

CableLabs®
SaFI Specifications

Interactive Application Messaging Specification

CL-SaFI-IAM-I01-090626

ISSUED

Notice

This document is the result of a cooperative effort undertaken at the direction of Cable Television Laboratories, Inc. for the benefit of the cable industry and its customers. This document may contain references to other documents not owned or controlled by CableLabs. Use and understanding of this document may require access to such other documents. Designing, manufacturing, distributing, using, selling, or servicing products, or providing services, based on this document may require intellectual property licenses from third parties for technology referenced in this document.

Neither CableLabs nor any member company is responsible to any party for any liability of any nature whatsoever resulting from or arising out of use or reliance upon this document, or any document referenced herein. This document is furnished on an "AS IS" basis and neither CableLabs nor its members provides any representation or warranty, express or implied, regarding the accuracy, completeness, non-infringement, or fitness for a particular purpose of this document, or any document referenced herein.

© Copyright 2008-2009 Cable Television Laboratories, Inc.
All rights reserved.

Document Status Sheet

Document Control Number:	CL-SaFI-IAM-I01-090626			
Document Title:	Interactive Application Messaging Specification			
Revision History:	I01 – Released 6/26/09			
Date:	June 26, 2009			
Status:	Work in Progress	Draft	Issued	Closed
Distribution Restrictions:	Author Only	CL/Member	CL/Member/ Vendor	Public

Key to Document Status Codes:

Work in Progress	An incomplete document, designed to guide discussion and generate feedback that may include several alternative requirements for consideration.
Draft	A document in specification format considered largely complete, but lacking review by Members and vendors. Drafts are susceptible to substantial change during the review process.
Issued	A stable document, which has undergone rigorous member and vendor review and is suitable for product design and development, cross-vendor interoperability, and for certification testing.
Closed	A static document, reviewed, tested, validated, and closed to further engineering change requests to the specification through CableLabs.

Trademarks:

CableLabs[®], DOCSIS[®], EuroDOCSIS[™], eDOCSIS[™], M-CMTS[™], PacketCable[™], EuroPacketCable[™], PCMM[™], CableHome[®], CableOffice[™], OpenCable[™], OCAP[™], CableCARD[™], M-Card[™], DCAS[™], tru2way[™], and Cable PC[™] are trademarks of Cable Television Laboratories, Inc.

Contents

1	SCOPE	1
1.1	Introduction and Purpose	1
1.2	Requirements	1
2	REFERENCES	2
2.1	Normative References	2
2.2	Informative References.....	2
2.3	Reference Acquisition	2
3	TERMS AND DEFINITIONS	3
4	ABBREVIATIONS AND ACRONYMS	4
5	OVERVIEW	5
5.1	General Context	5
5.1.1	<i>Reference Architecture</i>	5
5.1.2	<i>Interface Description</i>	5
5.2	Transport Protocols.....	6
5.3	Data Model	6
5.4	Identifiers.....	6
5.5	Timing Model	7
6	INTERACTIVE APPLICATION MESSAGING REQUIREMENTS	8
6.1	Requirements on interoperable interactive applications	8
6.2	Requirements on compliant network systems.....	8
6.3	Data Model	8
6.3.1	<i>Application Message</i>	8
6.3.2	<i>Application Message Header</i>	8
6.3.3	<i>Application Message Block</i>	8
6.3.4	<i>Geographic Code</i>	9
6.3.5	<i>Application Report Block</i>	9
7	INTERACTIVE APPLICATION MESSAGING DATA MODEL SCHEMA	10
8	HTTP ENCODING OF DATA MODEL	11
APPENDIX I	XML ENCODING OF DATA MODEL (INFORMATIVE)	13

Figures

Figure 5-1 - Interactive Application Messaging Reference Architecture Diagram.....5
Figure 8-1 - Application Message Block11
Figure 8-2 - Application Report Block11
Figure 8-3 - Repeating Blocks12

1 SCOPE

1.1 Introduction and Purpose

The purpose of this document is to define requirements for an Interactive Application Messaging (IAM) platform. The IAM platform is composed of the content of messages generated by applications on a cable receiver and transported to a network component within an MSO system, a logical architecture for the generation and delivery of such messages, and any other related requirements necessary to implement the platform.

