by the @version attribute having a value of “1.1”, communicating with a “1.3” or greater capable ADM *may* observe the following:

- An ADM response or acknowledgement message indicating an error as the core:StatusCode/@detail attribute is set to the value core:UnsupportedProtocol. This detail code indicates the ADM or the queried ADM endpoint does not support “1.1” messages (i.e., the ADM or the ADM endpoint is a “1.2” or greater interface).

- Any successful ADM response message which contains core:Address elements ***shall*** restrict the endpoints listed to those which are PODM “1.1” capable.

When a “1.3” or greater ADM accepts a “1.1” ADS registration, the ADM ***shall*** guarantee the message exchanges with the ADS do not use the PODM “1.2” (i.e., the PlacementOpportunityV2 element or any other element defined within this appendix) in any message. How a “1.3” ADM handles this translation is outside the scope of this standard. In addition, a "1.4" ADM ***shall not*** use the EntertainmentNPT or SpotNPT and it ***shall*** use the replacement PlayPositions element and its children.

A “1.2” capable ADS, signaled by the @version attribute having a value of “1.2”, communicating with a “1.3” or greater capable ADM *may* observe the following:

- An ADM response or acknowledgement message indicating an error as the core:StatusCode/@detail attribute is set to the value core:UnsupportedProtocol. This detail code indicates the ADM or the queried ADM endpoint does not support “1.2” messages (i.e., the ADM or the ADM endpoint is a “1.3” or greater interface).

- Any successful ADM response message which contains core:Address elements ***shall*** restrict the endpoints listed to those which are PODM “1.2” capable.

When a “1.3” ADM accepts a “1.2” ADS registration, the ADM ***shall*** guarantee the message exchanges with the ADS use the PODM “1.2” (i.e., the PlacementOpportunityV2 element or any other element defined within this appendix) in any message. In addition, a "1.4" ADM ***shall*** not use the EntertainmentNPT or SpotNPT and it ***shall*** use the replacement PlayPositions element and its children.

D.5 PODM Attributes Types and Common Types (NORMATIVE)

The PODM includes all attribute types and common types as defined by the normatively included reference specifications. Additionally, the following are defined by this specification.

D.5.1 poGroupIndexAttrType Attribute Type

The poGroupIndexAttrType, a xsd:unsignedInteger typically referred to as the poGroupIndex attribute, allows one or more Placement Opportunities to be identified as members of a related or common group. The value zero indicates not used or no group. The index value typically starts at one and increment by one. Placement Opportunity grouping criteria are outside the scope of this specification.

D.5.2 unitsIdentificationAttrType Attribute Type

The unitsIdentificationAttrType is a core:nonEmptyStringType identifier which *may* have one of the values from Table 3. The supplied string provides measurement context for specified quantitative measured values.

Value	Description
pixels	Measurement units defined as display resolution location pixels.
percentage	Measurement units defined as a resolution independent percentage of space.

…	User defined measurement units that are outside of this specification's scope. The string ***shall*** be prefixed with the text "private:".

Table 25: unitsIdentificationAttrType Values

D.6 PODM Element Details (NORMATIVE)

The PODM leverages a subset of elements from several SCTE 130 specifications along with new elements in defining the data model composition. The new PODM elements are listed in Table 26 and are detailed in subsequent document sections.

Element	Description
ContentGroupFilter	Generic means of describing an asset set using a gis:BasicQueryFilter or gis:AdvancedQueryFilter element.
Interactive	Provides an extension for interactive specific information.
LinearAvailBindingV2	The revised SCTE 130-3 LinearAvailBinding element extended to supply additional functionality.
MenuCategory	Displayable menu folder identifier.
OpportunityBindingV2	The revised SCTE 130-3 OpportunityBinding element extended to supply additional functionality.
OpportunityConstraintsV2	The rights and constraints of the placements within this Placement Opportunity. This element is an extension of the adm:OpportunityConstraints element.
PlacementOpportunityTerms	The business terms relevant to the Placement Opportunity.
PlacementOpportunityV2	The revised SCTE 130-3 Placement Opportunity definition extended to supply new and additional functionality. This element is an extension of the adm:PlacementOpportunity element.
Plane	Visibility details relevant to a Placement Opportunity's screen real estate.
Programming	The referenced content stream containing the Placement Opportunity.
Size	Screen real estate space identification.
SpatialPosition	Screen real estate consumable by a Placement Opportunity.
TopCorner	Coordinate details specific identifying a spatial position.

Table 26: PODM Elements

D.6.1 ContentGroupFilter

The ContentGroupFilter element provides a generic means of describing a set of assets, feeds, or other form of content. It *may* use either the gis:BasicQueryFilter or the gis:AdvancedQueryFilter as a means of describing the set. If this element is present, the containing Placement Opportunity is available as described for all content in the set description. If a service endpoint were to query for content within this set, the described Placement Opportunity *may* be present in the result set.

The XML schema diagram for this element is as follows:

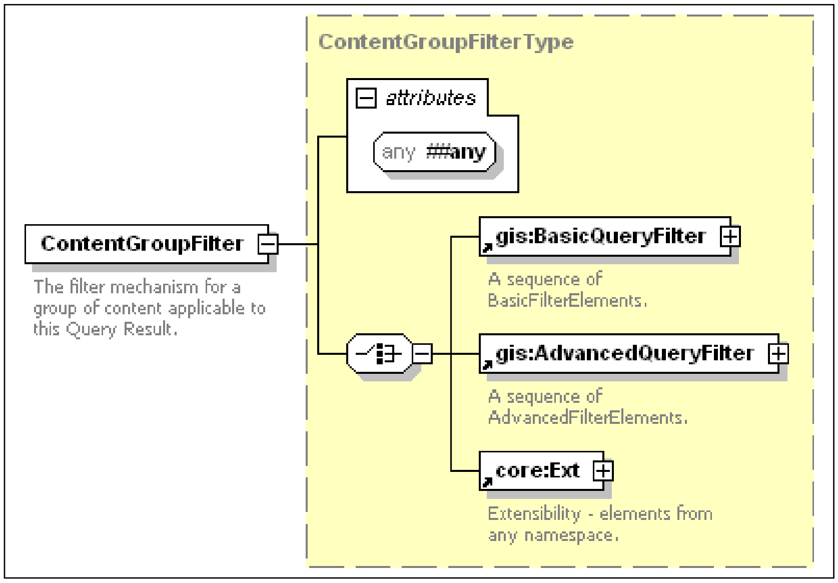

ContentGroupFilterType
attributes
any ##any
ContentGroupFilter
The filter mechanism for a group of content applicable to this Query Result.
gis:BasicQueryFilter
A sequence of BasicFilterElements.
gis:AdvancedQueryFilter
A sequence of AdvancedFilterElements.
core:Ext
Extensibility - elements from any namespace.

Figure 103: ContentGroupFilter Element Schema

The ContentGroupFilter element semantics are as follows:

@##any [Optional] — Additional attributes from any namespace.

gis:BasicQueryFilter [Choice] — Describes the set of content using a gis:BasicQueryFilter element. See [SCTE130-8] for additional information.

gis:AdvancedQueryFilter [Choice] — Describes the set of content using the gis:AdvancedQueryFilter element. See [SCTE130-8] for additional information.

core:Ext [Choice] — Any elements from other namespace. See [SCTE130-2] for additional information.

D.6.2 Interactive

The Interactive element is an extension point for all definitions of interactive rights, if any, that are allowed. Such examples *may* include “telescope”, “rfi”, and "bookmark". Constructs *may* be used to specify "pause ads" or restricted trick play. All such definitions are intentionally left to the implementation.

The XML schema diagram for this element is as follows:

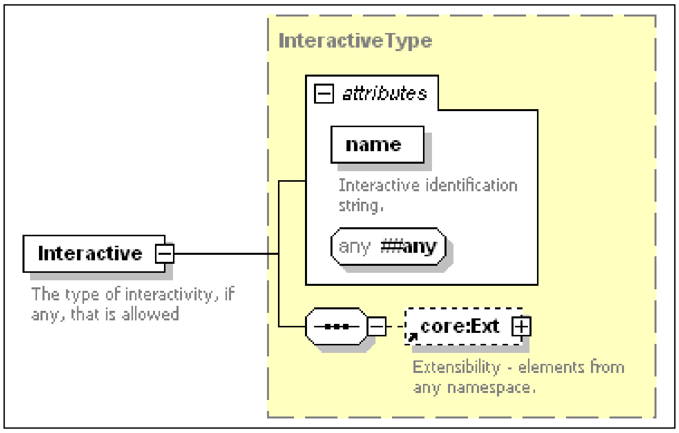

InteractiveType
attributes
name
Interactive identification string.
any ##any
Interactive
The type of interactivity, if any, that is allowed
core:Ext
Extensibility - elements from any namespace.

