AdCel API

- Introduction
- Ad Requests
- Request Parameters for Banners & Interstitials
    - Required parameters
    - Optional parameters
    - Adunit and size parameter
    - Interstitials
- Video Tag Integration
    - Required parameters
    - Optional Parameters
    - Additional Video Parameters
    - Creative Capability
    - User Data
    - Location Encoding
- Server-side Requests
- Ad Responses
    - No Ad Available
    - Sample Ad Response With Tracking Image
    - Sample Raw HTML Ad Response
    - Sample Ad Response Without Tracking Image
- Ad Display
    - Mobile Websites
    - Mobile Client Applications
- Appendices
    - Appendix A: Content Categories
    - Appendix B: Sample Location Encoding
    - Appendix C: VAST event tracking

Unknown macro: 'hideelements-macro'

Introduction

The Verve AdCel™ advertising API enables mobile websites and client applications to serve advertising to mobile users. AdCel allows ad networks to be dynamically provisioned, prioritized, and turned on or off without affecting mobile site or client application integrations.

AdCel's interface is a simple HTTP GET to request an advertisement, which returns a response containing advertisement delivery information.

Ad Requests

A basic mobile display advertisement request is an HTTP GET request. There are several different request methods that return different styles of ad responses. Choose whichever method is most convenient for you.

1. banner - an API call that returns adcel proprietary XML, which provides information about an ad, including the banner image URL, image alt text, the click through URL, and possibly ad text. Also can return "raw response" from an ad network instead. Eg.

```
http://adcel.vrvm.com/banner?p=ptnr&b=sampletag&c=999&ip=132.42.45.2&
lat=40.753800&long=-73.983354
```

2. adtag.js - returns javascript which is will be embedded into a page where the adTag was placed. It uses "document.write" to place the information into the HTML page. This is not meant to be called by a client directly, but rather by our own generated adTag code that the client has embedded into their page. The adTag code to be embedded by the client is generated in the dashboard. Eg.

```
http://adcel.vrvm.com/adtag.js?p=ptnr&b=sampletag&c=999&lat=40.753800
&long=-73.983354
```

3. htmlad - an API call that is nearly identical to "banner". The difference being that instead of returning AdCel proprietary XML, the actual HTML that will display the ad is returned. This means that the client calling to the API can simply copy the results into their ad space as is, rather than having to parse our response, and build the ad themselves. Eg.

```
http://adcel.vrvm.com/htmlad?p=ptnr&b=sampletag&c=999&lat=40.753800&l
ong=-73.983354
```

4. vast - an API call that returns a VAST xml document. Unlike the other endpoints, this format is only intended to be used in VAST compliant video viewers. AdCel places event tracking calls in the various video viewing lifecycle (see Appendix B)

```
https://adcel.vrvm.com/vast?p=ptnr&b=vastvrvtest&c=999&adunit=vastlin
ear&lat=40.753800&long=-73.983354
```

Request Parameters for Banners & Interstitials

These URLs support a number of parameters, some of which must always be included in a request and others which are optional and can be used when additional data (such as user location) are available.

Optional parameters that are most highly valued by Verve are parameters which deal with user location (lat/long/ll/z) and user identifiers (ui/uis)

Required parameters

Parameter	Name	Description
p	Portal Keyword	This is a Verve-assigned value identifying the distribution portal on which the advertisement is being displayed. In most cases, a mobile website accessed directly using the publisher's hostname is on the default portal (keyword: "def"). The portal will change if the site is accessed through a mobile carrier's "deck". Mobile client applications are typically segregated by platform (e.g., iPhone, BlackBerry, Windows Mobile, etc.) using portals.
b	Partner Keyword	This is a Verve-assigned value identifying the publisher of the mobile content. Note that there's a one-to-one relationship between a mobile site and a publisher. For example, for the Daily Planet's mobile site, the partner keyword might be "dailyplanet".
c	Content Category	Content category ID. This is an integer which identifies the type of content available on the page for which an ad is requested. For example, a page of sports news will have one category ID, while a page of movie reviews will have another. See Appendix A for a complete list of categories and codes. Multiple c values can be passed on the querystring as c=value1&c=value2&c=value3
ui	User Identifier	Unique identifier for a user, commonly from an HTTP cookie (mobile web) or some sort of device ID (apps). If a ui value is passed, the uis parameter should also be set. ui value is case sensitive. Unhashed IDFA (Apple ID) should be upper case. All other ui values, included hashed IDFA's, should be lower case.
uis	User Identifier Source/Type	Code indicating the source or type of ID passed in the "ui" parameter. (Ignored when the "ui" parameter is not present.) By default and in the absence of this parameter, the user identifier is assumed to be a randomly-generated ID. iOS id's must be passed as uppercase and Android id's in lower case. See User Identifier Guidance for the values to code various identifiers.
appid	App Bundle ID	Unique key that identifies the app (e.g. iOS Bundle ID or Android App ID or Package name). This is usually in the following format: com.example.myapp