1.2 Requirements

Throughout this document, the words that are used to define the significance of particular requirements are capitalized. These words are:

"SHALL"	This word means that the item is an absolute requirement of this specification.
"SHALL NOT"	This phrase means that the item is an absolute prohibition of this specification.
"SHOULD"	This word means that there may exist valid reasons in particular circumstances to ignore this item, but the full implications should be understood and the case carefully weighed before choosing a different course.
"SHOULD NOT"	This phrase means that there may exist valid reasons in particular circumstances when the listed behavior is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behavior described with this label.
"MAY"	This word means that this item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because it enhances the product, for example; another vendor may omit the same item.

2 REFERENCES

2.1 Normative References

In order to claim compliance with this specification, it is necessary to conform to the following standards and other works as indicated, in addition to the other requirements of this specification. Notwithstanding, intellectual property rights may be required to use or implement such normative references.

- [SaFI COM XSD] CL-SaFI-COM-I01.xsd, June 26, 2009, Cable Television Laboratories, Inc.
[IAM XSD] CL-SaFI-IAM-I01.xsd, June 26, 2009, Cable Television Laboratories, Inc.
[RFC3986] IETF RFC 3986, Uniform Resource Identifier (URI): Generic Syntax, URN Namespace.

2.2 Informative References

This document uses the following informative references.

- [SMS] Service Measurement Summary Interface Specification, CL-SP-SaFI-SMS-I01-090626, June 26, 2009, Cable Television Laboratories, Inc.
[CIP] Campaign Information Package Specification, CL-SP-SaFI-CIP-I01-090626, June 26, 2009, Cable Television Laboratories, Inc.
[IAF] Interactive Application Fulfillment Summary Interface Specification, CL-SP-SaFI-IAF-I01-090626, June 26, 2009, Cable Television Laboratories, Inc.
[IAM EXMPL] CL-SaFI-IAM-I01-example1.xml, June 26, 2009, Cable Television Laboratories, Inc.

2.3 Reference Acquisition

- Cable Television Laboratories, Inc., 858 Coal Creek Circle, Louisville, CO 80027; Phone +1-303-661-9100; Fax +1-303-661-9199; <http://www.cablelabs.com>
- Internet Engineering Task Force (IETF) Secretariat, 46000 Center Oak Plaza, Sterling, VA 20166, Phone +1-571-434-3500, Fax +1-571-434-3535, <http://www.ietf.org>
- W3C, <http://www.w3.org>

3 TERMS AND DEFINITIONS

This specification uses the following terms:

Service Measurement Information about the reach and usage of a campaign.

4 ABBREVIATIONS AND ACRONYMS

This specification uses the following abbreviations:

AMB	Application Message Block
ARB	Application Report Block
CAAS	Common Advanced Advertising Systems
EPSID	Enhanced Program Sequence ID, a small integer identifying a unique Enhanced Package or Enhanced Package Element within a specific Programmed Event
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol over Secure Socket Layer
PEID	Programmed Event ID, a globally-unique identifier for a Programmed Event
SaFI	Stewardship and Fulfillment Interfaces. A collection of interfaces defined by CableLabs to support advanced services on multiple cable systems
SOAP	Simple Object Access Protocol; as of SOAP 1.2 it no longer represents an acronym
STB	Set-top Box
WSDL	Web Services Description Protocol

5 OVERVIEW

5.1 General Context

The IAM platform provides a critical interface between interoperable applications and MSO systems. Interoperable applications are applications that are distributed to more than one MSO. Messages instantiated by interoperable applications must therefore take the same form, regardless of the MSO system in which it is executing. This specification defines what form these messages may take and describes how MSO system components decode them.

Messages support two primary MSO-CAAS interfaces: the Service Measurement Summary [SMS] and Interactive Application Fulfillment [IAF] Interfaces. Service Measurement contains information about the execution of a campaign, including viewership of applications and individual overlays, while Application Fulfillment carries the results of voting/polling and RFI overlays. Both of these higher-level interfaces are supplied with specific application event and overlay data through the Interactive Application Messaging interface.

5.1.1 Reference Architecture

The following diagram illustrates a systems view of the IAM platform. This diagram is derived from work produced by the Advanced Advertising Interfaces team, and represents an advertising-centric view of the platform. The IAM platform will be generalized such that applications unrelated to advertising can use the same platform.

The IAM platform is represented by the blocks labeled "App Messaging Payload" and "App Messaging Post" in Figure 5-1.

