



Digital Video Ad Serving Template (VAST) Version 2.0

Released November 2009

This document has been developed by the IAB Digital Video Committee.

Document Version	2.0
Author	Ari Paparo - Google
Date	11/11/2009
Summary of Changes from Previous Version	<ul style="list-style-type: none">• New root node "VAST"• Version number for root node and AdSystem• Refactoring of resourceTypes for Companion, NonLinear and Wrapper• Support for interactive MediaFiles• Move apiFramework from element to attribute• Alignment of tracking event names with IAB video reporting standards• Clarification of documentation• Introduction of Creative element to delineate multi-part creatives within a single Ad• Tracking of creative elements and companions• Change VASTAdTagURL to VASTAdTagURI

About the IAB Digital Video Committee:

The Digital Video Committee of the IAB is comprised of over 145 member companies actively engaged in the creation and execution of digital video advertising. One of the goals of the committee is to implement a comprehensive set of guidelines, measurement, and creative options for interactive video advertising. The committee works to educate marketers and agencies on the strength of digital video as a marketing vehicle. A full list of Committee member companies can be found at: http://www.iab.net/member_center/35088?iabid=a033000000s0p4AAA

This document can be found on the IAB website at: <http://www.iab.net/vast>

IAB Contact Information:

Jeremy Fain
Vice President of Industry Services, IAB
212-380-4724
jeremy@iab.net

Table of Contents

SUMMARY AND SCOPE	4
DEFINITIONS	5
ADDITIONAL RECOMMENDATIONS	6
RATIONALE	6
METHODS OF AD SERVING	7
AD REQUEST	7
IMPRESSION TRACKING	8
<i>Wrapper Ads</i>	<i>8</i>
VAST AD RESPONSE	9
CLICK TRACKING	15
USING EXTENSIONS	15
APPENDIX A: AD TAG FORMAT RECOMMENDATION	16

Summary and Scope

The scope of this project is to develop and publish a standard XML-based ad response for in-stream video as well as an XML Schema Definition (“XSD”) for developers. This Video Ad Serving Template (“VAST”) is meant to accommodate the majority of current practices within the online advertising business.

This document is designed for any on-demand video player where the ad response is parsed prior to play. For example, use of this standard would be appropriate within an Adobe Flash player if the ad response was requested and parsed in ActionScript but would not be appropriate if contained directly within a SMIL playlist directly sent to the player. It may be possible to use this XML format for applications other than on-demand viewing such as live video streaming, downloadable video players, set-top boxes, etc, but those applications are explicitly beyond the scope of the current effort.

The goal of this specification is to be compatible with any video player framework that is scriptable. It will be up to each Secondary Ad Server to develop its own implementation of the standard and it will be up to each publisher or vendor to implement the standard in their Primary Ad Server and their Video Players.

It is assumed in this document that Video Players will attempt to align impression tracking with the [IAB Digital Video Impression Guidelines](#).

This document is intended to support both Linear Video Ads (such as “pre-rolls”), Non-linear Video Ads (such as “overlays”) and Companion ads as defined in the [IAB Digital Video Ad Format Guidelines](#). Many Non-linear ads include complex interaction with the underlying Video Player, and thus the VAST standard may not be entirely sufficient for implementing such ads across ad servers at this time. It is also important to note that VAST does not specify the positioning or timing of the ads within a video player; it is left to the video player’s itself to determine this since the player is the entity with the understanding of the context in which the ads will appear.

Note: The current version of VAST is 2.0. VAST 1.0 is deprecated and it is recommended that all participants in the industry upgrade their implementations as soon as feasible.

Definitions

Companion Ad: Commonly text, display ads, rich media, or skins that wrap around the video experience. These ads come in a number of sizes and shapes and typically run alongside or surrounding the video player.

InLine Ad: VAST document that includes all the elements necessary to display the visual experience of the ad.

Linear Video Ad: The ad is presented before, in the middle of, or after the video content is consumed by the user, in very much the same way a TV commercial can play before, during or after the chosen program.

Non-linear Video Ad: The ad runs concurrently with the video content so the users see the ad while viewing the content. Non-linear video ads can be delivered as text, graphical ads, or as video overlays.