Optional parameters

Parameter	Name	Description
adunit	Ad Unit	Used to retrieve an ad specifically crafted for the target medium. Required for non-MMA banners. (see below). List of acceptable ad units are mma (default), banner, inter, tinter, vastlinear (only supported on the /vast endpoint), vast nonlinear (only supported on the /vast endpoint).
size	Ad Size	Specifies the size of the ad you want returned. Required for non-MMA banners or where an explicit mma size is required. See below for list of acceptable sizes and the ad units.
lat	Latitude	Latitude location of the user making the ad request. Note the "ll" parameter is the preferred method in which to supply geolocation information. Please pass the most accurate location known, which is generally GPS data pulled from the device.
long	Longitude	Longitude location of the user making the ad request. Please pass the most accurate location known, which is generally GPS data pulled from the device.
ll	Lat/Long (Encoded)	Latitude and longitude of the user, encoded. The location of the user at the time of the advertising request. Availability of this parameter is, of course, platform- dependent and optional. It must be excluded if the user does not consent to provide location information in some manner. However, if location is available, it should be included. The parameter name is two lower-case Ls, and is encoded in a special format (described later in this document). The location's resolution does not have to be particularly precise, however a minimum target of 3km is recommended. Also, for a particular user session, it's sufficient to obtain the location only once. I.e., location does not need to be looked-up for every ad request during a session. Please pass the most accurate location known, which is generally GPS data pulled from the device.
latlong	Lat/Long (Plain-text)	Latitude and longitude of the user, plain text. The value of this parameter should adhere to the format <latitude>,<longitude>. The data supplied here should adhere to the same rules as the above lat/long parameters, this is simply a convient method of providing location for those integrations where it is not possible to split the lat/long into separate parameters.
la	Location Age	Age of the supplied location (either lat/long or ll), in minutes. Please omit if the age is unknown.
lacc	Location Accuracy	Accuracy of the supplied location (either lat/long or ll), in meters. Please omit if the accuracy is unknown.
z	Postal Code	Zip or Zip+4 postal code describing the location of the content being viewed (not necessarily the postal code of the user's location). The value should be a zero leading 5 digit postal code. Examples: 07048, 21228, 90210-1234,
ip	Device's IP Address	IP address of the client/device on which the requested ad will be displayed. Mandatory for server-side requests; optional when requests come directly from a user's device.
age	Age	The user's age used for demographic targeting.
zip	Zip	The user's zip code.
gender	Gender	The user's gender used for demographic targeting. Acceptable values: "male", "female", "m", "f". If unknown, do not pass the key/value.
site	Site/App Name	The site on which the ad is presented. On mobile web, this would be the hostname of the mobile site; in the app world, it's a unique key that identifies the app (e.g., the iOS bundle ID).
pos	position	position in which ad is to appear on page. Acceptable values are "top","inline", and "bottom"
ei	extra info	semi-colon separated list of arbitrary key-value pairs. Example: color=blue;shape=square; (note, the value should be url encoded, as it is a query parameter value)
ua	User Agent	User Agent should be passed in the header. If that is not possible, pass an encoded user agent string here.
model	Device Model	Model information for the requesting device when otherwise unavailable from the user agent (i.e., iOS, primarily). An iPhone 6 is referenced as "iPhone7,2" and an iPhone 6 Plus is"iPhone7,1". Use the identifier as prescribed in the following list: https://theiphonewiki.com/wiki/Models
hwmdl	Hardware Model	Manufacturer's designation for the device hardware model. At present, this should only be populated for iOS using the "hw.model" sysctl. For example, the GSM iPhone 4 would use the value "N90AP". Refer to the internal name designed in this list: https://theiphonewiki.com/wiki/Models On iOS, this parameter is preferred to the "model" parameter, but should be omitted if the information is unavailable.