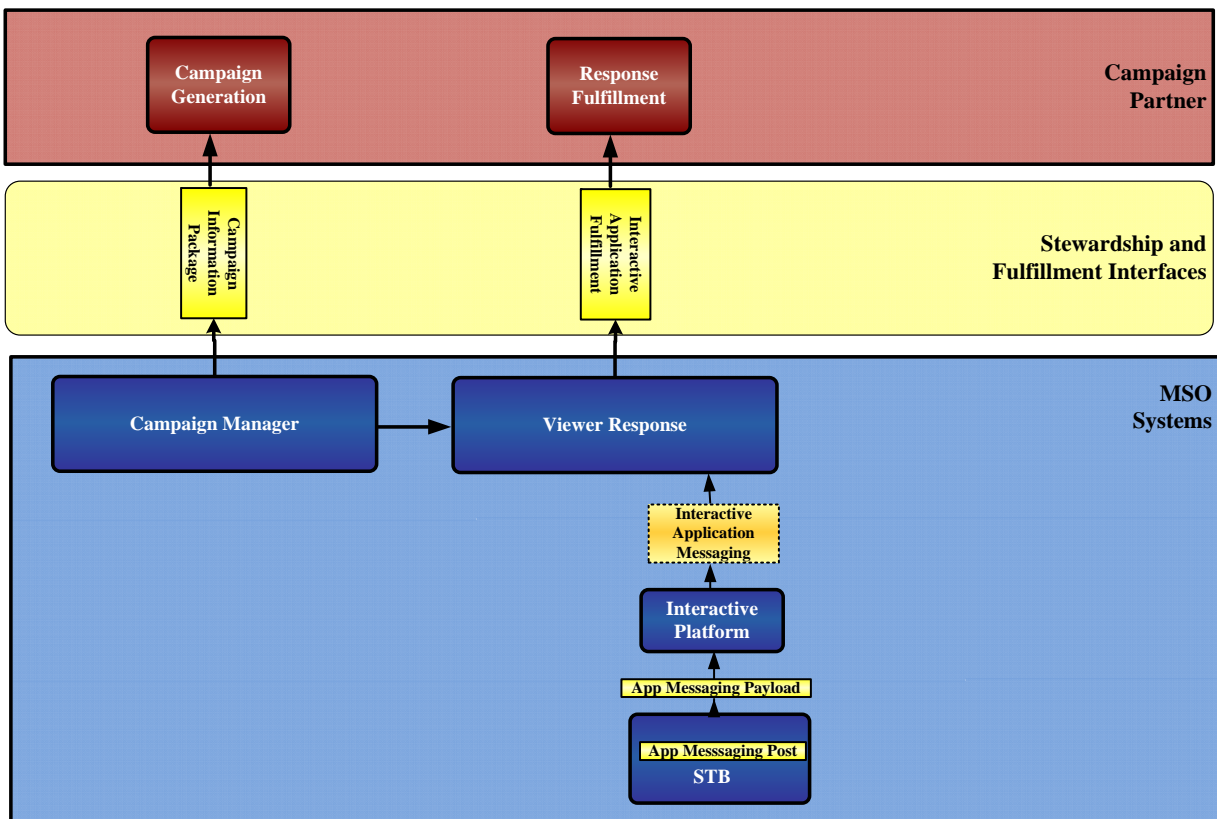


Figure 5-1 - Interactive Application Messaging Reference Architecture Diagram

5.1.2 Interface Description

The Interactive Application Messaging platform includes two components: the App Messaging Post and the App Messaging Payload indicated in Figure 5-1.

The messaging payload is formally defined by an XML schema and conveys information from an STB to MSO network components. The encoding of the payload may be defined by the MSO, using any protocol supported by the STB and other network components, but the data model nevertheless conforms to this specification, and the URL address used by the application must reference an MSO or CAAS component and will be the same irrespective of the MSO system within which the application executes. It is the responsibility of the MSO to ensure this URL resolves to the proper system address for its respective network.

The format of a message expressed as an HTTP POST is described herein. The specific API calls necessary to construct and initiate a message are not defined. Both ETV and OCAP applications MAY generate messages in this format.