Primary Ad Server: First ad serving system called by the Video Player or other framework. It is assumed that in most cases a publisher will make all initial ad requests through their Primary Ad Server (whether homegrown or third party), then redirect to other ad servers as needed.

Secondary Ad Server: Ad server used by an ad network or by the buyer of ads to serve creative, track results and optimize creatives.

VAST (Video Ad Serving Template): XML document format describing an ad to be displayed in, over, or around a Video Player or a Wrapper pointing to a downstream VAST document to be requested.

Video Player: Environment in which in-stream video content is played. The Video Player may be built by the publisher or provided by a vendor.

Wrapper Ad: VAST document that points to another VAST document from a different server.

Additional Recommendations

In addition to the XML ad response standard detailed below, two optional recommendations have been identified to assist in the successful delivery of video ads:

1. A set of key-value pairs that can be added to the ad tag request in order to reduce errors resulting from incorrect ad responses. For example, if a publisher only wishes to receive Windows Media-formatted video ads or wants to receive ads of a certain duration, an ad serving vendor may wish to define the ad tag in such a way as to only provide ads with those specifications. Because the syntax of ad tags differs significantly between ad servers, this is an optional, recommended aspect of the project and not a formal requirement. This recommendation is included in this document as Appendix A.
2. A method of communication between the ad itself and the Video player in which it is displayed. This communication is important because both Linear and Non-Linear video ads can be interactive, and such user interaction will generally affect the activity of the Video player. For example, when a user clicks on an overlay ad the user may be presented with more information about the advertiser, while the video content is expected to pause. Currently, each publisher and technology vendor has implemented this type of communication in a non-standardized way, resulting in additional work for all parties when bringing campaigns live. This recommendation, called VPAID, was published by the IAB in a separate document available at <http://www.iab.net/vpaid>.

Rationale

The development of a standard method for in-stream ad serving is a necessary precursor to the acceptance of third party serving by publishers. The lack of third party ad serving within Video Players currently is causing two types of inefficiency: Publishers are not able to readily use ad networks to sell unsold video ad inventory, and buyers are not able to use their current reporting and optimization tools for in-stream ads.

The adoption of the VAST standard is expected to bring immediate benefits to all parties in the video ecosystem:

Party	Expected Benefits
Publishers	<ul style="list-style-type: none"> • Increase yield by using ad networks to sell unsold inventory • Reduce friction with buyers by allowing third-party ad tags within video players
Ad Networks/Exchanges	<ul style="list-style-type: none"> • More easily onboard new publishers without need for technical integration • Reduce friction with buyers by allowing third-party ad tags
Agencies/Advertisers	<ul style="list-style-type: none"> • Utilize existing investment in ad serving, reporting, optimization tools
Technology Vendors	<ul style="list-style-type: none"> • Build for a single technical standard instead of multiple proprietary standards

Methods of Ad Serving

It is anticipated that publishers will allow one of two methods for in-stream ad serving:

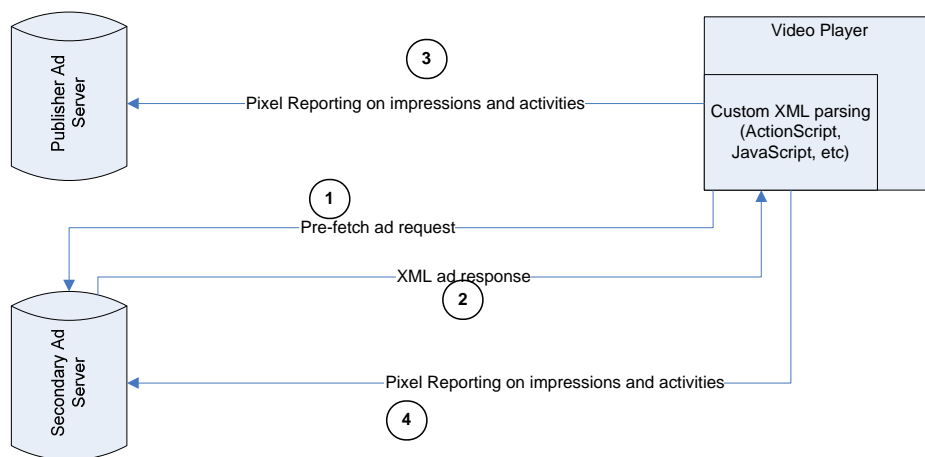
- Hard-coded video and tracking URIs into the player or Primary Ad Server
- Dynamically redirecting from their ad server to one or more other ad servers

This project only focuses on the dynamically served method. The hard coded method can easily be customized by traffickers based on individual publisher requirements and does not require standardization.