dguid	device guid	A Verve-assigned identifier assigned to the device type making the request. It is used to override the "User-Agent" HTTP request header. Note that this parameter is typically only used in conjunction with the Verve Registration API for mobile client applications. If blank, omitted, or an unknown value, the "User-Agent" HTTP request header is used to identify the device type. Ignore unless directed otherwise.
db	display block id	Ignore unless directed otherwise.
pm	partner module id	Ignore unless directed otherwise.
iframe	disable iframe wrapping	If this parameter is set to false, responses from certain ad networks will not be wrapped within an iframe. This parameter should not be relied on to ensure that there is never or is always an iframe response. This is generally an internal Verve parameter for testing. Ignore unless directed otherwise.
cc	creative capability	Define the creative capability of your request. "cc=1": for MRAID1 support "cc=2": for MRAID2 support (From the client Apps this value will be just integer 2)
nwk	Ad Network	char[]: The production ID of the ad network (ex: dfp even in staging would be '41'). Skips the chain resolution and directly calls the network.
flt	Flight	char[]: Unique flight identifier of some sort (passed through to ad network DSL) Requires nwk parameter is present. Identifies the flight id this call will use for creative tag retrieval.
ctg	Creative Tag	char[]: Unique creative identifier of some sort (passed through to ad network DSL) Requires nwk and flt parameters. Identifies the creative id this call will use for creative tag retrieval.
sc	Secure Creative	s : requires secure creative ns : requires insecure creative x or n : secure creative agnostic (can handle both secure or unsecure)
ou	Referer Override	The ou parameter should be used for mobile web requests, specifying the page url that the ad was requested on. Note: AdCel will use the referer header value for this, if this override is not specified.
appver	App Version	This is the current version number of your app
usprivacy	US Privacy Regulatory Context	(Used for CCPA) Regulatory context governing personal data contained within the bid request and any subsequent related transactions. Format defined by: https://iabtechlab.com/wp-content/uploads/2019/11/U.S.-Privacy-String-v1.0-IAB-Tech-Lab.pdf

When using the Verve Registration API, typically the “b” (partner keyword), “p” (portal keyword), and “ua” or “dguid” parameters are pre-populated in the AdCel base URL value returned upon successful registration. Other values such as “pm” (partner module ID), “db” (display block ID), and “c” (content category ID) are available from the Content API in either the content hierarchy or listing.

If the implementer is not using the Verve Registration or Content APIs, as is usually the case for mobile website applications, Verve will supply the values required. In cases where the AdCel integration is for content linked from a Verve-hosted mobile site, the partner and portal keywords and display block, partner module, and content category IDs are usually static.

Adunit and size parameter

By default, all ad requests return auto-sized MMA banners. An adunit and size is required for non-mma banners or if a specific mma size is required. Requests for the iPad portal (p=ipad) must set an adunit value.

Adunit Value	Size Values	Description	Example

mma	Size optional. Largest size served based on device capability.
	320x50, 300x50
		Auto-sized MMA banner. Only set mma and the size if a specific mma size is required.	Auto sized mma banner. MMA adunit and size not required.
			<script>
			vrv = document.vrv || {};
			vrv.b = 'sampletag';
			vrv.p = 'ptnr';
			vrv.c = '999';
			</script>
			<script src="https://ad.vrvm.com/custom/verve/pixels/vervetag/vervetag.js">
			For a specific mma size:
			<script>
			vrv = document.vrv || {};
			vrv.b = 'sampletag';
			vrv.p = 'ptnr';
			vrv.c = '999';
			vrv.adunit = 'mma';
			vrv.size = '300x50';
			</script>
			<script src="https://ad.vrvm.com/custom/verve/pixels/vervetag/vervetag.js">

banner	728x90, 300x250	Tablet ad. Requires tablet portal ('p') values:
		• iPad: 'ipad' (vrv.p = 'ipad')
		• Android Tablet: 'apad' (vrv.p = 'apad')
			<script>
			vrv = document.vrv || {};
			vrv.b = 'sampletag';
			vrv.p = 'ipad';
			vrv.c = '999';
			vrv.adunit = 'banner';
			vrv.size = '728x90';
			</script>
			<script src="https://ad.vrvm.com/custom/verve/pixels/vervetag/vervetag.js">

banner	300x250	Mobile phone banner. Requires mobile device portal ('p') values:
		• iPhone: 'iphn' (vrv.p = 'iphn')
		• Android: 'anap' (vrv.p = 'anap')
			<script>
			vrv = document.vrv || {};
			vrv.b = 'sampletag';
			vrv.p = 'iphn';
			vrv.c = '999';
			vrv.adunit = 'banner';
			vrv.size = '300x250';
			</script>
			<script src="https://ad.vrvm.com/custom/verve/pixels/vervetag/vervetag.js">

inter	320x416, 300x250, 320x480	Mobile interstitial. Full size (320x416) or small size (300x250)	<script>
			vrv = document.vrv || {};
			vrv.b = 'sampletag';
			vrv.p = 'iphn';
			vrv.c = '999';
			vrv.adunit = 'inter';
			vrv.size = '320x416';
			</script>
			<script src="https://ad.vrvm.com/custom/verve/pixels/vervetag/vervetag.js">

Interstitials

To request an interstitial ad, the adunit needs be set to 'inter' for mobile phones or 'tinter' for tablets. Refer to the associated sizes in the adunit section above.