Figure 104: Interactive Element Schema

The Interactive element semantics are as follows:

@name [Required, core:nonEmptyStringType] — An identification string for this interactive element. See [SCTE130-2] for additional information.

@##any [Optional] — Additional attributes from any namespace.

core:Ext [Optional] — Any elements from any namespace. See [SCTE130-2] for additional information.

D.6.3 LinearAvailBindingV2

The LinearAvailBindingV2 element extends the adm:LinearAvailBindingType by including the @poGroupIndex attribute enabling Placement Opportunities to be associated with the index value acting as the group identifier.

The XML schema diagram for the LinearAvailBindingV2 element is illustrated in Figure 105.

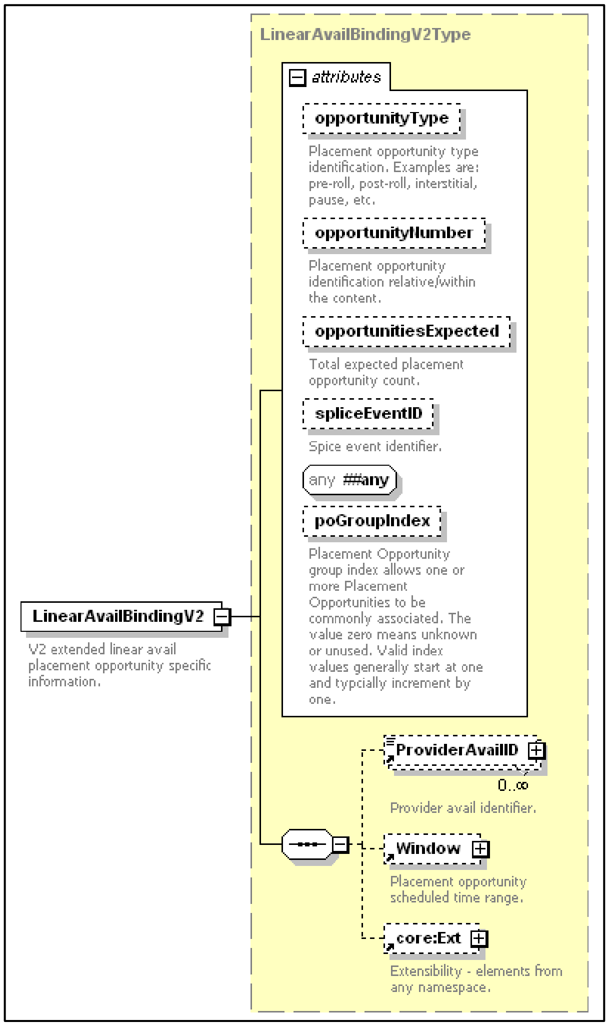

LinearAvailBindingV2Type
attributes
opportunityType
Placement opportunity type identification. Examples are: pre-roll, post-roll, interstitial, pause, etc.
opportunityNumber
Placement opportunity identification relative/within the content.
opportunitiesExpected
Total expected placement opportunity count.
spliceEventID
Spice event identifier.
any ##any
poGroupIndex
Placement Opportunity group index allows one or more Placement Opportunities to be commonly associated. The value zero means unknown or unused. Valid index values generally start at one and typcially increment by one.
LinearAvailBindingV2
V2 extended linear avail placement opportunity specific information.
ProviderAvailID
0..∞
Provider avail identifier.
Window
Placement opportunity scheduled time range.
core:Ext
Extensibility - elements from any namespace.

Figure 105: LinearAvailBindingV2 Element Schema

Beyond the adm:LinearAvailBinding element's original definition, see Section 14.15 for additional information, the LinearAvailBindingV2 element *may* contain the following attribute:

@poGroupIndex [Optional, poGroupIndexAttrType] — Placement Opportunity group index allows one or more Placement Opportunities to be identified as associated. See Section D.5.1 for additional information.

D.6.4 MenuCategory

The MenuCategory element contains a displayable menu folder identifier.

The XML schema diagram for this element is as follows:

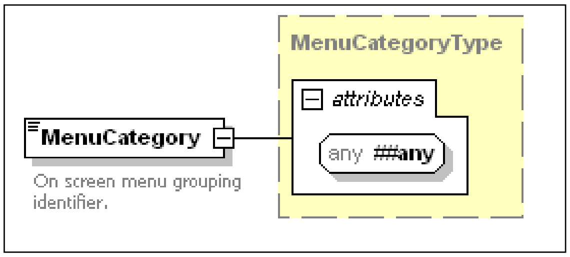


Figure 106: MenuCategory Element Schema

The MenuCategory element semantics are as follows:

@##any [Optional] — Additional attributes from any namespace.

The MenuCategory element's value is a core:nonEmptyStringType containing the displayable menu folder identifier.

D.6.5 OpportunityBindingV2

The OpportunityBindingV2 element extends the adm:OpportunityBindingType by including the @poGroupIndex attribute enabling Placement Opportunities to be associated with the index value acting as the group identifier.

The XML schema for the OpportunityBindingV2 element is illustrated in Figure 107.

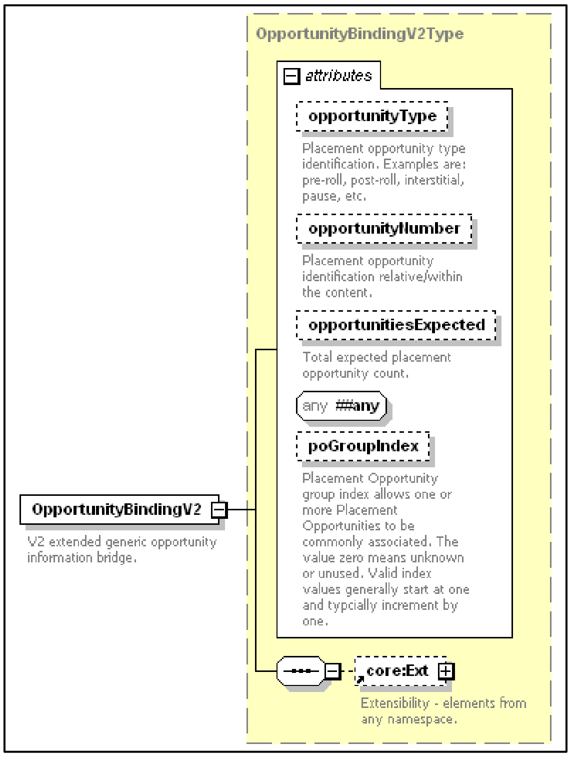

OpportunityBindingV2Type
attributes
opportunityType
Placement opportunity type identification. Examples are: pre-roll, post-roll, interstitial, pause, etc.
opportunityNumber
Placement opportunity identification relative/within the content.
opportunitiesExpected
Total expected placement opportunity count.
any ##any
poGroupIndex
Placement Opportunity group index allows one or more Placement Opportunities to be commonly associated. The value zero means unknown or unused. Valid index values generally start at one and typcially increment by one.
OpportunityBindingV2
V2 extended generic opportunity information bridge.
core:Ext
Extensibility - elements from any namespace.

Figure 107: OpportunityBindingV2 Element Schema

Beyond the adm:OpportunityBinding element's original definition, see Section 14.24 for additional information, the OpportunityBindingV2 element *may* contain the following attribute:

@poGroupIndex [Optional, poGroupIndexAttrType] — Placement Opportunity group index allows one or more Placement Opportunities to be identified as associated. See Section D.5.1 for additional information.

D.6.6 OpportunityConstraintsV2

The OpportunityConstraintsV2 element contains the rights and constraints of placements within a Placement Opportunity. This element extends the adm:OpportunityConstraintsType.

