SCTE · ISBE

STANDARDS

Digital Video Subcommittee

AMERICAN NATIONAL STANDARD

ANSI/SCTE 130-9 2020

Recommended Practices for SCTE 130 Digital Program Insertion—Advertising Systems Interfaces

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.

Table of Contents

List of Figures

List of Tables

List of Examples

1   Introduction

The goal of this recommended practices document is to serve as an informational enhancement to SCTE 130, Digital Program Insertion—Advertising Systems Interfaces. SCTE 130 is necessarily brief in many areas in order to maintain conciseness and accuracy. This document serves as a companion to SCTE 130.

This document makes reference to versions of SCTE 130 documents from both the first edition (now *deprecated*) and second edition of the standard, as outlined in Section 2.   For further details concerning the differences between the first and second editions of SCTE 130, see SCTE130-1 2013: Digital Program Insertion—Advertising Systems Interfaces Part 1 – Overview.

2   Informative References

At the time of publication, the editions indicated below were current. All standards are subject to revision, and parties to agreement based on this document are encouraged to apply the most recent editions of the documents listed below when appropriate.

[CLADI1-1]	CableLabs Asset Distribution Interface Specification Version 1.1: MD-SP-ADI1.1-I04-060505, http://www.cablelabs.com/specifications/MD-SP-ADI1.1-I04-060505.pdf
[SCTE130-1]	SCTE 130-1 20xx: Digital Program Insertion—Advertising Systems Interfaces Part 1 – Advertising Systems Overview.
[SCTE130-2]	SCTE 130-2 20xx: Digital Program Insertion—Advertising Systems Interfaces Part 2 – Core Data Elements.
[SCTE130-3]	SCTE 130-3 20xx: Digital Program Insertion—Advertising Systems Interfaces Part 3 – Ad Management Service Interface.
[SCTE130-4]	SCTE 130-4 2015: Digital Program Insertion—Advertising Systems Interfaces Part 4 – Content Information Service.
[SCTE130-5]	SCTE 130-5 2016: Digital Program Insertion—Advertising Systems Interfaces Part 5 – Placement Opportunity Information Service.
[SCTE130-6]	SCTE 130-6 20xx: Digital Program Insertion—Advertising Systems Interfaces Part 6 – Subscriber Information Service.
[SCTE130-7]	SCTE 130-7 2015: Digital Program Insertion – Advertising Systems Interfaces Part 7 – Message Transport.
[SCTE130-8]	SCTE 130-8 20xx: Digital Program Insertion —Advertising Systems Interfaces Part 8—General Information Service.
[SCTE 130-10]	SCTE 130-10 20xx: Digital Program Insertion – Advertising Systems Interfaces Part 10 – Stream Restriction Data Model.
[SCTE35]	ANSI/SCTE 35 2019r1—Digital Program Insertion Cueing Message for Cable.

3   Glossary 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 1, SCTE 130 Part 2, SCTE 130 Part 3, SCTE 130 Part 4, SCTE 130 Part 5, SCTE 130 Part 6, SCTE 130 Part 7 and SCTE 130 Part 8 definitions are included herein. See [SCTE130-1], [SCTE130-2], [SCTE130-3], , [SCTE130-4], [SCTE130-5], [SCTE130-6], [SCTE130-7], [SCTE130-8], and [SCTE130-10] for additional information.

This document additionally defines the following terms:

Intermediary: A SCTE 130 compliant server deployed between one or more ADMs and one or more ADSs.

4   Abbreviations

All SCTE 130 Part 1, SCTE 130 Part 2, SCTE 130 Part 3, SCTE 130 Part 4, SCTE 130 Part 5, SCTE 130 Part 6, SCTE 130 Part 7 and SCTE 130 Part 8 abbreviations are included herein. See [SCTE130-1], [SCTE130-2], [SCTE130-3], [SCTE130-4], [SCTE130-5], [SCTE130-6], [SCTE130-7], [SCTE130-8], and [SCTE130-10] for additional information.

5   Overview

This document provides an informative (i.e., non-normative) set of recommended practices for SCTE 130, a standardized and extensible message based interface defining a minimal set of cooperating logical services necessary to communicate placement opportunities, placement decisions, and placement related event data necessary for accountability measurements. Use cases are presented for placement decision messaging, information service messaging, as well as several advanced topics that warranted a more in depth discussion than was offered in the core SCTE 130 suite of standards.

5.1   Scope of the Recommended Practices Document

This document is an informational companion to the SCTE 130 suite of standards. It is not in itself a specification or a standard. The information within is intended as guidelines and recommendations for implementers of the standard. Where this document contradicts with the SCTE 130 suite of standards, the SCTE 130 suite of standards take precedence.

5.2 Purpose

While the standards encompassing SCTE 130 have sufficient depth for their respective services, they individually lack the breadth necessary to understand the entire ecosystem. This document seeks to provide a complete view of the SCTE 130 architecture, describes the interaction among all of the services via common use cases, and provides end-to-end messaging examples to show how the data flows through the system throughout the course of a viewing session.

5.3 Example Messaging and UUIDs

This document contains many example messages, all of which have the @messageId attribute. The data type of the @messageId attribute is messageIdAttrType, which Section 11.2.2 (idAttrType Attribute Type) of [SCTE130-2] defines as:

> ...a service channel unique identifier for a message. Every SCTE 130 message instance shall have a service channel unique value and shall not be empty. The attribute should be a Universally Unique Identifier as defined by RFC 4122 . . ..

For readability and formatting purposes, this document uses values like "uuid-x" instead of a true UUID.

Other identifiers, such as PlacementOpportunity/@id, Query/@queryId, and Cursor/@cursorId, are all of different data types, but should also be UUIDs. For the same reasons stated in the previous paragraph, these attributes contain values such as "uuid-x" or "query-x".

6 Structure and Summary

This document contains three main sections: Placement Decision Usage Patterns, Information Service Usage Patterns, and Advanced Topics. Throughout all sections of this document, whenever a message or message pair is described, a small sequence diagram is used to show which service initiates the message and the order in which the messages are sent. Certain attributes or elements within some of the example messages are highlighted for emphasis, with explanations of their significance immediately before or after the message.

Placement Decision Usage Patterns Summary

The Placement Decision Usage Patterns section contains the majority of the ADS/ADM message examples in the document. Three complete VOD use cases are displayed, as well as registration, and both ADM and ADS initiated registration. All of the top level sections in Placement Decision Usage Patterns start with a diagram showing the different services and the way data flows through the ecosystem, followed by a sequence diagram that shows the specific messages traveling between services and the order in which they are sent. See Section 7 (Placement Decision Usage Patterns) for details.

ADS/ADM Registration Messaging

This section details the registration messaging between the ADM and ADS. ListADMServicesRequest/Response, ListADSRegistrationRequest/Response, and ADSRegistrationRequest/Response are covered with example messaging, as well as the @version attribute and its relation to PODM support. See Section 7.1 (ADS/ADM Registration) for details.

ADS-Centric VOD

In this scenario, the ADS is responsible for making calls to the information services and using that data to aid in its placement decisions. After the placement, event data is obtained by the ADM and sent to the ADS. The complete lifecycle of the placement is shown along with example messages. See Section 7.2 (ADS-centric VOD) for details.

ADM-Centric VOD

In this scenario, the ADM is responsible for making calls to the information services and using that data to decorate the PlacementRequest, which it then forwards to the ADS. After the placement, event data is obtained by the ADM and sent to the ADS. The complete lifecycle of the placement is shown along with example messages. See Section 7.3 (ADM-centric VOD) for details.

VOD with an Intermediary

For the Intermediary use case, an ADM sends a bare bones PlacementRequest to an Intermediary. The Intermediary is responsible for making calls to the information services and using that data to decorate the PlacementRequest before forwarding it to the ADS. After the placement, event data is obtained by the ADM and travels through the Intermediary before reaching the ADS. The complete lifecycle of the placement is shown along with example messages. See Section 7.4 (VOD with an Intermediary) for details.

ADS Initiated Deregistration

This section details the ADS initiated deregistration messaging between the ADM and ADS. ADSDeregisterRequest/Response is covered in detail, including the difference in behavior of the ADM based on the presence of the @registrationRef attribute. See Section 7.5 (ADS Initiated Deregistration) for details.

ADM Initiated Deregistration

This section details the ADM initiated deregistration messaging between the ADM and ADS. ADSDeregistrationNotification/Acknowledgement is covered in detail, including the difference in behavior of the ADS based on the presence of the @registrationRef attribute. See Section 7.6 (ADM Initiated Deregistration) for details.

Information Service Usage Pattern Summary

For each of the three Information Services, four use cases are explored that aid a client in retrieving data: Feature Discovery, Notifications, Queries, and Cursors. There is some overlap in the Query section with some of the VOD Placement Decision Usage Patterns, but these sections also detail querying with a non-unique identifier, such as a regular expression. Each of the four use cases contain a sequence diagram that shows the specific messages traveling between services and the order in which they are sent. See Section 8 (Information Service Usage Patterns) for details.

Advanced Topics Summary

Some topics warranted a more in depth discussion than was offered in the SCTE 130 suite of standards, and the purpose of this section is to explore those advance topics and expand on the explanations in the existing SCTE 130 suite of standards. See Section 9 (Advanced Topics) for details.

7 Placement Decision Usage Patterns

This section contains the majority of the ADS/ADM message examples contained in the document. Three complete VOD use cases are displayed, as well as registration, and both ADM and ADS initiated registration. All of the top level sections in Placement Decision Usage Patterns start with a diagram showing the different services and the way data flows through the ecosystem, followed by a sequence diagram that shows the specific messages traveling between services and the order in which they are sent.

7.1 ADS/ADM Registration

This section details the registration messaging between the ADM and ADS. ListADMServicesRequest/Response, ListADSRegistrationRequest/Response, and ADSRegistrationRequest/Response are covered with example messaging, as well as the @version attribute and it's relation to PODM support.
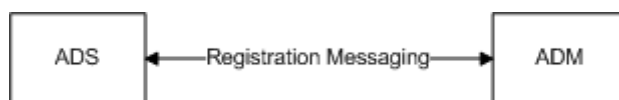
Figure 1: Registration Messaging Overview

The ADS is responsible for initiating registration messaging. The ADS queries the ADM for available services via the ListADMServicesRequest, determines if it already has registrations with a ListADSRegistrationRequest, then registers with an ADSRegistrationRequest. The sequence diagram below shows registration message flow between the ADM and the ADS.
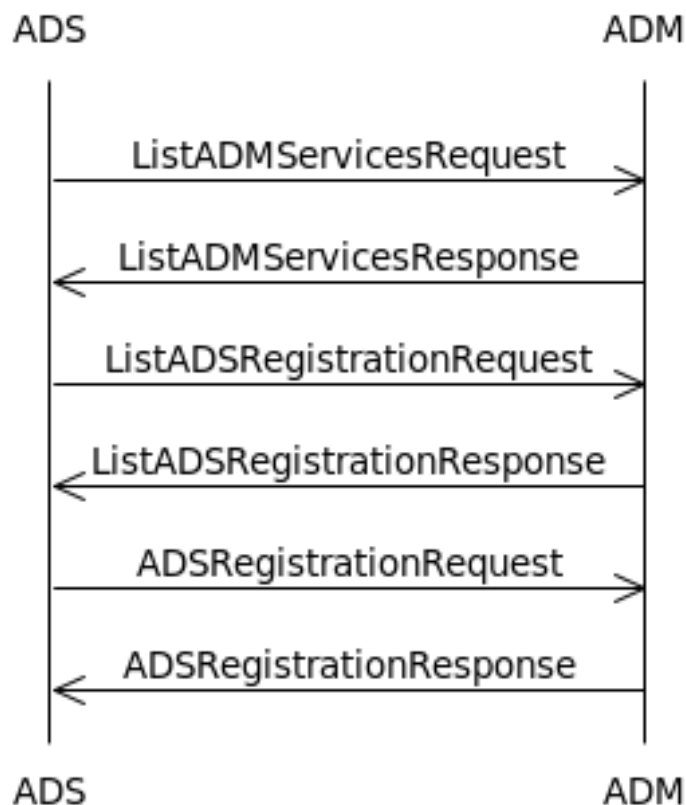
Figure 2: Registration Sequence Diagram

7.1.1 ListADMServicesRequest/ListADMServicesResponse

The ADS first initiates a ListADMServicesRequest to learn about the capabilities and services offered by the ADM. For more detail on the ListADMServices messages, see Section 11.2 (List ADM Placement Services) of [SCTE130-3].

Figure 3: List ADM Placement Services Message Exchange

Below is an example ListADMServicesRequest initiated by the ADS.

```
<ListADMServicesRequest messageId="uuid-1" version="1.1"
identity="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5"
system="ads"/>
```

Example 1

In the following example, the ADM's ListADMServicesResponse contains a single service for preRoll ads (highlighted below).

```
<ListADMServicesResponse messageId="uuid-2" version="1.1"
identity="ADMLogicalService_1170FEF4-C19B-450E-B624-421B23F525F4"
system="adm" messageRef="uuid-1">
  <core:StatusCode class="0"/>
  <ADMCapabilities placementStatusOnlyRegistration="false"/>
  <ServiceDescription>
    <SystemContext/>
    <Service id="Service1" opportunityType="preRoll" />
  </ServiceDescription>
  <core:Callout>
    <core:Address>www.adshark.com/allMessages</core:Address>
  </core:Callout>
</ListADMServicesResponse>
```

Example 2

The value of the version attribute can signify use of the Placement Opportunity Data Model (PODM). An ADS can query the ADM with @version="1.2" (highlighted below) to determine if the ADM supports PODM.

```
<ListADMServicesRequest messageId="uuid-3" version="1.2"
identity="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5"
system="ads"/>
```

Example 3

If the ADM supports PODM, it responds with @version="1.2" in the ListADMServiceResponse message. Otherwise, it responds with an error and a core:StatusCode/@detail attribute having the value core:UnsupportedProtocol or it responds with a successful ListADMServicesResponse message having the @version attribute set to the value "1.1".

Further discussion of the version attribute with respect to ADM/ADS messaging is located in Appendix D (THE PLACEMENT OPPORTUNITY DATA MODEL (NORMATIVE)) of [SCTE130-3].

7.1.2   ListADSRegistrationRequest/ListADSRegistrationResponse

Before the ADS registers for an ADM offered service, it should check the ADM to ensure it has not already registered for that service by using a ListADSRegistrationRequest. For more detail on the ListADSRegistration message, see Section 11.4 (List ADS Registrations) of [SCTE130-3].

Figure 4: ADM Registration Status Message Exchange

Below is a typical ListADSRegistrationRequest that will request all registrations for the ADS, due to the lack of a @registrationRef attribute in the message.

```
<ListADSRegistrationRequest messageId="uuid-4" version="1.1"
identity="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5"
system="ads"/>
```

Example 4

This ADM's ListADSRegistrationResponse shows that the ADS has no active registrations on the ADM.

```
<ListADSRegistrationResponse messageId="uuid-5" version="1.1"
identity="ADMLogicalService_1170FEF4-C19B-450E-B624-421B23F525F4"
system="adm" messageRef="uuid-4">
  <core:StatusCode class="0">
    <core:Note>There are currently no ADS registrations.</core:Note>
  </core:StatusCode>
</ListADSRegistrationResponse>
```

Example 5

Alternatively, there may already be registrations on the ADM as illustrated in the following example.

```
<ListADSRegistrationResponse messageId="uuid-6" version="1.1"
identity="ADMLogicalService_1170FEF4-C19B-450E-B624-421B23F525F4"
system="adm" messageRef="uuid-4">
  <core:StatusCode class="0"/>
  <ADSRegistrationRequest messageId="uuid-7" version="1.1"
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="Service4" opportunityType="postRoll">
        <Scope>Local</Scope>
      </Service>
    </ServiceDescription>
    <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="uuid-8" version="1.1" 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 6

It is important to note that, although the standard does not prohibit the ADM from returning a complete list of all active registrations, in practice it should only return registrations matching the identity of the ADS currently querying. An ADS should not see another ADS's registration in a ListADSRegistrationResponse, regardless of the presence of the @registrationRef attribute.

7.1.3   ADSRegistrationRequest/ADSRegistrationResponse

After the ADS sends an ADSRegistrationRequest to an ADM, the ADM may block and verify the Callouts before responding with an ADSRegistrationResponse. For more detail on the ADSRegistration messages, see Section 11.3 (Registration) of [SCTE130-3].

Figure 5: Registration Message Exchange

```
<ADSRegistrationRequest messageId="uuid-9" version="1.1" system="ads"
identity="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5">
  <ServiceDescription>
    <SystemContext/>
    <Service id="Service1" opportunityType="preRoll" />
  </ServiceDescription>
  <core:Callout>
  <core:Address>www.adshark.com/allMessages</core:Address>
  </core:Callout>
</ADSRegistrationRequest>
```

Example 7

Now, the ADM initiates a ServiceCheckRequest to verify the Callout. For more detail on the ServiceCheck messages, see Section 10.1 (Service Check Messages) of [SCTE130-2].

```
<ServiceCheckRequest messageId="uuid-10"
version="1.1" identity="ADMLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F4" system="adm"/>
```

Example 8

The ADS will respond with a ServiceCheckResponse, verifying that the endpoint it defined in the ADSRegistrationRequest is able to receive messages in the future.

```
<ServiceCheckResponse messageId="uuid-11"
version="1.1" identity="ADSLogicalService_1170FEF4-C19B-450E-B624-
421B23F525F5" system="ads" messageRef="uuid-10">
  <StatusCode class="0"/>
</ServiceCheckResponse>
```

Example 9

The ADM then responds with an ADSRegistrationResponse that acknowledges the registration.

```
<ADSRegistrationResponse messageId="uuid-12" version="1.1"
identity="ADMLogicalService_1170FEF4-C19B-450E-B624-421B23F525F4"
system="adm" messageRef="uuid-9">
  <core:StatusCode class="0"/>
</ADSRegistrationResponse>
```

Example 10

The value of the version attribute can signify use of the Placement Opportunity Data Model (PODM). An ADS can attempt to register with the ADM with @version="1.2" in order to utilize the PODM.

```
<ADSRegistrationRequest messageId="uuid-13" version="1.2" system="ads"
identity="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5">
  <ServiceDescription>
    <SystemContext/>
    <Service id="Service1" opportunityType="preRoll" />
  </ServiceDescription>
  <core:Callout>
  <core:Address>www.adshark.com/allMessages</core:Address>
  </core:Callout>
</ADSRegistrationRequest>
```

Example 11

If the ADM supports PODM, it responds with @version="1.2" in the ListADMServiceResponse message. Otherwise, it responds with an error and a core:StatusCode/@detail attribute having the value core:UnsupportedProtocol or it responds with a successful ListADMServicesResponse message having the @version attribute set to the value "1.1".

Further discussion of the version attribute with respect to ADM/ADS messaging is located in Appendix D (THE PLACEMENT OPPORTUNITY DATA MODEL (NORMATIVE)) of [SCTE130-3].

7.2 ADS-centric VOD
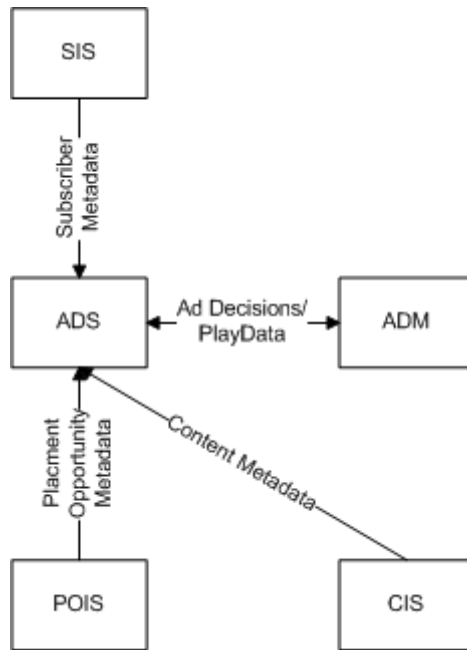
Figure 6: ADS-centric VOD Messaging Overview

In an ADS-centric view of SCTE 130 (shown above), the ADS is responsible for communication with the other SCTE 130 information services and the ADM will place the ad decisions dictated by the ADS. Registration is required between the ADS and ADM (outlined in Section 0) before the placement messaging shown in the sequence diagram below is attempted.
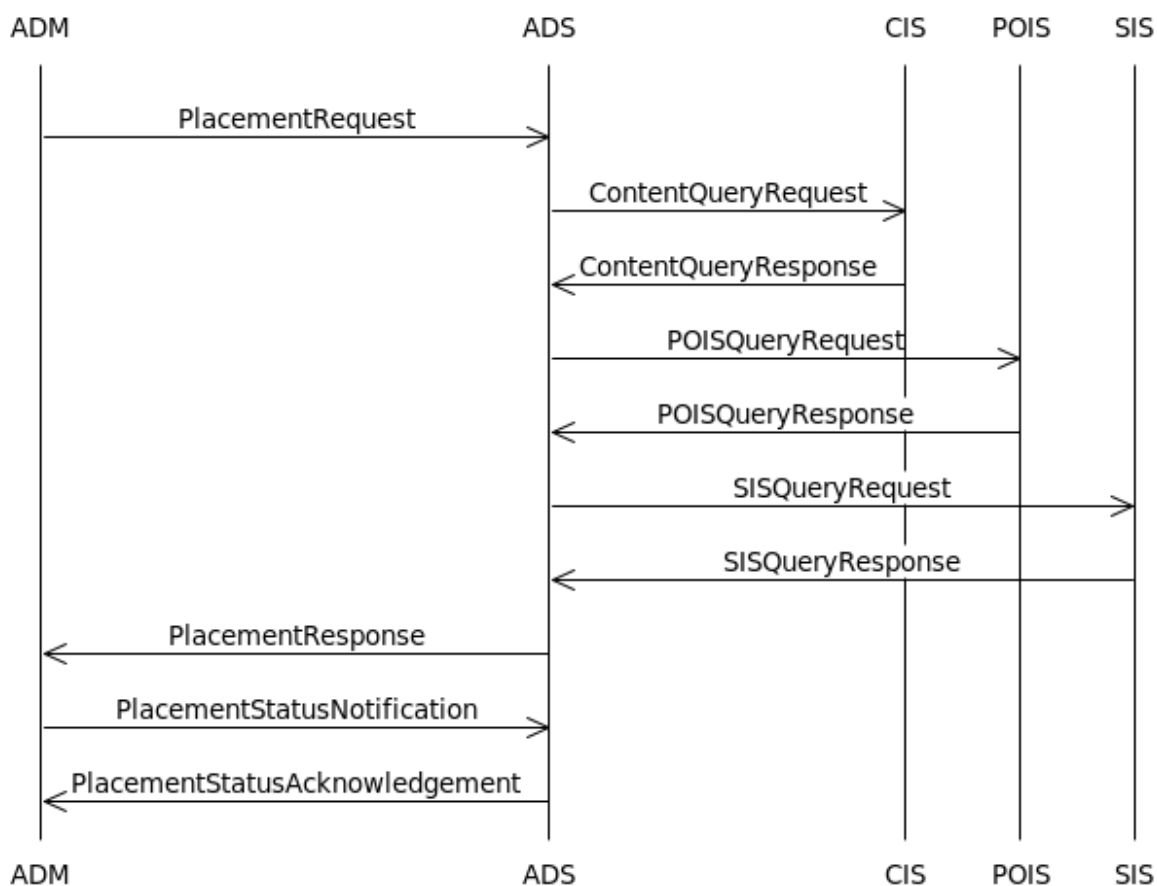
Figure 7: ADS-centric VOD Sequence Diagram

7.2.1   PlacementRequest

At the start of a VOD session, the ADM will query the ADS with a PlacementRequest. For more detail on the Placement messages, see Section 12.1 (Placement Messaging) of [SCTE130-3].
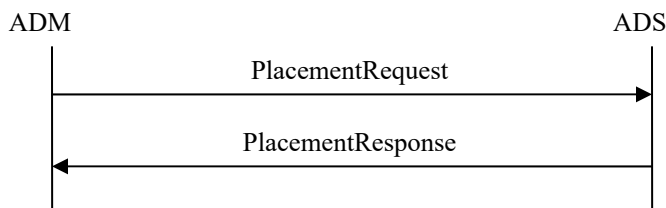
Figure 8: ADS-centric VOD Placement Request Message Exchange

The ADM's PlacementRequest contains one PlacementOpportunity as required by the schema, but the ADM has no metadata for the content, placement opportunity, or subscriber other than the client device's MAC address and the provider ID and asset ID of the entertainment content. This leaves the ADS responsible for querying information services (if available) to make an informed ad decision.

```
<PlacementRequest messageId="uuid-14" version="1.1"
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>
    <TerminalAddress type="MAC">03-55-78-90-03-02</TerminalAddress>
  </Client>
  <PlacementOpportunity id="uuid-15" serviceRegistrationRef="ServiceRef1">
  </PlacementOpportunity>
</PlacementRequest>
```

Example 12

7.2.2   ContentQueryRequest/ContentQueryResponse

Before making an ad decision and returning a PlacementResponse to the ADM, the ADS may query the CIS for additional information about the entertainment asset to aid in its decision. For more detail on the ContentQuery messages, see Section 10.15 (Content Query Request and Response) and Appendix E.2 (Content Query Request and Response) of [SCTE130-4].
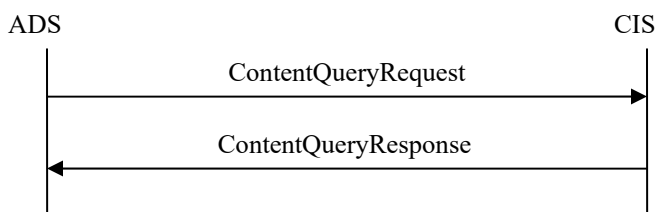
Figure 9: ADS-Centric VOD Content Query Message Exchange

In order to retrieve the complete metadata for the asset, the @expandOutput attribute is required to be set to true.

```
<ContentQueryRequest messageId="uuid-16" system="sys-1" version ="1.1"
    identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
    <ContentQuery expandOutput="true" contentQueryId="1">
       <core:ContentDataModel type="CLADI_1.1">URI</core:ContentDataModel>
       <QueryFilter>
           <FilterElement name="Provider_ID"  value="entertainment.com" />
           <FilterElement name="Asset_ID"  value="ABCD01234567890123456" />
       </QueryFilter>
    </ContentQuery>
</ContentQueryRequest>
```

Example 13

The CIS will respond with the full metadata in the specified content data model, CLADI 1.1 [CLADI 1.1].

```
<ContentQueryResponse messageId="uuid-17" system="acs-1" version="1.1"
     identity="40DA910E-01AF-5050-C7EA-5D7B4A475312" messageRef="uuid-16">
    <core:StatusCode class="0"/>
    <ContentQueryResult resultSetSize="1" contentQueryRef="1">
       <BasicQueryResultList>
           <core:Content>
               <core:ContentLocation
mediaAvailable="true">URI</core:ContentLocation>
               <core:Ext>
                   <ADI>
                       <Metadata> <AMS Asset_Name="..."/> …</Metadata>
                   </ADI>
               </core:Ext>
           </core:Content>
       </BasicQueryResultList>
    </ContentQueryResult>
</ContentQueryResponse>
```

Example 14

For readability, the actual CLADI 1.1 [CLADI 1.1] metadata has been removed and posted below.