Ad Request

The anticipated sequence of ad requests from the Video Player in the user's browser is shown in the simplified diagram below:

Generic Description of Third Party Ad Serving for In-Stream



1. A publisher's system (e.g. ad server, client-side video player, or other mechanism) makes an ad request to the Secondary Ad Server. The ad tag to the Secondary Ad Server may be present within a playlist, hard-coded into the player environment, returned from the ad server as a redirect, or otherwise derived. If retrieved from a Primary Ad Server, the request to the Secondary Ad Server will generally not immediately track an impression since the IAB guideline for video impression requires tracking post-buffering.
2. The Secondary Ad Server can either respond with a Wrapper XML pointing to another Secondary Ad Server or respond with a VAST-formatted XML document describing the ad to be shown. Both the Wrapper XML and the VAST XML are described later in this document. It will be up to the Publisher to determine business rules about the maximum number of jumps between Secondary Ad Servers that are allowed in order to optimize user experience. With each chained Secondary Ad Server the overall size of data transferred and latency will increase, so many publishers may wish to restrict the number of redirects to 2 or fewer. Optionally, the Video Player can validate business rules around the XML response and choose whether or not to display the returned ad. For example, if the player only accepts 15 second video slots and the Secondary Ad Server returns a 30 second spot, it can be thrown away and an error report generated. As discussed later in this document, ad servers may choose to adopt tagging conventions in order to reduce the likelihood of such errors.
3. Based on the XML response, a set of tracking URIs will be requested by the Video Player for reporting purposes. Different ad servers and Video Players may support different sets of metrics and therefore different combinations of reporting.
4. Both the Primary Ad Server and all Secondary Ad Servers must record impressions. See the Impression Tracking section of this document for more on this subject.

Impression Tracking

In order to support IAB compliant impression tracking (see: [IAB Digital Video Ad Measurement Guidelines](#)) the ad requests for video ads need to delay the impression tracking until after the video has completed buffering. However, the standard method of redirecting between ad servers using HTTP 302 responses does not allow for each ad server to communicate to the video player the proper impression URI to request.

For example, when the Video Player makes a request from a Primary Ad Server, a 302 is returned to the ad tag of the Secondary Ad Server. The Secondary Ad Server responds with XML that includes a URI for impression tracking. The Video Player requests this URI post-buffering so the Secondary Ad Server records an IAB compliant video ad impression. But how does the Primary Ad Server know when to record an impression?

There are three possible methods for overcoming this limitation:

1. Wrap each ad server redirect response in additional XML including relevant tracking URIs ("XML Wrapper Method").
2. Use tag-based syntax to include impression and click tracking in the request in much the same way as rich media works today ("Rich Media Method").
3. Require that all Secondary Ad Servers include fields in their trafficking interfaces for the entry of a number of reporting URIs from publishers and networks ("Multiple URI Method").

Although any of the three methods are acceptable, the VAST specification includes direct support of the XML Wrapper Method through the <wrapper> element. The XML Wrapper method is the preferred method due to its power and extensibility. It will be up to each Publisher to determine what policies are put in place to avoid excessive redirecting or large file size in XML responses.

Wrapper Ads

The flow of the XML Wrapper method is very similar to the simplified diagram shown above with some important differences:

1. The Primary Ad Server or content management system is the first to receive a request.
2. The Primary Ad Server responds with a VAST XML document with a Wrapper ad including the impression URI, other tracking URIs, and click tracking URIs to be requested by the video player, along with the ad tag to the Secondary Ad Server from which the ad will be served.
3. The Video Player requests the ad tag from the Secondary Ad Server.
4. The Secondary Ad Server responds with the VAST document containing an InLine Ad, or alternatively can respond with a VAST document containing a second Wrapper Ad. Potentially there could be a third or fourth set of URIs as in the case where an ad network redirects to an agency ad server. This implies that the Video Player scripts need to keep track of multiple tracking URIs per event, per ad unit.
5. The Secondary Ad Server URIs are requested when certain events take place.
6. The Primary Ad Server URIs are requested when certain events take place. These URIs are determined by the first Wrapper Ad returned.