The XML schema diagram for this element is as follows:

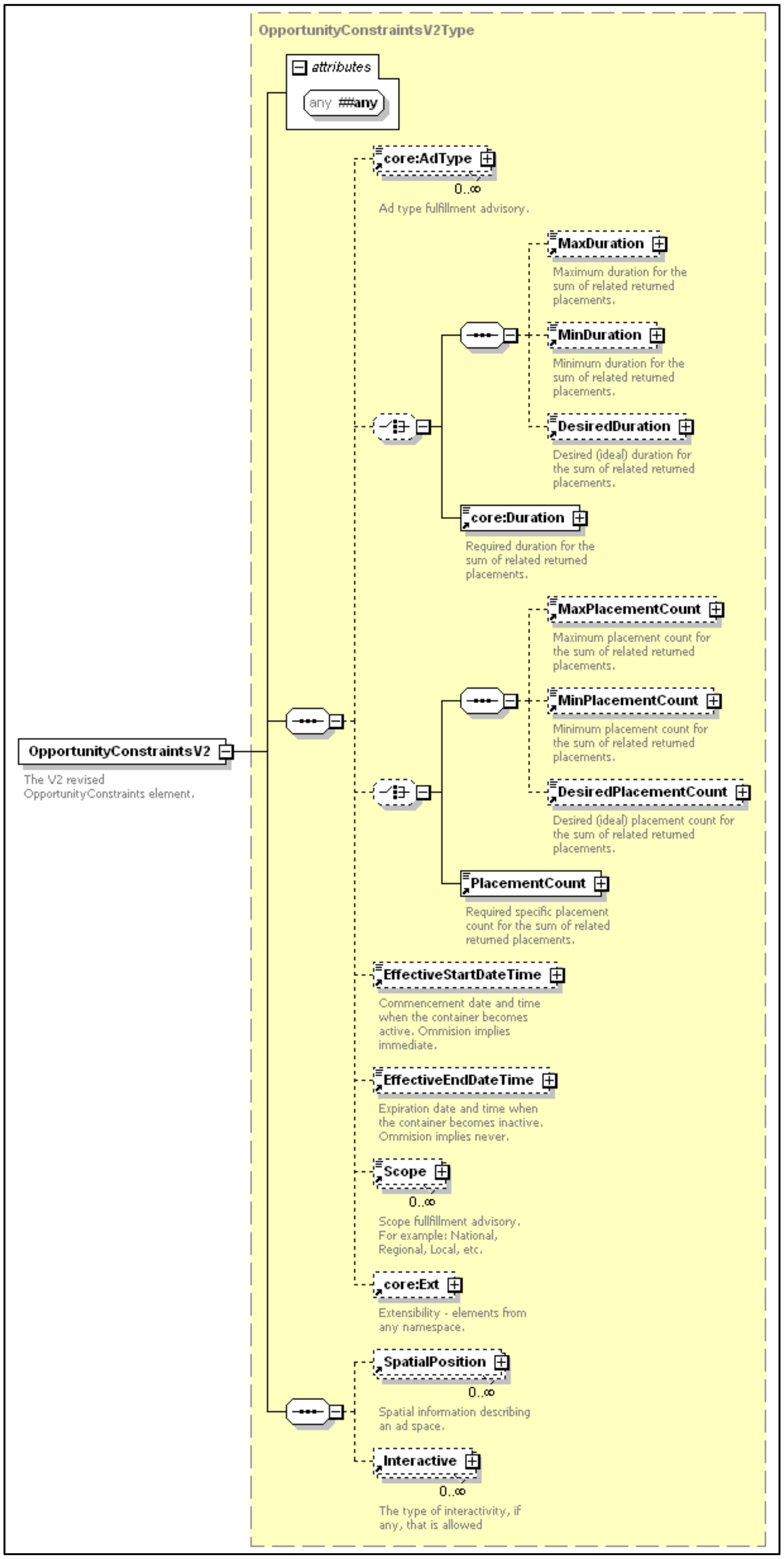

OpportunityConstraintsV2Type
attributes
any ##any
core:AdType
0..∞
Ad type fulfillment advisory.
MaxDuration
Maximum duration for the sum of related returned placements.
MinDuration
Minimum duration for the sum of related returned placements.
DesiredDuration
Desired (ideal) duration for the sum of related returned placements.
core:Duration
Required duration for the sum of related returned placements.
MaxPlacementCount
Maximum placement count for the sum of related returned placements.
MinPlacementCount
Minimum placement count for the sum of related returned placements.
DesiredPlacementCount
Desired (ideal) placement count for the sum of related returned placements.
PlacementCount
Required specific placement count for the sum of related returned placements.
OpportunityConstraintsV2
The V2 revised OpportunityConstraints element.
EffectiveStartDateTime
Commencement date and time when the container becomes active. Ommision implies immediate.
EffectiveEndDateTime
Expiration date and time when the container becomes inactive. Ommision implies never.
Scope
0..∞
Scope fullfillment advisory. For example: National, Regional, Local, etc.
core:Ext
Extensibility - elements from any namespace.
SpatialPosition
0..∞
Spatial information describing an ad space.
Interactive
0..∞
The type of interactivity, if any, that is allowed

Figure 108: OpportunityConstraintsV2 Element Schema

Beyond the adm:OpportunityConstraints element's original definition (Section 14.25) , the OpportunityConstraintsV2 element *may* contain one or more of the following elements:

SpatialPosition [Optional] — Zero or more descriptions of the position and portion of the opportunity relative to screen real estate. See Section D.6.12 for additional information.

Interactive [Optional] — Zero or more interactive characteristic descriptions. See Section D.6.2 for additional information.

D.6.7 PlacementOpportunityTerms

The PlacementOpportunityTerms element contains the business terms for how this Placement Opportunity should be billed, what, if any, revenue splits are, and who *may* have exclusive rights to sell the opportunity.

The XML schema diagram for this element is as follows:

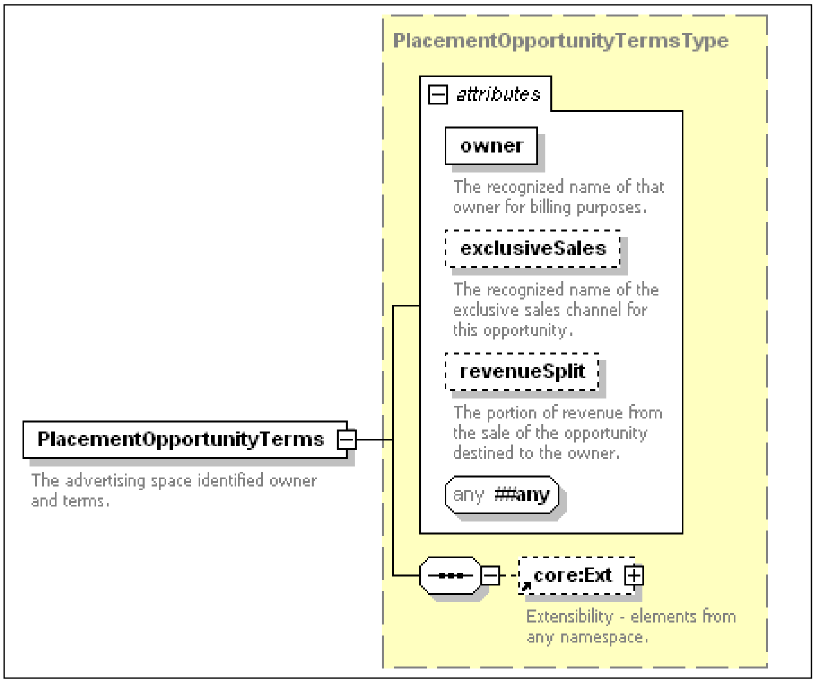

PlacementOpportunityTermsType
attributes
owner
The recognized name of that owner for billing purposes.
exclusiveSales
The recognized name of the exclusive sales channel for this opportunity.
revenueSplit
The portion of revenue from the sale of the opportunity destined to the owner.
any ##any
PlacementOpportunityTerms
The advertising space identified owner and terms.
core:Ext
Extensibility - elements from any namespace.

Figure 109: PlacementOpportunityTerms Element Schema

The following are the PlacementOpportunityTerms element's semantics.

@owner [Required, core:nonEmptyString] — The recognized owner name for billing purposes.

@exclusiveSales [Optional, xsd:boolean] — Set to true if the @owner attribute specified entity has exclusive rights to sell this Placement Opportunity. If the attribute is omitted, the default value ***shall*** be false.

@revenueSplit [optional, xsd:nonNegativeInteger] — The percentage portion of sale revenue for the opportunity destined to the owner identified by the @owner attribute. The attributes range is 0 to 100. If the attribute is omitted, the default value ***shall*** be 100 (or all revenue).