```
<ADI>
  <Metadata>
    <AMS Asset_Class="package" Asset_ID="ABCD01234567890123343" Asset_Name="SCTE 130: The Movie"
Creation_Date="2008-02-11" Description="SCTE 130: The Movie" Product="MOD"
Provider="entertainment" Provider_ID="entertainment.com" Version_Major="1" Version_Minor="0"/>
    <App_Data App="MOD" Name="Metadata_Spec_Version" Value="CableLabsVOD1.1"/>
    <App_Data App="MOD" Name="Provider_Content_Tier" Value="ENTERTAINMENT_NEWRELEASE"/>
  </Metadata>
  <Asset>
    <Metadata>
      <AMS Asset_Class="title" Asset_ID="ABCD01234567890123344" Asset_Name="SCTE 130: The Movie"
Creation_Date="2008-02-11" Description="SCTE 130: The Movie" Product="MOD"
Provider="entertainment" Provider_ID="entertainment.com" Version_Major="1" Version_Minor="0"/>
      <App_Data App="MOD" Name="Title" Value="SCTE 130: The Movie"/>
      <App_Data App="MOD" Name="Title_Brief" Value="SCTE 130: The Movie"/>
      <App_Data App="MOD" Name="Type" Value="title"/>
      <App_Data App="MOD" Name="Year" Value="2009"/>
      <App_Data App="MOD" Name="Director" Value="Bryan Santangelo"/>
      <App_Data App="MOD" Name="Actors_Display" Value="John Mick, Corey Brown"/>
      <App_Data App="MOD" Name="Actors" Value="John Mick"/>
      <App_Data App="MOD" Name="Actors" Value="Corey Brown"/>
      <App_Data App="MOD" Name="Genre" Value="Action & Adventure"/>
      <App_Data App="MOD" Name="Category" Value="On Demand/High Definition/HD Movies"/>
      <App_Data App="MOD" Name="Category" Value="On Demand/Movies/HD Movies"/>
      <App_Data App="MOD" Name="Category" Value="On Demand/Movies/Movies By Title/Movies S - Z"/>
      <App_Data App="MOD" Name="Category" Value="On Demand/Movies/New Releases/High Definition"/>
      <App_Data App="MOD" Name="Category" Value="On Demand/What's Hot/HD Movies"/>
      <App_Data App="MOD" Name="Billing_ID" Value="60671"/>
      <App_Data App="MOD" Name="Box_Office" Value="24000000"/>
      <App_Data App="MOD" Name="Closed_Captioning" Value="Y"/>
      <App_Data App="MOD" Name="Contract_Name" Value="ENT"/>
      <App_Data App="MOD" Name="Display_As_Last_Chance" Value="7"/>
      <App_Data App="MOD" Name="Display_As_New" Value="7"/>
      <App_Data App="MOD" Name="Display_Run_Time" Value="02:03"/>
      <App_Data App="MOD" Name="Distributor_Name" Value="ENT"/>
      <App_Data App="MOD" Name="Distributor_Royalty_Minimum" Value="2.59"/>
      <App_Data App="MOD" Name="Distributor_Royalty_Percent" Value="65.0"/>
      <App_Data App="MOD" Name="Licensing_Window_End" Value="2015-01-01"/>
      <App_Data App="MOD" Name="Licensing_Window_Start" Value="2008-03-21"/>
      <App_Data App="MOD" Name="Maximum_Viewing_Length" Value="01:00:00"/>
      <App_Data App="MOD" Name="Preview_Period" Value="300"/>
      <App_Data App="MOD" Name="Provider_QA_Contact" Value="help@ent.com"/>
      <App_Data App="MOD" Name="Rating" Value="R"/>
      <App_Data App="MOD" Name="Run_Time" Value="02:03:58"/>
      <App_Data App="MOD" Name="Studio" Value="SCTE"/>
      <App_Data App="MOD" Name="Studio_Name" Value="SCTE"/>
      <App_Data App="MOD" Name="Suggested_Price" Value="5.99"/>
      <App_Data App="MOD" Name="Summary_Short" Value="The adventures of a DVS subcommittee"/>
    </Metadata>
    <Asset>
      <Metadata>
        <AMS Asset_Class="movie" Asset_ID="ABCD01234567890123345" Asset_Name="SCTE 130: The Movie"
Creation_Date="2008-02-11" Description="SCTE 130: The Movie" Product="MOD"
Provider="entertainment" Provider_ID="entertainment.com" Version_Major="1" Version_Minor="0"/>
        <App_Data App="MOD" Name="Audio_Type" Value="Stereo"/>
        <App_Data App="MOD" Name="Type" Value="movie"/>
        <App_Data App="MOD" Name="HDContent" Value="Y"/>
      </Metadata>
      <Content Value="entertainment.com::TVNX10084127342::On_Demand.mpeg"/>
    </Asset>
    <Asset>
      <Metadata>
        <AMS Asset_Class="preview" Asset_ID="ABCD01234567890123346" Asset_Name="SCTE 130: The
Movie" Creation_Date="2008-02-11" Description="SCTE 130: The Movie" Product="MOD"
Provider="entertainment" Provider_ID="entertainment.com" Version_Major="1" Version_Minor="0"/>
        <App_Data App="MOD" Name="Audio_Type" Value="Stereo"/>
        <App_Data App="MOD" Name="Rating" Value="R"/>
        <App_Data App="MOD" Name="Run_Time" Value="00:05:00"/>
        <App_Data App="MOD" Name="Type" Value="preview"/>
      </Metadata>
      <Content Value="entertainment.com::TVNX10084127343::On_Demand.mpeg"/>
```

```
      </Asset>
    </Asset>
</ADI>
```

<div align="center">**Example 15**</div>

With all of the calls to the SCTE 130 information services (CIS, SIS, POIS), an ADS may prefer to register for notifications and cache metadata in an effort to improve performance.

7.2.3   POISQueryRequest/POISQueryResponse

The ADS now needs to obtain additional metadata for any placement opportunities that may exist in the entertainment. For more detail on the POISQuery messages, see Section 11.14 (POISQueryRequest and Response Messages) and Appendix A.3 (POISQueryRequest and Response Message Examples) of [SCTE130-5].
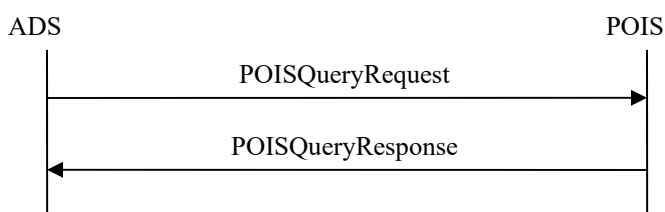
<div align="center">**Figure 10: ADS-Centric VOD POIS Query Message Exchange**</div>

Using BasicQueryFilters, the ADS can query for PlacementOpportunity information for a specified asset.

```
<POISQueryRequest messageId="uuid-18" system="POISClient" identity="40DA910E-01AF-5050-C7EA-
5D7B4A475757" version="1.0">
        <gis:Query queryId="pois-98765" expandOutput="true" uniqueQualifierNameRef="P3-PODM">
                <gis:ServiceDataModel>"http://www.scte.org/schemas/130-
3/2008a/adm/podm"</gis:ServiceDataModel>
                <gis:BasicQueryFilter>
                        <gis:BasicFilterElement value="ABCD0123456789012345" name="assetId"/>
                </gis:BasicQueryFilter>
                <gis:BasicQueryFilter>
                        <gis:BasicFilterElement value="entertainment.com" name="providerId"/>
                </gis:BasicQueryFilter>
        </gis:Query>
</POISQueryRequest>
```

<div align="center">**Example 16**</div>

The POIS then responds with a POISQueryResponse that includes all of the PlacementOpportunity metadata it has for the entertainment. The example below illustrates that the content contains one 30-second, local, preRoll placement opportunity available for ad placement.

```
<POISQueryResponse version="1.0" messageId="uuid-19" messageRef="uuid-18"
system="POISServer" identity="50DA910E-01AF-5050-C7EA-5D7B4A475759">
      <core:StatusCode class="0"/>
      <gis:QueryResult resultSetSize="1" queryRef="pois-98765">
            <podm:PODMBasicQueryResult>
                  <podm:PlacementOpportunityV2 serviceRegistrationRef="-"
id="uuid-20">
                        <adm:Entertainment>
                              <core:Content>
                                    <core:AssetRef
providerID="entertainment.com" assetID="ABCD0123456789012345"/>
                              </core:Content>
                        </adm:Entertainment>
                        <adm:OpportunityBinding opportunityType="preRoll"
opportunitiesExpected="1" opportunityNumber="1"/>
                        <podm:OpportunityConstraintsV2>
                              <core:Duration>PT30S</core:Duration>
                              <adm:Scope>Local</adm:Scope>
                        </podm:OpportunityConstraintsV2>
                  </podm:PlacementOpportunityV2>
            </podm:PODMBasicQueryResult>
      </gis:QueryResult>
</POISQueryResponse>
```

Example 17

While the previous section mentioned that caching and registering for notifications may improve performance, this may not be advisable in all situations due to the nature of some VOD ecosystems. In a situation where ad decisions are not all determined at the start of a VOD session and requested in real time as they occur in the stream, the ADS could be overwhelmed with notifications if the POIS is manipulating PlacementOpportunity metadata based on time of day.

7.2.4   SISQueryRequest/SISQueryResponse

The ADS may also obtain additional profile data for the subscriber initiating the VOD session. For more detail on the SISQuery messages, see Section 10.14 (SISQueryRequest and SISQueryResponse) and Appendix A.3 (SIS Query Request and Response) of [SCTE130-6].
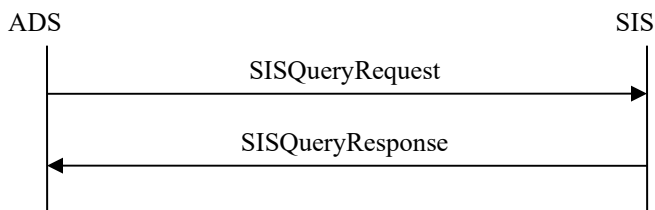
Figure 11: ADS-Centric VOD SIS Query Message Exchange

Using a UniqueQualifier with the client terminal's MAC address, the ADS can query for subscriber information.

```
<SISQueryRequest messageId="uuid-21" system="SISClient" version="1.0"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
      <gis:Query queryId="233" expandOutput="false">
      <gis:ServiceDataModel>http://SuperDemograpics.com</gis:ServiceDataModel
>
            <gis:UniqueQualifier>
                  <gis:Qualifier name="MACAddress" value="03-55-78-90-03-
02"/>
            </gis:UniqueQualifier>
      </gis:Query>
</SISQueryRequest>
```

Example 18

The SIS then responds with a SISQueryResponse that includes audience qualifiers it has for the queried subscriber.

```
<SISQueryResponse messageId="uuid-22" system="SISServer" version="1.0"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475312" messageRef="uuid-21">
      <core:StatusCode class="0"/>
      <gis:QueryResult queryRef="233" resultSetSize="1"
totalResultSetSize="1">
            <gis:BasicQueryResult>
                  <gis:QualifierSet>
                        <gis:Qualifier name="MACAddress" value="03-55-78-90-
03-02"/>
                        <gis:Qualifier name="Age" value="Under20"/>
                        <gis:Qualifier name="Income" value="Under50K"/>
                        <gis:Qualifier name="ZipCode" value="01720"/>
                        <gis:Qualifier name="CreditLimit" value="100.00"/>
                        <gis:Qualifier name="SportsInterest"
value="Basketball"/>
                  </gis:QualifierSet>
            </gis:BasicQueryResult>
      </gis:QueryResult>
</SISQueryResponse>
```

Example 19

7.2.5   PlacementResponse

Armed with all of the entertainment asset metadata, PlacementOpportunity parameters, and subscriber audience qualifiers the ADS can now make an informed ad decision. For more detail on the Placement messages, see Section 12.1 (Placement Messaging) of [SCTE130-3].
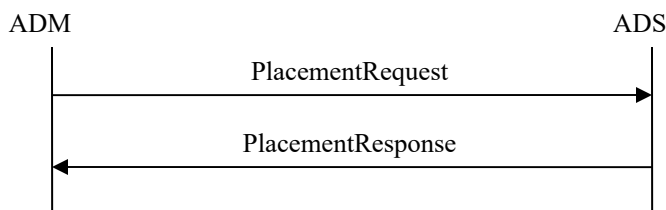
Figure 12: ADS-centric VOD Placement Response Message Exchange

The ADS responds to the ADM with an ad placement for the preRoll placement opportunity.

```
<PlacementResponse messageId="uuid-23" version="1.1"
identity="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5"
system="ads" messageRef="uuid-14">
  <core:StatusCode class="0"/>
  <PlacementDecision id="PD1" placementOpportunityRef="5B8ECCED-979B-421E-
8A6E-E2192B115162">
    <Placement id="uuid-20" action="fill">
      <core:Content>
        <core:AssetRef providerID="adsrus.com"
assetID="ZXYW0123456789012345"/>
        <core:Tracking>uuid-24</core:Tracking>
      </core:Content>
    </Placement>
  </PlacementDecision>
</PlacementResponse>
```

Example 20

The VOD system is responsible for the ad being inserted before the entertainment content starts.

7.2.6   PlacementStatusNotification/PlacementStatusAcknowledgement

After the VOD session ends, event data is required to be returned to the ADS from the ADM via PlacementStatusNotification. For more detail on the PlacementStatus messages, see Section 12.3 (Placement Status Messaging) of [SCTE130-3].
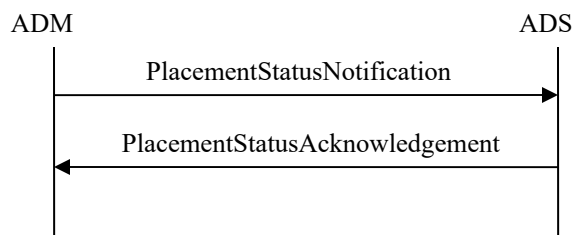
Figure 13: ADS-centric VOD Placement Status Message Exchange

The PlacementStatusNotification contains startPlacement and endPlacement events for the preRoll ad that was played prior to the entertainment content.

```
<PlacementStatusNotification messageId="uuid-25" version="1.1"
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>uuid-24</core:Tracking>
          </core:Content>
          <SpotNPT scale="1">0</SpotNPT>
        </Spot>
      </PlacementStatusEvent>
      <PlacementStatusEvent type="endPlacement" time="2007-01-
09T14:20:30.0Z">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="adsrus.com"
assetID="ZXYW0123456789012345"/>
            <core:Tracking>uuid-24</core:Tracking>
          </core:Content>
          <SpotNPT scale="1">30</SpotNPT>
        </Spot>
      </PlacementStatusEvent>
    </Events>
  </PlayData>
</PlacementStatusNotification>
```

Example 21

The ADS responds with a PlacementStatusAcknowledgement.

```
<PlacementStatusAcknowledgement messageId="uuid-26" version="1.1"
identity="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5"
system="ads" messageRef="uuid-25" >
  <core:StatusCode class="0"/>
</PlacementStatusAcknowledgement>
```

Example 22

7.3   ADM-centric VOD
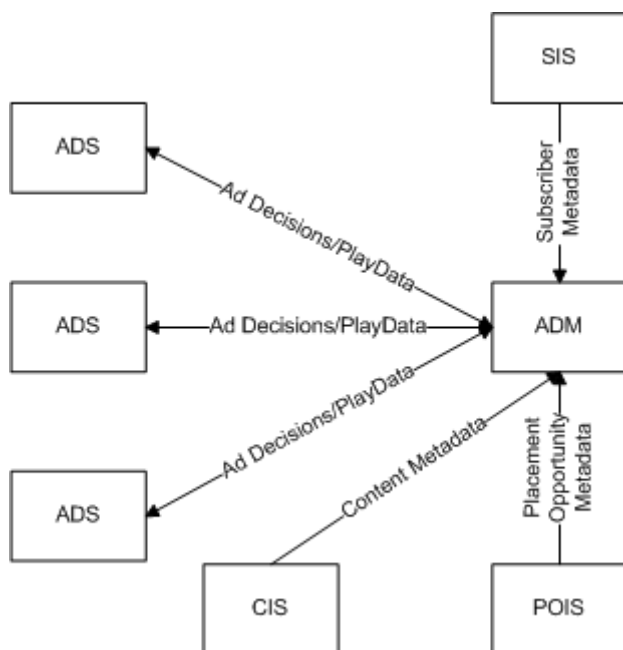
Figure 14: ADM-centric VOD Messaging Overview

In an ADM-centric view of SCTE 130 (shown above), the ADM is responsible for communication with the other SCTE 130 information services to obtain entertainment asset metadata, details of Placement Opportunities, and subscriber profile information. In this model the ADM may issue requests to multiple ADSs which have registered for various services and/or PO types/scopes which the ADM is thus able to distinguish. The ADM will place the ad decisions determined by the ADS. Registration is required between each ADS and the ADM (outlined in Section 0) before the placement messaging in the sequence diagram below is attempted. In the case of multiple ADSs, the registration process is repeated for each server.
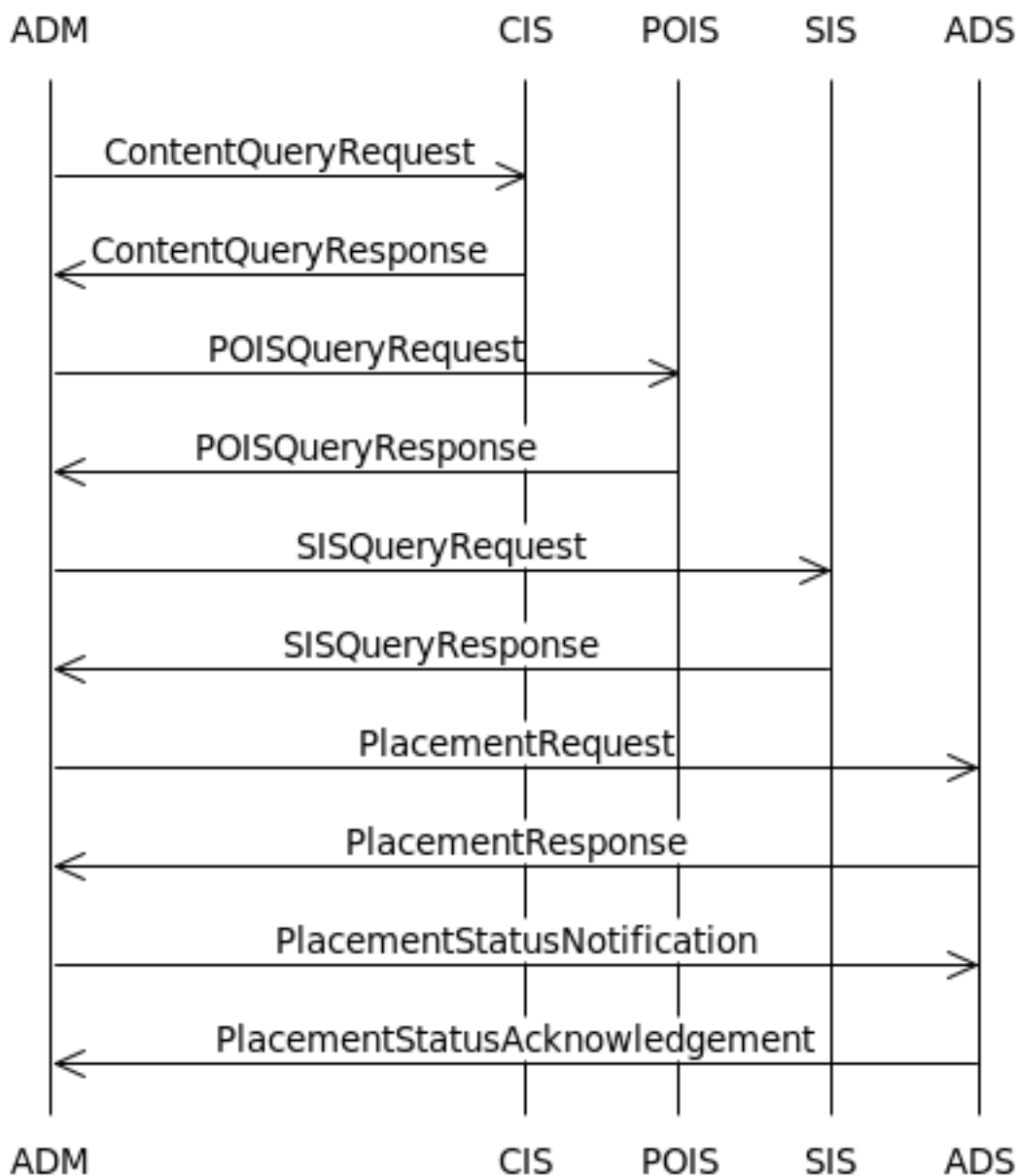
Figure 15: ADM-centric VOD Sequence Diagram

7.3.1   ContentQueryRequest/ContentQueryResponse

Before sending a PlacementRequest to the ADS, the ADM may query the CIS for additional information about the entertainment asset to provide the ADS with more metadata to aid in its decision.
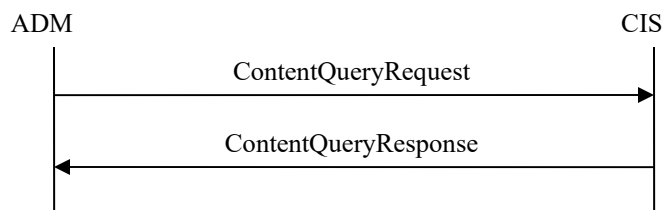
Figure 16: ADM-centric VOD Content Query Message Exchange

In order to retrieve the complete metadata for the asset, the @expandOutput attribute is required to be set to true.

```
<ContentQueryRequest messageId="uuid-27" system="sys-1" version ="1.1"
    identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
    <ContentQuery expandOutput="true" contentQueryId="1">
       <core:ContentDataModel type="CLADI_1.1">URI</core:ContentDataModel>
       <QueryFilter>
            <FilterElement name="Provider_ID"  value="entertainment.com" />
            <FilterElement name="Asset_ID"  value="ABCD01234567890012345" />
       </QueryFilter>
    </ContentQuery>
</ContentQueryRequest>
```

Example 23

The CIS will respond with the full metadata in the specified content data model, CLADI 1.1.

```
<ContentQueryResponse messageId="uuid-28" system="acs-1" version="1.1"
     identity="40DA910E-01AF-5050-C7EA-5D7B4A475312" messageRef="uuid-27">
    <core:StatusCode class="0"/>
    <ContentQueryResult resultSetSize="1" contentQueryRef="1">
       <BasicQueryResultList>
            <core:Content>
                <core:ContentLocation
mediaAvailable="true">URI</core:ContentLocation>
                <core:Ext>
                    <ADI>
                        <Metadata> <AMS Asset_Name="..."/> …</Metadata>
                    </ADI>
                </core:Ext>
            </core:Content>
       </BasicQueryResultList>
    </ContentQueryResult>
</ContentQueryResponse>
```

Example 24

For readability, the actual CLADI 1.1 metadata has been removed and posted below. The Product, highlighted below, will correspond to the ProductType in the PlacementRequest sent by the ADM.

```
<ADI>
  <Metadata>
    <AMS Asset_Class="package" Asset_ID="ABCD01234567890123343" Asset_Name="SCTE 130: The Movie"
Creation_Date="2008-02-11" Description="SCTE 130: The Movie" Product="MOD"
Provider="entertainment" Provider_ID="entertainment.com" Version_Major="1" Version_Minor="0"/>
    <App_Data App="MOD" Name="Metadata_Spec_Version" Value="CableLabsVOD1.1"/>
    <App_Data App="MOD" Name="Provider_Content_Tier" Value="ENTERTAINMENT_NEWRELEASE"/>
  </Metadata>
  <Asset>
    <Metadata>
      <AMS Asset_Class="title" Asset_ID="ABCD01234567890123344" Asset_Name="SCTE 130: The Movie"
Creation_Date="2008-02-11" Description="SCTE 130: The Movie" Product="MOD"
Provider="entertainment" Provider_ID="entertainment.com" Version_Major="1" Version_Minor="0"/>
      <App_Data App="MOD" Name="Title" Value="SCTE 130: The Movie"/>
      <App_Data App="MOD" Name="Title_Brief" Value="SCTE 130: The Movie"/>
      <App_Data App="MOD" Name="Type" Value="title"/>
      <App_Data App="MOD" Name="Year" Value="2009"/>
      <App_Data App="MOD" Name="Director" Value="Bryan Santangelo"/>
      <App_Data App="MOD" Name="Actors_Display" Value="John Mick, Corey Brown"/>
      <App_Data App="MOD" Name="Actors" Value="John Mick"/>
      <App_Data App="MOD" Name="Actors" Value="Corey Brown"/>
      <App_Data App="MOD" Name="Genre" Value="Action & Adventure"/>
      <App_Data App="MOD" Name="Category" Value="On Demand/High Definition/HD Movies"/>
      <App_Data App="MOD" Name="Category" Value="On Demand/Movies/HD Movies"/>
      <App_Data App="MOD" Name="Category" Value="On Demand/Movies/Movies By Title/Movies S - Z"/>
      <App_Data App="MOD" Name="Category" Value="On Demand/Movies/New Releases/High Definition"/>
      <App_Data App="MOD" Name="Category" Value="On Demand/What's Hot/HD Movies"/>
      <App_Data App="MOD" Name="Billing_ID" Value="60671"/>
      <App_Data App="MOD" Name="Box_Office" Value="24000000"/>
      <App_Data App="MOD" Name="Closed_Captioning" Value="Y"/>
      <App_Data App="MOD" Name="Contract_Name" Value="ENT"/>
      <App_Data App="MOD" Name="Display_As_Last_Chance" Value="7"/>
      <App_Data App="MOD" Name="Display_As_New" Value="7"/>
      <App_Data App="MOD" Name="Display_Run_Time" Value="02:03"/>
      <App_Data App="MOD" Name="Distributor_Name" Value="ENT"/>
      <App_Data App="MOD" Name="Distributor_Royalty_Minimum" Value="2.59"/>
      <App_Data App="MOD" Name="Distributor_Royalty_Percent" Value="65.0"/>
      <App_Data App="MOD" Name="Licensing_Window_End" Value="2015-01-01"/>
      <App_Data App="MOD" Name="Licensing_Window_Start" Value="2008-03-21"/>
      <App_Data App="MOD" Name="Maximum_Viewing_Length" Value="01:00:00"/>
      <App_Data App="MOD" Name="Preview_Period" Value="300"/>
      <App_Data App="MOD" Name="Provider_QA_Contact" Value="help@ent.com"/>
      <App_Data App="MOD" Name="Rating" Value="R"/>
      <App_Data App="MOD" Name="Run_Time" Value="02:03:58"/>
      <App_Data App="MOD" Name="Studio" Value="SCTE"/>
      <App_Data App="MOD" Name="Studio_Name" Value="SCTE"/>
      <App_Data App="MOD" Name="Suggested_Price" Value="5.99"/>
      <App_Data App="MOD" Name="Summary_Short" Value="The adventures of a DVS subcommittee"/>
    </Metadata>
    <Asset>
      <Metadata>
        <AMS Asset_Class="movie" Asset_ID="ABCD01234567890123345" Asset_Name="SCTE 130: The Movie"
Creation_Date="2008-02-11" Description="SCTE 130: The Movie" Product="MOD"
Provider="entertainment" Provider_ID="entertainment.com" Version_Major="1" Version_Minor="0"/>
        <App_Data App="MOD" Name="Audio_Type" Value="Stereo"/>
        <App_Data App="MOD" Name="Type" Value="movie"/>
        <App_Data App="MOD" Name="HDContent" Value="Y"/>
      </Metadata>
      <Content Value="entertainment.com::TVNX10084127342::On_Demand.mpeg"/>
    </Asset>
    <Asset>
      <Metadata>
        <AMS Asset_Class="preview" Asset_ID="ABCD01234567890123346" Asset_Name="SCTE 130: The
Movie" Creation_Date="2008-02-11" Description="SCTE 130: The Movie" Product="MOD"
Provider="entertainment" Provider_ID="entertainment.com" Version_Major="1" Version_Minor="0"/>
        <App_Data App="MOD" Name="Audio_Type" Value="Stereo"/>
        <App_Data App="MOD" Name="Rating" Value="R"/>
        <App_Data App="MOD" Name="Run_Time" Value="00:05:00"/>
        <App_Data App="MOD" Name="Type" Value="preview"/>
      </Metadata>
      <Content Value="entertainment.com::TVNX10084127343::On_Demand.mpeg"/>
```

```
    </Asset>
  </Asset>
</ADI>
```

<p align="center">Example 25</p>

The figure below illustrates the link between the data returned in the ContentQueryResponse and the PlacementRequest. The top section is an excerpt from the ContentQueryResponse in Example 25 (specifically, the metadata excerpt in Example 25), and the bottom section is from the PlacementRequest in Example 30.

```
        <AMS Asset_Class="movie" Asset_ID="ABCD01234567890123459" Asset_Name="SCTE 130: The Movie"
Creation_Date="2008-02-11" Description="SCTE 130: The Movie" Product="MOD"
Provider="entertainment" Provider ID="entertainment.com" Version_Major="1" Version Minor="0"/>


        <Service id=" Service1">
          <ProductType>MOD</ProductType>
        </Service>
```

<p align="center">Figure 17: ADM-centric VOD CIS/PlacementRequest Link</p>

With all of the calls to the SCTE 130 information services (CIS, SIS, POIS), an ADM may prefer to register for notifications and cache metadata in an effort to improve performance.

7.3.2   POISQueryRequest/POISQueryResponse

The ADM now needs to query the POIS to determine if any placement opportunities exist in the content.
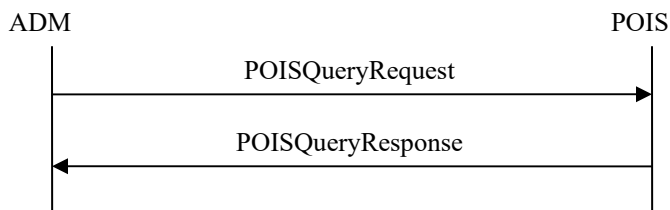
<p align="center">Figure 18: ADM-centric VOD POIS Query Message Exchange</p>

Using BasicQueryFilters, the ADM can query for PlacementOpportunity information for a specified asset.

```
<POISQueryRequest messageId="uuid-29" system="POISClient" identity="40DA910E-
01AF-5050-C7EA-5D7B4A475757" version="1.0">
     <gis:Query queryId="pois-98765" expandOutput="true"
uniqueQualifierNameRef="P3-PODM">
          <gis:ServiceDataModel>http://www.scte.org/schemas/130-
3/2008a/adm/podm</gis:ServiceDataModel>
          <gis:BasicQueryFilter>
               <gis:BasicFilterElement value="ABCD0123456789012345"
name="assetId"/>
          </gis:BasicQueryFilter>
          <gis:BasicQueryFilter>
               <gis:BasicFilterElement value="entertainment.com"
name="providerId"/>
          </gis:BasicQueryFilter>
     </gis:Query>
</POISQueryRequest>
```

Example 26

The POIS then responds with a POISQueryResponse that includes all of the metadata it has for the queried content. In this response, the ADM learns that one preRoll PlacementOpportunity exists. The Duration, Scope, and @opportunityType attribute highlighted below will correspond to the Duration, Scope, and @opportunityType attribute in the PlacementRequest sent by the ADM.