Video Tag Integration

The video tag Integration is the simplest form of integration that Verve supports for video. The Verve video tag is a simple call to a Verve VAST endpoint, which will return a VAST-compliant video ad response.

```
https://adcel.vrvm.com/vast?p=ptnr&b=vastvrvtest&c=999&adunit=vastlinear&lat=40.753800&long=-73.983354
```

Required parameters

There are the parameters that are required in every tag.

Parameter	Name	Description
p	Portal Keyword	This is a Verve-assigned value identifying the distribution portal on which the advertisement is being displayed. In most cases, a mobile website accessed directly using the publisher's hostname is on the default portal (keyword: "def"). The portal will change if the site is accessed through a mobile carrier's "deck". Mobile client applications are typically segregated by platform (e.g., iPhone, BlackBerry, Windows Mobile, etc.) using portals.
b	Partner Keyword	This is a Verve-assigned value identifying the publisher of the mobile content. Note that there's a one-to-one relationship between a mobile site and a publisher. For example, for the Daily Planet's mobile site, the partner keyword might be "dailyplanet".
c	Content Category	This is an integer which identifies the type of content available on the page for which an ad is requested. For example, a page of sports news will have one category ID, while a page of movie reviews will have another. See Appendix A for a complete list of categories and codes. Multiple c values can be passed on the querystring as c=value1&c=value2&c=value3
ui	User Identifier	Device ID. If a ui value is passed, the uis parameter should also be set. ui value is case sensitive. Unhashed IDFA (Apple ID) should be upper case. All other ui values, including hashed IDFA's, should be lower case.
uis	User Identifier Source	Code indicating the source or type of ID passed in the "ui" parameter. (Ignored when the "ui" parameter is not present.) By default, and in the absence of this parameter, the user identifier is assumed to be a randomly-generated ID. iOS id's must be passed as uppercase and Android id's in lower case. Refer to this section to understand the uis mapping: User Identifier Guidance. If you cannot properly set a uis value, please talk with your account manager for alternate ways to pass this data.
appid	App Bundle Id	Unique key that identifies the app (e.g. iOS Bundle ID or Android App ID or Package name). This is usually in the following format: com.example.myapp
adunit	Ad Unit	Valid values; vastlinear (only supported on the /vast endpoint), vastnonlinear (only supported on the /vast endpoint).
cc	Container Capability	Indicates the container capabilities for the vidoe ad request. VAST and/or VPAID support, including supported version, should be communicated: e.g. cc=vast2.0&cc=vpaid1.0
videoPlacement	Video Placement	Allowed values: stream, banner, article, feed, floating • stream: Played before, during or after the streaming video content that the consumer has requested (e.g., Pre-roll, Mid-roll, Post-roll) • banner: Exists within a web banner that leverages the banner space to deliver a video experience as opposed to another static or rich media format. The format relies on the existence of display ad inventory on the page for its delivery. • article: Loads and plays dynamically between paragraphs of editorial content; existing as a standalone branded message • feed: Found in content, social, or product feeds • floating: Covers the entire or a portion of screen area, but is always on screen while displayed (i.e. cannot be scrolled out of view)
vph	Video Player Height	The video player height. Example value: 200.
vpw	Video Player Width	The video player width. Example value: 300.

Optional Parameters

If your application/platform captures accurate location or advertiser ids, passing this data on the request can greatly increase the monetization of your inventory. Simply insert the parameters you wish to pass in the request in your ad tag as show below. Your account manager will create a custom tag for your specific integration.