The Wrapper Ad includes a subset of descriptive and tracking elements from the InLine Ad, along with an extra element for the ad tag of the downstream ad server.

VAST Ad Response

Defining the standardized XML response is the point of this project. A sample XML document and an XSD schema are available on the IAB website.

Some clarification and description about the XML response:

1. The top-level element in the VAST XML definition is the <Ad>. An "Ad" contains some combination of video, companions, and non-linear units for a single advertiser.
2. A single VAST response may include multiple Ads from multiple advertisers. It will be up to the Video Player to determine the order, timing, placement, etc for the multiple ads. However, the player should generally respect the sequential order of the Ad elements within a VAST response.
3. The VAST response does not contain information on the placement or timing of the elements. It is up to the Video Player to determine the optimal inclusion points of the ads.
4. An Ad can either be of type <Inline>, meaning it contains all the elements necessary to display the visual experience, or of type <Wrapper>, which points to a downstream VAST document that must be requested from another server.
5. The XML response may indicate that no ads of any type are available. This would be indicated by the absence of any Ads.
6. An Ad may include one or more <Creative> elements that are part of a single execution within a <Creatives> container. For example, an Ad may include a linear video element with a set of companion banners; this would be reflected by two Creative elements.
7. Three types of Creatives are supported and indicated by the elements <Linear>, <NonLinearAds>, and <CompanionAds>. The Creative element takes an optional "sequence" attribute that indicates the suggested order in which the Creatives should be displayed. If two Creative elements are intended to be shown at the same time they should share the same sequence number.
8. An Ad includes a single required <Impression> element that should be requested as soon as any creative element within the Ad is displayed.
9. When there are multiple Creative elements it may be desired to track the display of each element separately. An additional tracking element is included creative elements and should be requested as soon as the respective element is displayed. For example, suppose an Ad includes a linear video and a non-linear overlay and the video is played first. When the video is played the player should request both the impression URI and the creativeView tracking URI for the video. The player should then request the creativeView tracking URI for the overlay as soon as that element becomes visible.
10. The Linear element defines a video, image, or interactive element that is to be played within the video view area.
11. The CompanionAds element may include a set of one or more companion banners of different pixel dimensions or technologies. If companions are intended to be shown at different times within the playback then multiple Creatives should be used, optionally with distinct sequence hints.
12. The NonLinearAds element may include one or more non-linear elements representing a single creative concept. If the intention is to show different non-linears at different points in the playback then multiple Creatives should be used.
13. Publishers may choose different levels of support for VAST within their Video Players. For example, a publisher may choose to only allow tracking by a Secondary Ad Server, in which case only the tracking section of the XML would be relevant.
14. For latency or infrastructure reasons some publishers may not wish to allow Secondary Ad Servers to serve the video files themselves. The XML document may be used in this scenario with media file URIs pointing to server locations hosted by the publisher infrastructure.
15. An optional error URI is provided so the various ad servers can be informed if the ad did not play for any reason. Note, most ad servers do not currently support notification functionality of this type, but it is included as a future capability.
16. All ad servers need to provide, at a minimum, a single impression tracking URI. Multiple Impression elements may appear in sequence within their respective sections. For example, an Inline Ad may contain two sequential Impression elements, one for the ad server and the other for a 3rd party affiliated with the campaign.