```
<POISQueryResponse version="1.1" messageId="uuid-30" messageRef="uuid-29"
system="POISServer" identity="50DA910E-01AF-5050-C7EA-5D7B4A475759">
     <core:StatusCode class="0"/>
     <gis:QueryResult resultSetSize="1" queryRef="pois-98765">
          <podm:PODMBasicQueryResult>
               <podm:PlacementOpportunityV2 serviceRegistrationRef="-"
id="uuid-31">
                    <adm:Entertainment>
                         <core:Content>
                              <core:AssetRef
providerID="entertainment.com" assetID="ABCD0123456789012345"/>
                         </core:Content>
                    </adm:Entertainment>
                    <adm:OpportunityBinding opportunityType="preRoll"
opportunitiesExpected="1" opportunityNumber="1"/>
                    <podm:OpportunityConstraintsV2>
                         <core:AdType>30-Second-Spot</core:AdType>
                         <core:Duration>PT30S</core:Duration>
                         <adm:Scope>Local</adm:Scope>
                    </podm:OpportunityConstraintsV2>
               </podm:PlacementOpportunityV2>
          </podm:PODMBasicQueryResult>
     </gis:QueryResult>
</POISQueryResponse>
```

Example 27

The figure below illustrates the link between the data returned in the POISQueryResponse and the PlacementRequest. The top section is an excerpt from the POISQueryResponse in Example 27, and the bottom section is from the PlacementRequest in Example 30.

```
<podm:PlacementOpportunityV2 serviceRegistrationRef="-" id="THS000000000000000009">
    <adm:Entertainment>
        <core:Content>
            <core:AssetRef providerID="entertainment.com" assetID="ABCD0123456789012345"/>
        </core:Content>
    </adm:Entertainment>
    <adm:OpportunityBinding opportunityType="preRoll" opportunitiesExpected="1" opportunityNumber="1"/>
    <podm:OpportunityConstraintsV2>
        <core:AdType>30-Second-Spot</core:AdType>
        <core:Duration>PT30S</core:Duration>
        <adm:Scope>Local</adm:Scope>
    </podm:OpportunityConstraintsV2>
</podm:PlacementOpportunityV2>

<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>
            <Duration>PT30S</Duration>
        </OpportunityConstraints>
    </PlacementControl>
</PlacementOpportunity>
```

Figure 19: ADM-centric VOD POIS/PlacementRequest Link

While the previous section mentioned that caching and registering for notifications may improve performance, this may not be advisable in all situations due to the nature of some VOD ecosystems. In a situation where ad decisions are not all determined at the start of a VOD session and requested in real time as they occur in the stream, the ADM could be overwhelmed with notifications if the POIS is manipulating PlacementOpportunity metadata based on time of day.

7.3.3 SISQueryRequest/SISQueryResponse

The ADM may also obtain additional profile data for the subscriber initiating the VOD session.

ADM   SIS

          SISQueryRequest →

          ← SISQueryResponse
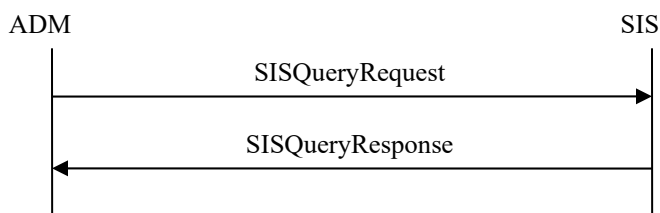
Figure 20: ADM-centric VOD SIS Query Message Exchange

Using a UniqueQualifier with the client terminal's MAC address, the ADM can query for subscriber information.

```
<SISQueryRequest messageId="uuid-32" system="SISClient" version="1.0"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
      <gis:Query queryId="233" expandOutput="false">
      <gis:ServiceDataModel>http://SuperDemograpics.com</gis:ServiceDataModel
>
            <gis:UniqueQualifier>
                  <gis:Qualifier name="MACAddress" value="03-55-78-90-03-
02"/>
            </gis:UniqueQualifier>
      </gis:Query>
</SISQueryRequest>
```

Example 28

The SIS then responds with a SISQueryResponse that includes audience qualifiers it has for the queried subscriber. The Qualifiers, highlighted below, will correspond to TargetCodes in the PlacementRequest sent by the ADM.

```
<SISQueryResponse messageId="uuid-33" system="SISServer" version="1.0"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475312" messageRef="uuid-32">
      <core:StatusCode class="0"/>
      <gis:QueryResult queryRef="233" resultSetSize="1"
totalResultSetSize="1">
            <gis:BasicQueryResult>
                  <gis:QualifierSet>
                        <gis:Qualifier name="MACAddress" value="03-55-78-90-
03-02"/>
                        <gis:Qualifier name="Age" value="Under20"/>
                        <gis:Qualifier name="Income" value="Under50K"/>
                        <gis:Qualifier name="ZipCode" value="01720"/>
                        <gis:Qualifier name="CreditLimit" value="100.00"/>
                        <gis:Qualifier name="SportsInterest"
value="Basketball"/>
                  </gis:QualifierSet>
            </gis:BasicQueryResult>
      </gis:QueryResult>
</SISQueryResponse>
```

Example 29

The figure below illustrates the link between the data returned in the SISQueryResponse and the PlacementRequest. The top section is an excerpt from the SISQueryResponse in Example 29, and the bottom section is from the PlacementRequest in Example 30.

```
<gis:Qualifier name="Age" value="Under20">
<gis:Qualifier name="Income" value="Under50K"/>
<gis:Qualifier name="ZipPlusFour" value="01720"/>
<gis:Qualifier name="CreditLimit" value="100.00"/>
<gis:Qualifier name="SportsInterest" value="Basketball"/>


<TargetCode key="Age">Under20<TargetCode/>
<TargetCode key="Income">Under50K<TargetCode/>
<TargetCode key="ZipPlusFour">01720<TargetCode/>
<TargetCode key="CreditLimit">100.00<TargetCode/>
<TargetCode key="SportsInterest">Basketball<TargetCode/>
```

Figure 21: ADM-centric VOD SIS/PlacementRequest Link

With all of the calls to the SCTE 130 information services (CIS, SIS, POIS), an ADM may prefer to register for notifications and cache metadata in an effort to improve performance.

7.3.4   PlacementRequest/PlacementResponse

Armed with all of the entertainment asset metadata, PlacementOpportunity parameters, and subscriber audience qualifiers the ADM can now send complete PlacementRequests to each of the registered ADSs by distinguishing values such as PO type, scope, etc.

ADM   ADS

PlacementRequest

PlacementResponse
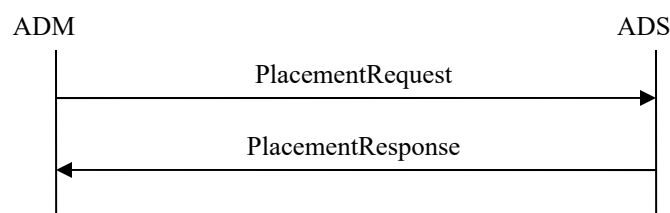
Figure 22: ADM-centric VOD Placement Message Exchange

The ADM's PlacementRequest contains one PlacementOpportunity for a preRoll.

```
<PlacementRequest messageId="uuid-34" version="1.1"
identity="ADMLogicalService_1170FEF4-C19B-450E-B624-421B23F525F4"
system="adm" updatesAllowed="true">
  <Service id=" Service1">
    <ProductType>MOD</ProductType>
  </Service>
  <Entertainment>
    <core:Content>
      <core:AssetRef providerID="entertainment.com"
assetID="ABCD0123456789012345"/>
    </core:Content>
  </Entertainment>
  <Client>
    <TargetCode key="Age">Under20</TargetCode>
    <TargetCode key="Income">Under50K</TargetCode>
    <TargetCode key="ZipCode">01720</TargetCode>
    <TargetCode key="CreditLimit">100.00</TargetCode>
    <TargetCode key="SportsInterest">Basketball</TargetCode>
  </Client>
  <PlacementOpportunity id="uuid-35" serviceRegistrationRef="ServiceRef1">
    <OpportunityBinding opportunityType="preRoll"/>
    <OpportunityConstraints>
      <Scope>Local</Scope>
    </OpportunityConstraints>
    <PlacementControl id="uuid-36" action="fill">
      <OpportunityConstraints>
        <core:Duration>PT30S</core:Duration>
      </OpportunityConstraints>
    </PlacementControl>
  </PlacementOpportunity>
</PlacementRequest>
```

Example 30

The ADS responds to the ADM with ad placements for the preRoll placement opportunity.

```
<PlacementResponse messageId="uuid-37" version="1.1"
identity="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5"
system="ads" messageRef="uuid-34">
  <core:StatusCode class="0"/>
  <ADMData><!--Hidden secret sauce--></ADMData>
  <PlacementDecision id="uuid-38" placementOpportunityRef="uuid-35">
    <Placement id="uuid-39" action="fill" placementControlRef="uuid-36">
      <core:Content>
        <core:AssetRef providerID="adsrus.com"
assetID="ZXYW0123456789012345"/>
        <core:Tracking>uuid-40</core:Tracking>
      </core:Content>
    </Placement>
  </PlacementDecision>
</PlacementResponse>
```

Example 31

7.3.5 PlacementStatusNotification/PlacementStatusAcknowledgement

After the VOD session ends, event data is required to be returned to the ADS from the ADM via PlacementStatusNotification.
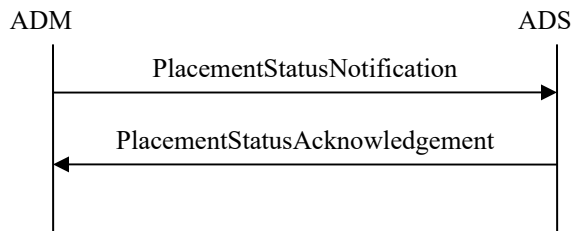
Figure 23: ADM-centric VOD Placement Status Message Exchange

```
<PlacementStatusNotification messageId="uuid-41" version="1.1"
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>uuid-40</core:Tracking>
          </core:Content>
          <SpotNPT scale="1">0</SpotNPT>
        </Spot>
      </PlacementStatusEvent>
      <PlacementStatusEvent type="endPlacement" time="2007-01-
09T14:20:30.0Z">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="adsrus.com"
assetID="ZXYW0123456789012345"/>
            <core:Tracking>uuid-40</core:Tracking>
          </core:Content>
          <SpotNPT scale="1">30</SpotNPT>
        </Spot>
      </PlacementStatusEvent>
    </Events>
  </PlayData>
</PlacementStatusNotification>
```

Example 32

The ADS responds with a PlacementStatusAcknowledgement.

```
<PlacementStatusAcknowledgement messageId="uuid-42" version="1.1"
identity="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5"
system="ads" messageRef="uuid-41" >
  <core:StatusCode class="0"/>
</PlacementStatusAcknowledgement>
```

Example 33

7.4 VOD with an Intermediary
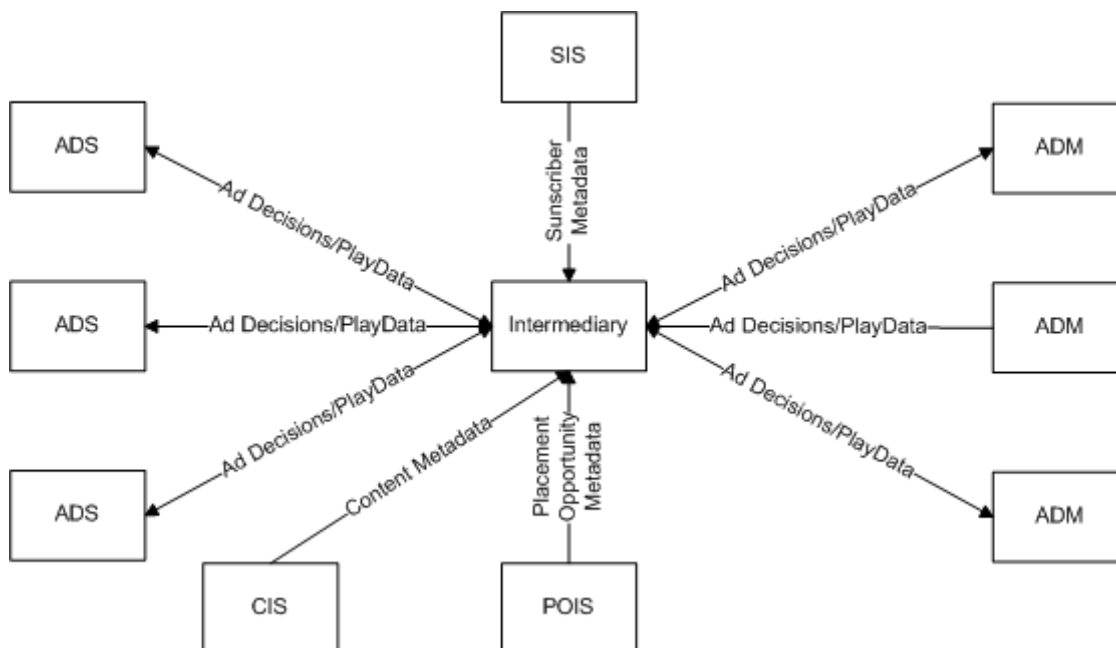
Figure 24: VOD with an Intermediary Messaging Overview

In some systems, an intermediary server may be deployed between one or more ADMs and one or more ADSs. The Intermediary is responsible for communication with the other SCTE 130 information services to obtain entertainment asset metadata, details of Placement Opportunities, and subscriber profile information. In this model (shown above) the Intermediary may issue requests to multiple ADSs which have registered for various services and/or PO types/scopes which the Intermediary is thus able to distinguish. The ADM will place the ad decisions determined by the Intermediary. Registration is required between each ADS and the Intermediary (outlined in Section 0) before the placement messaging in the sequence diagram below is attempted. In the case of multiple ADSs, the registration process is repeated for each server. How an Intermediary reconciles a difference in PODM support between an ADM and ADS is not addressed in this document.
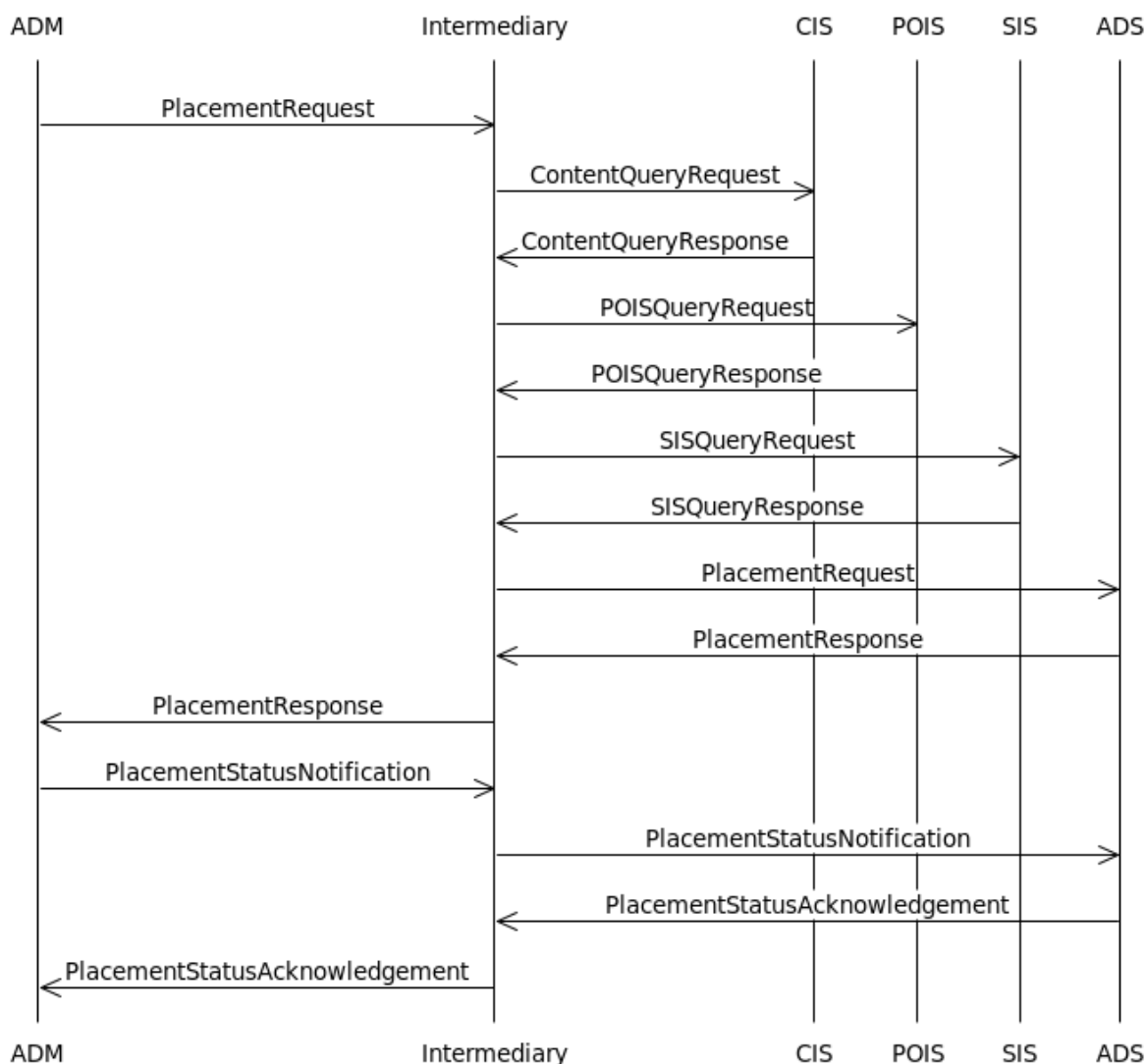
Figure 25: VOD with an Intermediary Sequence Diagram

7.4.1   PlacementRequest

At the start of a VOD session, the ADM will query the Intermediary with a PlacementRequest. The ADM's PlacementRequest contains one PlacementOpportunity as required by the schema, but the ADM has no metadata for the content, placement opportunity, or subscriber other than the client terminal's MAC address and the entertainment content provider ID and asset ID. This leaves the Intermediary responsible for querying information services (if available) to give the ADS the data it needs to make an informed ad decision.
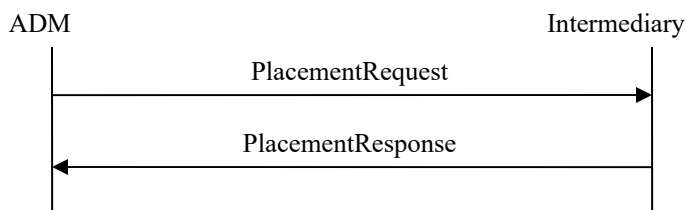
Figure 26: VOD with an Intermediary Placement Request Message Exchange

The ADM's PlacementRequest contains one PlacementOpportunity for a preRoll.

```
<PlacementRequest messageId="uuid-43" version="1.1"
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>
    <TerminalAddress type="MAC">03-55-78-90-03-02</TerminalAddress>
  </Client>
  <PlacementOpportunity id="uuid-44" serviceRegistrationRef="ServiceRef1">
  </PlacementOpportunity>
</PlacementRequest>
```

Example 34

7.4.2   ContentQueryRequest/ContentQueryResponse

Before sending a PlacementRequest to the ADS, the Intermediary may query the CIS for additional information about the entertainment asset to provide the ADS with more metadata to aid in its decision.
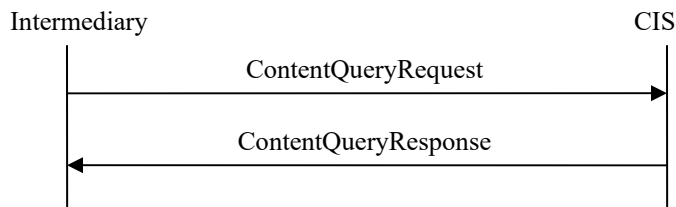
Figure 27: VOD with an Intermediary Content Query Message Exchange

In order to retrieve the complete metadata for the asset, the @expandOutput attribute is required to be set to true.

```
<ContentQueryRequest messageId="uuid-45" system="sys-1" version ="1.1"
    identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
    <ContentQuery expandOutput="true" contentQueryId="1">
       <core:ContentDataModel type="CLADI_1.1">URI</core:ContentDataModel>
       <QueryFilter>
           <FilterElement name="Provider_ID"  value="entertainment.com" />
           <FilterElement name="Asset_ID"  value="ABCD0123456789012345" />
       </QueryFilter>
    </ContentQuery>
</ContentQueryRequest>
```

Example 35

The CIS responds with the full metadata in the specified content data model, CLADI 1.1.

```
<ContentQueryResponse messageId="uuid-46" system="acs-1" version="1.1"
     identity="40DA910E-01AF-5050-C7EA-5D7B4A475312" messageRef="uuid-45">
    <core:StatusCode class="0"/>
    <ContentQueryResult resultSetSize="1" contentQueryRef="1">
       <BasicQueryResultList>
           <core:Content>
               <core:ContentLocation
mediaAvailable="true">URI</core:ContentLocation>
               <core:Ext>
                   <ADI>
                       <Metadata> <AMS Asset_Name="..."/> …</Metadata>
                   </ADI>
               </core:Ext>
           </core:Content>
       </BasicQueryResultList>
    </ContentQueryResult>
</ContentQueryResponse>
```

Example 36

For readability, the actual CLADI 1.1 metadata has been removed and posted below. The Product, highlighted below, will correspond to the ProductType in the PlacementRequest sent by the Intermediary.

```
<ADI>
  <Metadata>
    <AMS Asset_Class="package" Asset_ID="ABCD01234567890123 43" Asset_Name="SCTE 130: The Movie"
Creation_Date="2008-02-11" Description="SCTE 130: The Movie" Product="MOD"
Provider="entertainment" Provider_ID="entertainment.com" Version_Major="1" Version_Minor="0"/>
    <App_Data App="MOD" Name="Metadata_Spec_Version" Value="CableLabsVOD1.1"/>
    <App_Data App="MOD" Name="Provider_Content_Tier" Value="ENTERTAINMENT_NEWRELEASE"/>
  </Metadata>
  <Asset>
    <Metadata>
      <AMS Asset_Class="title" Asset_ID="ABCD01234567890123 44" Asset_Name="SCTE 130: The Movie"
Creation_Date="2008-02-11" Description="SCTE 130: The Movie" Product="MOD"
Provider="entertainment" Provider_ID="entertainment.com" Version_Major="1" Version_Minor="0"/>
      <App_Data App="MOD" Name="Title" Value="SCTE 130: The Movie"/>
      <App_Data App="MOD" Name="Title_Brief" Value="SCTE 130: The Movie"/>
      <App_Data App="MOD" Name="Type" Value="title"/>
      <App_Data App="MOD" Name="Year" Value="2009"/>
      <App_Data App="MOD" Name="Director" Value="Bryan Santangelo"/>
      <App_Data App="MOD" Name="Actors_Display" Value="John Mick, Corey Brown"/>
      <App_Data App="MOD" Name="Actors" Value="John Mick"/>
      <App_Data App="MOD" Name="Actors" Value="Corey Brown"/>
      <App_Data App="MOD" Name="Genre" Value="Action & Adventure"/>
      <App_Data App="MOD" Name="Category" Value="On Demand/High Definition/HD Movies"/>
      <App_Data App="MOD" Name="Category" Value="On Demand/Movies/HD Movies"/>
      <App_Data App="MOD" Name="Category" Value="On Demand/Movies/Movies By Title/Movies S - Z"/>
      <App_Data App="MOD" Name="Category" Value="On Demand/Movies/New Releases/High Definition"/>
      <App_Data App="MOD" Name="Category" Value="On Demand/What's Hot/HD Movies"/>
      <App_Data App="MOD" Name="Billing_ID" Value="60671"/>
      <App_Data App="MOD" Name="Box_Office" Value="24000000"/>
      <App_Data App="MOD" Name="Closed_Captioning" Value="Y"/>
      <App_Data App="MOD" Name="Contract_Name" Value="ENT"/>
      <App_Data App="MOD" Name="Display_As_Last_Chance" Value="7"/>
      <App_Data App="MOD" Name="Display_As_New" Value="7"/>
      <App_Data App="MOD" Name="Display_Run_Time" Value="02:03"/>
      <App_Data App="MOD" Name="Distributor_Name" Value="ENT"/>
      <App_Data App="MOD" Name="Distributor_Royalty_Minimum" Value="2.59"/>
      <App_Data App="MOD" Name="Distributor_Royalty_Percent" Value="65.0"/>
      <App_Data App="MOD" Name="Licensing_Window_End" Value="2015-01-01"/>
      <App_Data App="MOD" Name="Licensing_Window_Start" Value="2008-03-21"/>
      <App_Data App="MOD" Name="Maximum_Viewing_Length" Value="01:00:00"/>
      <App_Data App="MOD" Name="Preview_Period" Value="300"/>
      <App_Data App="MOD" Name="Provider_QA_Contact" Value="help@ent.com"/>
      <App_Data App="MOD" Name="Rating" Value="R"/>
      <App_Data App="MOD" Name="Run_Time" Value="02:03:58"/>
      <App_Data App="MOD" Name="Studio" Value="SCTE"/>
      <App_Data App="MOD" Name="Studio_Name" Value="SCTE"/>
      <App_Data App="MOD" Name="Suggested_Price" Value="5.99"/>
      <App_Data App="MOD" Name="Summary_Short" Value="The adventures of a DVS subcommittee"/>
    </Metadata>
    <Asset>
      <Metadata>
        <AMS Asset_Class="movie" Asset_ID="ABCD01234567890123 45" Asset_Name="SCTE 130: The Movie"
Creation_Date="2008-02-11" Description="SCTE 130: The Movie" Product="MOD"
Provider="entertainment" Provider_ID="entertainment.com" Version_Major="1" Version_Minor="0"/>
        <App_Data App="MOD" Name="Audio_Type" Value="Stereo"/>
        <App_Data App="MOD" Name="Type" Value="movie"/>
        <App_Data App="MOD" Name="HDContent" Value="Y"/>
      </Metadata>
      <Content Value="entertainment.com::TVNX10084127342::On_Demand.mpeg"/>
    </Asset>
    <Asset>
      <Metadata>
        <AMS Asset_Class="preview" Asset_ID="ABCD01234567890123 46" Asset_Name="SCTE 130: The
Movie" Creation_Date="2008-02-11" Description="SCTE 130: The Movie" Product="MOD"
Provider="entertainment" Provider_ID="entertainment.com" Version_Major="1" Version_Minor="0"/>
        <App_Data App="MOD" Name="Audio_Type" Value="Stereo"/>
        <App_Data App="MOD" Name="Rating" Value="R"/>
        <App_Data App="MOD" Name="Run_Time" Value="00:05:00"/>
        <App_Data App="MOD" Name="Type" Value="preview"/>
      </Metadata>
      <Content Value="entertainment.com::TVNX10084127343::On_Demand.mpeg"/>
```

```
      </Asset>
    </Asset>
</ADI>
```

Example 37

The figure below illustrates the link between the data returned in the ContentQueryResponse and the PlacementRequest. The top section is an excerpt from the ContentQueryResponse in Example 37, and the bottom section is from the PlacementRequest in Example 42.

```
        <AMS Asset_Class="movie" Asset_ID="ABCD0123456789012345" Asset_Name="SCTE 130: The Movie"
Creation_Date="2008-02-11" Description="SCTE 130: The Movie" Product="MOD"
Provider="entertainment" Provider_ID="entertainment.com" Version_Major="1" Version_Minor="0"/>


        <Service id=" Service1">
          <ProductType>MOD</ProductType>
        </Service>
```

Figure 28: Intermediary-centric VOD CIS/PlacementRequest Link

With all of the calls to the SCTE 130 information services (CIS, SIS, POIS), an Intermediary may prefer to register for notifications and cache metadata in an effort to improve performance.

7.4.3   POISQueryRequest/POISQueryResponse

The Intermediary now needs to obtain additional metadata for the PlacementOpportunity in the entertainment.
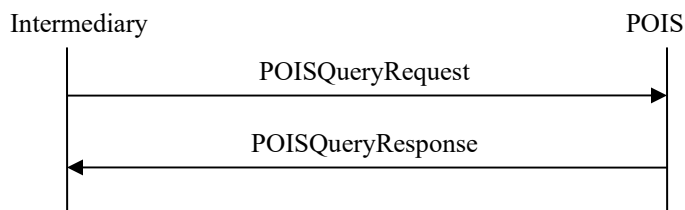
Figure 29: VOD with an Intermediary POIS Query Message Exchange

Using BasicQueryFilters, the Intermediary can query for PlacementOpportunity information for a particular asset.