Parameter	Name	Description
ll	Lat/Long (Encoded)	Latitude and longitude of the user, encoded. The location of the user at the time of the advertising request. Availability of this parameter is, of course, platform- dependent and optional. It must be excluded if the user does not consent to provide location information in some manner. However, if location is available, it should be included. The parameter name is two lower-case Ls, and is encoded in a special format (described later in this document). The location's resolution does not have to be particularly precise, however a minimum target of 3km is recommended. Also, for a particular user session, it's sufficient to obtain the location only once. I.e., location does not need to be looked-up for every ad request during a session. Please pass the most accurate location known, which is generally GPS data pulled from the device.
lat	Latitude	Latitude location of the user making the ad request. Note the "ll" parameter is the preferred method in which to supply geolocation information. Please pass the most accurate location known, which is generally GPS data pulled from the device.
long	Longitude	Longitude location of the user making the ad request. Please pass the most accurate location known, which is generally GPS data pulled from the device.
latlong	Lat/Long (Plain-text)	Latitude and longitude of the user, plain text. The value of this parameter should adhere to the format <latitude>,<longitude>. The data supplied here should adhere to the same rules as the above lat/long parameters, this is simply a convient method of providing location for those integrations where it is not possible to split the lat/long into separate parameters.
hwmdl	Hardware Model	Manufacturer's designation for the device hardware model. At present, this should only be populated for iOS using the "hw.model" sysctl. For example, the GSM iPhone 4 would use the value "N90AP". Refer to the internal name designed in this list: https://theiphonewiki.com/wiki/Models On iOS, this parameter is preferred to the "model" parameter, but should be omitted if the information is unavailable.
duration	Duration	Specifies the duration in seconds of the ad you want returned. Valid values: 15 Second Video: "15" or 30 Second Video: "30", etc.
appver	App Version	This is the current version number of your app

Additional Video Parameters

In addition to the relevant parameters above (location/device/supply/etc), the following parameters may be used to further filter the potential demand to be requested.

Note: previously, the size parameter was used to define the duration. It has now been separated into a new field, 'duration'. Duration passed in the size parameter will still be honored, but this is less favorable than using the duration parameter.

Many of the parameters defined here have been derived from the OpenRTB specification.

Parameter	Name	Type	Examples	Required?	Default Value	Description
minDuration	Minimum Duration	integer (seconds)	15	no	any	The minimum length (in seconds) of the advertisement requested.
maxDuration	Maximum Duration	integer (seconds)	60	no	any	The maximum length (in seconds) of the advertisement requested.
videoMime	Video Mime Type	string	video/mp4	no	video/mp4	The mime type allowed for video delivery (e.g.: video/mp4, video/x-flv)
deliveryType	Delivery Type	String	progressive	no	progressive	Supported delivery methods. Allowed values: progressive, streaming, download
skip	Allow Skip	boolean (true|false)	true	no	false	Indicates if the player will allow the video to be skipped.
skipMin	Skip Minimum Seconds	integer (seconds)	10	no	0	Videos of total duration greater than this number of seconds can be skippable; only applicable if the ad is skippable

skipAfter	Skip After Seconds	integer (seconds)	10	no	0	Number of seconds a video must play before skipping is enabled; only applicable if the ad is skippable.
startDelay	Start Delay Seconds	integer (seconds)	10	no	0	Indicates the start delay in seconds for pre-roll, mid-roll, or post-roll ad placements. 0 is pre-roll >0 is Mid-Roll (value indicates start delay in second)
boxingAllowed	Letterboxing Allowed	boolean (true|false)	false	no	true	Indicates if letter-boxing of 4:3 content into a 16:9 window is allowed
autoPlay	Auto Play	boolean (true|false)	true	no	any	Indicates if the video is allowed to auto play
audioOnStart	Audio On Start	boolean (true|false)	true	no	any	Indicates if the video's audio is allowed to be on when starting

Creative Capability

Creative capability may now be used to define the version of vast/vpaid that the requester may need, and this can be passed via the creative capablity parameter. Multiple creative capability query parameters may be passed.

Allowed values for the creative capability parameter (cc) include: vast1.0, vast2.0, vast3.0, vpaid1.0, vpaid2.0

For example, a player that is capable of rendering a vpaid 1.0 creative delivered via vast would build a request with the following: cc=vast2.0&cc=vpaid1.0

This integration only supports vpaid delivery over javascript.

User Data

The "ei" (extra information) parameter is used to pass unique ad-hoc user data on ad request to AdCel.

Value	Description
age	Integer age of user
gender	User gender. Acceptable values include "m", "f", "male", or "female"
l	Language of the user. Acceptable values are "en" (English) and "es" (Spanish)

The ei parameter is a semi-colon separated list of values. For example, "ei=age=21;gender=f".