17. The XML document supports extensive tracking if available from the Video Player. Tracking is based on the [IAB Digital Video In-Stream Ad Metrics Definitions](#) document.
18. The Wrapper Ad must include the URI of a Secondary Ad Server's VAST response.
19. The Wrapper Ad may include separately served Companions or NonLinear Creatives from either the primary or Secondary Ad Servers. For example, instead of putting all assets into a single VAST InLine response, a Secondary Ad Server might provide an ad tag for an InLine response for the linear video ad, and a separate ad tag pointing to a standard banner ad for a Companion.
20. The Wrapper Ad may include any number of Tracking URIs to allow Companions to be served from an InLine response but tracked separately from the Impression. It may also include Tracking elements for separately tracking Linear or NonLinear views or events. The server providing the Wrapper Ad response may not know exactly which creative elements are to be delivered downline in InLine ads; in this case the Wrapper should include placeholders for the maximum set of Creatives that could be played within the Video Player.
21. A single click through on the ad is expected, but multiple named click tracking URIs (CustomClick) can be provided to allow per-publisher customization. In addition, separate URIs can be provided for tracking the click (ClickTracking) and for the destination page to open upon a click-through (ClickThrough).
22. Any number of URIs can be provided for a single video ad within the MediaFiles section of the response, but it is assumed that all versions of the video represent the same creative unit with the same duration, Ad-ID (ISCI code), etc. Bandwidth is indicated per media file using the bitrate attribute. It will be up to Video Players to determine which media files with which bitrates are appropriate for their users. Image ads or interactive ads can be included in the MediaFiles section with appropriate Mime types. If multiple linear concepts are desired (e.g. pre-roll and post-roll) then multiple Creatives should be used.
23. Ad elements can specify an AdParameters element to have specified parameters passed to them from the player. For example, a non-linear ad may require that tracking information be passed at run-time.
24. The apiFramework element indicates the method in which the video player can communicate to the various ad assets. Common values for this element are "[VPAID](#)" and "clickTag". For ad hoc solutions other values may be used as needed.
25. Extensions elements allow for customization or for ad server specific features (e.g. geo data, unique identifiers).
26. It is preferable that the ad server delivering the Wrapper Ad busts the cache on the AdTagURI parameter by including a random number in the URI at the time of delivery.
27. Most Secondary Ad Servers support the inclusion of a parameter in their ad tags (generally "click=") that will redirect all clicks through the Primary Ad Server to enable counting. In the case of VAST, the value of this parameter (which is set by the Primary Ad Server) should be placed in the VideoClicks-ClickTracking element of the Wrapper Ad response and parsed into the ad tags contained within the InLine response.
28. All URIs should be wrapped in CDATA blocks

XML summary for VAST ad server response:

Element	Attributes	Values	Required	Notes
VAST		Root node	Yes	
	version	String	Yes	Current version is 2.0
Ad	id	String	Yes	Top-level element, wraps each ad in the response
InLine	None	None	No	Second-level element surrounding complete ad data for a single ad
AdSystem	None	String	Yes	Indicates source ad server
	version	String	No	Internal version used by ad system
AdTitle	None	String	Yes	Common name of ad
Description	None	String	No	Longer description of ad
Survey	None	URI	No	URI of request to survey vendor

Element	Attributes	Values	Required	Notes
Error	None	URI	No	URI to request if ad does not play due to error
Impression	None	URI	Yes	URI to track impression
	id	String	No	Optional identifier
Creatives	None	None	Yes	Container for one or more Creative elements
Creative				Wraps each creative element
	id	String	No	Optional identifier
	sequence	Integer	No	The preferred order in which multiple Creatives should be displayed
	AdID	String	No	Ad-ID for the creative (formerly ISCI)
Linear			No	
Duration	None	Time	Yes	Duration, in standard time format hh:mm:ss
TrackingEvents			No	
Tracking		URI	No	URI to track various events during playback
	event	creativeView, start, midpoint, firstQuartile, thirdQuartile, complete, mute, unmute, pause, rewind, resume, fullscreen, expand, collapse, acceptInvitation, close	Yes	The name of the event to track for the Linear element. The creativeView should always be requested when present.
AdParameters		String	No	Data to be passed into the video ad.
VideoClicks				
ClickThrough	None	URI	No	URI to open as destination page when user clicks on the video
ClickTracking		URI	No	URI to request for tracking purposes when user clicks on the video
CustomClick		URI	No	URIs to request on custom events such as hotspotted video
	id	String	No	Optional identifier
MediaFiles			Yes	
MediaFile		URI	Yes	Location of linear file
	id	String	No	Optional identifier
	delivery	streaming, progressive	Yes	Method of delivery of ad
	type	String	Yes	MIME type. Popular MIME types include, but are not limited to "video/x-ms-wmv" for Windows Media, and