```
<POISQueryRequest messageId="uuid-47" system="POISClient" identity="40DA910E-
01AF-5050-C7EA-5D7B4A475757" version="1.1">
      <gis:Query queryId="pois-98765" expandOutput="true"
uniqueQualifierNameRef="P3-PODM">
            <gis:ServiceDataModel>"http://www.scte.org/schemas/130-
3/2008a/adm/podm"</gis:ServiceDataModel>
            <gis:BasicQueryFilter>
                  <gis:BasicFilterElement value="ABCD0123456789012345"
name="assetId"/>
            </gis:BasicQueryFilter>
            <gis:BasicQueryFilter>
                  <gis:BasicFilterElement value="entertainment.com"
name="providerId"/>
            </gis:BasicQueryFilter>
      </gis:Query>
</POISQueryRequest>
```

Example 38

The POIS then responds with a POISQueryResponse that includes all of the metadata it has for
the queried content. In this response, the Intermediary learns that one preRoll
PlacementOpportunity exists. The Duration, Scope, and @opportunityType attribute highlighted
below will correspond to the Duration, Scope, and @opportunityType attribute in the
PlacementRequest sent by the Intermediary.

```
<POISQueryResponse version="1.1" messageId="uuid-48" messageRef="uuid-47"
system="POISServer" identity="50DA910E-01AF-5050-C7EA-5D7B4A475759">
      <core:StatusCode class="0"/>
      <gis:QueryResult resultSetSize="1" queryRef="pois-98765">
            <podm:PODMBasicQueryResult>
                  <podm:PlacementOpportunityV2 serviceRegistrationRef="-"
id="uuid-49">
                        <adm:Entertainment>
                              <core:Content>
                                    <core:AssetRef
providerID="entertainment.com" assetID="ABCD0123456789012345"/>
                              </core:Content>
                        </adm:Entertainment>
                        <adm:OpportunityBinding opportunityType="preRoll"
opportunitiesExpected="1" opportunityNumber="1"/>
                        <podm:OpportunityConstraintsV2>
                              <core:AdType>30-Second-Spot</core:AdType>
                              <core:Duration>PT30S</core:Duration>
                              <adm:Scope>Local</adm:Scope>
                        </podm:OpportunityConstraintsV2>
                  </podm:PlacementOpportunityV2>
            </podm:PODMBasicQueryResult>
      </gis:QueryResult>
</POISQueryResponse>
```

Example 39

The figure below illustrates the link between the data returned in the POISQueryResponse and the PlacementRequest. The top section is an excerpt from the POISQueryResponse in Example 39, and the bottom section is from the PlacementRequest in Example 42.

```
<podm:PlacementOpportunityV2 serviceRegistrationRef="-" id="THS000000000000000009">
    <adm:Entertainment>
        <core:Content>
            <core:AssetRef providerID="entertainment.com" assetID="ABCD0123456789012345"/>
        </core:Content>
    </adm:Entertainment>
    <adm:OpportunityBinding opportunityType="preRoll" opportunitiesExpected="1" opportunityNumber="1"/>
    <podm:OpportunityConstraintsV2>
        <core:AdType>30-Second-Spot</core:AdType>
        <core:Duration>PT30S</core:Duration>
        <adm:Scope>Local</adm:Scope>
    </podm:OpportunityConstraintsV2>
</podm:PlacementOpportunityV2>

<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>
            <Duration>PT30S</Duration>
        </OpportunityConstraints>
    </PlacementControl>
</PlacementOpportunity>
```

Figure 30: Intermediary-centric VOD POIS/PlacementRequest Link

While the previous section mentioned that caching and registering for notifications may increase performance, this may not be advisable in all situations due to the nature of some VOD ecosystems. In a situation where ad decisions are not all determined at the start of a VOD session and requested in real time as they occur in the stream, the Intermediary could be overwhelmed with notifications if the POIS is manipulating PlacementOpportunity metadata based on time of day.

7.4.4 SISQueryRequest/SISQueryResponse

The Intermediary now needs to obtain additional metadata for the subscriber initiating the VOD session.
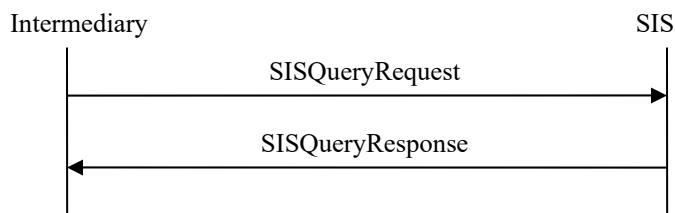
Figure 31: VOD with an Intermediary SIS Query Message Exchange

Using a UniqueQualifier, the Intermediary can query for subscriber information.

```
<SISQueryRequest messageId="uuid-50" system="SISClient" version="1.0"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
      <gis:Query queryId="233" expandOutput="false">
      <gis:ServiceDataModel>http://SuperDemograpics.com</gis:ServiceDataModel
>
            <gis:UniqueQualifier>
                  <gis:Qualifier name="MACAddress" value="03-55-78-90-03-
02"/>
            </gis:UniqueQualifier>
      </gis:Query>
</SISQueryRequest>
```

Example 40

The SIS then responds with a SISQueryResponse that includes metadata it has for the queried subscriber. The Qualifiers, highlighted below, will correspond to TargetCodes in the PlacementRequest sent by the Intermediary.

```
<SISQueryResponse messageId="uuid-51" system="SISServer" version="1.0"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475312" messageRef="uuid-50">
      <core:StatusCode class="0"/>
      <gis:QueryResult queryRef="233" resultSetSize="1"
totalResultSetSize="1">
            <gis:BasicQueryResult>
                  <gis:QualifierSet>
                        <gis:Qualifier name="MACAddress" value="03-55-78-90-
03-02"/>
                        <gis:Qualifier name="Age" value="Under20"/>
                        <gis:Qualifier name="Income" value="Under50K"/>
                        <gis:Qualifier name="ZipCode" value="01720"/>
                        <gis:Qualifier name="CreditLimit" value="100.00"/>
                        <gis:Qualifier name="SportsInterest"
value="Basketball"/>
                  </gis:QualifierSet>
            </gis:BasicQueryResult>
      </gis:QueryResult>
</SISQueryResponse>
```

Example 41

The figure below illustrates the link between the data returned in the SISQueryResponse and the PlacementRequest. The top section is an excerpt from the SISQueryResponse in Example 41, and the bottom section is from the PlacementRequest in Example 42.

```
<gis:Qualifier name="Age" value="Under20">
<gis:Qualifier name="Income" value="Under50K"/>
<gis:Qualifier name="ZipPlusFour" value="01720"/>
<gis:Qualifier name="CreditLimit" value="100.00"/>
<gis:Qualifier name="SportsInterest" value="Basketball"/>


<TargetCode key="Age">Under20<TargetCode/>
<TargetCode key="Income">Under50K<TargetCode/>
<TargetCode key="ZipPlusFour">01720<TargetCode/>
<TargetCode key="CreditLimit">100.00<TargetCode/>
<TargetCode key="SportsInterest">Basketball<TargetCode/>
```

Figure 32: Intermediary-centric VOD SIS/PlacementRequest Link

7.4.5 PlacementRequest/PlacementResponse

Armed with all of the entertainment asset and PlacementOpportunity metadata, the Intermediary can now send a complete PlacementRequest to the ADS.
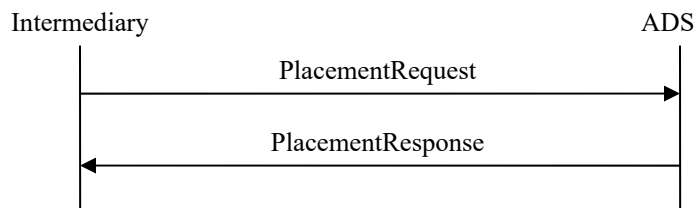
Figure 33: VOD with an Intermediary Placement Message Exchange

The Intermediary's PlacementRequest contains one PlacementOpportunity for a preRoll.

```
<PlacementRequest messageId="uuid-52" version="1.1"
identity="ADMLogicalService_1170FEF4-C19B-450E-B624-421B23F525F4"
system="adm" updatesAllowed="true">
  <Service id="Service1">
    <ProductType>MOD</ProductType>
  </Service>
  <Entertainment>
    <core:Content>
      <core:AssetRef providerID="entertainment.com"
assetID="ABCD0123456789012345"/>
    </core:Content>
  </Entertainment>
  <Client>
    <TargetCode key="Age">Under20</TargetCode>
    <TargetCode key="Income">Under50K</TargetCode>
    <TargetCode key="ZipCode">01720</TargetCode>
    <TargetCode key="CreditLimit">100.00</TargetCode>
    <TargetCode key="SportsInterest">Basketball</TargetCode>
  </Client>
  <PlacementOpportunity id="uuid-53" serviceRegistrationRef="ServiceRef1">
    <OpportunityBinding opportunityType="preRoll"/>
    <OpportunityConstraints>
      <Scope>Local</Scope>
    </OpportunityConstraints>
    <PlacementControl id="uuid-54" action="fill">
      <OpportunityConstraints>
        <core:Duration>PT30S</core:Duration>
      </OpportunityConstraints>
    </PlacementControl>
  </PlacementOpportunity>
</PlacementRequest>
```

Example 42

The ADS responds to the Intermediary with ad placements for the preRoll placement opportunity.

```
<PlacementResponse messageId="uuid-55" version="1.1"
identity="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5"
system="ads" messageRef="uuid-52">
  <core:StatusCode class="0"/>
  <ADMData><!--Hidden secret sauce--></ADMData>
  <PlacementDecision id="uuid-56" placementOpportunityRef="uuid-53">
    <Placement id="uuid-57" action="fill" placementControlRef="uuid-54">
      <core:Content>
        <core:AssetRef providerID="adsrus.com"
assetID="ZXYW0123456789012345"/>
        <core:Tracking>uuid-58</core:Tracking>
      </core:Content>
    </Placement>
  </PlacementDecision>
</PlacementResponse>
```

Example 43

With all of the calls to the SCTE 130 information services (CIS, SIS, POIS), an Intermediary may prefer to register for notifications and cache metadata in an effort to improve performance.

7.4.6   PlacementResponse

The Intermediary can now respond to the ADM.
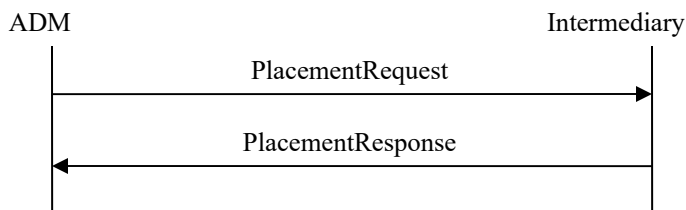
Figure 34: VOD with an Intermediary Placement Response Message Exchange

The Intermediary responds to the ADM with ad placements for the preRoll placement opportunity.

```
<PlacementResponse messageId="uuid-59" version="1.1"
identity="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5"
system="ads" messageRef="uuid-43">
  <core:StatusCode class="0"/>
  <ADMData><!--Hidden secret sauce--></ADMData>
  <PlacementDecision id="uuid-60" placementOpportunityRef="uuid-44">
    <Placement id="uuid-61" action="fill">
      <core:Content>
        <core:AssetRef providerID="adsrus.com"
assetID="ZXYW0123456789012345"/>
        <core:Tracking>uuid-62</core:Tracking>
      </core:Content>
    </Placement>
  </PlacementDecision>
</PlacementResponse>
```

Example 44

7.4.7   PlacementStatusNotification

After the VOD session ends, event data is required to be propagated to the Intermediary and ADS from the ADM via PlacementStatusNotification.
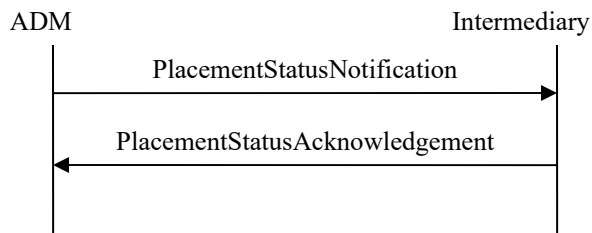
Figure 35: VOD with an Intermediary Placement Status Notification Message Exchange

```
<PlacementStatusNotification messageId="uuid-63" version="1.1"
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>uuid-62</core:Tracking>
          </core:Content>
          <SpotNPT scale="1">0</SpotNPT>
        </Spot>
      </PlacementStatusEvent>
      <PlacementStatusEvent type="endPlacement" time="2007-01-
09T14:20:30.0Z">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="adsrus.com"
assetID="ZXYW0123456789012345"/>
            <core:Tracking>uuid-62</core:Tracking>
          </core:Content>
          <SpotNPT scale="1">30</SpotNPT>
        </Spot>
      </PlacementStatusEvent>
    </Events>
  </PlayData>
</PlacementStatusNotification>
```

Example 45

The Intermediary is responsible for propagating this information to the ADS.

7.4.8   PlacementStatusNotification/PlacementStatusAcknowledgement

Now that the Intermediary has the event data, it is responsible for notifying the appropriate ADS(s). In the event of multiple ownership, the Intermediary may divide the ADM's PlacementStatusNotification and only notify individual ADSs with event data related to the ads they placed.
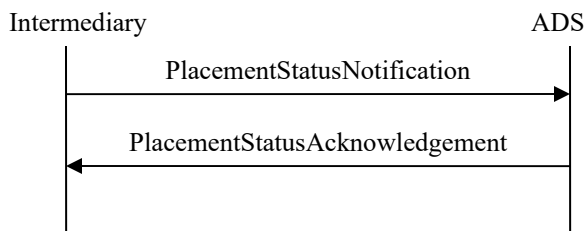
Figure 36: VOD with an Intermediary Placement Status Message Exchange

```
<PlacementStatusNotification messageId="uuid-64" version="1.1"
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>uuid-58</core:Tracking>
          </core:Content>
          <SpotNPT scale="1">0</SpotNPT>
        </Spot>
      </PlacementStatusEvent>
      <PlacementStatusEvent type="endPlacement" time="2007-01-
09T14:20:30.0Z">
        <Spot>
          <core:Content>
            <core:AssetRef providerID="adsrus.com"
assetID="ZXYW0123456789012345"/>
            <core:Tracking>uuid-58core:Tracking>
          </core:Content>
          <SpotNPT scale="1">30</SpotNPT>
        </Spot>
      </PlacementStatusEvent>
    </Events>
  </PlayData>
</PlacementStatusNotification>
```

Example 46

The ADS responds with a PlacementStatusAcknowledgement.

```
<PlacementStatusAcknowledgement messageId="uuid-65" version="1.1"
identity="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5"
system="ads" messageRef="uuid-64" >
  <core:StatusCode class="0"/>
</PlacementStatusAcknowledgement>
```

Example 47

7.4.9   PlacementStatusAcknowledgement

The Intermediary then responds to the ADM with the PlacementStatusAcknowledgement.

Figure 37: VOD with an Intermediary Placement Status Acknowledgement Message Exchange

```
<PlacementStatusAcknowledgement messageId="uuid-66" version="1.1"
identity="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5"
system="ads" messageRef="uuid-63" >
  <core:StatusCode class="0"/>
</PlacementStatusAcknowledgement>
```

Example 48

7.5 ADS Initiated Deregistration

This section details the ADS initiated deregistration messaging between the ADM and ADS.
ADSDeregisterRequest/Response is covered in detail, including the difference in behavior of the
ADM based on the presence of the @registrationRef attribute.
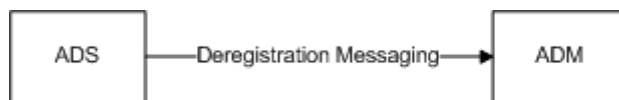
Figure 38: ADS Initiated Deregistration Messaging Overview

Once an ADS initiates deregistration with an ADM, the ADM will stop sending PlacementRequests and associated messaging to the ADS. The sequence diagram below shows ADS initiated deregistration message flow between the ADM and the ADS.

Figure 39: ADS Initiated Deregistration Sequence Diagram

7.5.1   ADSDeregisterRequest/ADSDeregisterResponse

Figure 40: ADS Initiated Deregistration Message Exchange

An ADS can deregister for all active registrations or a specific registration. Below is an ADSDeregisterRequest for all registrations, which is signified by the lack of the @registrationRef attribute. For more detail on the ADSDeregister messages, see Section 11.5 (Deregistration) of [SCTE130-3].

```
<ADSDeregisterRequest messageId="uuid-67" version="1.1"
identity="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5"
system="ads"/>
```

Example 49

The ADM responds with an ADSDeregisterResponse.

```
<ADSDeregisterResponse messageId="uuid-68" version="1.1"
identity="ADMLogicalService_1170FEF4-C19B-450E-B624-421B23F525F4"
system="adm" messageRef="uuid-67">
  <core:StatusCode class="0"/>
</ADSDeregisterResponse>
```

Example 50

Below is an example of an ADS that is deregistering for a specific registration. Note the inclusion of the @registrationRef attribute, which refers to a specific registration that the ADS has initiated in the past.

```
<ADSDeregisterRequest messageId="uuid-69" version="1.1"
identity="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5"
system="ads" registrationRef="uuid-70"/>
```

Example 51

The ADM responds with an ADSDeregisterResponse.

```
<ADSDeregisterResponse messageId="uuid-71" version="1.1"
identity="ADMLogicalService_1170FEF4-C19B-450E-B624-421B23F525F4"
system="adm" messageRef="uuid-69">
  <core:StatusCode class="0"/>
</ADSDeregisterResponse>
```

Example 52

7.6 ADM Initiated Deregistration

This section details the ADM initiated deregistration messaging between the ADM and ADS. ADSDeregistrationNotification/Acknowledgement is covered in detail, including the difference in behavior of the ADS based on the presence of the @registrationRef attribute.
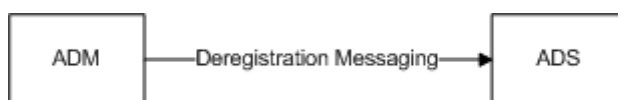
Figure 41: ADM Initiated Deregistration Messaging Overview

The sequence diagram below shows ADM initiated deregistration message flow between the ADM and the ADS.
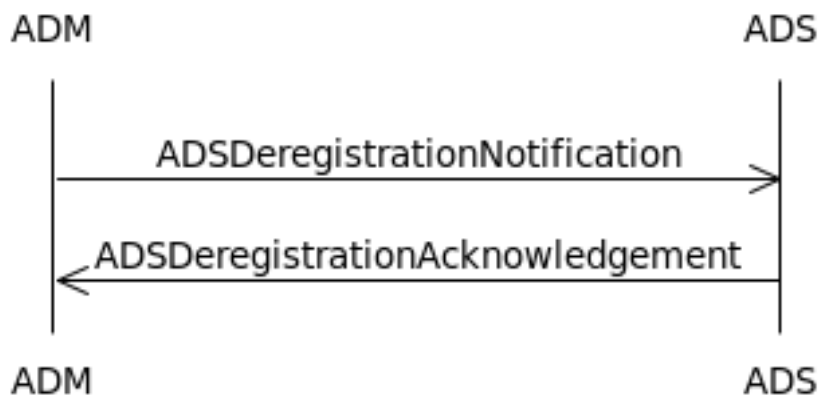
Figure 42: ADM Initiated Deregistration Sequence Diagram

7.6.1 ADSDeregistrationNotification/ADSRegistrationAcknowledgement
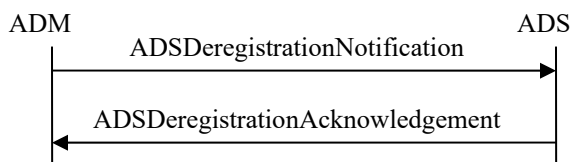
Figure 43: ADM Initiated Deregistration Message Exchange

Like the ADS initiated deregistration, an ADM can deregister for all active registrations or a specific registration. Below is an ADSDeregistrationNotification for all registrations, which is signified by the lack of the @registrationRef attribute. For more detail on the ADSDeregistration messages, see Section 11.5 (Deregistration) of [SCTE130-3].

```
<ADSDeregistrationNotification messageId="uuid-72" version="1.1"
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 53

The ADS responds with an ADSDeregistrationAcknowledgement.

```
<ADSDeregistrationAcknowledgement messageId="uuid-73" version="1.1"
identity="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5"
system="ads" messageRef="uuid-72">
  <core:StatusCode class="0"/>
</ADSDeregistrationAcknowledgement>
```

Example 54

Like the ADS initiated deregistration, the inclusion of the @registrationRef attribute allows an ADM to deregister a specific ADS registration.

```
<ADSDeregistrationNotification messageId="uuid-74" version="1.1"
identity="ADMLogicalService_1170FEF4-C19B-450E-B624-421B23F525F4"
system="adm" registrationRef="uuid-75">
  <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>
```
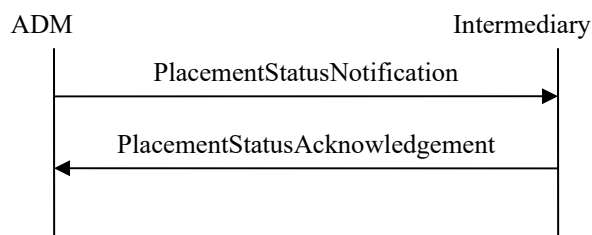
Example 55

The ADS responds with an ADSDeregistrationAcknowledgement.

```
<ADSDeregistrationAcknowledgement messageId="uuid-76" version="1.1"
identity="ADSLogicalService_1170FEF4-C19B-450E-B624-421B23F525F5"
system="ads" messageRef="uuid-74">
  <core:StatusCode class="0"/>
</ADSDeregistrationAcknowledgement>
```

Example 56

8   Information Service Usage Patterns

For each of the three Information Services, four use cases are explored that aid a client in retrieving data: Feature Discovery, Notifications, Queries, and Cursors. There is some overlap in the Query section with some of the VOD Placement Decision Usage Patterns, but these sections also detail querying with a non-unique identifier, such as a regular expression. Each of the four use cases contain a sequence diagram that shows the specific messages traveling between services and the order in which they are sent.

Since an ADS, ADM, Intermediary, or any other SCTE 130 compliant server may make calls to the information services, "Client" will be used as a generic identifier for any client of the CIS, POIS, or SIS.

8.1 SIS

This section describes the four major use cases of the SIS: Feature Discovery, Notifications, Queries, and Cursors.
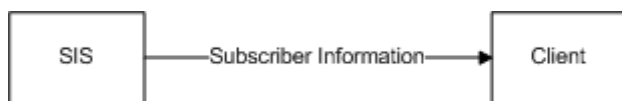
Figure 44: SIS Overview

8.1.1 SIS Feature Discovery

The sequence diagram below illustrates feature discovery message flow between a Client and an SIS.
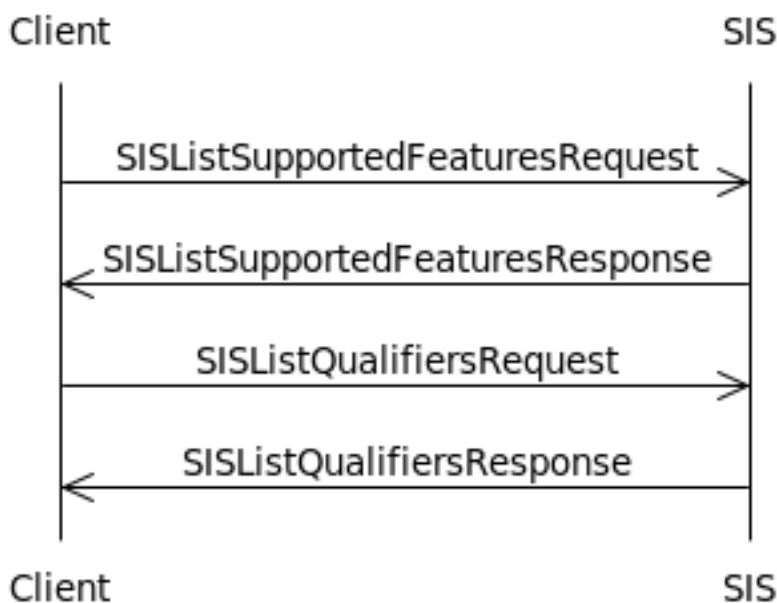
Figure 45: SIS Discovery Sequence Diagram

While there is no required registration process that occurs before queries between a Client and an SIS can begin, there are mechanisms for determining endpoints and qualifiers.

8.1.1.1 *SISListSupportedFeaturesRequest/SISListSupportedFeaturesResponse*
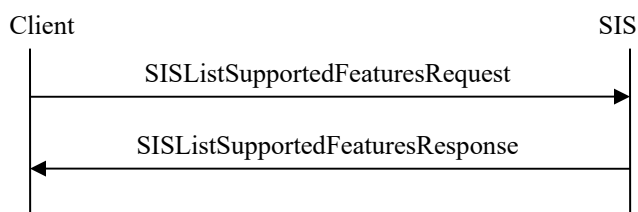
Figure 46: SIS List Supported Features Message Exchange

In order to determine the endpoints, data models, and query languages supported by the SIS, the Client sends an SISListSupportedFeaturesRequest to the SIS. For more detail on the SISListSupportedFeatures messages, see Section 10.7 (SISListSupportedFeaturesRequest and SISListSupportedFeaturesResponse) of [SCTE130-6], and Section 11.1 (ListSupportedFeaturesRequest and ListSupportedFeaturesResponse Message Types) and Appendix E.1 (Listing Supported Features) of [SCTE130-8].

```
<SISListSupportedFeaturesRequest messageId="uuid-77" system="SISClient"
version="1.0" identity="40DA910E-01AF-5050-C7EA-5D7B4A475311"/>
```

Example 57

The SIS responds with an SISListSupportedFeaturesResponse.

```
<SISListSupportedFeaturesResponse messageId="uuid-78" system="SISServer" version="1.0"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475312" messageRef="uuid-77">
      <core:StatusCode class="0"/>
      <core:Callout>
            <core:Address type="SOAP1.1">http://10.250.30.22/SISServer</core:Address>
      </core:Callout>
      <gis:ServiceDataModelProfile>
            <gis:ServiceDataModel>http://SuperDemograpics.com</gis:ServiceDataModel>
            <gis:AdvancedQueryLanguage>XPath</gis:AdvancedQueryLanguage>
            <gis:AdvancedQueryLanguage>XQuery</gis:AdvancedQueryLanguage>
      </gis:ServiceDataModelProfile>
      <gis:ServiceDataModelProfile>
       <gis:ServiceDataModel>http://AdZoneDemograpics.com</gis:ServiceDataModel>
      </gis:ServiceDataModelProfile>
</SISListSupportedFeaturesResponse>
```

Example 58

8.1.1.2 SISListQualifiersRequest/SISListQualifiersResponse

Figure 47: SIS List Qualifiers Message Exchange

In order to determine the qualifiers offered by the SIS, the Client sends an SISListSupportedFeaturesRequest to the SIS. In the example below, the Client is asking specifically for the qualifiers supported by the "http://SuperDemographics.com" data model. For more detail on the SISListQualifiers messages, see Section 10.8 (SISListQualifiersRequest and SISListQualifiersResponse) of [SCTE130-6], and Section 11.2 (ListQualifiersRequest and ListQualifiersResponse Message Types) and Appendix E.2 (Listing Qualifiers) of [SCTE130-8].

```
<SISListQualifiersRequest messageId="uuid-79" system="SISClient" version
="1.0" identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
  <gis:ServiceDataModel>http://SuperDemographics.com</gis:ServiceDataModel>
</SISListQualifiersRequest>
```

Example 59

The SIS responds with an SISListQualifiersResponse listing all available qualifiers for the particular data model.

```
<SISListQualifiersResponse messageId="uuid-80" system="SISServer" version="1.0"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475312" messageRef="uuid-79">
      <core:StatusCode class="0"/>
      <gis:BasicQueryDataModelDescription>
            <gis:ServiceDataModel>http://SuperDemographics.com</gis:ServiceDataModel>
            <gis:UniqueQualifierDeclaration>
                  <gis:QualifierDeclaration name="MACAddress"/>
            </gis:UniqueQualifierDeclaration>
            <gis:QualifierDescription name="Age" valueType="enumeration">
                  <gis:EnumerationValue>UnderTwenty</gis:EnumerationValue>
                  <gis:EnumerationValue>TwentyToForty</gis:EnumerationValue>
                  <gis:EnumerationValue>FortyToSixty</gis:EnumerationValue>
                  <gis:EnumerationValue>OverSixty</gis:EnumerationValue>
            </gis:QualifierDescription>
            <gis:QualifierDescription name="Income" valueType="enumeration">
                  <gis:EnumerationValue>Under50K</gis:EnumerationValue>
                  <gis:EnumerationValue>50Kto100K</gis:EnumerationValue>
                  <gis:EnumerationValue>100Kto200K</gis:EnumerationValue>
                  <gis:EnumerationValue>Over200K</gis:EnumerationValue>
            </gis:QualifierDescription>
            <gis:QualifierDescription name="ZipCode" valueType="string">
                  <gis:MaxLength>10</gis:MaxLength>
            </gis:QualifierDescription>
            <gis:QualifierDescription name="MACAddress" valueType="string">
                  <gis:MaxLength>17</gis:MaxLength>
            </gis:QualifierDescription>
            <gis:QualifierDescription name="CreditLimit" valueType="float"/>
            <gis:QualifierDescription name="SportsInterest" valueType="enumeration">
                  <gis:EnumerationValue>Hockey</gis:EnumerationValue>
                  <gis:EnumerationValue>Football</gis:EnumerationValue>
                  <gis:EnumerationValue>Baseball</gis:EnumerationValue>
                  <gis:EnumerationValue>Basketball</gis:EnumerationValue>
                  <gis:EnumerationValue>Soccer</gis:EnumerationValue>
                  <gis:EnumerationValue>Tennis</gis:EnumerationValue>
                  <gis:EnumerationValue>Fishing</gis:EnumerationValue>
                  <gis:EnumerationValue>Hunting</gis:EnumerationValue>
                  <gis:EnumerationValue>None</gis:EnumerationValue>
            </gis:QualifierDescription>
      </gis:BasicQueryDataModelDescription>
</SISListQualifiersResponse>
```

Example 60

8.1.2   SIS Notifications

The sequence diagram below illustrates notification message flow between a Client and an SIS.
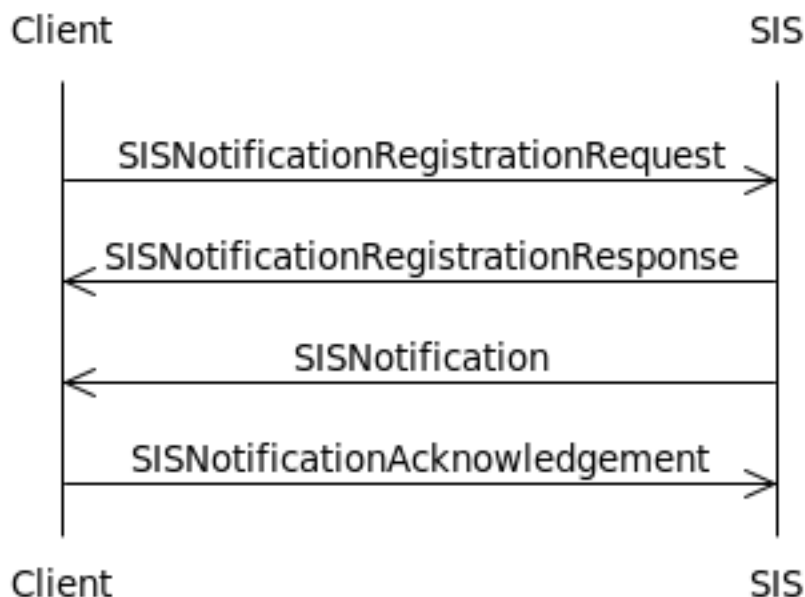
Figure 48: SIS Notification Sequence Diagram

Subscriber data is not static, and subscribers may be added or removed from the system. The notification messaging allows a Client to register once, then automatically receive notifications for any changes related to the previously defined subscribers. This function enables a Client to cache and selectively invalidate subscribers.