@##any [Optional] — Additional attributes from any namespace.

core:Ext [Optional] — Any elements from any namespaces. See [SCTE130-2] for additional information.

D.6.8 PlacementOpportunityV2

The PlacementOpportunityV2 element is the PODM root element. This element defines the space and is the lead for all static and dynamic rules of what a placement *may* be, where in the stream it can exist, and what can be done within that opportunity. In effect, it defines the space and the capabilities for a given potential content stream. The PlacementOpportunityV2 element *may* be substituted for the adm:PlacementOpportunity element as it extends the currently defined adm:PlacementOpportunity element.

The XML schema diagram for this element is illustrated in Figure 110.

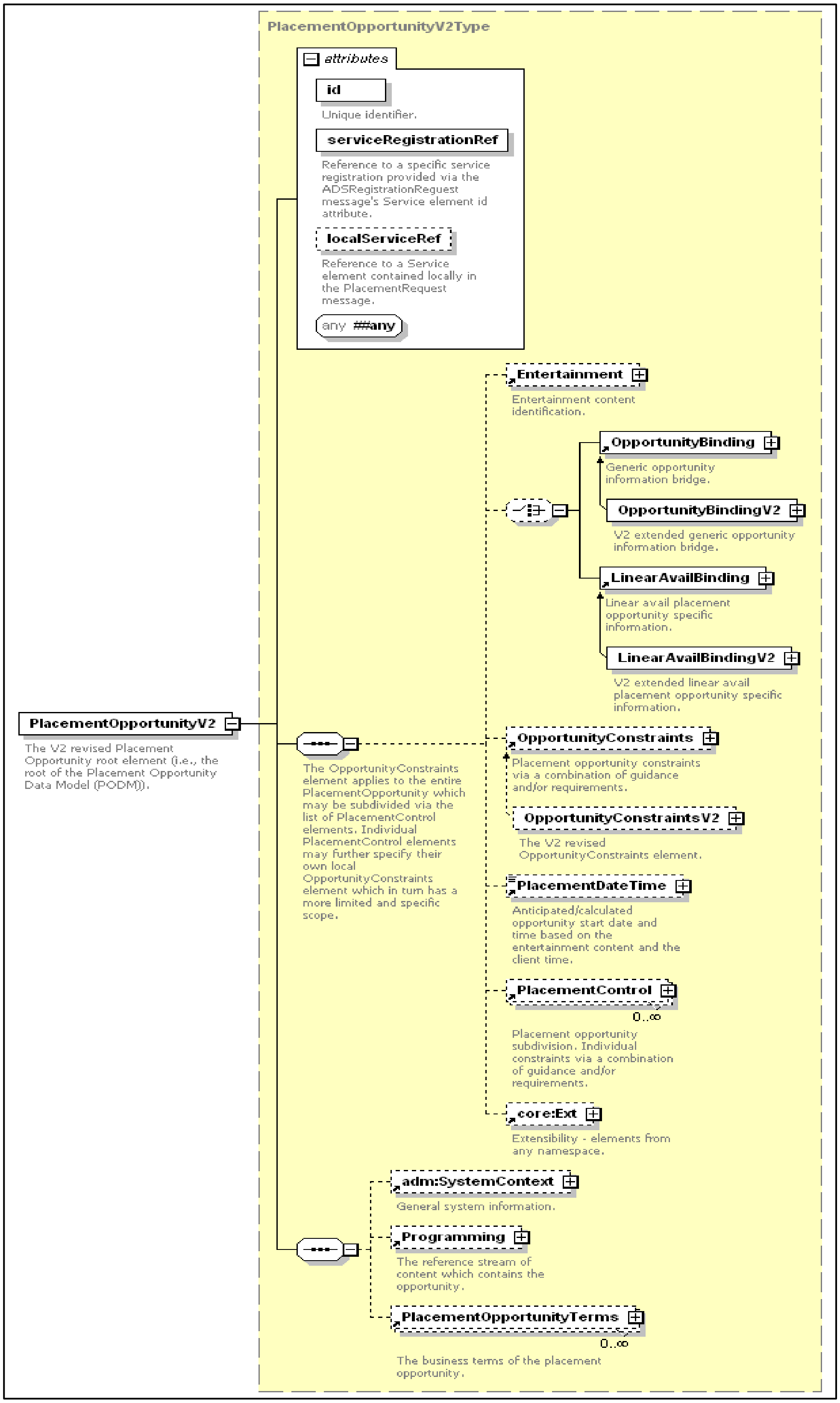

PlacementOpportunityV2Type
attributes
id
Unique identifier.
serviceRegistrationRef
Reference to a specific service registration provided via the ADSRegistrationRequest message's Service element id attribute.
localServiceRef
Reference to a Service element contained locally in the PlacementRequest message.
any ##any
Entertainment
Entertainment content identification.
OpportunityBinding
Generic opportunity information bridge.
OpportunityBindingV2
V2 extended generic opportunity information bridge.
LinearAvailBinding
Linear avail placement opportunity specific information.
LinearAvailBindingV2
V2 extended linear avail placement opportunity specific information.
PlacementOpportunityV2
The V2 revised Placement Opportunity root element (i.e., the root of the Placement Opportunity Data Model (PODM)).
The OpportunityConstraints element applies to the entire PlacementOpportunity which may be subdivided via the list of PlacementControl elements. Individual PlacementControl elements may further specify their own local OpportunityConstraints element which in turn has a more limited and specific scope.
OpportunityConstraints
Placement opportunity constraints via a combination of guidance and/or requirements.
OpportunityConstraintsV2
The V2 revised OpportunityConstraints element.
PlacementDateTime
Anticipated/calculated opportunity start date and time based on the entertainment content and the client time.
PlacementControl
0..∞
Placement opportunity subdivision. Individual constraints via a combination of guidance and/or requirements.
core:Ext
Extensibility - elements from any namespace.
adm:SystemContext
General system information.
Programming
The reference stream of content which contains the opportunity.
PlacementOpportunityTerms
0..∞
The business terms of the placement opportunity.

Figure 110: PlacementOpportunityV2 Element Schema

The following are the PlacementOpportunityV2 element's semantics:

@id [Required, core:idAttrType] — The @id attribute, also known as a Placement Opportunity identifier or POID, uniquely identifies the Placement Opportunity within the scope of the @identity attribute from the enclosing top level parent messaging element, and ***shall not*** be empty. See Section 14.26.1.1 and [SCTE130-2] for additional information.

@serviceRegistrationRef [Optional, core:idAttrType] —When present in ADM/ADS message exchanges, this attribute ***shall*** contain the value as defined bythis document. For other message exchanges where this attribute has insufficient context, the attribute may be omitted; if included in such cases, the attribute's value ***shall*** be "–". See Section 12.1 and [SCTE130-2] for additional information.

@localServiceRef [Optional, core:idAttrType] — For ADM/ADS message exchanges, this attribute ***shall*** contain the value as defined by this document. For other message exchanges where this attribute has insufficient context, the attribute ***shall not*** be used. See Section 12.1 and Section 13.1.4 for additional information.

@##any [Optional] — Additional attributes from any namespace.

adm:Entertainment [Optional] — The reference stream of content containing the Placement Opportunity. This element ***shall*** be used when only one core:Content element is present. If more than one core:Content element is required, the adm:Entertainment elements under the Programming element ***shall*** be utilized and this element ***shall*** be omitted. In the event both this adm:Entertainment element and the adm:Entertainment elements under the Programming element are present, this element ***shall*** take precedence. See Section 13.2.1 for additional information.

adm:OpportunityBinding [Optional] — The relative position and existence of the Placement Opportunity. See Section 14.24 for additional information.

OpportunityBindingV2 [Optional] — This enhanced version of the adm:OpportunityBinding element *may* only be present in the PlacementOpportunityV2 element. See Section D.6.5 for additional information.

adm:LinearAvailBinding [Optional] — The relative position and existence of the Placement Opportunity. This element ***shall*** be present when the Placement Opportunity applies to linear content. See Section 14.15 for additional information.