5.2 Transport Protocols

This messaging specification may be supported by ETV, OCAP, and other application formats. For ETV, MSO ETV User Agents and associated MSO network components define the transport protocol used to transmit messages from an STB to MSO network components. On ETV implementations, return channel messaging is modeled on HTTP, but for performance reasons, an MSO MAY choose to implement another protocol. All OCAP-enabled networks support HTTP; therefore, OCAP platforms that make use of this specification SHALL use HTTP as the transport.

Other MSO network components MAY adhere to this specification; for instance, a proprietary or HTTP form of a message MAY be transformed into an XML representation of the data model.

5.3 Data Model

An XML schema is defined that formally describes the data that may be generated by interoperable interactive applications and transmitted to MSO system components. An HTTP encoding of the data model is described for use by applications.

Based upon the Schema defined in Section 7, MSO system components MAY easily transform the HTTP format into XML or some other useful format.

5.4 Identifiers

The data model contains several identifiers. These identifiers may be resolved to campaign elements described in the campaign description associated with the application, or otherwise used to disambiguate message elements. The identifiers are:

UnitID – This value may be retrieved by an application through a well-known API. It indicates a unique device on an MSO's network and can be used to calculate unique responses. The semantics of the value are defined by the MSO; it may represent a MAC address, or an obfuscated value set by the MSO.

ProgrammedEventID (PEID) – This value is accessed by an application through a well-known resource transmitted with the application. The value is assigned during the stewardship process and is included as an element within the Campaign Description associated with the application instance. The PEID uniquely identifies the programmed event associated with the application instance, e.g., the Gators/Razorbacks football game broadcast at 4 pm ET, October 17, 2008.

EnhancementPackageSequenceID (EPSID) – This value is accessed by an application through a well-known resource transmitted with the application. The value is assigned during the stewardship process and is included as an element within the Campaign Description associated with the application instance. The EPSID identifies a "package" within an application. A package is a group of User Interface "overlays" or other application components. For instance, a sequence of overlays presenting an atomic vote/response group might be identified with an EPSID distinct from other overlays presented by the application.

EventID – An EventID indicates a discreet component or event within the context of a "package" identified by an EPSID.

5.5 Timing Model

The data model supports a timing model for accurately communicating the real-time value at which certain events transpire.

Messages that contain instrumentation information *MAY* include both the display time and the termination time of the overlay. Termination may occur when a viewer selects the "exit" key, chooses a response from an interactive overlay, or the overlay times out. The termination time allows calculation of "dwell time" on a given UI component. Each message contains an absolute time value in the Timestamp element of the ApplicationMessageHeaderGroup element. Subsequent time values are relative to the Timestamp.

Messages that contain lifecycle and response information include a single Timestamp at which the event occurred.

6 INTERACTIVE APPLICATION MESSAGING REQUIREMENTS

This section defines requirements for the Interactive Application Messaging platform.

6.1 Requirements on interoperable interactive applications

Interoperable interactive applications SHALL comply with this specification by instantiating HTTP commands that conform to the data model and encoding rules defined herein. The data model is formally defined in Section 7, and encoding rules for HTTP transport are defined in Section 8.

6.2 Requirements on compliant network systems

Since interoperable interactive applications execute unchanged on all MSO systems, URLs utilized by interoperable interactive applications to post messages conformant to this specification SHALL be supported on all MSOs systems. Compliant URL forms are defined in Section 8.

6.3 Data Model

An XML schema is defined in Appendix I to fully describe the data elements that will be transmitted between client device and the MSO. This schema is provided as an Informative reference.

6.3.1 Application Message

The Application Message is the root element in the Interactive Application Message data model. It provides a container for all data being transferred.

It contains the following data units.

- Application Message Header – Contains common addressing elements to all messages.
- Application Message Block – Repeating wrapper element for individual messages.

6.3.2 Application Message Header

The Application Message Header contains common data elements to all Application Message Blocks being reported within a single Application Message.

It contains the following data units.

- Minimum Schema Version – Lowest compatible schema/protocol version to be used with a message.
- Unit ID – Unique identifier for client device.
- Geographic Code – One of either zip code or system code used to report location information on client/client device.
- PEID – Programming Event Identifier used to uniquely identify a scheduled program. TBD – can this field be made smaller, using registered namespace.
- Timestamp – Reference timestamp used by all Application Report Blocks. Each individual Event will report its time as an offset of this timestamp.

6.3.3 Application Message Block

The Application Message Block is a repeating container used to encapsulate messages associated with a common EPSID. It can contain multiple sub-blocks called Application Report Blocks.