8.1.2.1 *SISNotificationRegistrationRequest/ SISNotificationRegistrationResponse*
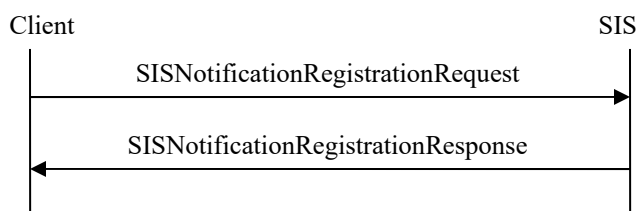
Figure 49: SIS Notification Registration Message Exchange

In the message below, the Client is registering for notifications for all subscribers that are added, updated, or deleted that are under twenty. For more detail on the SISNotificationRegistration messages, see Section 10.10 (SISNotificationRegistrationRequest and SISNotificationRegistrationResponse) and Appendix A.4 (SIS Notification Registration Request and Notification) of [SCTE130-6], and Section 11.4 (NotificationRegistrationRequest and NotificationRegistrationResponse Message Types) and Appendix E.4 (Notification Registration Examples) of [SCTE130-8].

```
<SISNotificationRegistrationRequest messageId="uuid-81"
system="SISClient" version="1.0" identity="40DA910E-01AF-5050-C7EA-
5D7B4A475311">
 <core:Callout>
  <core:Address type="SOAP1.1">http://10.250.30.77/SISClient</core:Address>
 </core:Callout>
 <gis:Query queryId="query-10">
  <gis:ServiceDataModel>http://SuperDemographics.com</gis:ServiceDataModel>
  <gis:BasicQueryFilter>
   <gis:BasicFilterElement name="Age" value="UnderTwenty"
valueIsRegex="false"/>
  </gis:BasicQueryFilter>
 </gis:Query>
</SISNotificationRegistrationRequest>
```
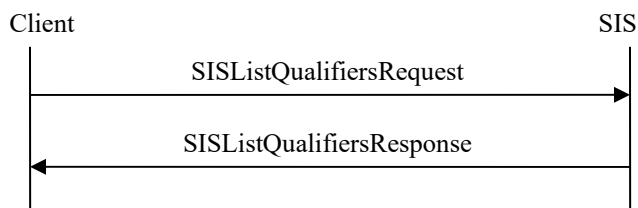
Example 61

The SIS acknowledges the registration with an SISNotificationRegistrationResponse.

```
<SISNotificationRegistrationResponse messageId="uuid-82"
system="SISServer" version="1.0" identity="40DA910E-01AF-5050-C7EA-5D7B4A475312"
messageRef="uuid-81">
  <core:StatusCode class="0"/>
</SISNotificationRegistrationResponse>
```

Example 62

8.1.2.2   SISNotification/SISNotificationAcknowledgement
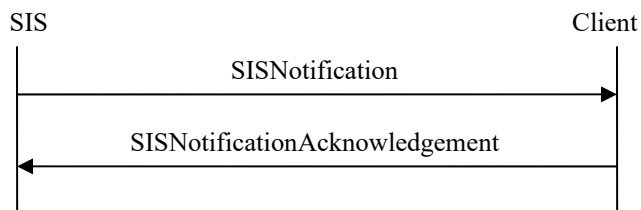
Figure 50: SIS Notification Message Exchange

Now that the Client has registered with the SIS, the SIS is now responsible for notifying the
Client of any changes related to the subscribers that meet the qualifier in the registration. This is
accomplished via SISNotifications. The @noticeType attribute, highlighted below, is used to
denote the type of notification. In the example below, the SIS is notifying the Client of five new
subscribers under twenty that were added to the system. For more detail on the SISNotification
messages, see Section 10.11 (SISNotification and SISNotificationAcknowledgement) and
Appendix A.4 (SIS Notification Registration Request and Notification) of [SCTE130-6], and

Section 11.5 (Notification and NotificationAcknowledgement Message Types) and Appendix E.5 (Notification) of [SCTE130-8].

```
<SISNotification noticeType="new" messageId="uuid-83" system="SISServer"
version="1.0" identity="40DA910E-01AF-5050-C7EA-5D7B4A475312">
 <gis:QueryResult resultSetSize="5" queryRef="query-10"
totalResultSetSize="5">
  <gis:BasicQueryResult>
   <gis:UniqueQualifier>
    <gis:Qualifier name="MACAddress" value="00-1e-c2-01-d3-2d"/>
   </gis:UniqueQualifier>
   <gis:UniqueQualifier>
    <gis:Qualifier name="MACAddress" value="00-1e-52-74-2e-6a"/>
   </gis:UniqueQualifier>
   <gis:UniqueQualifier>
    <gis:Qualifier name="MACAddress" value="00-1e-52-74-2e-6d"/>
   </gis:UniqueQualifier>
   <gis:UniqueQualifier>
    <gis:Qualifier name="MACAddress" value="00-50-56-c0-00-01"/>
   </gis:UniqueQualifier>
   <gis:UniqueQualifier>
    <gis:Qualifier name="MACAddress" value="00-50-56-c0-00-08"/>
   </gis:UniqueQualifier>
  </gis:BasicQueryResult>
 </gis:QueryResult>
</SISNotification>
```

Example 63

The Client acknowledges that it received the notification with an SISAcknowledgement.

```
<SISNotificationAcknowledgement messageId="uuid-84" system="SISClient"
version="1.0" identity="40DA910E-01AF-5050-C7EA-5D7B4A475311" messageRef="uuid-
83">
  <core:StatusCode class="0"/>
</SISNotificationAcknowledgement>
```

Example 64

8.1.3   SIS Queries

The sequence diagram below illustrates query message flow between a Client and an SIS.
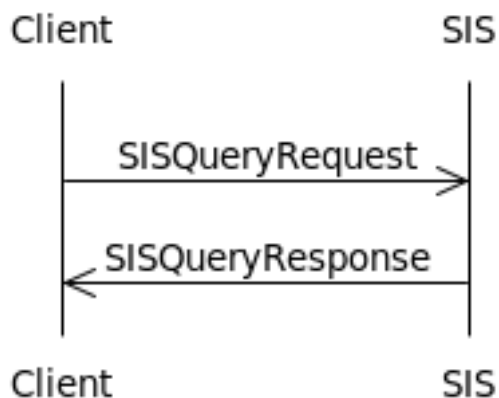
Figure 51: SIS Query Sequence Diagram

The Client is responsible for initiating queries to the SIS, and the SIS will respond with the relevant subscriber information.

8.1.3.1   *SISQueryRequest/SISQueryResponse*

Figure 52: SIS Query Message Exchange

8.1.3.1.1   Unique Identifier

Throughout this document, examples with information services have been shown with a unique identifier, with the intention of the calling SCTE130 component receiving a result set size of 1 back. With the SIS, a MAC address is unique to a particular subscriber's hardware. This is ideal in a situation where the Client is not caching data to potentially improve performance and is only interested in the subscriber's data that is relevant to the PlacementRequest it is attempting to service. The example from Section 7.2.4, which uses a UniqueQualifier with the subscriber's MAC address, is below.

```
<SISQueryRequest messageId="uuid-85" system="SISClient" version="1.0"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
      <gis:Query queryId="233" expandOutput="false">
      <gis:ServiceDataModel>http://SuperDemograpics.com</gis:ServiceDataModel
>
            <gis:UniqueQualifier>
                  <gis:Qualifier name="MACAddress" value="03-55-78-90-03-
02"/>
            </gis:UniqueQualifier>
      </gis:Query>
</SISQueryRequest>
```

Example 65

The SIS then responds with a SISQueryResponse that includes metadata it has for the queried subscriber.

```
<SISQueryResponse messageId="uuid-86" system="SISServer" version="1.0"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475312" messageRef="uuid-85">
      <core:StatusCode class="0"/>
      <gis:QueryResult queryRef="233" resultSetSize="1"
totalResultSetSize="1">
            <gis:BasicQueryResult>
                  <gis:QualifierSet>
                        <gis:Qualifier name="MACAddress" value="03-55-78-90-
03-02"/>
                        <gis:Qualifier name="Age" value="UnderTwenty"/>
                        <gis:Qualifier name="Income" value="Under50K"/>
                        <gis:Qualifier name="ZipCode" value="01720"/>
                        <gis:Qualifier name="CreditLimit" value="100.00"/>
                        <gis:Qualifier name="SportsInterest"
value="Basketball"/>
                  </gis:QualifierSet>
            </gis:BasicQueryResult>
      </gis:QueryResult>
</SISQueryResponse>
```

Example 66

8.1.3.1.2 Non-Unique Identifier

Calling out to an information service at the beginning of every viewing session in a VOD system make could be considered inefficient, particularly in systems in which there are tight time constraints. An alternate method of gathering subscriber information is to use a non-unique identifier to get information on all subscribers that fall under a particular qualifier.

For this example, assume that the Client is only interested in doing age based targeting on subscribers under the age of twenty. The qualifier that was discovered in the SISListQualifiersResponse in Example 60 can now be used to obtain data for all subscribers that are identified with that qualifier.

```
<SISQueryRequest messageId="uuid-87" system="SISClient" version="1.0"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
      <gis:Query queryId="234" expandOutput="true">
      <gis:ServiceDataModel>http://SuperDemograpics.com</gis:ServiceDataModel
>
            <gis:BasicQueryFilter>
                  <gis:BasicFilterElement name="Age" value="UnderTwenty"
valueIsRegex="false"/>
            </gis:BasicQueryFilter>
      </gis:Query>
</SISQueryRequest>
```

Example 67

The SIS responds with all subscribers under the age of twenty.

```
<SISQueryResponse messageId="uuid-88" system="SISServer" version="1.0"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475312" messageRef="uuid-87">
      <core:StatusCode class="0"/>
      <gis:QueryResult queryRef="234" resultSetSize="5" totalResultSetSize="5">
            <gis:BasicQueryResult>
                  <gis:QualifierSet>
                        <gis:Qualifier name="MACAddress" value="00-1e-c2-01-d3-
2d"/>
                        <gis:Qualifier name="Age" value="UnderTwenty"/>
                        <gis:Qualifier name="Income" value="Under50K"/>
                        <gis:Qualifier name="ZipCode" value="01720"/>
                        <gis:Qualifier name="CreditLimit" value="100.00"/>
                        <gis:Qualifier name="SportsInterest" value="Basketball"/>
                  </gis:QualifierSet>
                  <gis:QualifierSet>
                        <gis:Qualifier name="MACAddress" value="00-1e-52-74-2e-
6a"/>
                        <gis:Qualifier name="Age" value="UnderTwenty"/>
                        <gis:Qualifier name="Income" value="Under50K"/>
                        <gis:Qualifier name="ZipCode" value="01847"/>
                        <gis:Qualifier name="CreditLimit" value="200.00"/>
                        <gis:Qualifier name="SportsInterest" value="Football"/>
                  </gis:QualifierSet>
                  <gis:QualifierSet>
                        <gis:Qualifier name="MACAddress" value="00-1e-52-74-2e-
6d"/>
                        <gis:Qualifier name="Age" value="UnderTwenty"/>
                        <gis:Qualifier name="Income" value="Under50K"/>
                        <gis:Qualifier name="ZipCode" value="01847"/>
                        <gis:Qualifier name="CreditLimit" value="100.00"/>
                        <gis:Qualifier name="SportsInterest" value="Tennis"/>
                  </gis:QualifierSet>
                  <gis:QualifierSet>
                        <gis:Qualifier name="MACAddress" value="00-50-56-c0-00-
01"/>
                        <gis:Qualifier name="Age" value="UnderTwenty"/>
                        <gis:Qualifier name="Income" value="Under50K"/>
                        <gis:Qualifier name="ZipCode" value="01720"/>
                        <gis:Qualifier name="CreditLimit" value="500.00"/>
                        <gis:Qualifier name="SportsInterest" value="Football"/>
                  </gis:QualifierSet>
                  <gis:QualifierSet>
                        <gis:Qualifier name="MACAddress" value="00-50-56-c0-00-
08"/>
                        <gis:Qualifier name="Age" value="UnderTwenty"/>
                        <gis:Qualifier name="Income" value="Under50K"/>
                        <gis:Qualifier name="ZipCode" value="01754"/>
                        <gis:Qualifier name="CreditLimit" value="50.00"/>
                        <gis:Qualifier name="SportsInterest" value="Fishing"/>
                  </gis:QualifierSet>
            </gis:BasicQueryResult>
      </gis:QueryResult>
</SISQueryResponse>
```
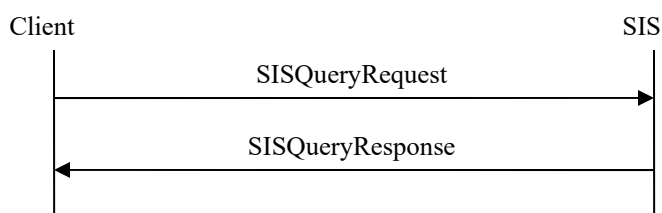
Example 68

A regex can also be used to query the SIS. The example below would return all subscribers that have a MAC address assigned to them.

```
<SISQueryRequest messageId="uuid-89" system="SISClient" version="1.0"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
      <gis:Query queryId="234" expandOutput="true">
      <gis:ServiceDataModel>http://SuperDemograpics.com</gis:ServiceDataModel
>
            <gis:BasicQueryFilter>
                  <gis:BasicFilterElement name="MACAddress" value=".*"
valueIsRegex="true"/>
            </gis:BasicQueryFilter>
      </gis:Query>
</SISQueryRequest>
```

Example 69

8.1.4   SIS Cursors

The sequence diagram below illustrates cursor message flow between a Client and an SIS.
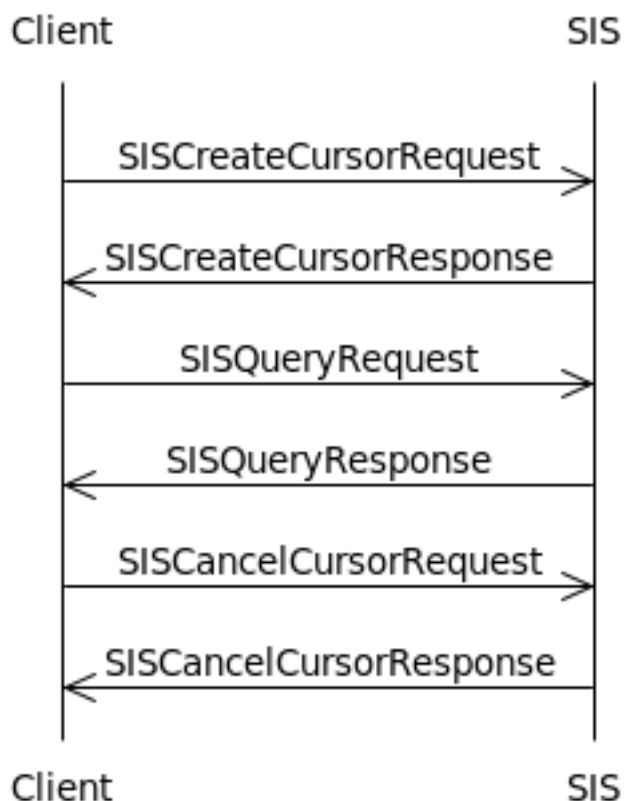
Figure 53: SIS Cursor Sequence Diagram

When a Client queries an SIS with a non-unique identifier, such as the examples in Section 8.1.3.1.2, the amount of data in the response could be quite large. To obtain the data in chunks, the Client can set up a cursor and walk the data set, requesting a subset of the data over multiple SISQueryRequests until it has all of the data.

8.1.4.1 *SISCreateCursorRequest/SISCreateCursorResponse*
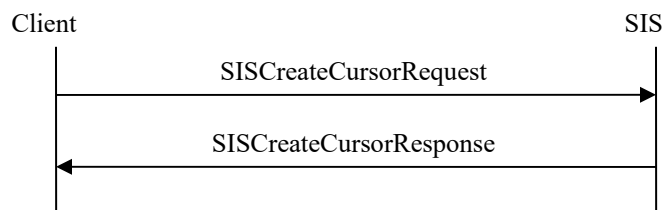
Figure 54: SIS Create Cursor Message Exchange

In the example below, the Client is creating a cursor for all subscribers under twenty. For more detail on the SISCreateCursor messages, see Section 10.12 (SISCreateCursorRequest and SISCreateCursorRequest Response) of [SCTE130-6], and Section 11.6 (CreateCursorRequest and CreateCursorResponse Message Types) and Appendix D.1 (Creating Cursors) of [SCTE130-8].

```
<SISCreateCursorRequest messageId="uuid-90" version="1.0" system="SISClient"
cursorId="cursor-1" cursorExpires="2011-08-10T12:00:00Z" identity="40DA910E-
01AF-5050-C7EA-5D7B4A475311">
  <gis:Query queryId="query-1" expandOutput="false">
   <gis:ServiceDataModel>http://SuperDemographics.com</gis:ServiceDataModel>
   <gis:BasicQueryFilter>
    <gis:BasicFilterElement name="Age" value="UnderTwenty"
valueIsRegex="false"/>
   </gis:BasicQueryFilter>
  </gis:Query>
</SISCreateCursorRequest>
```

Example 70

The SIS acknowledges the successful creation of the cursor with an SISCreateCursorResponse, which includes the number of total records that the query returned.

```
<SISCreateCursorResponse messageId="uuid-91" version="1.0" system="SISServer"
messageRef="uuid-90" totalResultSetSize="100" cursorExpires="2009-08-
10T12:00:00.0Z" identity="40DA910E-01AF-5050-C7EA-5D7B4A475312">
  <core:StatusCode class="0"/>
</SISCreateCursorResponse>
```

Example 71

8.1.4.2 SISQueryRequest/SISQueryResponse
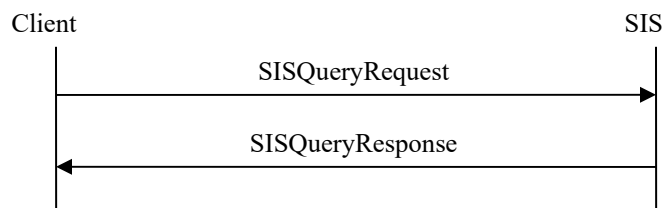
Figure 55: SIS Query Message Exchange

To walk the data in the cursor, the Client sends an SISQueryRequest to the SIS, causing the SIS to return an SISQueryResponse with the specified number of subscribers. In the example below, the Client is requesting ten subscribers, starting at index 0. A subsequent call would start at index 10 and request ten subscribers, and this would be repeated until the Client has retrieved all of the data from the SIS for the cursor.

```
<SISQueryRequest messageId="uuid-92" version="1.0" system="SISClient"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
  <gis:Cursor startIndex="0" count="10" cursorRef="cursor-1" queryRef="query-
1"/>
</SISQueryRequest>
```

Example 72

8.1.4.3 SISCancelCursorRequest/SISCancelCursorResponse
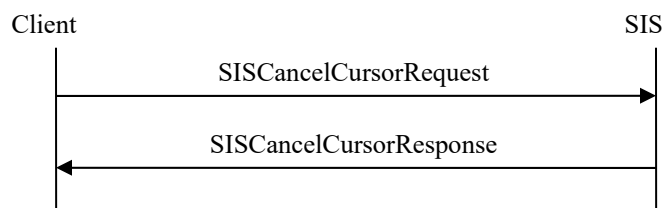
Figure 56: SIS Cancel Cursor Message Exchange

Once the Client has obtained all of the data it needs from a cursor, it should cancel it with the Client via the SISCancelCursorRequest. For more detail on the SISCancelCursor messages, see Section 10.13 (SISCancelCursorRequest and SISCancelCursorResponse) of [SCTE130-6], and Section 11.7 (CancelCursorRequest and CancelCursorResponse Message Types) and Appendix D.3 (Canceling Existing Cursors) of [SCTE130-8].

```
<SISCancelCursorRequest messageId="uuid-93" version="1.0" system="SISClient"
cursorRef="cursor-1" identity="40DA910E-01AF-5050-C7EA-5D7B4A475311"/>
```

Example 73

The SIS responds with an SISCancelCursorResponse.

```
<SISCancelCursorResponse messageId="uuid-94" version="1.0" system="SISServer"
messageRef="uuid-93" identity="40DA910E-01AF-5050-C7EA-5D7B4A475312">
  <core:StatusCode class="0"/>
</SISCancelCursorResponse>
```

Example 74

8.2 POIS

This section describes the four major use cases of the POIS: Feature Discovery, Notifications, Queries, and Cursors.

Figure 57: POIS Overview

8.2.1 POIS Feature Discovery

The sequence diagram below illustrates feature discovery message flow between a Client and a POIS.
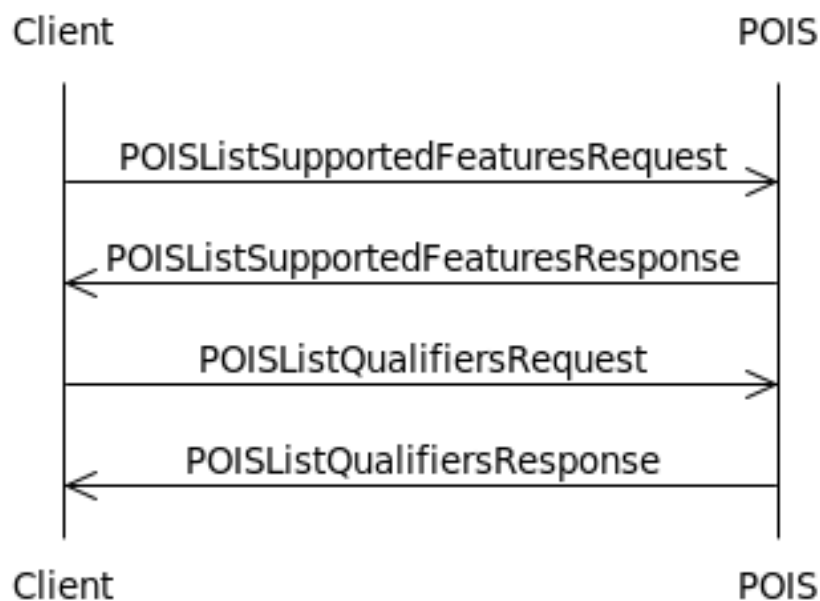
Figure 58: POIS Discovery Sequence Diagram

While there is no required registration process that occurs before queries between a Client and a POIS can begin, there are mechanisms for determining endpoints and qualifiers.

8.2.1.1   *POISListSupportedFeaturesRequest/ POISListSupportedFeaturesResponse*
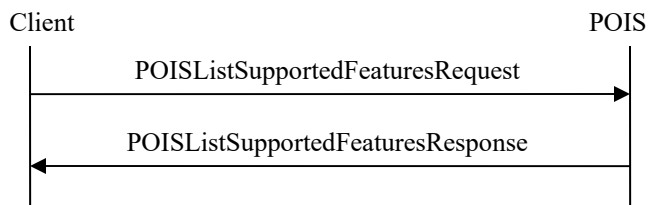
Figure 59: POIS List Supported Features Message Exchange

In order to determine the endpoints, data models, and query languages supported by the POIS, the Client sends a POISListSupportedFeaturesRequest to the POIS. For more detail on the POISListSupportedFeatures messages, see Section 11.7 (POISListSupportedFeaturesRequest and Response Messages) and Appendix A.1 (POISListSupportedFeaturesRequest and Response Message Examples) of [SCTE130-5], and Section 11.1 (ListSupportedFeaturesRequest and ListSupportedFeaturesResponse Message Types) and Appendix E.1 (Listing Supported Features) of [SCTE130-8].

```
<POISListSupportedFeaturesRequest messageId="uuid-95" system="POISClient"
version="1.0" identity="40DA910E-01AF-5050-C7EA-5D7B4A475311"/>
```

Example 75

The POIS responds with a POISListSupportedFeaturesResponse.

```
<POISListSupportedFeaturesResponse messageId="uuid-96" system="POISServer"
version="1.0" identity="40DA910E-01AF-5050-C7EA-5D7B4A475312"
messageRef="uuid-95">
      <core:StatusCode class="0"/>
      <core:Callout>
            <core:Address
type="SOAP1.1">http://10.250.30.22/POISServer</core:Address>
      </core:Callout>
      <gis:ServiceDataModelProfile>
            <gis:ServiceDataModel>http://www.scte.org/schemas/130-
3/2008a/adm/podm</gis:ServiceDataModel>
      </gis:ServiceDataModelProfile>
</POISListSupportedFeaturesResponse>
```

Example 76

8.2.1.2 POISListQualifiersRequest/POISListQualifiersResponse

Figure 60: POIS List Qualifiers Message Exchange

In order to determine the qualifiers offered by the POIS, the Client sends a POISListSupportedFeaturesRequest to the POIS. In the example below, the Client is asking specifically for the qualifiers supported by the "http://www.scte.org/schemas/130-3/2008a/adm/podm" data model. For more detail on the POISListQualifiers messages, see Section 11.8 (POISListQualifiersRequest and Response Messages) and Appendix A.2 (POISListUniqueQualifiersRequest and Response Message Examples) of [SCTE130-5], and Section 11.2 (ListQualifiersRequest and ListQualifiersResponse Message Types) and Appendix E.2 (Listing Qualifiers) of [SCTE130-8].

```
<POISListQualifiersRequest messageId="uuid-97" system="POISClient" version
="1.0" identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
  <gis:ServiceDataModel>http://www.scte.org/schemas/130-
3/2008a/adm/podm</gis:ServiceDataModel>
</POISListQualifiersRequest>
```

Example 77

The POIS responds with a POISListQualifiersResponse listing all available qualifiers for the particular data model.

```
<POISListQualifiersResponse messageId="uuid-98" system="POISServer"
version="1.0" identity="40DA910E-01AF-5050-C7EA-5D7B4A475312"
messageRef="uuid-97">
      <core:StatusCode class="0"/>
      <gis:BasicQueryDataModelDescription>
           <gis:ServiceDataModel>http://www.scte.org/schemas/130-
3/2008a/adm/podm</gis:ServiceDataModel>
           <gis:UniqueQualifierDeclaration uniqueQualifierName="P3-PODM">
                <gis:QualifierDeclaration name="poid"/>
           </gis:UniqueQualifierDeclaration>
           <gis:QualifierDescription name="poid" valueType="string"/>
      </gis:BasicQueryDataModelDescription>
</POISListQualifiersResponse>
```

Example 78

8.2.2   POIS Notifications

The sequence diagram below illustrates notification message flow between a Client and a POIS.
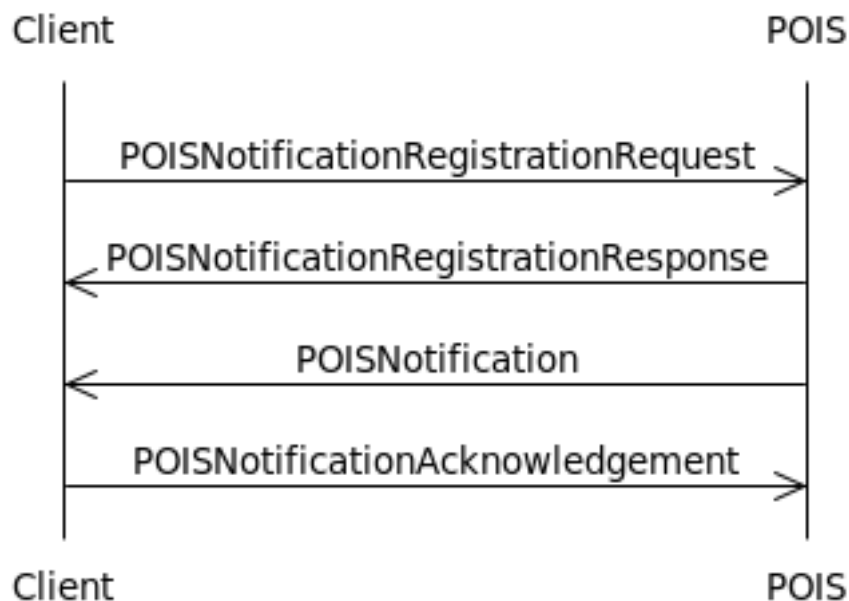
Figure 61: POIS Notification Sequence Diagram

Placement opportunity data is not necessarily static, and placement opportunities may be added or removed from the system. The notification messaging allows a Client to register once, then automatically receive notifications for any changes related to the previously defined placement opportunities. This functionality enables a Client to cache and selectively invalidate placement opportunities.