LinearAvailBindingV2 [Optional] — This enhanced version of the adm:LinearAvailBinding element *may* only be present in the PlacementOpportunityV2 element. See Section D.6.3 for additional information.

adm:OpportunityConstraints [Optional] — The original SCTE-130 defined element. See Section 14.25 for additional information. **OpportunityConstraintsV2 [Optional]** — This enhanced version of the adm:OpportunityConstraints element *may* only be present in the PlacementOpportunityV2 element. This element extends the current adm:OpportunityConstraints element by defining additional placement rights and constraints within this Placement Opportunity. See Section D.6.6 for additional information.

adm:PlacementDateTime [Optional] — For ADM/ADS message exchanges, this element ***shall*** contain the value as defined by this document. For other message exchanges where this element has insufficient context, the element ***shall not*** be used. See Section 14.30 for additional information. **adm:PlacementControl [Optional]** — This element includes relative positioning of specific Placement Opportunities within a stream. See Section 14.28 for additional information.

core:Ext [Optional] — Any elements from any namespaces. See [SCTE130-2] for additional information.

adm:SystemContext [Optional] — System environment information typically refining the placement service identification. The element's usage ***shall*** be as defined by [SCTE 130-3] and the element ***shall*** only be present when the containing element's schema does not offer this element (i.e., any parent element does not contain the adm:SystemContext element). For example, this element ***shall not*** be present when the PlacmentOpportunityV2 element is contained in a adm:PlacementRequest message. The parent element ***shall*** always take precedence when there is a conflict. See Section 14.52 for additional information. **Programming [Optional] —** A container holding references to content streams containing this Placement Opportunity. This element ***shall*** be used when more than one core:Content element is needed. If only one core:Content element is required, the standalone adm:Entertainment element previously described in this semantics section ***shall*** be utilized and this element ***shall*** be omitted. See Section D.6.10 for additional information.

PlacementOpportunityTerms [Optional] — The Placement Opportunity business terms. See Section D.6.7 for additional information.

D.6.9 Plane

This element contains the relevant details on how "show-able" an area of a screen is. As multiple spaces *may* be defined for a single Placement Opportunity, or multiple Placement Opportunities *may* exist simultaneously, some spaces *may* intersect. This defines where one space exists relative to another in the z plane, and the level of translucence or opacity.

The XML schema diagram for this element is as follows:

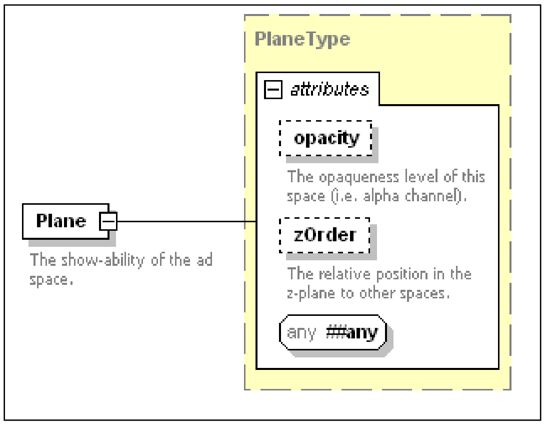

PlaneType
attributes
opacity
The opaqueness level of this space (i.e. alpha channel).
zOrder
The relative position in the z-plane to other spaces.
any ##any
Plane
The show-ability of the ad space.

Figure 111: Plane Element Schema

The Plane element contains the following semantics:

@opacity [optional, xsd:nonNegativeInteger] — The level of opaqueness of the space (i.e. alpha channel). A value of 100 is completely opaque, and a value of 0 is completely transparent. If the attribute is missing, the default value ***shall*** be 100 (i.e., completely opaque).

@zOrder [optional, xsd:nonNegativeInteger] — The relative position in the z-plane to other spaces. The @zOrder attribute value specifies the bottom-to-top stacking order of the spaces and the attributes minimum value ***shall*** be 1. The value 0, which ***shall*** be reserved, indicates the bottom of the stack and spaces are stacked bottom-to-top in order of increasing @zOrder attribute value. Spaces with identical @zOrder attribute values have unspecified relative stacking order. If the attribute is omitted, the default @zOrder attribute value ***shall*** be 1. Video essence assumes a @zOrder attribute value of 0 (making it the bottom of the stack).

@##any [Optional] — Additional attributes from any namespace.

D.6.10 Programming

The Programming element describes the content reference stream containing the Placement Opportunity if there is more than one reference stream or if the stream is non-video. If there is more than one reference stream or if the stream is non-video, the Programming element ***shall*** be used to describe the content reference stream containing the Placement Opportunity (and the adm:Entertainment element ***shall not*** be used). If there is only one reference stream and it is video, the adm:Entertainment element under the PlacementOpportunityV2 element ***shall*** describe the reference content stream. If neither the adm:Entertainment under PlacementOpportunityV2 element or the Programming element are recognized by the content delivery system, the Placement Opportunity does not exist with respect to said content delivery system. Table 27 defines how these elements ***shall*** be employed based on the reference streams present.

Reference Streams Present	Placement Opportunities Reference Stream
Single, video	The adm:Entertainment element under the PlacementOpportunityV2 element
Single, non-video	The Programming element
Multiple streams	The Programming element

Table 27: Placement Opportunity Reference Stream Identification

The XML schema diagram for this element is as follows:

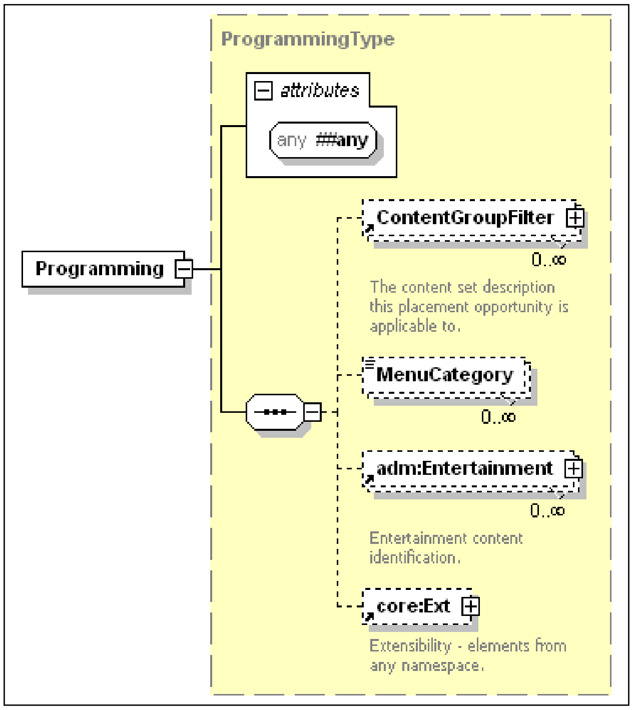

ProgrammingType
attributes
any ##any
Programming
ContentGroupFilter
0..∞
The content set description this placement opportunity is applicable to.
MenuCategory
0..∞
adm:Entertainment
0..∞
Entertainment content identification.
core:Ext
Extensibility - elements from any namespace.

Figure 112: Programming Element Schema

The Programming element semantics are defined as follows:

@##any [Optional] — Additional attributes from any namespace.

ContentGroupFilter [Optional] — The set description of content to which this Placement Opportunity is applicable. See Section D.6.1 for additional information.

MenuCategory [Optional] — The category in the on screen guide where the Placement Opportunity *may* be constrained (i.e., this Placement Opportunity only exists for content available within this category). See Section D.6.4 for additional information.

adm:Entertainment [Optional] — The content this Placement Opportunity *may* exist within, before, or after. This element ***shall*** only be used if more than one adm:Entertainment element is required. If a single adm:Entertainment element is required, the adm:Entertainment element under the PlacementOpportunityV2 element ***shall*** be used. See [SCTE130-3] for additional information.

core:Ext [Optional] — Any elements from namespaces. See [SCTE130-2] for additional information.

D.6.11 Size

The Size element contains the relevant details about the size of the Placement Opportunity space.

The XML schema diagram for this element is as follows:

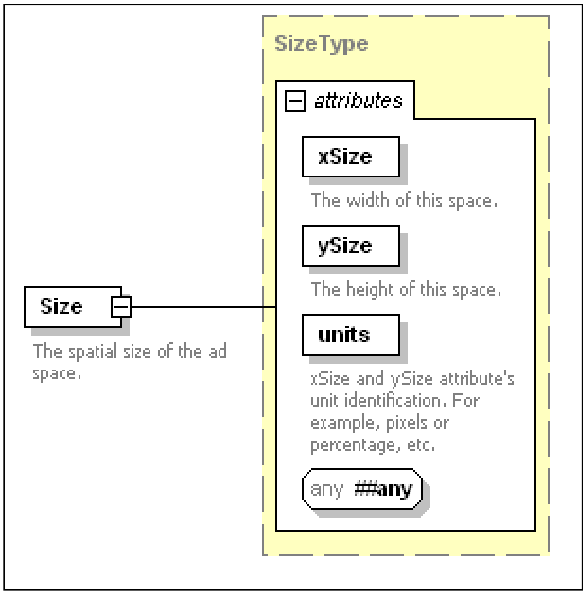

SizeType
attributes
xSize
The width of this space.
ySize
The height of this space.
units
xSize and ySize attribute's unit identification. For example, pixels or percentage, etc.
any ##any
Size
The spatial size of the ad space.