Location Encoding

In order to both economize data transfer and obfuscate user-specific location information otherwise transferred in plain-text, the latitude and longitude parameter (“ll”) may be encoded using a method derived from that used by Google Maps. This method takes the pair of signed floats and encodes them into a single ASCII string. The method can be described as follows:

1. Multiply both values by 100000 and convert the result to an integer. (Note that this truncates the precision to 5 decimal places.)
2. Shift the value left by one bit.
3. Invert negative numbers.
4. Split the value into 5-bit sections, starting from the right.
5. Reverse the order of the sections.
6. Bitwise OR each value with 0x20 if another bit chunk follows.
7. Add 0x3F (ASCII '?') to each value.
8. Convert each value to its ASCII equivalent.
9. Translate reserved URL characters to ASCII 0 to 9 (see sample code in the appendix).

An example of this encoding is:

```
+37.33168900, -122.03073100: _jzbF8cygV
```

Server-side Requests

Frequently, an ad request must be resolved “server-side” on behalf of the mobile device. This is typically because an AdCel response cannot be handled directly by a mobile browser. However, server-side requests should be avoided where possible, such as when developing mobile client applications.

Ultimately, the AdCel advertisement request needs to mimic a request from the actual device as much as possible. Both AdCel and the ad networks with which it integrates depend upon a number of request characteristics -- the device’s IP address and HTTP request headers, in particular -- to properly target and control ad delivery. When server\- side requests are not implemented properly, some of these characteristics are lost and ad targeting becomes less effective.

Both mobile devices and carriers add meaningful HTTP request headers that help identify the device, the device’s capabilities, the carrier, and, on occasion, the mobile subscriber. To the fullest extent possible, the server making an AdCel request should mimic these headers.

For example, the server receives a request from a mobile device for a mobile website page. The server should store the headers included in this request. When the server is generating its request of AdCel for a mobile advertisement, with three exceptions, it should populate the ad request with the headers from the original mobile device page request. The three headers it should not include in the AdCel request are: "Connection", "Host", and "X-Forward-For".

Examples of headers that are meaningful for ad targeting include: "Accept", "Accept-Encoding", "Accept-Language", "msisdn", "Referer", "User-Agent", and "x-up-subno".

Similarly, the server must pass through the mobile device's IP address via the AdCel request’s "ip" GET parameter.

To summarize:

1. Ideally, the server will include all original page request headers in the AdCelrequest except “Connection”, “Host”, and “X-Forwarded-For”. The minimum requirement is that “User-Agent” is included with the device’s user agent string.
2. The server must include the device’s IP address in the AdCel request’s "ip" GET parameter.
3. If Google AdSense ads are or might be used in the implementation, the AdCel reqeust’s “ou” GET parameter must be populated with the encoded URL of the mobile page request.

Ad Responses

At present, AdCel supports a single, XML-formatted response type. The requester must parse this response to extract the information necessary to display the returned advertisement.

In its most basic form, a successful ad response includes:

1. A URL to a banner image, appropriately sized for the mobile device;
2. A response, or "clickthrough", URL that sends the user to a mobile landing page for the advertisement.
3. Alternate and "lead-in" text for the advertisement.
4. Optionally, an audit, or tracking, image URL. Frequently, an ad network will include this 3rd-party URL in order to allow independent auditing of advertisement delivery. These URLs ordinarily return a 1x1 transparent GIF image.
5. Optionally, a snippet of "raw" HTML that should be used to display the advertisement. Some ad networks require that the mobile browser display the advertisement in a particular way. Google AdSense for Mobile is currently the only ad network using this feature.

A successful ad request will have an HTTP status code of 200 for its response. A non-200 status code should be treated as an error and the response body discarded. Thefollowing sections provide examples of successful ad responses.

No Ad Available

When no advertisement is available for the request, AdCel simply returns an empty response. Note, however, that the response is still considered "successful" and will have an HTTP status code of 200. The "Content-Type" header is "text/plain".

There are, of course, many reasons why an ad may not be available for a given request. For example, active campaigns may not be targeted to match the parameter values passed in for the request or campaigns may be "paced" to control the rate at which they’re delivered.