8.2.2.1   *POISNotificationRegistrationRequest/ POISNotificationRegistrationResponse*
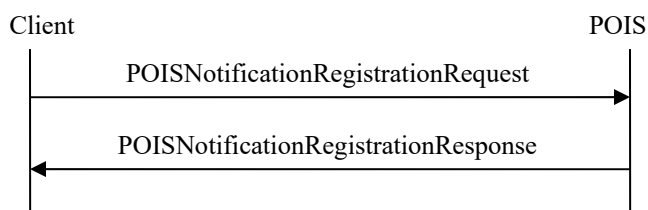
Figure 62: POIS Notification Registration Message Exchange

In the message below, the Client is registering for notifications for all preRoll placement opportunities that are added, updated, or deleted. For more detail on the POISNotificationRegistration messages, see Section 11.10 (POISNotificationRegistrationRequest and Response Messages) of [SCTE130-5], and Section

11.4 (NotificationRegistrationRequest and NotificationRegistrationResponse Message Types)
and Appendix E.4 (Notification Registration Examples) of [SCTE130-8].

```
<POISNotificationRegistrationRequest messageId="uuid-99"
system="POISClient" version="1.0"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
 <core:Callout>
  <core:Address type="SOAP1.1">http://10.250.30.77/POISClient</core:Address>
 </core:Callout>
 <gis:Query queryId="query-10">
      <gis:ServiceDataModel>"http://www.scte.org/schemas/130-
3/2008a/adm/podm"</gis:ServiceDataModel>
      <gis:BasicQueryFilter>
           <gis:BasicFilterElement value="preRoll" name="position"/>
      </gis:BasicQueryFilter>
 </gis:Query>
</POISNotificationRegistrationRequest>
```
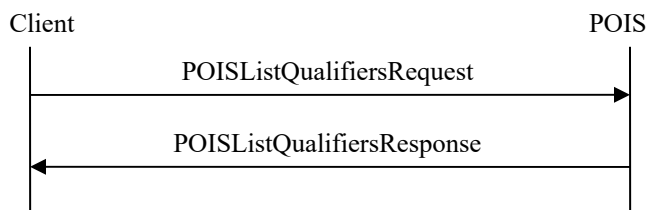
Example 79

The POIS acknowledges the registration with a POISNotificationRegistrationResponse.

```
<POISNotificationRegistrationResponse messageId="uuid-100"
system="POISServer" version="1.0"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475312" messageRef="uuid-99">
  <core:StatusCode class="0"/>
</POISNotificationRegistrationResponse>
```

Example 80

8.2.2.2 POISNotification/POISNotificationAcknowledgement
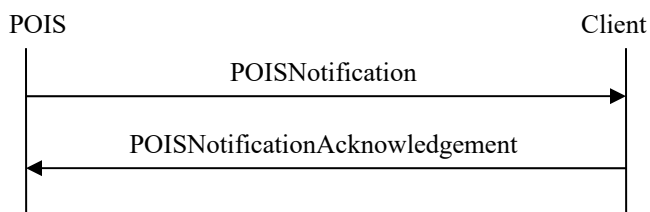
Figure 63: POIS Notification Message Exchange

Now that the Client has registered with the POIS, the POIS is responsible for notifying the Client
of any changes related to the PlacementOpportunities meeting the qualifier in the registration.
This is accomplished via POISNotifications. The @noticeType attribute, highlighted below, is
used to denote the type of notification. In the example below, the SIS is notifying the Client of a
new PlacementOpportunity that was added to the system. For more detail on the
POISNotification messages, see Section 11.11 (POISNotification and Acknowledgement

Messages) of [SCTE130-5], and Section 11.5 (Notification and NotificationAcknowledgement Message Types) and Appendix E.5 (Notification) of [SCTE130-8].

```
<POISNotification noticeType="new" messageId="uuid-101" system="POISServer"
version="1.0" identity="40DA910E-01AF-5050-C7EA-5D7B4A475312">
 <gis:QueryResult resultSetSize="1" queryRef="query-10" totalResultSetSize="2">
        <podm:PODMBasicQueryResult>
        <podm:PlacementOpportunityV2 serviceRegistrationRef="-" id="uuid-102">
                    <adm:Entertainment>
                            <core:Content>
                                    <core:AssetRef providerID="entertainment.com"
assetID="ABCD0123456789012345"/>
                            </core:Content>
                    </adm:Entertainment>
                    <adm:OpportunityBinding opportunityType="preRoll"
opportunitiesExpected="2" opportunityNumber="1"/>
                    <podm:OpportunityConstraintsV2>
                            <core:AdType>30-Second-Spot</core:AdType>
                            <core:Duration>PT30S</core:Duration>
                            <adm:Scope>Local</adm:Scope>
                    </podm:OpportunityConstraintsV2>
            </podm:PlacementOpportunityV2>
        </podm:PODMBasicQueryResult>
  </gis:QueryResult>
</POISNotification>
```

Example 81

The Client acknowledges that it received the notification with a POISAcknowledgement.

```
<POISNotificationAcknowledgement messageId="uuid-103"
system="POISClient" version="1.0" identity="40DA910E-01AF-5050-C7EA-5D7B4A475311"
messageRef="uuid-101">
  <core:StatusCode class="0"/>
</POISNotificationAcknowledgement>
```

Example 82

8.2.3 POIS Queries

The sequence diagram below illustrates query message flow between a Client and a POIS.
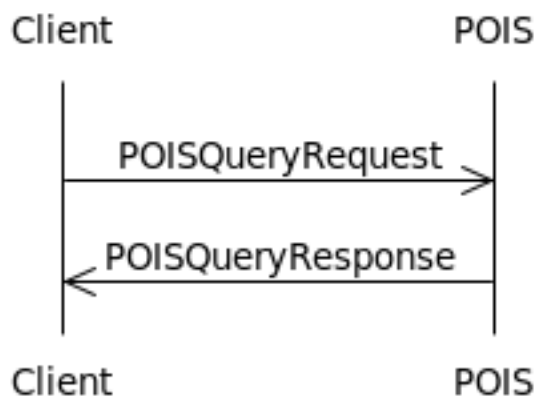
Figure 64: POIS Query Sequence Diagram

The Client is responsible for initiating queries to the POIS, and the POIS responds with the relevant placement opportunity information.

8.2.3.1   POISQueryRequest/POISQueryResponse

Figure 65: POIS Query Message Exchange

8.2.3.1.1   Unique Identifier

Throughout this document, examples with information services have been shown with a unique identifier, with the intention of the calling SCTE130 component receiving a result set size of 1 back. With the POIS, a combination of providerId and assetId is unique to a particular piece of content. This is ideal in a situation where the Client is not caching data to potentially improve performance and is only interested in the placement opportunities that are relevant to the PlacementRequest it is attempting to service. The example from Section 7.2.3, which uses filters with the assetId and providerId, is below.

```
<POISQueryRequest messageId="uuid-104" system="POISClient" identity="40DA910E-01AF-5050-C7EA-
5D7B4A475757" version="1.0">
        <gis:Query queryId="pois-98765" expandOutput="true" uniqueQualifierNameRef="P3-PODM">
                <gis:ServiceDataModel>"http://www.scte.org/schemas/130-
3/2008a/adm/podm"</gis:ServiceDataModel>
                <gis:BasicQueryFilter>
                        <gis:BasicFilterElement value="ABCD0123456789012345" name="assetId"/>
                </gis:BasicQueryFilter>
                <gis:BasicQueryFilter>
                        <gis:BasicFilterElement value="entertainment.com" name="providerId"/>
                </gis:BasicQueryFilter>
        </gis:Query>
</POISQueryRequest>
```

Example 83

The POIS then responds with a POISQueryResponse that includes metadata it has for the queried asset.

```
<POISQueryResponse version="1.1" messageId="uuid-105" messageRef="uuid-104"
system="POISServer" identity="50DA910E-01AF-5050-C7EA-5D7B4A475759">
     <core:StatusCode class="0"/>
     <gis:QueryResult resultSetSize="1" queryRef="pois-98765">
          <podm:PODMBasicQueryResult>
               <podm:PlacementOpportunityV2 serviceRegistrationRef="-"
id="uuid-106">
                    <adm:Entertainment>
                         <core:Content>
                              <core:AssetRef
providerID="entertainment.com" assetID="ABCD01234567890012345"/>
                         </core:Content>
                    </adm:Entertainment>
                    <adm:OpportunityBinding opportunityType="preRoll"
opportunitiesExpected="1" opportunityNumber="1"/>
                    <podm:OpportunityConstraintsV2>
                         <core:Duration>PT30S</core:Duration>
                         <adm:Scope>Local</adm:Scope>
                    </podm:OpportunityConstraintsV2>
               </podm:PlacementOpportunityV2>
          </podm:PODMBasicQueryResult>
     </gis:QueryResult>
</POISQueryResponse>
```

Example 84

8.2.3.1.2 Non-Unique Identifier

Calling out to an information service at the beginning of every VOD session in a VOD system make could be considered inefficient, particularly in systems in which there are tight time constraints. An alternate method of gathering placement opportunity information is to use a non-unique identifier to get information on all placement opportunities that fall under a particular filter.

For this example, assume that the Client is only interested in placing ads on preRoll and postRoll opportunities.

```
<POISQueryRequest messageId="uuid-107" system="POISClient"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475757" version="1.0">
            <gis:Query queryId="pois-98765" expandOutput="true"
uniqueQualifierNameRef="P3-PODM">
            <gis:ServiceDataModel>"http://www.scte.org/schemas/130-
3/2008a/adm/podm"</gis:ServiceDataModel>
            <gis:BasicQueryFilter>
                <gis:BasicFilterElement value="preRoll" name="position"/>
            </gis:BasicQueryFilter>
            <gis:BasicQueryFilter>
                <gis:BasicFilterElement value="postRoll" name="position"/>
            </gis:BasicQueryFilter>
      </gis:Query>
</POISQueryRequest>
```

Example 85

The POIS responds with all preRoll and postRoll placement opportunities.


```
<POISQueryResponse version="1." messageId="uuid-108" messageRef="uuid-107" system="POISServer"
identity="50DA910E-01AF-5050-C7EA-5D7B4A475759">
      <core:StatusCode class="0"/>
      <gis:QueryResult resultSetSize="2568" queryRef="pois-98765">
            <podm:PODMBasicQueryResult>
                  <podm:PlacementOpportunityV2 serviceRegistrationRef="-" id="uuid-109">
                        <adm:Entertainment>
                              <core:Content>
                                    <core:AssetRef providerID="tbs.com"
assetID="TVN000000000000000009"/>
                              </core:Content>
                        </adm:Entertainment>
                        <adm:OpportunityBinding opportunityType="preRoll"
opportunitiesExpected="2" opportunityNumber="1"/>
                        <podm:OpportunityConstraintsV2>
                              <core:AdType>30-Second-Spot</core:AdType>
                              <core:Duration>PT30S</core:Duration>
                        </podm:OpportunityConstraintsV2>
                  </podm:PlacementOpportunityV2>
                  <!-- -->
                  <!-- 2566 more results not shown. -->
                  <!-- -->
                  <podm:PlacementOpportunityV2  serviceRegistrationRef="-" id="uuid-110">
                        <adm:Entertainment>
                              <core:Content>
                                    <core:AssetRef providerID="tbs.com"
assetID="TVN000000000000000009"/>
                              </core:Content>
                        </adm:Entertainment>
                        <adm:OpportunityBinding opportunityType="postRoll"
opportunitiesExpected="2" opportunityNumber="2"/>
                        <podm:OpportunityConstraintsV2>
                              <core:AdType>60-Second-Spot</core:AdType>
                              <core:Duration>PT60S</core:Duration>
                        </podm:OpportunityConstraintsV2>
                  </podm:PlacementOpportunityV2>
            </podm:PODMBasicQueryResult>
      </gis:QueryResult>
</POISQueryResponse>
```
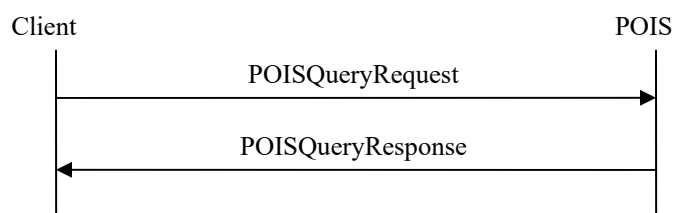
Example 86

A regex can also be used to query the POIS. The example below would return all placement opportunities that have a position assigned to them.

```
<POISQueryRequest messageId="uuid-111" system="POISClient"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475757" version="1.0">
        <gis:Query queryId="pois-98765" expandOutput="true"
uniqueQualifierNameRef="P3-PODM">
        <gis:ServiceDataModel>"http://www.scte.org/schemas/130-
3/2008a/adm/podm"</gis:ServiceDataModel>
        <gis:BasicQueryFilter>
            <gis:BasicFilterElement value=".*" name="position"
valueIsRegex="true"/>
        </gis:BasicQueryFilter>
    </gis:Query>
</POISQueryRequest>
```

Example 87

8.2.4   POIS Cursors

The sequence diagram below illustrates cursor message flow between a Client and a POIS.

Figure 66: POIS Cursor Sequence Diagram

When a Client queries a POIS with a non-unique identifier, such as the examples in Section 8.2.3.1.2, the amount of data in the response could be quite large. To obtain the data in chunks, the Client can set up a cursor and walk the data set, requesting a subset of the data over multiple POISQueryRequests until it has all of the data.

8.2.4.1 POISCreateCursorRequest/POISCreateCursorResponse
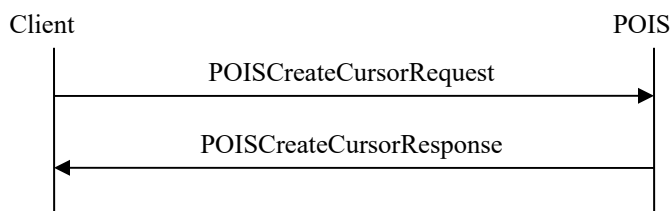
Figure 67: POIS Create Cursor Message Exchange

In the example below, the Client is creating a cursor for all preRoll placement opportunities. For more detail on the POISCreateCursor messages, see Section 11.12 (POISCreateCursorRequest and Response Messages) of [SCTE130-5], and Section 11.6 (CreateCursorRequest and CreateCursorResponse Message Types) and Appendix D.1 (Creating Cursors) of [SCTE130-8].

```
<POISCreateCursorRequest messageId="uuid-112" version="1.0"
system="POISClient" cursorId="cursor-1" cursorExpires="2011-08-10T12:00:00Z"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
  <gis:Query queryId="query-1" expandOutput="true">
     <gis:ServiceDataModel>"http://www.scte.org/schemas/130-
3/2008a/adm/podm"</gis:ServiceDataModel>
     <gis:BasicQueryFilter>
           <gis:BasicFilterElement value="preRoll" name="position"/>
     </gis:BasicQueryFilter>
  </gis:Query>
</POISCreateCursorRequest>
```
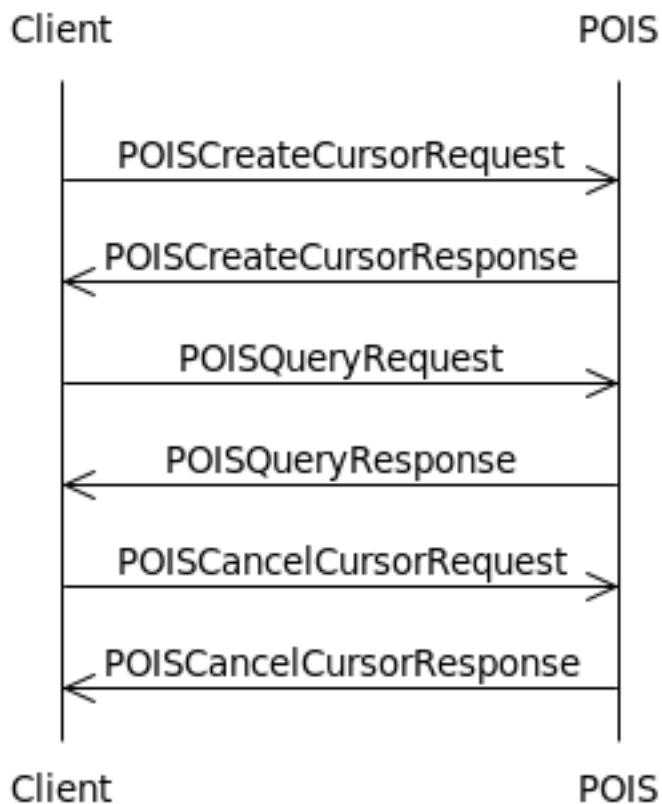
Example 88

The POIS acknowledges the successful creation of the cursor with a POISCreateCursorResponse, which includes the number of total records that the query returned.

```
<POISCreateCursorResponse messageId="uuid-113" version="1.0"
system="POISServer" messageRef="uuid-112" totalResultSetSize="100"
cursorExpires="2009-08-10T12:00:00.0Z" identity="40DA910E-01AF-5050-C7EA-
5D7B4A475312">
  <core:StatusCode class="0"/>
</POISCreateCursorResponse>
```

Example 89

8.2.4.2 POISQueryRequest/POISQueryResponse
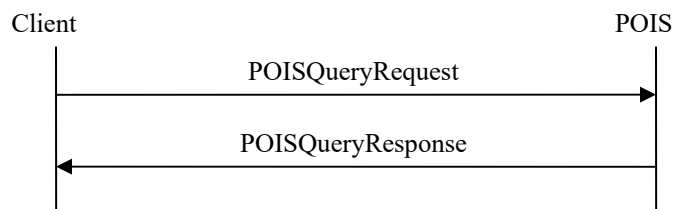
Figure 68: POIS Query Message Exchange

To walk the data in the cursor, the Client sends a POISQueryRequest to the POIS, which causes the POIS to return a POISQueryResponse with the specified number of placement opportunities. In the example below, the Client is requesting ten placement opportunities, starting at index 0. A subsequent call would start at index 10 and request ten placement opportunities, and this would be repeated until the Client has retrieved all of the data from the POIS for the cursor.

```
<POISQueryRequest messageId="uuid-114" version="1.0" system="POISClient"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
  <gis:Cursor startIndex="0" count="10" cursorRef="cursor-1" queryRef="query-
1"/>
</POISQueryRequest>
```

Example 90

8.2.4.3 POISCancelCursorRequest/POISCancelCursorResponse
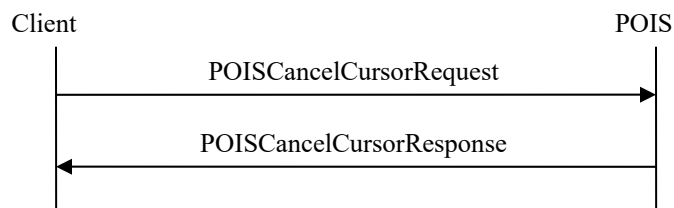
Figure 69: POIS Cancel Cursor Message Exchange

Once the Client has obtained all of the data it needs from a cursor, it should cancel it with the Client via the POISCancelCursorRequest. For more detail on the POISCancelCursor messages, see Section 11.13 (POISCancelCursorRequest and Response Messages) of [SCTE130-5], and Section 11.7 (CancelCursorRequest and CancelCursorResponse Message Types) and Appendix D.3 (Canceling Existing Cursors) of [SCTE130-8].

```
<POISCancelCursorRequest messageId="uuid-115" version="1.0"
system="POISClient" cursorRef="cursor-1" identity="40DA910E-01AF-5050-C7EA-
5D7B4A475311"/>
```

Example 91

The POIS responds with a POISCancelCursorResponse.

```
<POISCancelCursorResponse messageId="uuid-116" version="1.0"
system="POISServer" messageRef="uuid-115" identity="40DA910E-01AF-5050-C7EA-
5D7B4A475312">
  <core:StatusCode class="0"/>
</POISCancelCursorResponse>
```

Example 92

8.3   CIS

This section describes the four major use cases of the CIS: Feature Discovery, Notifications, Queries, and Cursors.

Figure 70: CIS Overview

8.3.1   CIS Feature Discovery

The sequence diagram below illustrates feature discovery message flow between a Client and a CIS.

Figure 71: CIS Discovery Sequence Diagram

While there is no required registration process that occurs before queries between a Client and a CIS can begin, there are mechanisms for determining endpoints and supported query languages.

8.3.1.1 *ListSupportedFeaturesRequest/ListSupportedFeaturesResponse*
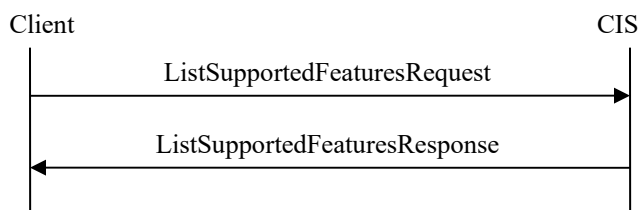
Figure 72: CIS List Supported Features Message Exchange

In order to determine the endpoints, data models, and query languages supported by the CIS, the Client sends a ListSupportedFeaturesRequest to the CIS. For more detail on the ListSupportedFeatures messages, see Section 10.9 (List Supported Features Request and Response) and Appendix E.1 (List Supported Features Request and Response) of [SCTE130-4].

```
<ListSupportedFeaturesRequest messageId="uuid-117" system="sys-1"
version="1.1" identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
</ListSupportedFeaturesRequest>
```

Example 93

The CIS responds with a ListSupportedFeaturesResponse.

```
<ListSupportedFeaturesResponse messageId="uuid-118" system="sys-cis"
version="1.1" identity="40DA910E-01AF-5050-C7EA-5D7B4A475312"
messageRef="uuid-117">
   <core:StatusCode class="0"/>
   <core:Callout>
       <core:Address type="SOAP">http://10.250.30.22/ADSServer</core:Address>
   </core:Callout>
   <DataModelList>
       <core:ContentDataModel type="CLADI_1.1">URI </core:ContentDataModel>
   </DataModelList>
   <AdvancedQueryLanguageList>
       <AdvancedQueryLanguage version="1.0">XPath</AdvancedQueryLanguage>
       <AdvancedQueryLanguage version="1.0">XQuery</AdvancedQueryLanguage>
   </AdvancedQueryLanguageList>
</ListSupportedFeaturesResponse>
```

Example 94

8.3.2   CIS Notifications

The sequence diagram below illustrates notification message flow between a Client and a CIS.
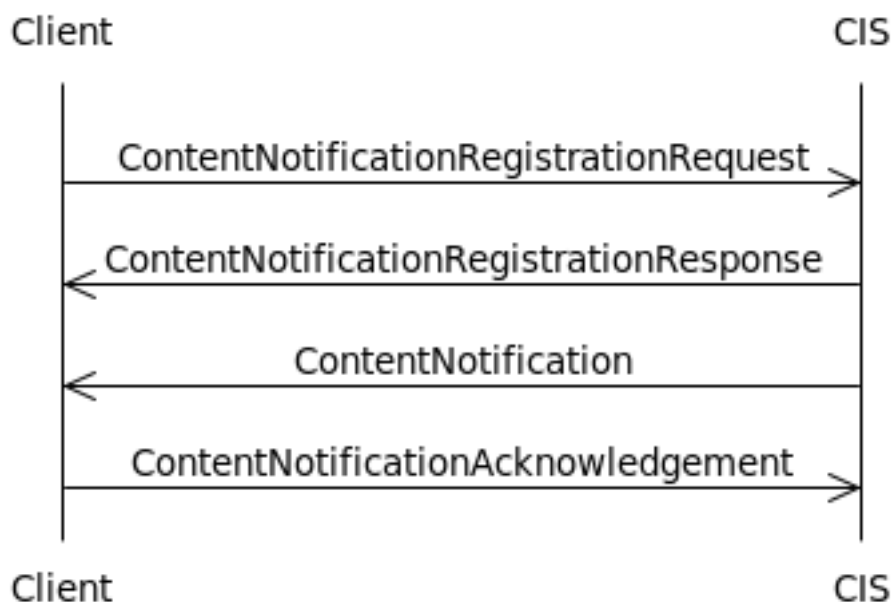
Figure 73: CIS Notification Sequence Diagram

Content data is not necessarily static, and content may be added or removed from the system. The notification messaging allows a Client to register once, then automatically receive notifications for any changes related to the previously defined content. This functionality enables a Client to cache and selectively invalidate content.

8.3.2.1 ContentNotificationRegistrationRequest/ ContentNotificationRegistrationResponse

Figure 74: CIS Notification Registration Message Exchange

In the example below, the Client is registering for notifications for all provider IDs, which is essentially all content on the system. For more detail on the ContentNotificationRegistration messages, see Section 10.11 (Content Notification Registration Request and Response) and Appendix E.3 (Content Notification Registration Request) of [SCTE130-4].

```
<ContentNotificationRegistrationRequest messageId="uuid-119" system="sys-1" version
="1.1" identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
    <core:Callout message="ContentNotification">
        <core:Address type="SOAP">http://10.250.30.22/ADSServer</core:Address>
    </core:Callout>
    <ContentNotificationSelector queryId="query-1">
       <core:ContentDataModel type="CLADI_1.1">URI</core:ContentDataModel>
       <QueryFilter>
            <FilterElement name="Provider_ID" value=".*" valueIsRegex="true"/>
       </QueryFilter>
    </ContentNotificationSelector>
</ContentNotificationRegistrationRequest>
```

Example 95

The CIS responds with a ContentNotificationRegistrationResponse that acknowledges the registration.

```
<ContentNotificationRegistrationResponse messageId="uuid-120"
system="CISServer" version="1.0" identity="40DA910E-01AF-5050-C7EA-5D7B4A475312"
messageRef="uuid-119">
  <core:StatusCode class="0"/>
</ContentNotificationRegistrationResponse>
```

Example 96

8.3.2.2 ContentNotification/ContentNotificationAcknowledgement

CIS   Client

ContentNotification

ContentNotificationAcknowledgement
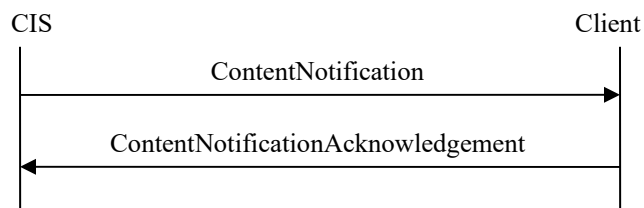
Figure 75: CIS Notification Message Exchange

Whenever a piece of content on the CIS is added, deleted, updated, the CIS initiates a
ContentNotification. The attribute @type="new" denotes that a new asset has been added. For
more detail on the ContentNotification messages, see Section 10.12 (Content Notification and
Acknowledgement) and Appendix E.4 (Content Notification) of [SCTE130-4].

```
<ContentNotification type="new" messageId="uuid-121" system="acs-1" version ="1.1"
    identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
    <ContentQueryResult resultSetSize="1" contentQueryRef="1">
       <BasicQueryResultList>
          <core:Content>
              <core:AssetRef assetID="XXXX000000000001" providerID="max.com"/>
              <core:ContentLocation mediaAvailable="true"></core:ContentLocation>
          </core:Content>
       </BasicQueryResultList>
    </ContentQueryResult>
</ContentNotification>
```
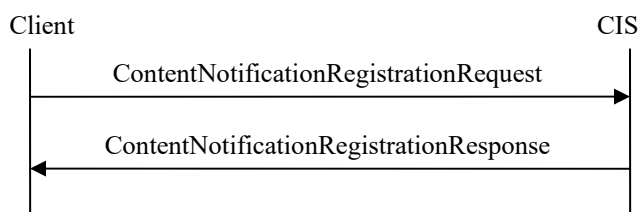
Example 97

Notice that the notification only includes the AssetRef. It will be up to the Client to follow up
with a ContentQueryRequest with @expandOutput="true" to obtain the full metadata, as
previously shown in Example 13.