Figure 113: Size Element Schema

The Size element semantics are as follows:

@xSize [required, xsd:nonNegativeInteger] — The width of this space in the units specified by the @units attribute.

@ySize [required, xsd:nonNegativeInteger] — The height of this space in the units specified by the @units attribute.

@units [required, unitsIdentificationAttrType] — Identification of the measurement unit system for which @xSize and @ySize attributes values are referencing. Typical examples are pixels or percentage. See Section D.5.2 for additional information.

@##any [Optional] — Additional attributes from any namespace.

D.6.12 SpatialPosition

The SpatialPosition element identifies the screen real estate that *may* be consumed by a Placement Opportunity. This position *may* be occupied by an overlay within a larger Placement Opportunity, or a Placement Opportunity over the entertainment content. The absence of a SpatialPosition element indicates the Placement Opportunity ***shall*** have access to the entire screen.

The XML schema diagram for this element is as follows:

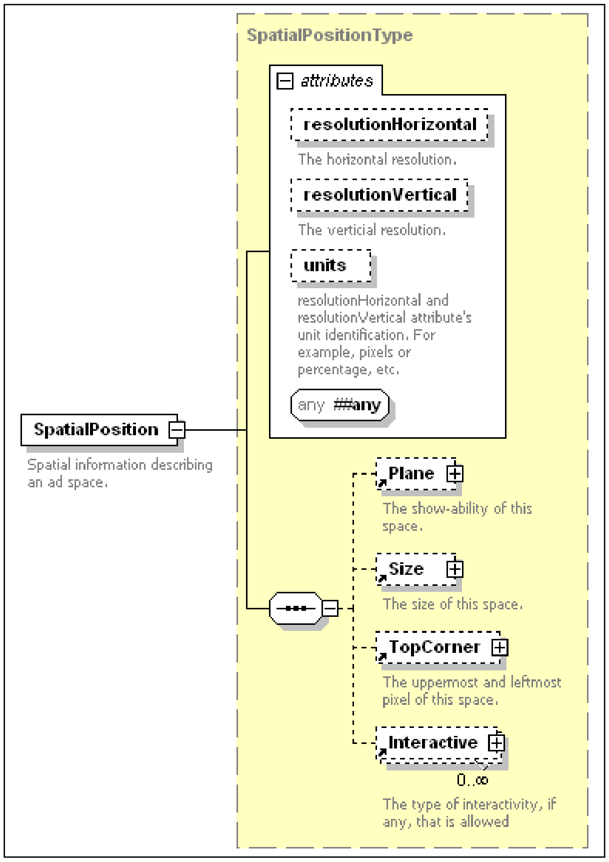

SpatialPositionType
attributes
resolutionHorizontal
The horizontal resolution.
resolutionVertical
The verticial resolution.
units
resolutionHorizontal and resolutionVertical attribute's unit identification. For example, pixels or percentage, etc.
any ##any
SpatialPosition
Spatial information describing an ad space.
Plane
The show-ability of this space.
Size
The size of this space.
TopCorner
The uppermost and leftmost pixel of this space.
Interactive
0..∞
The type of interactivity, if any, that is allowed

Figure 114: SpatialPosition Element Schema

The SpatialPosition element semantics are as follows:

@resolutionHorizontal [optional, xsd:nonNegativeInteger] — The expected width of the screen for this SpatialPosition element in the units specified by the @units attribute.

@resolutionVertical [optional, xsd:nonNegativeInteger] — The expected width of the screen for this SpatialPosition element in the units specified by the @units attribute.

@units [optional, unitsIdentificationAttrType] — Identification of the measurement unit system for which @resolutionHorizontal and @resolutionVertical attributes values are referencing. Typical examples are pixels or percentage. See Section D.5.2 for more information and the default value if the attribute is omitted.

@##any [Optional] — Additional attributes from any namespace.

Plane [Optional] — The show-ability of this space. See Section D.6.9 for additional information.

Size [Optional] — The size of this space. See D.6.11 for additional information. The Size element's @xPosition attribute value ***shall not*** exceed the maximum value of the @resolutionHorizontal attribute if present. The Size element's @yPosition attribute values ***shall not*** exceed the maximum value of the @resolutionVertical attribute if present.

TopCorner [Optional] — The uppermost and leftmost identification of this space. See Section D.6.13 for additional information.

Interactive [Optional] — Zero or more elements describing the interactivity characteristics of this space. See Section D.6.2 for additional information.

D.6.13 Top Corner

The TopCorner element contains the relevant details on where a spatial position exists on the screen.

The XML schema diagram for this element is as follows:

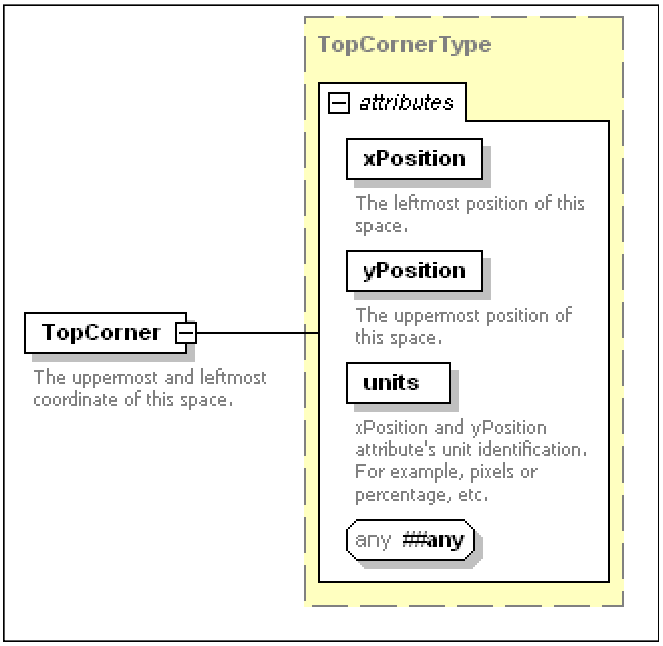

TopCornerType
attributes
xPosition
The leftmost position of this space.
yPosition
The uppermost position of this space.
units
xPosition and yPosition attribute's unit identification. For example, pixels or percentage, etc.
any ##any
TopCorner
The uppermost and leftmost coordinate of this space.

Figure 115: TopCorner Element Schema

The TopCorner element semantics are as follows:

@xPosition [required, xsd:nonNegativeInteger] — The leftmost position of this space in the units specified by the @units attribute.

@yPosition [required, xsd:nonNegativeInteger] — The uppermost position of this space in the units specified by the @units attribute.

@units [required, unitsIdentificationAttrType] — Identification of the measurement unit system for which @xSize and @ySize attributes values are referencing. Typical examples are pixels or percentage. See Section D.5.2 for additional information.

@##any [Optional] — Additional attributes from any namespace.