Element	Attributes	Values	Required	Notes
				"video/x-flv" for Flash Video.
	bitrate	Integer	No	Bitrate of encoded video in Kbps
	width	Integer	Yes	Pixel dimensions of video
	height	Integer	Yes	Pixel dimensions of video
	scalable	Boolean	No	Whether it is acceptable to scale the image.
	maintainAspectRatio	Boolean	No	Whether the ad must have its aspect ratio maintained when scaled
	apiFramework	String	No	The apiFramework defines the method to use for communication if the MediaFile is interactive.
CompanionAds				
Companion			No	Any number of companions in any desired pixel dimensions.
	id	String	No	Optional identifier
	width	Integer	Yes	Pixel dimensions of companion
	height	Integer	Yes	Pixel dimensions of companion
	expandedWidth	Integer	No	Pixel dimensions of expanding companion ad when in expanded state
	expandedHeight	Integer	No	Pixel dimensions of expanding companion ad when in expanded state
	apiFramework	String	No	The apiFramework defines the method to use for communication with the companion
StaticResource		URI	No	URI to a static file, such as an image or SWF file
	creativeType	String	Yes	Mime type of static resource
IFrameResource	None	URI	No	URI source for an IFrame to display the companion element
HTMLResource	None	CDATA	No	HTML to display the companion element
TrackingEvents			No	
Tracking		URI	No	URI to track view of Companion
	event	creativeView	Yes	The creativeView should always be requested when present. For Companions creativeView is the only supported event.
CompanionClickThrough	None	URI	No	URI to open as destination page when user clicks on the companion

Element	Attributes	Values	Required	Notes
AltText	None	String	No	Alt text to be displayed when companion is rendered in HTML environment.
AdParameters		String	No	Data to be passed into the companion ads
NonLinearAds				
NonLinear			No	Any number of non-linear assets in any desired pixel dimensions.
	id	String	No	Optional identifier
	width	Integer	Yes	Pixel dimensions of companion
	height	Integer	Yes	Pixel dimensions of companion
	expandedWidth	Integer	No	Pixel dimensions of expanding nonlinear ad when in expanded state
	expandedHeight	Integer	No	Pixel dimensions of expanding nonlinear ad when in expanded state
	scalable	Boolean	No	Whether it is acceptable to scale the image
	maintainAspectRatio	Boolean	No	Whether the ad must have its aspect ratio maintained when scaled
	minSuggestedDuration	Time	No	Suggested duration to display non-linear ad, typically for animation to complete. Expressed in standard time format hh:mm:ss
	apiFramework	String	No	The apiFramework defines the method to use for communication with the non-linear element
StaticResource		URI	No	URI to a static file, such as an image or SWF file
	creativeType	String	Yes	Mime type of static resource
IFrameResource	None	URI	No	URI source for an IFrame to display the non-linear element
HTMLResource	None	CDATA	No	HTML to display the non-linear element
TrackingEvents			No	
Tracking		URI	No	URI to track various events during playback
	event	creativeView, start, midpoint, firstQuartile, thirdQuartile, complete, mute, unmute, pause,	Yes	The name of the event to track. The creativeView should always be requested when present.

Element	Attributes	Values	Required	Notes
		rewind, resume, fullscreen, expand, collapse, acceptInvitation, close		
NonLinearClickThrough	None	URI	No	URI to open as destination page when user clicks on the non-linear ad unit
AdParameters		String	No	Data to be passed into the video ad.
Extensions				
Extension	type	Any	No	Any valid XML may be included in the Extensions node. See section below for suggested naming conventions of types.
Wrapper	None	None	No	Second-level element surrounding wrapper ad pointing to Secondary ad server.
AdSystem	Same as above in Inline section			
VASTAdTagURI	None	URI	Yes	URI of ad tag of downstream Secondary Ad Server
Error	Same as above in Inline section			
Impression	Same as above in Inline section			
Creatives	Same as above in Inline section			
Creative	Same as above in Inline section			
Linear	Same as above in Inline section			
TrackingEvents	Same as above in Inline section			
VideoClicks	Same as above in Inline section			
CompanionAds	Same as above in Inline section			
Companion	Same as above in Inline section			
NonLinearAds	Same as above in Inline section			
TrackingEvents	Same as above in Inline section			
NonLinear	Same as above in Inline section			
Extensions	Same as above in Inline section			

Click Tracking

The video asset only allows a single primary click through URI with optional additional custom click-throughs. As such, each Secondary Ad Server can provide its own URI for click tracking and these URIs can be requested by the player as any other tracking URI.