The Client responds with a ContentNotificationAcknowledgement.

```
<ContentNotificationAcknowledgement messageId="uuid-122"
system="CISClient" version="1.0" identity="40DA910E-01AF-5050-C7EA-5D7B4A475311"
messageRef="uuid-121">
  <core:StatusCode class="0"/>
</ContentNotificationAcknowledgement>
```

Example 98

8.3.3 CIS Queries

The sequence diagram below illustrates query message flow between a Client and a CIS.
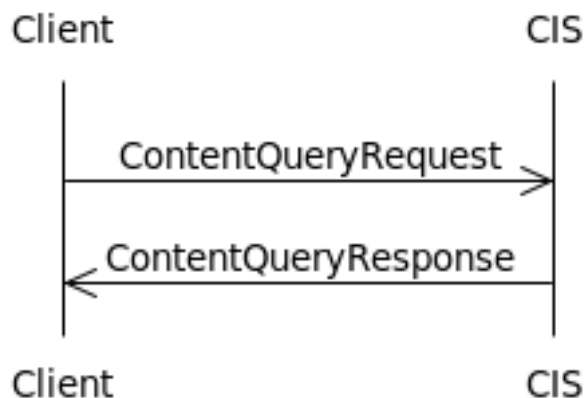
Figure 76: CIS Query Sequence Diagram

The Client is responsible for initiating queries to the CIS, and the CIS will respond with the relevant content information.

8.3.3.1 ContentQueryRequest/ContentQueryResponse
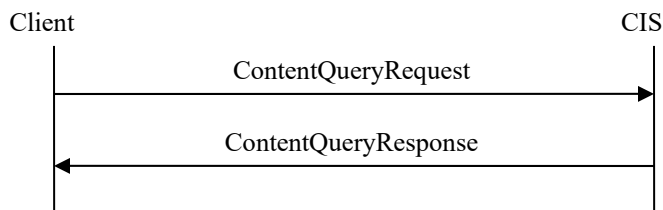
Figure 77: CIS Query Message Exchange

8.3.3.1.1 Unique Identifier

Throughout this document, examples with information services have been shown with a unique identifier, with the intention of the calling SCTE130 component receiving a result set size of 1 back. With the CIS, a combination of providerId and assetId is unique to a particular piece of content. This is ideal in a situation where the Client is not caching data to potentially improve performance and is only interested in the content that is relevant to the PlacementRequest it is attempting to service. The example from Section 7.2.2, which uses filters with the assetId and providerId, is below.

```
<ContentQueryRequest messageId="uuid-123" system="sys-1" version ="1.1"
    identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
    <ContentQuery expandOutput="true" contentQueryId="1">
       <core:ContentDataModel type="CLADI_1.1">URI</core:ContentDataModel>
       <QueryFilter>
            <FilterElement name="Provider_ID"  value="entertainment.com" />
            <FilterElement name="Asset_ID"  value="ABCD0123456789012345" />
       </QueryFilter>
    </ContentQuery>
</ContentQueryRequest>
```

Example 99

The CIS responds with the full metadata in the specified content data model, CLADI 1.1.

```
<ContentQueryResponse messageId="uuid-124" system="acs-1" version="1.1"
     identity="40DA910E-01AF-5050-C7EA-5D7B4A475312" messageRef="uuid-123">
    <core:StatusCode class="0"/>
    <ContentQueryResult resultSetSize="1" contentQueryRef="1">
       <BasicQueryResultList>
            <core:Content>
                <core:ContentLocation
mediaAvailable="true">URI</core:ContentLocation>
                <core:Ext>
                   <ADI>
                       <Metadata> <AMS Asset_Name="..."/> …</Metadata>
                   </ADI>
                </core:Ext>
            </core:Content>
       </BasicQueryResultList>
    </ContentQueryResult>
</ContentQueryResponse>
```

Example 100

8.3.3.1.2 Non-Unique Identifier

Calling out to an information service at the beginning of every viewing session in a VOD system make could be considered inefficient, particularly in systems in which there are tight time constraints. An alternate method of gathering content information is to use a non-unique identifier to get information on all content that fall under a particular filter.

For this example, assume that the Client is interested in all content on the CIS. The Client can query for this by using a regex, calling for all "Provider_ID" with a value of ".*".

```
<ContentQueryRequest messageId="uuid-125" system="sys-1" version ="1.1"
   identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
   <ContentQuery contentQueryId="1">
     <core:ContentDataModel type="CLADI_1.1">URI </core:ContentDataModel>
     <QueryFilter>
          <FilterElement name="Provider_ID" value=".*" valueIsRegex="true"/>
     </QueryFilter>
   </ContentQuery>
</ContentQueryRequest>
```

Example 101

The CIS will respond with all content. The full list has been intentionally omitted here.

```
<ContentQueryResponse messageId="uuid-126" system="sys-cis" version="1.1"
    identity="40DA910E-01AF-5050-C7EA-5D7B4A475312" messageRef="uuid-125">
   <core:StatusCode class="0"/>
   <ContentQueryResult resultSetSize="100" contentQueryRef="1">
      <BasicQueryResultList>
          <core:Content>
              <core:AssetRef assetID="ABCD0123456789012345" providerID="com.com"/>
              <core:ContentLocation mediaAvailable="true">URI</core:ContentLocation>
          </core:Content>
               <!-- .... -->
      </BasicQueryResultList>
   </ContentQueryResult>
</ContentQueryResponse>
```

Example 102

8.3.4 CIS Cursors

The sequence diagram below illustrates cursor message flow between a Client and a CIS.
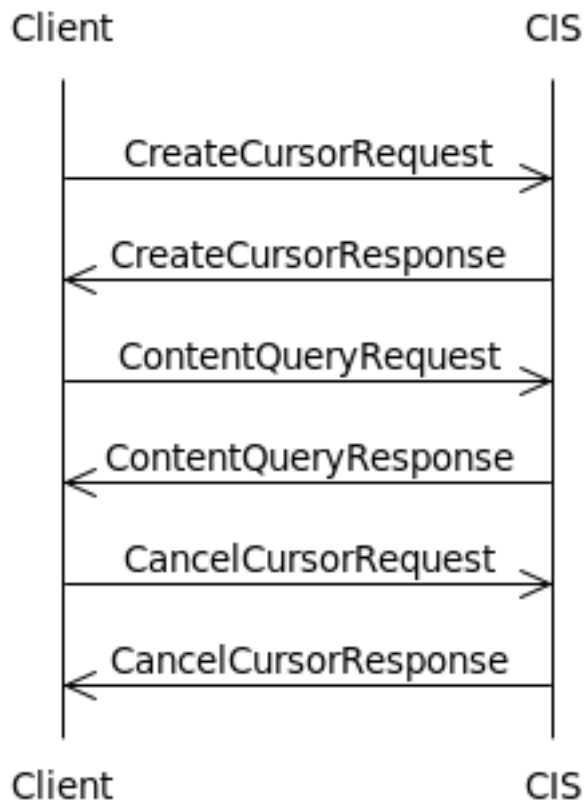
Figure 78: CIS Cursor Sequence Diagram

When a Client queries a CIS with a non-unique identifier, such as the examples in Section 8.3.3.1.2, the amount of data in the response could be quite large. To obtain the data in chunks, the Client can set up a cursor and walk the data set, requesting a subset of the data over multiple ContentQueryRequests until it has all of the data.

8.3.4.1 CreateCursorRequest/CreateCursorResponse
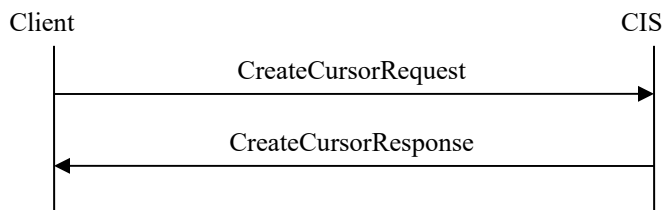
Figure 79: CIS Create Cursor Message Exchange

In the example below, the Client is creating a cursor for all asset metadata with a Provider_ID. For more detail on the CreateCursor messages, see Section 10.13 (Create Cursor Request and Response) and Appendix D.1 (Creating Cursors) of [SCTE130-4].

```
<CreateCursorRequest messageId="uuid-127" version="1.1" system="CIS"
cursorId="cursor-1" cursorExpires="2011-08-10T12:00:00Z" identity="40DA910E-
01AF-5050-C7EA-5D7B4A475311">
 <ContentQuery contentQueryId="query-1" expandOutput="false">
  <core:ContentDataModel type="CLADI_11">URI</core:ContentDataModel>
  <QueryFilter>
   <FilterElement name="Provider_ID" value=".*\.com"
valueIsRegex="true" />
  </QueryFilter>
 </ContentQuery>
</CreateCursorRequest>
```

Example 103

The CIS acknowledges the successful creation of the cursor with a CreateCursorResponse, which
includes the number of total records that the query returned.

```
<CreateCursorResponse messageId="uuid-128" version="1.1" system="CIS"
messageRef="uuid-127" resultSetSize="100" cursorExpires="2011-08-
10T12:00:00.0Z" identity="40DA910E-01AF-5050-C7EA-5D7B4A475312">
    <core:StatusCode class="0"/>
</CreateCursorResponse>
```

Example 104

8.3.4.2   *ContentQueryRequest/ContentQueryResponse*
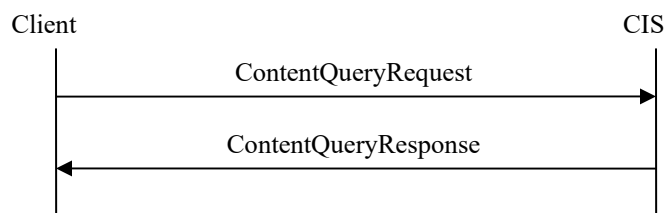
Figure 80: CIS Query Message Exchange

To walk the data in the cursor, the Client will send a ContentQueryRequest to the CIS, which
causes the CIS to return a CISQueryResponse with the specified number of assets. In the
example below, the Client is requesting ten assets, starting at index 0. A subsequent call would
start at index 10 and request ten assets, and this would be repeated until the Client has retrieved
all of the data from the CIS for the cursor.

```
<ContentQueryRequest messageId="uuid-129" version="1.1" system="CIS"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
   <Cursor startIndex="0" count="10" cursorRef="cursor-1"
contentQueryRef="query-1"/>
</ContentQueryRequest>
```

Example 105

8.3.4.3   CancelCursorRequest/CancelCursorResponse
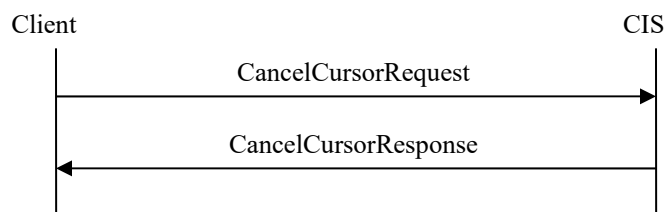
Figure 81: CIS Cancel Cursor Message Exchange

Once the Client has obtained all of the data it needs from a cursor, it should cancel it with the Client via the CancelCursorRequest. For more detail on the CancelCursor messages, see Section 10.14 (Cancel Cursor Request and Response) and Appendix D.3 (Canceling Existing Cursors) of [SCTE130-4].

```
<CancelCursorRequest messageId="uuid-130" version="1.1" system="client1"
cursorRef="cursor-1" identity="40DA910E-01AF-5050-C7EA-5D7B4A475311">
</CancelCursorRequest>
```

Example 106

The CIS responds with a CancelCursorResponse.

```
<CancelCursorResponse messageId="uuid-131" version="1.1" system="CIS"
messageRef="uuid-130" identity="40DA910E-01AF-5050-C7EA-5D7B4A475312">
    <core:StatusCode class="0"/>
</CancelCursorResponse>
```

Example 107

9 Advanced Topics

Some topics warranted a more in depth discussion than was offered in the SCTE 130 suite of standards, and the purpose of this section is to explore those advance topics and expand on the explanations in the existing SCTE 130 suite of standards.

9.1 Fill versus Replace

The Placement element contains an @action attribute that controls the way that the ADM handles the given Placement.
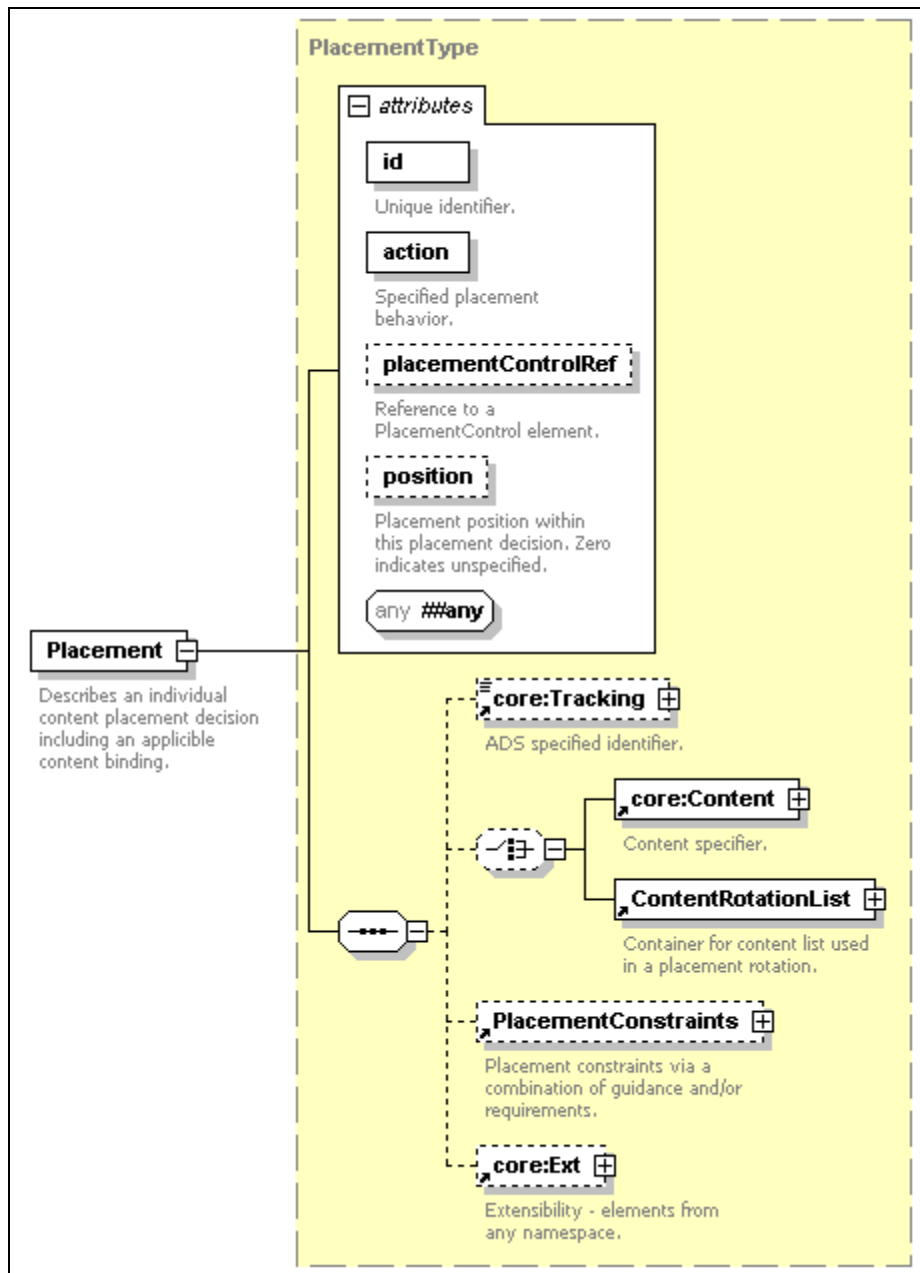
Figure 82: Placement Element

The valid values for @action are listed in the table below, taken from [SCTE130-3].

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 1: Values for the @action attribute

The two most common values for the @action attribute are fill and replace, which this section will detail. With a replace, the placement will overwrite a portion of the entertainment asset, and fill will insert an ad at the designated NPT, as if the entertainment asset had been paused, then resumed after the placement was displayed.

In the figure below, an entertainment asset is shown with no Placements, but a potential insertion point is noted. Note: Figures are not to scale.

BOS

NPT = x,
Insertion Point

EOS
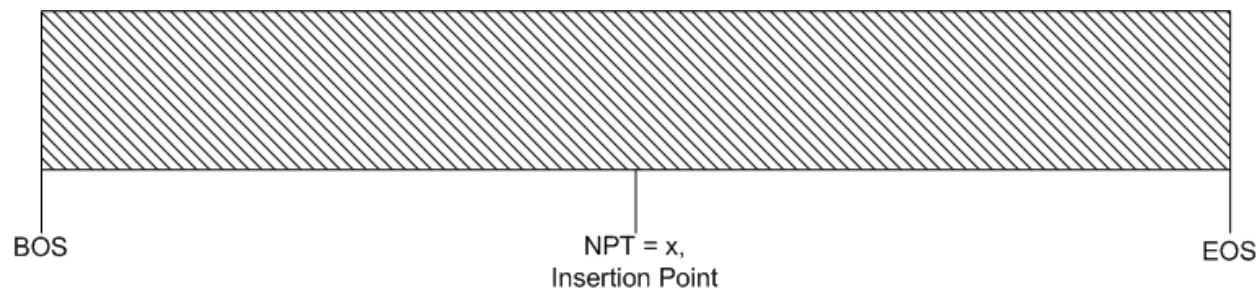
Figure 83: Contiguous Entertainment Asset

With a fill Placement, none of the entertainment asset will be omitted and the ad will be displayed in its entirety before resuming the entertainment.
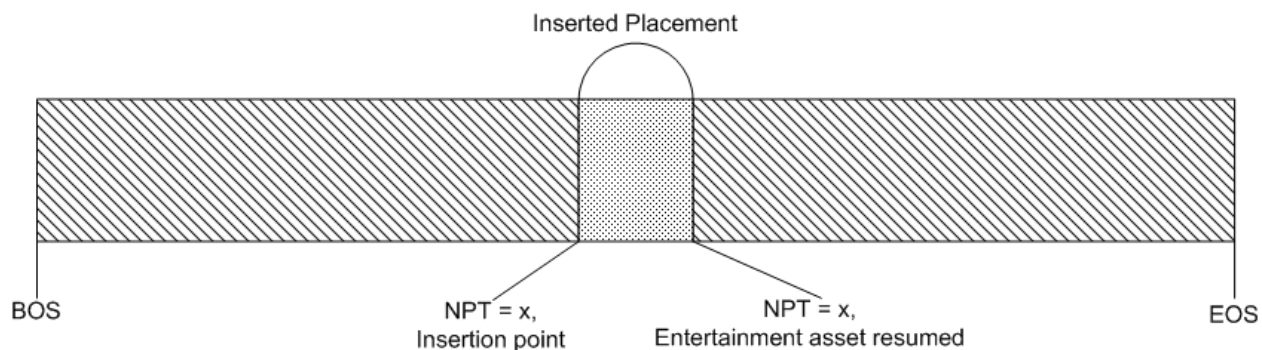
Figure 84: Entertainment Asset with a fill Placement

With a replace, a portion of the entertainment asset will be omitted. In the example below, a 30 second ad is replacing 30 seconds of the entertainment asset.
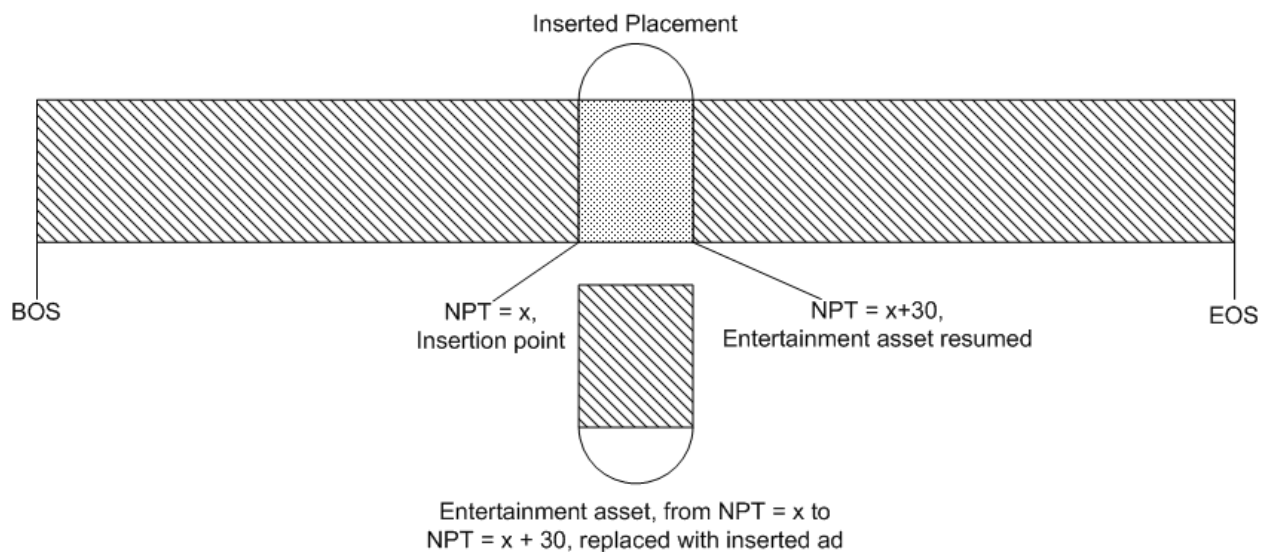
Figure 85: Entertainment Asset with a replace Placement

With the replace, the subscriber is not seeing the 30 seconds of the entertainment content that the Placement is overwriting. This is particularly useful when a piece of content has baked in ads. The ads may have to display for a period of time due to contractual obligations, but can be replaced with updated or better targeted ads once the contractual obligations have been met.

9.2 midRolls

In the OpportunityBinding element, there is an @opportunityType attribute that identifies the type of the PlacementOpportunity.
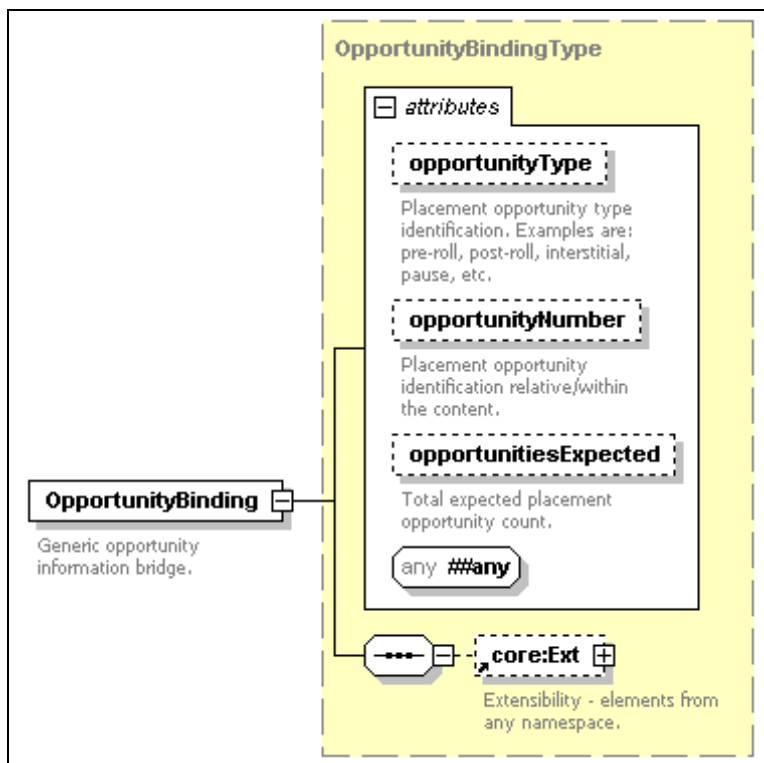
Figure 86: OpportunityBinding Element

The valid values for @opportunityType are listed in the table below, taken from [SCTE130-3].

Value	Description
preRoll	Prior to the start of the entertainment content.
postRoll	Post the end of the entertainment content.
interstitial	Anywhere within the 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 2: Values for the @opportunityType attribute

A midRoll is defined as a placement opportunity occurring during the play out of an entertainment asset. While typically associated to linear broadcasting, midRolls are not limited to this context and can occur in VOD as well. The insertion points described in Section 9.1 are considered midRoll placement opportunities.

This was a term that was often used interchangeably with the term interstitial, but interstitial has evolved into describing an opportunity that resides between pieces of entertainment content, while midRoll has kept the definition above. [SCTE130-3] has not caught up to this industry practice (notice the absence of midRoll in the above chart), so to avoid confusion, "private:midRoll" should be used when constructing midRoll placement opportunities.

midRoll placement opportunities are more complex to determine in VOD since they reside within the entertainment content. The insertion points need to make sense within the content and not reside in the middle of a line of dialog or a scene that doesn't lend itself well to insertion. How these placements are determined is outside of the scope of SCTE 130, but this can be accomplished by utilizing SCTE 35 cue messages, or some other sort of out of band metadata ingestion mechanism.

9.3 Playlisting

Playlisting refers to the practice of returning a PlacementResponse comprising a list of entertainment assets or asset segments along with advertising assets. Elsewhere in this document and in the SCTE 130 suite of standards, the examples of ADS PlacementResponse messages include only information about advertising assets. However, it has become a common practice in

the industry to make use of playlist-style PlacementResponse messages. This section describes a recommended approach and messaging format for such playlists.

In this section, we will use the example of creating a playlist that describes ad insertion into the fictitious VOD entertainment asset depicted in the figure below. The asset has one midroll break with "baked in" ads beginning 288.00 seconds into the asset and ending at 319.00 seconds.
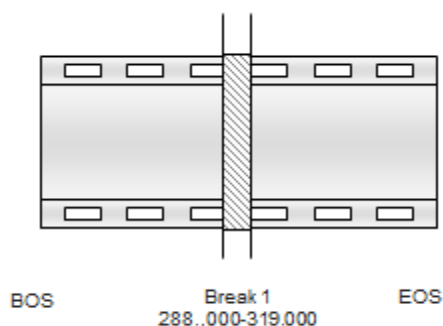
Figure 87: VOD entertainment asset prior to ad insertion

In this example, the baked in ads will be replaced with dynamically inserted ads, and one pre- and post-roll ad each will be included, as well. The entertainment asset is depicted again in the figure below, this time with the dynamically added pre-, mid-, and post-roll ads shown.
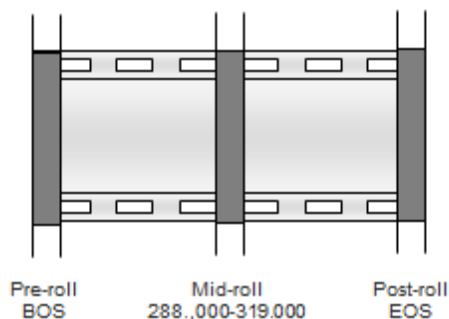
Figure 88: VOD entertainment asset following ad insertion

9.3.1   PlacementRequest

The PlacementRequest below is reproduced from 7.4.1, VOD with an Intermediary. A non-ADM-centric architecture is typical of 130 systems that use playlisting. Since a service other

than the ADM is returning a playlist to the ADM, the ADM would likely delegate responsibility for calling other SCTE 130 services, as well. In such cases, a very sparse PlacementRequest such as that used with an intermediary would be typical. Example 108 below includes only an asset ID and provider ID for locating content metadata, a MAC address for looking up any subscriber metadata, and an empty placement opportunity, usually signifying that a POIS will provide the placement opportunities for this request.

```
<PlacementRequest messageId="uuid-14" version="1.1" 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>
    <TerminalAddress type="MAC">03-55-78-90-03-02</TerminalAddress>
  </Client>
  <PlacementOpportunity id="uuid-15" serviceRegistrationRef="ServiceRef1">
  </PlacementOpportunity>
</PlacementRequest>
```

Example 108

9.3.2

9.3.3 PlacementResponse

Example 109 below shows a PlacementResponse that uses the PlayPositions structure to describe the regions of the entertainment asset.

```
<?xml version="1.0" encoding="UTF-8"?>
<PlacementResponse system="ads" identity="MY_SYSTEM" version="1.3" messageId="uuid-xx"
messageRef="uuid-14" xmlns="http://www.scte.org/schemas/130-3/2008a/adm"
xmlns:core="http://www.scte.org/schemas/130-2/2008a/core"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="
http://www.scte.org/schemas/SCTE_130-3_2013.xsd">
  <core:StatusCode class="0"/>
  <!-- One pre-roll ad -->
  <PlacementDecision placementOpportunityRef="6655f515-c544-4324-b446-8915ed8bcede"
id="4e0419b5-c66a-492b-9407-dabaf43ec003">
    <Placement action="fill" id="03-01">
      <core:Content>
        <core:AssetRef assetID="ADAD0000000000003005" providerID="advertiser1.com"/>
        <core:Tracking>uuid-xx</core:Tracking>
      </core:Content>
    </Placement>
  </PlacementDecision>
  <!-- Entertainment asset plays from 0.000-288.000 -->
  <PlacementDecision placementOpportunityRef="6655f515-c544-4324-b446-8915ed8bcede"
id="245931db-a087-4685-811c-b23d55d1d823">
    <Entertainment>
      <core:Content>
        <core:AssetRef assetID="ABCD01123456789012345" providerID="entertainment.com"/>
      </core:Content>
      <!-- EntertainmentNPT scale="1.0">0.000-288.000</EntertainmentNPT -->
```

```
      <PlayPositions>
        <PlayPositionStart scale="1.0">
          <NPTOffset>0.000</NPTOffset>
        </PlayPositionStart>
        <PlayPositionEnd scale="1.0">
          <NPTOffset>288.000</NPTOffset>
        </PlayPositionEnd>
      </PlayPositions>
    </Entertainment>
  </PlacementDecision>
  <!-- One midRoll ad -->
  <PlacementDecision placementOpportunityRef="6655f515-c544-4324-b446-8915ed8bcede"
id="9839470c-8a39-42cc-97d6-5db9a01f1773">

    <Placement action="replace" id="05-01">
      <core:Content>
        <core:AssetRef assetID="ADAD0000000000003005" providerID="advertiser3.com"/>
        <core:Tracking>uuid-xxxx</core:Tracking>
      </core:Content>
    </Placement>
  </PlacementDecision>
  <!-- Entertainment asset plays from 319.000-1258.000 -->
  <PlacementDecision placementOpportunityRef="6655f515-c544-4324-b446-8915ed8bcede"
id="b03322e9-c666-49ca-8e38-9d14fe1ddbbc">
    <Entertainment>
      <core:Content>
        <core:AssetRef assetID=" ABCD01234567890012345" providerID="
entertainment.com"/>
      </core:Content>
      <!-- EntertainmentNPT scale="1.0">319.000-1258.000</EntertainmentNPT -->
      <PlayPositions>
        <PlayPositionStart scale="1.0">
          <NPTOffset>319.000</NPTOffset>
        </PlayPositionStart>
        <PlayPositionEnd scale="1.0">
          <NPTOffset>1258.000</NPTOffset>
        </PlayPositionEnd>
      </PlayPositions>
    </Entertainment>
  </PlacementDecision>
  <!-- Post-roll ad -->
  <PlacementDecision placementOpportunityRef="6655f515-c544-4324-b446-8915ed8bcede"
id="19cd6414-4c91-4844-9611-c579ccd42c46">
    <Placement action="fill" id="37-01">
      <core:Content>
        <core:AssetRef assetID="TEST0502000000000002" providerID="advertiser2.com"/>
        <core:Tracking>upid-yy</core:Tracking>
      </core:Content>
    </Placement>
  </PlacementDecision>
</PlacementResponse>
```

Example 109

9.3.4   Incorporating Asset Metadata via ADI 3.0

Creating NPT-based playlists of the kind discussed in sections 9.3.2 and 9.3.3 requires break timing information expressed as NPT values. One specification-based approach to communicating such break information to ad decision systems is via ADI 3.0 signaling metadata.