APPENDIX E. PLACEMENT OPPORTUNITY DATA MODEL EXAMPLES (INFORMATIVE)

E.1 Basic Substitution Example

Example 72 illustrates an original SCTE130-3 adm:PlacementRequest message containing three adm:PlacementOpportunty elements.

```
<PlacementRequest messageId="admmsg1" version="1.4" identity="ADMLogicalService_1170FEF4-C19B-
450E-B624-421B23F525F4" system="adm" updatesAllowed="false">
  <!--System context supplies information such as session identification, etc.-->
  <SystemContext>
    <Session>SessionID1</Session>
  </SystemContext>
  <!--Entertainment content reference.-->
  <Entertainment>
    <core:Content>
      <core:AssetRef providerID="moviecompany.com" assetID="MVIE0123456789012345"/>
    </core:Content>
  </Entertainment>
  <Client>
    <!--The Client element contains the targeting information.-->
    <TerminalAddress type="MAC">01-02-03-04-05-06</TerminalAddress>
    <TargetCode key="ZipPlus4">12345-6789</TargetCode>
    <TargetCode key="Gender">Male</TargetCode>
  </Client>
  <PlacementOpportunity id="Op1" serviceRegistrationRef="RegisteredServiceMatchRef1">
    <OpportunityBinding opportunityType="preRoll" opportunityNumber="1"
opportunitiesExpected="3"/>
    <OpportunityConstraints>
      <DesiredDuration>PT60S</DesiredDuration>
      <Scope>National</Scope>
    </OpportunityConstraints>
  </PlacementOpportunity>
  <PlacementOpportunity id="Op2" serviceRegistrationRef="RegisteredServiceMatchRef2">
    <OpportunityBinding opportunityType="midRoll" opportunityNumber="2"
opportunitiesExpected="3"/>
    <OpportunityConstraints>
      <MaxDuration>PT31S</MaxDuration>
      <MinDuration>PT29S</MinDuration>
      <DesiredDuration>PT30S</DesiredDuration>
      <Scope>Local</Scope>
    </OpportunityConstraints>
  </PlacementOpportunity>
  <PlacementOpportunity id="Op3" serviceRegistrationRef="RegisteredServiceMatchRef3">
    <OpportunityBinding opportunityType="postRoll" opportunityNumber="3"
opportunitiesExpected="3"/>
    <OpportunityConstraints>
      <MaxDuration>PT60S</MaxDuration>
      <MinDuration>PT5S</MinDuration>
      <DesiredDuration>PT10S</DesiredDuration>
      <Scope>Local</Scope>
    </OpportunityConstraints>
  </PlacementOpportunity>
</PlacementRequest>
```

Example 72

Example 73 re-implements Example 1 and extends it using the PODM defined PlacementOpportunityV2 element and the OpportunityConstraintsV2 element.

```
<PlacementRequest messageId="admmsg1" version="1.4" identity="ADMLogicalService_1170FEF4-C19B-
450E-B624-421B23F525F4" system="adm" updatesAllowed="false">
  <!--System context supplies information such as session identification, etc.-->
  <SystemContext>
    <Session>SessionID1</Session>
  </SystemContext>
  <!--Entertainment content reference.-->
  <Entertainment>
    <core:Content>
      <core:AssetRef providerID="moviecompany.com" assetID="MVIE0123456789012345"/>
    </core:Content>
  </Entertainment>
  <Client>
    <!--The Client element contains the targeting information.-->
    <TerminalAddress type="MAC">01-02-03-04-05-06</TerminalAddress>
    <TargetCode key="ZipPlus4">12345-6789</TargetCode>
    <TargetCode key="Gender">Male</TargetCode>
  </Client>
  <podm:PlacementOpportunityV2 id="Op1" serviceRegistrationRef="RegisteredServiceMatchRef1">
    <OpportunityBinding opportunityType="preRoll" opportunityNumber="1"
opportunitiesExpected="3"/>
    <podm:OpportunityConstraintsV2>
      <DesiredDuration>PT60S</DesiredDuration>
      <Scope>National</Scope>
      <podm:Interactive name="BadMamboJamboInteractiveContent">
        <core:Ext>
          <core:Content>
            <core:AssetRef providerID="mambo.com" assetID="MAMB0334455667788991"/>
          </core:Content>
        </core:Ext>
      </podm:Interactive>
    </podm:OpportunityConstraintsV2>
  </podm:PlacementOpportunityV2>
  <podm:PlacementOpportunityV2 id="Op2" serviceRegistrationRef="RegisteredServiceMatchRef2">
    <OpportunityBinding opportunityType="midRoll" opportunityNumber="2"
opportunitiesExpected="3"/>
    <OpportunityConstraints>
      <MaxDuration>PT31S</MaxDuration>
      <MinDuration>PT29S</MinDuration>
      <DesiredDuration>PT30S</DesiredDuration>
      <Scope>Local</Scope>
    </OpportunityConstraints>
  </podm:PlacementOpportunityV2>
  <PlacementOpportunity id="Op3" serviceRegistrationRef="RegisteredServiceMatchRef3">
    <OpportunityBinding opportunityType="postRoll" opportunityNumber="3"
opportunitiesExpected="3"/>
    <podm:OpportunityConstraintsV2>
      <MaxDuration>PT60S</MaxDuration>
      <MinDuration>PT5S</MinDuration>
      <DesiredDuration>PT10S</DesiredDuration>
      <Scope>Local</Scope>
    </podm:OpportunityConstraintsV2>
  </PlacementOpportunity>
</PlacementRequest>
```

Example 73

Example 2 illustrates how the adm:PlacmentOpportunity and the podm:PlacementOpportunityV2 element *may* simultaneously be carried in a adm:PlacementRequest message which has the XML namespaces declared correctly.

Example 2 also illustrates a PlacementOpportunityV2 element *may* use either an adm:OpportunityConstraints element or a podm:OpportunityConstraintsV2 element inside its context.

APPENDIX F. PODM AND INFORMATION SERVICE QUERIES (NORMATIVE)

F.1 Introduction

The PODM by itself does not constitute an information service, as it does not define any messaging XML elements. Any information service *may* include or utilize the PODM with a Placement Opportunity Information Service (POIS) implementation being a common candidate information service. The information service is expected to advertise PODM availability using the standard listing mechanisms as defined in [SCTE130-8].

The Placement Opportunity Data Model has been defined separately from the information service query elements allowing the data model to be used independently of the information service inquiry syntax and semantics. Consequently, there are two XML schema files (i.e. XSD files). One schema file is for the data model and one file defines a strongly typed basic query result element. The PODM specific query semantics and basic query result element are defined herein. Also, the data model and the query semantics share the same common XML namespace as defined in Table 1. Thus, they are related and separated only by the schema file.

Any information service including the PODM as a supported data model ***shall*** follow the semantics defined in the pursuant sections herein.

F.2 Data Model and Query Support

F.2.1 PODM Identification

The PODM ***shall*** be identified (or referenced) using the XML namespace stipulated in Table 1. Consequently, the gis:ServiceDataModel element's value (i.e., the URI) ***shall*** be the PODM XML namespace from Table 1. Example 74 illustrates.

```
<gis:ServiceDataModel>http://www.scte.org/schemas/130-3/2013/adm/podm</gis:ServiceDataModel>
```

Example 74

F.2.2 PODM Unique Qualifiers

Table 5 lists the mandatory SCTE 130 Part 3 Placement Opportunity Data Model (PODM) unique qualifier set an information service ***shall*** implement for the PODM data model basic query. An information service *may* implement other unique qualifier sets. However, these differences require separate gis:UniqueQualifierDeclaration elements identified by alternate @uniqueQualifierName attributes.

The @uniqueQualifierName attribute value for the PODM unique qualifier set ***shall*** be the string "P3-PODM" (without the quotes). Example 75 illustrates. For forward compatibility associated with this standard evolving, the @uniqueQualifierName attribute should be (i.e., is highly recommended to be) included in all elements where the attribute *may* optionally be included.

```
<gis:UniqueQualifierDeclaration uniqueQualifierName="P3-PODM">
```

```
  …
</gis:UniqueQualifierDeclaration>
```

Example 75

The PODM unique qualifiers are listed in Table 5 and ***shall*** appear exactly as they appear in the Unique Qualifier Identifier column. These values are used in the following schema locations:

- UniqueQualifierDeclaration/QualifierDeclaration/@name
- QualifierDescription/@name
- UniqueQualifier/Qualifier/@name

Unique Qualifier Identifier	PODM Syntax Mapping	Type	Description
poid	PlacementOpporunityV2/@id	string	Placement Opportunity Identifier. The poid corresponds to the Placement Opportunity @id attribute defined to be an identity unique identifier for the Placement Opportunity within the SCTE 130 Part 3 Placement Opportunity Data Model.

Table 28: PODM Unique Qualifiers

F.2.3 PODM Basic Query Requests

When querying the PODM using a basic query, the Query element should (i.e., is highly recommended to) include both the gis:ServiceDataModel element and the @uniqueQualifierName attribute with the values specified in Section F.2.1 and Section F.2.2. The primary motivation is forward compatibility as this specification evolves.

F.2.4 PODM Basic Query Results

When a basic query is issued referencing the PODM as the gis:ServiceDataModel and the gis:ServiceDataModel element's @uniqueQualiferName is set to value specified in Section F.2.2, the expanded results are returned using the PODMBasicQueryResult element specified in Section F.3.1. See Section F.3.1 for additional information.