It contains the following data units.

- EPSID – Enhanced Program Sequence ID. This is a unique value within the scope of an associated PEID.
- Application Report Block – Contains data relating to a specific Event.

6.3.4 Geographic Code

The Geographic Code contains client device location information in the form of either a zip code or a system code.

It contains the following data units.

- System Code – A syscode is a geographic identifier allocated by agreement of MSOs.
- Zip Code – Zip code for client and/or client device

6.3.5 Application Report Block

Lowest level container used to report on individual Events.

It contains the following data units.

- Event ID – The ID used to uniquely identify a reportable Event within an EPSID. Events are defined by each application.
- User Input – Specific client response being reported.
- Parameters – Optional data field used to transmit additional information. Contents of this field and parsing rules defined by each application.
- Relative Time Attribute – The Relative Time Attribute is a common element used to represent time as an offset and optionally length as a duration of a common time. In this case, the common time is carried in the Application Message Header as the Timestamp element.

7 INTERACTIVE APPLICATION MESSAGING DATA MODEL SCHEMA

The formal data definition is found in [IAM XSD].

8 HTTP ENCODING OF DATA MODEL

Applications may generate messages that adhere to this specification by instantiating HTTP POST commands that encode elements as key/value pairs. The HTTP POST SHALL be sent to the URI defined by `ara.cablelabs.com/am`. The encoding scheme defines four keys, each with its own unique value string. The first two keys are contractions of their XML counter part and are used to carry a single value.

Key	Value (Full Name)
ms	MinSchemaVersion
uid	UnitID

The next two keys are more complex in that their values are actually a series of values. These values use positional references marked by delimiters to map back to the proper key. The table below provides the appropriate mapping and delimiters. Note that the Timestamp value is a Unix Time (seconds elapsed since 1970-01-01 24:00:00 UTC) representation of UTC. The duration timestamps of Offset and Interval are offsets in seconds from Timestamp.

Key	Full Name	Value
am	Application Message	GeoID-PEID-Timestamp
amb	Application Message Block	EPSID-offset-interval-EventID-UserInput-Parameters

All individual data elements are delimited by a single dash '-' character. It is important to note that the AMB block can contain repeating value blocks. These repeating value blocks can take two forms. First is a repeating Application Report Block, which is denoted by a single underscore character '_'. Second is a repeating Application Message Block, which is denoted by a double underscore '__'. For detailed illustration, see example and diagrams below.



Figure 8-1 - Application Message Block

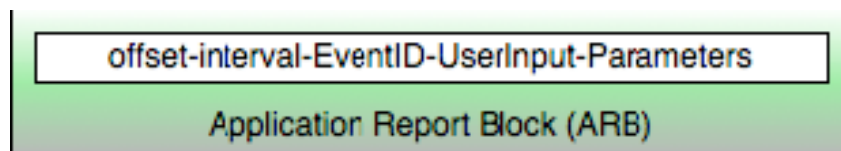


Figure 8-2 - Application Report Block

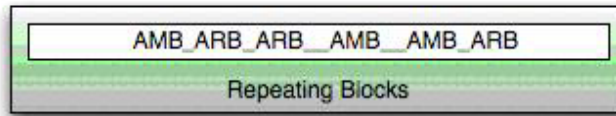


Figure 8–3 - Repeating Blocks

It is important to note that the '-' and '_' are special characters used in the message encoding. Any application leveraging this interface should take special care to ensure these characters either are not present in the data fields being transmitted over the interface or that they are properly encoded if they are being used in anyway except as field and message delimiters. The proper encoding should follow [RFC3986] encoding schema, where the character '-' is encoded as '%2d' and '_' is encoded as '%5f'.

An example message is provided using the XML sample included in Appendix I. **Note:** HTTP header information is included in the example but may be dropped depending on actual transport used between STB and headend network)

```
POST /am HTTP/1.1
Host: ara.cablelabs.com
User-Agent: MSO-UserAgent/1.0
Content-Length: 114
Content-Type: application/x-www-form-urlencoded
ms=1&
uid=0123456789ab&
am=80027-48taPS4lQVu7q6RjEA40kg== -1207276200&
amb=1-30--1-A-_60--2-Y-RED,XL__2-30-5—A-&
```

Appendix I XML Encoding of Data Model (Informative)

Examples of IAM data expressions can be found in [IAM EXMPL].