Typically, such information would be included in the asset metadata pitched to the ad decision system and stored in the CIS associated with the system. The component responsible for creating the playlist could then query the CIS to retrieve the NPT-based break information it needs to produce the playlist.

The signaling metadata for the asset depicted in Figure 87 above is shown in Example 110. The metadata contains one signal region, including a start point at 288.000 seconds and an end point at 319.000. This break duration matches the midroll break duration for the entertainment asset in our example exactly.

```
<SignalGroup>
  <SignalRegion xsi:type="SignalRegionType">
    <StartPoint>
      <NPTPointDescriptor nptPoint="288.000"/>
    </StartPoint>
    <EndPoint>
      <NPTPointDescriptor nptPoint="319.000"/>
    </EndPoint>
  </SignalRegion>
</SignalGroup>
```

Example 110

9.4 Stream Restriction

[SCTE 130-10] describes a data model for expressing stream restriction information. Stream restrictions are often used to prevent viewer actions such as fast forwarding or rewinding during a particular portion of video content. This section demonstrates use of stream restrictions to prevent viewers from fast forwarding through a specific ad, the mid-roll in the entertainment asset used in this example.

Example 111 shows the excerpted StreamRestrictionList element used to express fast forwarding restrictions on the mid-roll ad. The contents of the ScaleRange attribute indicate that the upper bound of the scale range is "1," or normal play speed, while the lower bound is unlimited. In plain language, the stream restriction indicates that the mid-roll ad may be played at normal speed, more slowly, or even rewound, but it may not be fast-forwarded. See [SCTE 130-10] for more details on stream restriction usage.

```
<StreamRestrictionList>
  <StreamRestriction>
    <ScaleRange lowerbound="-INF" upperbound="1" lowertest="unbounded" uppertest="lteq"/>
  </StreamRestriction>
</StreamRestrictionList>
```

Example 111: Stream restriction elements for preventing fast forward

This same stream restriction for fast forwarding is also demonstrated in context in Example 112, below (highlight). Here we have copied **Error! Reference source not found.** and extended the first mid-roll placement decision in the playlist. The decision now contains the stream restriction controls inside an extension (core:Ext) element as a child of the PlacementConstraints element, which is our recommended practice. This placement of the stream restriction indicates that fast forward is restricted for the first mid-roll ad alone. Other Placements do not have the same restriction.

```
<PlacementResponse system="ads" identity="MY_SYSTEM" version="1.1" messageId="uuid-xx"
messageRef="uuid-14">
  <core:StatusCode class="0"/>
    <!-- One pre-roll ad -->
  <PlacementDecision placementOpportunityRef="6655f515-c544-4324-b446-8915ed8bcede" id="4e0419b5-
c66a-492b-9407-dabaf43ec003">
    <Placement action="fill" id="03-01">
      <core:Content>
        <core:AssetRef assetID="ADAD0000000000003005" providerID="advertiser1.com"/>
        <core:Tracking>uuid-xx</core:Tracking>
      </core:Content>
    </Placement>
  </PlacementDecision>
  <!-- Entertainment asset plays from 0.000-288.000 -->
  <PlacementDecision placementOpportunityRef="6655f515-c544-4324-b446-8915ed8bcede" id="245931db-
a087-4685-811c-b23d55d1d823">
    <Entertainment>
      <core:Content>
        <core:AssetRef assetID="ABCD0123456789012345" providerID="entertainment.com"/>
      </core:Content>
      <EntertainmentNPT scale="1.0">0.000-288.000</EntertainmentNPT>
    </Entertainment>
  </PlacementDecision>
  <!-- One midRoll ad -->
  <PlacementDecision placementOpportunityRef="6655f515-c544-4324-b446-8915ed8bcede" id="9839470c-
8a39-42cc-97d6-5db9a01f1773">
    <PlacementConstraints>
      <core:Ext>
        <StreamRestrictionList>
          <StreamRestriction>
            <ScaleRange lowerbound="-INF" upperbound="1" lowertest="unbounded" uppertest="lteq"/>
          </StreamRestriction>
        </StreamRestrictionList>
      </core:Ext>
    </PlacementConstraints>
    <Placement action="replace" id="05-01">
      <core:Content>
        <core:AssetRef assetID="ADAD0000000000003005" providerID="advertiser3.com"/>
        <core:Tracking>uuid-xxxx</core:Tracking>
      </core:Content>
    </Placement>
  </PlacementDecision>
  <!-- Entertainment asset plays from 319.000-1258.000 -->
  <PlacementDecision placementOpportunityRef="6655f515-c544-4324-b446-8915ed8bcede" id="b03322e9-
c666-49ca-8e38-9d14fe1ddbbc">
    <Entertainment>
      <core:Content>
        <core:AssetRef assetID=" ABCD0123456789012345" providerID=" entertainment.com"/>
      </core:Content>
      <EntertainmentNPT scale="1.0">319.000-1258.000</EntertainmentNPT>
    </Entertainment>
  </PlacementDecision>
  <!-- Post-roll ad -->
  <PlacementDecision placementOpportunityRef="6655f515-c544-4324-b446-8915ed8bcede" id="19cd6414-
4c91-4844-9611-c579ccd42c46">
    <Placement action="fill" id="37-01">
      <core:Content>
        <core:AssetRef assetID="TEST0502000000000002" providerID="advertiser2.com"/>
```

```
        <core:Tracking>upid-yy</core:Tracking>
      </core:Content>
    </Placement>
  </PlacementDecision>
</PlacementResponse>
```

Example 112 – Stream Restrictions in a Placement Response

Finally, Example 113 illustrates the Stream Restriction extensions within the Placement Request borrowed from Example 108. Here we also recommend extending the PlacementConstraints element as a child of the PlacementOpportunity.

```
<PlacementRequest messageId="uuid-14" version="1.1" 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>
    <TerminalAddress type="MAC">03-55-78-90-03-02</TerminalAddress>
  </Client>
  <PlacementOpportunity id="uuid-15" serviceRegistrationRef="ServiceRef1">
    <PlacementConstraints>
      <core:Ext>
        <StreamRestrictionList>
          <StreamRestriction>
            <ScaleRange lowerbound="-INF" upperbound="1" lowertest="unbounded" uppertest="lteq"/>
          </StreamRestriction>
        </StreamRestrictionList>
      </core:Ext>
    </PlacementConstraints>
  </PlacementOpportunity>
</PlacementRequest>
```

Example 113

9.5 Multiple Placement Opportunity Ownership

Within a single entertainment asset, multiple placement opportunities may exist, and each of those opportunities could be filled by a different ADS. Also, a single PlacementOpportunity may be subdivided into many opportunities by creating multiple PlacementControl elements, and each of these subdivided opportunities can be serviced by different ADSs. Deciding which ADS receives the individual placement opportunities is multiple step process that requires coordination across the system. A common way to do this is with the Scope element, which is detailed in the figure below.
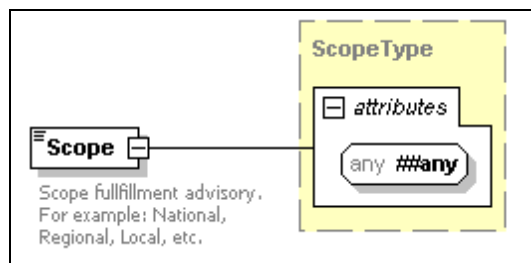
Figure 89: Scope Element

The valid values for Scope are listed in the table below, taken from [SCTE130-3].

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 3: Values for Scope

The following sections outline the steps required to set up and execute PlacementDecisions in a SCTE 130 environment that handles multiple ownership. Assume there exists two ADSs: a local ADS that is handling Local placement opportunities, and a national ADS that is handling National placement opportunities.

9.5.1   Registration

First, the ADSs are required to register with the ADM for their respective Scopes. The Scope element resides in ADSRegistrationRequest/ServiceDesrcription/Service.

Local ADS:

```
<ADSRegistrationRequest messageId="uuid-132" version="1.1" system="localADS"
identity="LocalADS">
  <ServiceDescription>
    <SystemContext/>
    <Service id="ServiceLocal">
      <Scope>Local</Scope>
    </Service>
  </ServiceDescription>
  <core:Callout>
  <core:Address>www.localads.com/allMessages</core:Address>
  </core:Callout>
</ADSRegistrationRequest>
```

Example 114

National ADS:

```
<ADSRegistrationRequest messageId="uuid-133" version="1.1"
system="nationalADS" identity="NationalADS">
  <ServiceDescription>
    <SystemContext/>
    <Service id="ServiceNational">
      <Scope>National</Scope>
    </Service>
  </ServiceDescription>
  <core:Callout>
  <core:Address>www.nationalads.com/allMessages</core:Address>
  </core:Callout>
</ADSRegistrationRequest>
```

Example 115

With these registrations in its system, the ADM will direct any Local placement opportunities to the local ADS, and any National opportunities to the national ads.

9.5.2   Placement

Now that the ADSs are registered with the ADM, it can begin to determine ownership and route PlacementRequests accordingly. The sequence diagram below shows placement messaging between the ADM and each ADS.
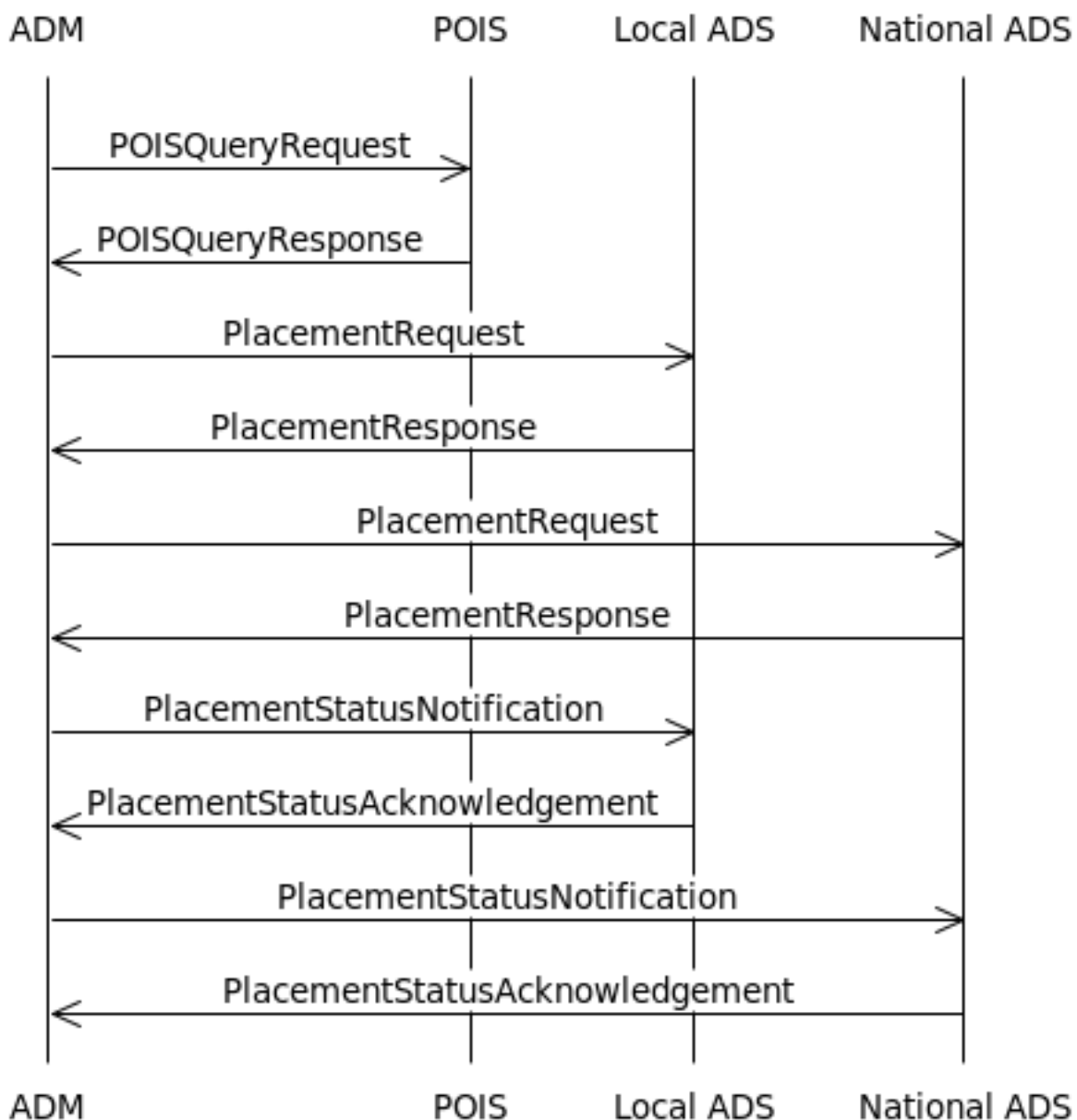
Figure 90: Multiple Ownership Placement Messaging Sequence Diagram

9.5.2.1   *POISQueryRequest/POISQueryResponse*

The ADM can query the POIS to determine if any placement opportunities exist in the content and in particular, the Scope of those placement opportunities.
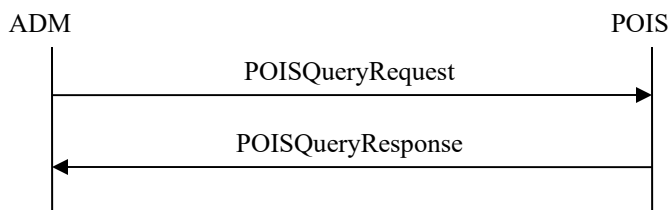
Figure 91: Multiple Ownership POIS Messaging

Using BasicQueryFilters, the ADM can query for PlacementOpportunity information for a particular asset.

```
<POISQueryRequest messageId="uuid-134" system="POISClient"
identity="40DA910E-01AF-5050-C7EA-5D7B4A475757" version="1.0">
      <gis:Query queryId="pois-98765" expandOutput="true"
uniqueQualifierNameRef="P3-PODM">
            <gis:ServiceDataModel>"http://www.scte.org/schemas/130-
3/2008a/adm/podm"</gis:ServiceDataModel>
            <gis:BasicQueryFilter>
                  <gis:BasicFilterElement value="ABCD0123456789012345"
name="assetId"/>
            </gis:BasicQueryFilter>
            <gis:BasicQueryFilter>
                  <gis:BasicFilterElement value="entertainment.com"
name="providerId"/>
            </gis:BasicQueryFilter>
      </gis:Query>
</POISQueryRequest>
```

Example 116

The POIS then responds with a POISQueryResponse that includes all of the metadata it has for the queried content. In this response, the ADM learns that two placement opportunities exist, one with a Local PlacementOpportunity, the other a National.

```
<POISQueryResponse version="1.1" messageId="uuid-135" messageRef="uuid-134"
system="POISServer" identity="50DA910E-01AF-5050-C7EA-5D7B4A475759">
      <core:StatusCode class="0"/>
      <gis:QueryResult resultSetSize="1" queryRef="pois-98765">
            <podm:PODMBasicQueryResult>
                  <podm:PlacementOpportunityV2 serviceRegistrationRef="-"
id="uuid-136">
                        <adm:Entertainment>
                              <core:Content>
                                    <core:AssetRef
providerID="entertainment.com" assetID="ABCD0123456789012345"/>
                              </core:Content>
                        </adm:Entertainment>
                        <adm:OpportunityBinding opportunityType="preRoll"
opportunitiesExpected="2" opportunityNumber="1"/>
                        <podm:OpportunityConstraintsV2>
                              <core:AdType>30-Second-Spot</core:AdType>
                              <core:Duration>PT30S</core:Duration>
                              <adm:Scope>Local</adm:Scope>
                        </podm:OpportunityConstraintsV2>
                  </podm:PlacementOpportunityV2>
                  <podm:PlacementOpportunityV2 serviceRegistrationRef="-"
id="uuid-137">
                        <adm:Entertainment>
                              <core:Content>
                                    <core:AssetRef
providerID="entertainment.com" assetID="ABCD0123456789012345"/>
                              </core:Content>
                        </adm:Entertainment>
                        <adm:OpportunityBinding opportunityType="postRoll"
opportunitiesExpected="2" opportunityNumber="2"/>
                        <podm:OpportunityConstraintsV2>
                              <core:AdType>30-Second-Spot</core:AdType>
                              <core:Duration>PT30S</core:Duration>
                              <adm:Scope>National</adm:Scope>
                        </podm:OpportunityConstraintsV2>
                  </podm:PlacementOpportunityV2>
            </podm:PODMBasicQueryResult>
      </gis:QueryResult>
</POISQueryResponse>
```
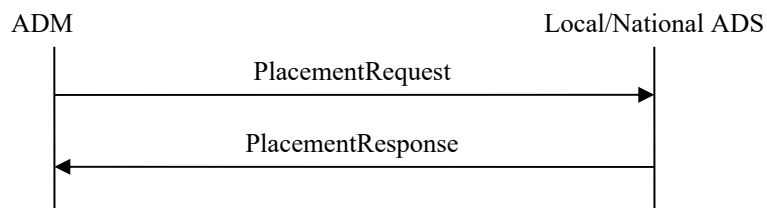
Example 117

Each of the placement opportunities in the POISQueryResponse will correspond to a different PlacementRequest, with each being sent to a different ADS.

9.5.2.2 *PlacementRequest*

Figure 92: Multiple Ownership Placement Messaging

Now the ADM can construct PlacementRequests for each PlacementOpportunity to send to their respective ADSs.

Local ADS:

```
<PlacementRequest messageId="uuid-138" version="1.1" identity="ADM"
system="ADM" updatesAllowed="false">
  <Entertainment>
    <core:Content>
      <core:AssetRef providerID="entertainment.com"
assetID="ABCD0123456789012345"/>
    </core:Content>
  </Entertainment>
  <PlacementOpportunity id="uuid-136" serviceRegistrationRef="LocalService">
    <OpportunityBinding opportunityType="preRoll"/>
    <OpportunityConstraints>
      <Scope>Local</Scope>
    </OpportunityConstraints>
    <PlacementControl id="uuid-139" action="fill">
      <OpportunityConstraints>
        <core:Duration>PT30S</core:Duration>
      </OpportunityConstraints>
    </PlacementControl>
  </PlacementOpportunity>
</PlacementRequest>
```

Example 118

National ADS:

```
<PlacementRequest messageId="uuid-140" version="1.1" identity="ADM"
system="ADM" updatesAllowed="false">
  <Entertainment>
    <core:Content>
      <core:AssetRef providerID="entertainment.com"
assetID="ABCD0123456789012345"/>
    </core:Content>
  </Entertainment>
  <PlacementOpportunity id="uuid-137"
serviceRegistrationRef="NationalService">
    <OpportunityBinding opportunityType="postRoll"/>
    <OpportunityConstraints>
      <Scope>National</Scope>
    </OpportunityConstraints>
    <PlacementControl id="uuid-141" action="fill">
      <OpportunityConstraints>
        <core:Duration>PT30S</core:Duration>
      </OpportunityConstraints>
    </PlacementControl>
  </PlacementOpportunity>
</PlacementRequest>
```
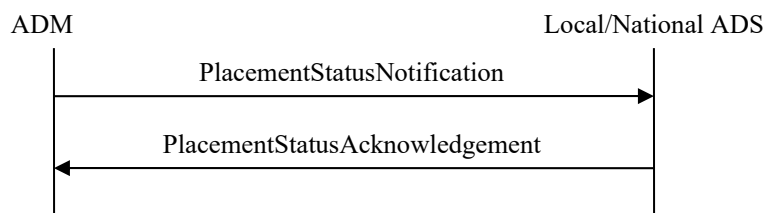
Example 119

Both messages could be sent concurrently, since there are no dependencies between them. Once both PlacementResponses have been returned to the ADM, the ADM is responsible for aggregating the PlacementDecision data and instructing the VOD back office to play the content.

9.5.2.3   *PlacementStatusNotification*

Figure 93: Multiple Ownership PSN Messaging

After the VOD session has completed, trick mode data is required to be propagated to the respective ADSs. Like with the PlacementRequest, the ADM will construct two separate PlacementStatusNotifications, one for each ADS, containing only play data from the Placement it was responsible.

9.6 Extension

9.6.1 Ext

From [SCTE130-2]:

> The Ext (extensibility) element allows zero or more elements from any namespace to be included. This element facilitates expansion, customization, and extensibility of the specification. Encapsulating elements from external namespaces into a single element allows filters, transforms, and other operations to be applied easily.
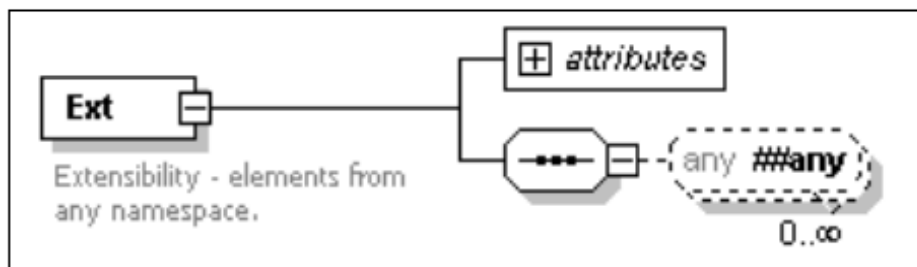

Figure 94: Ext Element

Virtually all complex Elements in the SCTE 130 schemas contain the Ext Element to allow for extension. A common use case for the Ext element has to do with stream restrictions for ad placements. The example in section 9.4 demonstrates and discusses such a usage of the Ext Element. Both the ADM and ADS are required to implement the stream restriction for it to work properly.

9.6.2 ##any

From [SCTE130-2]:

> When describing concrete XML schemas, an element wildcard (<xsd:any/>) is represented by the notation ##any. The ##any element refers to all namespaces including the namespace defined by this document. Any SCTE 130 or alternative namespace element may be included.

Like the Ext element, ##any is meant to facilitate expansion, customization, and extensibility of the standard, and both the client and server are required to implement the extension for it to properly work. The example below shows a Placement element with a custom @someOtherId attribute added to the Placement element. The contents of the PlacementConstraints have been omitted for brevity.

```
<Placement id="uuid-145" action="fill" placementControlRef="uuid-146" someOtherId="id12345">
  <core:Content>
    <core:AssetRef providerID="adsrus.com" assetID="ZXYW0123456789012345"/>
    <core:Tracking>uuid-147</core:Tracking>
  </core:Content>
  <PlacementConstraints>
    …
  </PlacementConstraints>
</Placement>
```

Example 120

9.7 Substitution

While [SCTE130-3] explains the mechanics of substitution, the use cases for it are not immediately obvious. Suppose an Intermediary is sitting between an ADM that does not support PODM, and an ADS that does. The Intermediary then receives a PlacementRequest from the ADM with one PlacementOpportunity. The Intermediary then queries the POIS to determine if any additional placement opportunities exist. Any placement opportunities received from the POIS will be PlacementOpportunityV2. The Intermediary can take all three placement opportunities and combine them in a single PlacementRequest before sending it to the ADS. In addition, the Intermediary has substituted an OpportunityConstraintsV2 into the PlacementOpportunity submitted by the ADM. This substitution allows the Intermediary the ability to provide additional data via the expanded OpportunityConstraintsV2 Element.

```
<PlacementRequest messageId="uuid-148" version="1.1" 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="uuid-149"
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="uuid-150"
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="uuid-151" 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 121

9.8 PlayData Variations

Within the PlacementStatusNotification message, play data can be represented in a variety of ways, at the discretion of the ADM. The ways these Events can be represented within the PlayData element are covered in detail in Section 14.33.2 (PlayData Element Examples) of [SCTE130-3], but the reasons why an ADM may choose each approach are not explained.

When doing simple VOD and only receiving Events from a single source at the end of a viewing session, it may make sense to use the PlacementStatusNotification/PlayData/EntertainmentScopedEvents or PlacementStatusNotification/PlayData/SpotScopedEvents. If an ADM receives Events from multiple sources, or the Events trickle in and are not obtained all at once, it may make sense for the ADM to use the outer Events structure, PlacementStatusNotification/PlayData/Events. Ultimately, the ADM can relay Event information to the ADS in either fashion, regardless of circumstances.

9.9   Service Monitoring and Recovery

There are several techniques a SCTE 130 service can use to monitor the other systems they interact with and recover gracefully from failures.

9.9.1   ServiceCheckRequest

The ServiceCheckRequest is a versatile message that can be used by any SCTE 130 service to verify endpoints and monitor a service's availability. For example, an ADM may use a ServiceCheckRequest on a set interval to verify that an ADS is still online, and throw an alert or stop sending PlacementRequests if it fails to respond, and resume when it successfully responds.

9.9.2   ListADSRegistrationRequest

For an ADS attempting to monitor an ADM, this is slightly more complicated. Availability is not the ADS's only concern, but state (registrations) as well. With respect to placement decision messaging, the ADS is a completely reactive service—once it registers with the ADM, it only responds to PlacementRequests and receives PlacementStatusNotifications. If an ADS did not receive a PlacementRequest for a period of time, it has no way of knowing if this is due to a failure in the ADM, or simply a low volume period with no placement opportunities. In order to ensure that the ADM has not failed in some fashion, the ADS may verify that the ADM still has its registration information by polling it with a ListADSRegistrationRequest on a set interval or after a period of inactivity.

If at some point an ADS fails to find its registration listed in a ListADSRegistrationRequest, it should re-register with the ADM. The ADS does not have to use the @resend attribute during the registration but how this effects outstanding PlacementStatusNotifications is implementation specific and outside the scope of this document.

9.9.3   @resend attribute

The @resend attribute allows a SCTE 130 service to resend a message that it believes may have never reached the destination server. The attribute's most common usage is during failure recovery. An ADM might send PlacementRequests and PlacementStatusNotifications that utilize the @resend attribute after an ADS outage. An ADS may utilize the ADSRegistrationRequest if the ADM somehow loses the ADS's original registration, and an information service could do the same for notification messaging.

While the use of the @resend attribute is always optional, there could be consequences for not using it. For example, imagine that an SIS has added a subscriber and is sending a SISNotification to an ADS, but it never responds with an SISNotificationAcknowlegement. The ADS receives the notification and crashes before responding. Once the ADS recovers, if the SIS attempts the notification again without resend, the ADS will think it is a duplicate and could return an error to the SIS. If the SIS used the @resend attribute, the ADS understands that the message could have already been received and can respond successfully. The same situation could occur with ADS registrations. If an ADM receives an ADSRegistrationRequest and crashes immediately without responding, subsequent ADSRegistrationRequests from the ADS after recovery would fail as duplicates. The ADS can use the @resend attribute, ListADSRegistrationRequest to make sure a registration exists, or de-register and re-register. All of the previously mentioned options are valid.

For further discussion of the @resend attribute and how it is utilized in messaging, see [SCTE130-2], Section 9.4.7 (High Availability and Message Retransmission) and Section 11.2.7 (resendAttrType Attribute Type).