Companion ads and Non-linear ads have an optional click through URI parameter. Whether this is required varies depending on the companion ad type. For example, an image companion will generally require a click through since it will be written on the page by the publisher. A script based HTML companion will generally include its own click-through when rendered on the page. It is important that Video Players implementing support for VAST account for these differences in order to avoid accidentally over-riding a click or preventing an element from being clickable.

In standard ad serving publishers track clicks by including a redirect URI as an extra parameter within the Secondary Ad Server's ad tag. However, since the companion ad tag in the video scenario is within the XML document this inclusion must be dynamic at the time of parsing. In order for the video player to know where within the companion ad tag to include the click string it is recommend that a token [CLICK] be included in the URIs.

Using Extensions

The VAST response allows any valid XML within the extensions element. Use of extensions will necessarily require offline coordination between VAST sender and VAST receiver. In order to simplify expected uses we suggest certain naming conventions for the type attribute of the extension element:

Extension element type	Usage
AdServer	Any information specific to the VAST ad server
CustomTracking	Custom tracking elements
Value	Data about the economic value or relative priority of the ad

Appendix A: Ad Tag Format Recommendation

In order to minimize the size of the XML response from Ad Servers and to reduce the possibility of video ads being returned that do not meet the publisher specs, it is recommended that Ad Servers implement certain pre-defined parameters in their ad tags. While the syntax of Ad Server tags differs between vendors and technologies, using common values within these tags will enable Primary Ad Servers and Video Players to simplify and automate executions.

It is preferable for Ad Servers to return XML that only includes information relevant to the request, however, there is no requirement that extraneous information be stripped out. For example, if a Video Player requests a Flash video, the response may include nodes for both a Flash video and a Windows Media video as well. Some Secondary Ad Servers may be unable to respond dynamically to these parameters and whether this is acceptable to Publishers will be a matter of business negotiation.

The recommended parameters are shown in the table below.

- Parameters are abbreviated to minimize URI length
- Parameters are case-sensitive.
- Multiple comma-separated values may be assigned to a parameter (e.g. VMaxd=15,30). Multiple values should be considered as an OR request, meaning that either value is acceptable.

Parameter	Acceptable Values	Notes
VMaxd	Any integer value	Maximum duration of video accepted, in seconds.
VPI	WMV, FLV, RA	Player frameworks accepted. These values do not reflect specific video file extensions, but rather are abbreviations for the three currently supported player types. As the VAST document is extended to other player types additional abbreviations may be added.
VHt	Any integer value	Expected height of video in pixels
VWd	Any integer value	Expected width of video in pixels
VBw	Any integer value	Maximum bandwidth of video requested in bits per second.
Vstrm	0 or 1	0 for progressive, 1 for streaming

Examples using a fictional Ad Server are shown below. The actual syntax of these requests is likely to differ between Ad Servers.

Sample Ad Tag	Explanation
<code>http://ad.server.com/site/content?random=1234</code>	Base ad tag with random number inserted. This ad tag will return any video ad within the Ad Server without regard to what the Video Player is expecting.
<code>http://ad.server.com/site/content?Vmaxd=30;random=1234</code>	The video duration is specified as 30 seconds and the Video Player expects the Ad Server to return an XML document specifying a 30-second spot. Since the player and bandwidth are not specified, the Secondary Ad Server can respond with ads meeting any or all such criteria.
<code>http://ad.server.com/site/content?VMaxd=30;VPI=WMP,FLV;random=1234</code>	In addition to requesting a 30-second spot, the Video Player indicates that it wants either a Windows Media Player-compatible video XML or a Flash-compatible video XML.