F.2.5 Advanced Query Language Support

An information service implementing the PODM should support advanced query language processing as defined in [SCTE130-8].

F.3 PODM Information Service Elements

Table 29 identifies the additional elements added by this specification beyond those defined by a typical information service.

Element	Description
PODMBasicQueryResult	A substitutable element for the gis:BasicQueryResultAbstract element which includes the podm:PlacementOpportunityV2 element.

Table 29: PODM Element Details

F.3.1 PODMBasicQueryResult

When a basic query is issued against the PODM and signaled as specified in Section F.2, the results ***shall*** be returned to the calling client encapsulated within a PODMBasicQueryResult element. This element ***shall*** be substituted for the gis:BasicQueryResultAbstract element under the gis:QueryResult element and it ***shall*** contain the immediate result set for the paired query. The gis:BasicQueryResult element ***shall not*** be used.

The XML schema for PODMBasicQueryResult is illustrated in Figure 116.

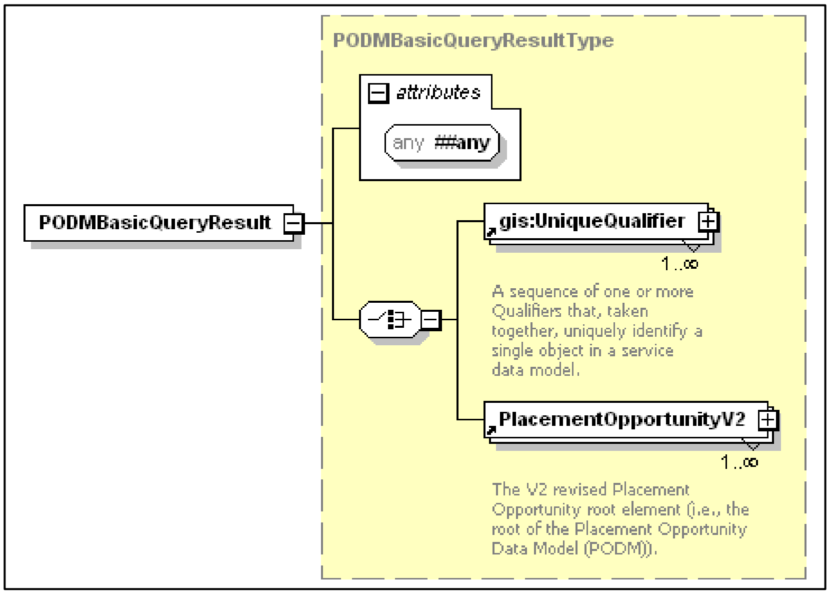

PODMBasicQueryResultType
attributes
any ##any
PODMBasicQueryResult
gis:UniqueQualifier
1..∞
A sequence of one or more Qualifiers that, taken together, uniquely identify a single object in a service data model.
PlacementOpportunityV2
1..∞
The V2 revised Placement Opportunity root element (i.e., the root of the Placement Opportunity Data Model (PODM)).

Figure 116: PODMBasicQueryResult Element XML Schema

The PODMBasicQueryResult element contains the following attributes and elements:

@##any [Optional] — Additional attributes from any namespace.

gis:UniqueQualifier [Required by Choice] — One or more gis:UniqueQualifier elements are returned as a response to a gis:Query element containing one or more gis:BasicFilterElement elements and the gis:Query element has the @expandOutput attribute set to the value "false". The gis:UniqueQualifier element ***shall*** contain the sequence of gis:Qualifier elements specified in Section F.2.2.

podm:PlacementOpportunityV2 [Required by Choice] — One or more podm:PlacementOpportunityV2 elements are returned as a response to a basic query when the originating gis:Query element contains either of the following:

- A gis:UniqueQualifer element.
- A sequence of gis:BasicFilterElement element in conjunction with gis:Query element's @expandOutput attribute being set to the value "true".

When the initiating gis:Query element contains a gis:UniqueQualifer element, only one podm:PlacementOpportunityV2 element ***shall*** be returned. If the inquiry result count evaluated to more than one, the query should be failed and a core:StatusCode element ***shall*** indicate the error condition using the value 8003 (Malformed Unique Qualifier). [Note: By definition, a unique qualifier ***shall*** resolve to one and only one object in the service data model. Thus, more than one result is considered an error.]

One or more podm:PlacementOpportunityV2 elements ***shall*** be returned when the gis:Query element contains a gis:BasicFilterElement and the gis:Query element's @expandOutput attribute is set to the value "true."

The following tables further refine the results dependent upon the initiating query's data.

Table 30 defines the possible result behaviors when the gis:QueryRequest element contains a gis:UniqueQualifier element.

gis:Query Element			**gis:QueryResult Element**		
					The PBQR element *shall* be present within the gis:QueryResult element when @resultsSetSize is non-zero.
@expand Output	**@result SetSize Only**	**@unique Qualifier NameRef**	**@total Result Set Size**	**@result SetSize**	**Elements present within the PBQR element when @resultsSetSize is non-zero.**
N/A	TRUE	Note 1	0 or 1	0	No elements ***shall*** be present.
FALSE	FALSE	Note 1	NP	0 or 1	A UniqueQualifier element whose contents conform to Section F.2.2.
TRUE	FALSE	Note 1	NP	0 or 1	A podm:PlacementOpportunityV2 element.

Table 30: gis:UniqueQualifier Query Result Behaviors

1. Note 1: The value should be (is highly recommend to be) the value specified in Section F.2.2 (i.e., “P3-PODM”).
2. N/A = Not Applicable (ignored if present)
3. NP = Not present (***shall not*** be returned)
4. PBQR = PODMBasicQueryResult element
5. AQR = gis:AdvancedQueryResult

Table 31 defines the possible result behaviors when the gis:QueryRequest element contains a gis:BasicQueryFilter sequence.

gis:Query Element			**gis:QueryResult Element**		
					The PBQR element *shall* be present within the gis:QueryResult element when @resultsSetSize is non-zero.
@expand Output	**@result SetSize Only**	**@unique Qualifier NameRef**	**@total Result Set Size**	**@result SetSize**	**Elements present within the PBQR element when @resultsSetSize is non-zero.**
N/A	TRUE	Note 1	0 to UB (Note 1)	0	No elements ***shall*** be present.
FALSE	FALSE	Note 1	NP	0 to UB (Note 2)	One to unbounded UniqueQualifier elements as defined by Section F.2.2.
TRUE	FALSE	Note 1	NP	0 to UB	One to unbounded podm:PlacementOpportunityV2 elements.

Table 31: gis:BasicQueryFilter Query Result Behaviors

6. Note 1: The value should be (is highly recommend to be) the value specified in Section F.2.2 (i.e., “P3-PODM”).
7. Note 2: If the same query had @expandOutput set to “true”, the value ***shall*** be the number of podm:PlacementOpportunityV2 elements returned.
8. N/A = Not Applicable (ignored if present)
9. NP = Not present (***shall not*** be returned)
10. PBQR = PODMBasicQueryResult element

Table 32 defines the possible result behaviors when the gis:QueryRequest element contains a gis:AdvancedQueryFilter sequence.

gis:Query Element			**gis:QueryResult Element**		
					The AQR element *shall* be present within the gis:QueryResult element when @resultsSetSize is non-zero.
@expand Output	**@result SetSize Only**	**@unique Qualifier NameRef**	**@total Result Set Size**	**@result SetSize**	**Elements present within the AQR element** **when @resultsSetSize is non-zero.**
FALSE	TRUE	Note 1	0 to UB	0	No elements *shall* be present.
TRUE	TRUE	Note 1	0 to UB		
FALSE	FALSE	Note 1	NP	0 to UB	1 to unbounded UniqueQualifier elements using the @uniqueQualifierNameRef identified unique qualifier
TRUE	FALSE	Note 1	NP	0 to UB (Note 2)	1 to unbounded AdvancedQueryResultData elements

Table 32: Advanced Qualifier Query Result Behaviors

11. Note 1: The value should be (is highly recommend to be) the value specified in Section F.2.2 (i.e., “P3-PODM”).

12. Note 2: If non-zero, the value ***shall*** be the number of AdvancedQueryResultData elements contained in the AdvancedQueryResult element.

13. NP = Not present (***shall not*** be returned)

14. AQR = gis:AdvancedQueryResult

This page intentionally blank.

#####