Sample Ad Response With Tracking Image

The following XML example is of an advertisement that includes a tracking image. This XML is presented in the response body, with a "Content-Type" of "application/xml".

```
<ad>
            <media>

<image_url>http://sponsor.vrvm.com/adcel/n=1/p=ptnr/b=sampletag/c=888/s=jy
qrt2xujx/m=817/x=4/ad_media.gif</image_url>
                <image_alt>Find the car you are looking for</image_alt>
            </media>
            <tracking>

<tracking_image_url>http://sponsor.vrvm.com/adcel/n=7/p=ptnr/b=sampletag/c
=888/s=O52bROsgC2/m=123/x=0/tracker_pixel.jpg
                </tracking_image_url>
            </tracking>
            <copy>
                <leadin>Find the car you are looking for</leadin>
            </copy>
            <clickthrough>

<url>http://sponsor.vrvm.com/adcel/n=1/p=ptnr/b=Lexington/c=888/s=jyqrt2xu
jx/fl=0/ad_click.htm</url>
            </clickthrough>
        </ad>
```

Sample Raw HTML Ad Response

AdCel will include raw HTML as CDATA values in the XML response for ad networks that have specific advertisement display requirements. Google AdSense for Mobile is the only ad network for which such responses are generated at present. However, the AdCel implementation should attempt to support this response type even if Google AdSense is not expected to be used for the target sites or applications as other ad networks may require this feature.

Occasionally, the raw HTML will contain a CDATA section of its own. In this case, the HTML data are escaped within the response element’s CDATA section (using character entity references), setting the attribute “escapedHtml” to “true”. If this attribute is omitted or “false”, the content of the response element’s CDATA section is escaped for XML only.

This response type is only applicable to mobile websites; implementation is not required for mobile client applications.

Note that, frequently, the raw HTML will be nothing more than an HTML comment. It is still necessary to include this comment in the page, as the ad network will crawl the mobile site to confirm that its specified HTML is presented verbatim. Following is an example response that includes such a comment:

```
<ad>
            <media>
                <image_url/>
                <image_alt/>
            </media>
            <copy>
                <leadin/>
            </copy>
            <clickthrough>
                <url/>
            </clickthrough>
            <rawResponse>
                <useRawResponse>true</useRawResponse>
                <response><![CDATA[
                    <!--
google_afm:ad_type=text_image:channel=7722988367:client=ca-mb-pub-55064272
09637323:format=mobile_single:markup=xhtml:oe=utf8:output=xhtml:ip=206.53.
153.54:url=http://m.dailyplanet.com:80/portalsearch.htm?terms=12841 -->
                    ]]>
                </response>
            </rawResponse>
        </ad>
```

This next example is of an actual ad that should be displayed using the raw HTML:

```
<ad>
            <media>
                <image_url/>
                <image_alt/>
            </media>
            <copy>
                <leadin/>
            </copy>
            <clickthrough>
                <url/>
            </clickthrough>
            <rawResponse>
                <useRawResponse>true</useRawResponse>
                <response><![CDATA[
                    <!--
google_afm:ad_type=text_image:channel=7722988367:client=ca-mb-pub-55064272
09637323:format=mobile_single:markup=xhtml:oe=utf8:output=xhtml:ip=192.249
.47.164:url=http://m.dailyplanet.com:80/index.jsp -->
                    <div style="background-color:#eeeeee; margin-top:3px;
border-top: 1px #555555 solid; margin-bottom:3px; border-bottom: 1px
#555555 solid;">
                        <div style="background-color:#eeeeee;
font-size:x-small; font-style:normal; font- weight:normal;
text-decoration:none; text-align:right; color: #777777;">
                            Ads by Google
                        </div>
                        <div style="background-color:#eeeeee;
font-size:small; font- style:normal; font-weight:normal;
text-decoration:none; padding-bottom: 3px;">
                            <span style="color: #000000; font-weight:
bold;">Baja Insurance</span><span
                                style="color:#000000;"> - Save money! -
</span><span style="color:#008000;">Got Coverage?</span><span
                                style="color:#000000;"> - </span><a

href="http://googleads.g.doubleclick.net/pagead/iclk?sa=l&amp;ai=BGM5vKJsV
Ssm8D5rIqAPprvysDMCNtwGwvRAQARgBII7xpw04AFCsxbzgAWDJBqABxpeT_QOyARRkaXNjb3
Zlcm91cnRvd24ubW9ia1vYmlsZV9zaW5nbGXIAQH8vZGlzY292ZXJvdXJ0b3duLm1vYmkvaW5k
ZXguanNwgAIB
sAIAqAMBsAMA6AMX9QMAAAAEsAQB&amp;num=1&amp;adurl=http://www.google.com/m/c
tc_ad%3Fphone%3D1-800-716-1804%26hl%3Den%26gl%3DUS"
                                style="color: #0000cc; font- size:small;
font-style:normal; font-weight:normal; text- decoration:underline;">Call:
                            1-800-716-1804</a></div>
                    </div>
                    ]]>
                </response>
            </rawResponse>
        </ad>
```

Sample Ad Response Without Tracking Image

Ad campaigns that do not require 3rd-party impression auditing won’t have a tracking image URL in their responses. The following example is of such an advertisement:

```
<ad>
            <media>

<image_url>http://event-img.qwapi.com/transcode?refresh=true&amp;location=
http://ad.qwapi.com/adserver/image%3fpath%3d28f487c4de18305db4cf1bc2f48922
d6.jpg&amp;qpub=0</image_url>
                <image_alt/>
            </media>
            <copy>
                <leadin>Its here - Get Skype for iPhone</leadin>
            </copy>
            <clickthrough>

<url>http://a.qwapi.com/c?t=-iE0j8m5plkPPMrgd7KfLQ&amp;iid=a0b04935165851b
29cc636a38ba87558&amp;aocid=76a23619f6264e0c9f7784785424f9e4&amp;test=0</u
rl>
            </clickthrough>
        </ad>
```

Ad Display

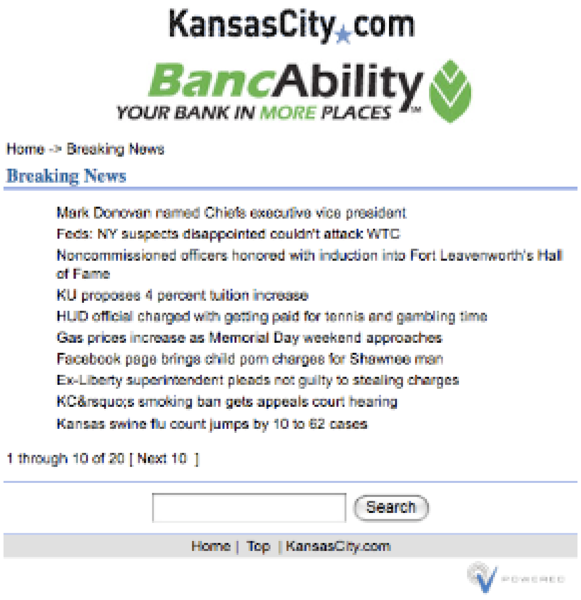
Mobile Websites

Verve Wireless, as a member of the Mobile Marketing Association, strives to conform to MMA guidelines for all its mobile advertising activities. Consequently, any mobile website integration with AdCel should display returned advertisements according to these guidelines. The MMA’s "Mobile Advertising Guidelines" document is available from: http://mmaglobal.com/mobileadvertising.pdf

Ad responses that include the raw HTML for advertisement display should be straightforward: simply include the contents of the CDATA value in the page.

Positioning of the ad should, of course, match the value passed in for the "pos" parameter in the request (defaulting to the top of the page).

The following screenshot is of a top-positioned banner. Note how it is placed between the masthead and news content.



Home -> Breaking News

Breaking News

Mark Donovan named Chiefs executive vice president
Feds: NY suspects disappointed couldn't attack WTC
Noncommissioned officers honored with induction into Fort Leavenworth's Hall of Fame
KU proposes 4 percent tuition increase
HUD official charged with getting paid for tennis and gambling time
Gas prices increase as Memorial Day weekend approaches
Facebook page brings child porn charges for Shawnee man
Ex-Liberty superintendent pleads not guilty to stealing charges
KC&rsquo;s smoking ban gets appeals court hearing
Kansas swine flu count jumps by 10 to 62 cases

1 through 10 of 20 [Next 10]

Home | Top | KansasCity.com

The screenshot uses the media.image_url, media.image_alt, clickthrough.url, and tracking.tracking_image_url elements of the AdCel response. Following is sample HTML code for generating the result shown in the screenshot:

```
<html>
        <head>
            ...
            <style type="text/css">
                ...
#bannerad {
                    text-align: center;
                }

                #bannerad img {
                    padding: 5px;
                }

                ...
            </style>
        </head>
        <body>
        ...
        <div id="bannerad"><img src="[tracking.tracking_image_url]"
width="1"
                                height="1"/><a
href="[clickthrough.url]"><img src="[media.image_url]"

alt="[media.image_alt]"/></a></div>
        ...
        </body>
        </html>
```

If tracking.tracking_image_url is not available or is empty, its “img” tag should not be included in the advertisement URL.

The copy.leadin element -- also known as a “text tagline” -- of the response is primarily a legacy device used to ensure that the advertisement is displayed on older mobile browsers that allowed the user to turn off graphics downloads. Lead-in text is typically displayed as centered immediately beneath the banner image and is clickable, linking to the value provided in the clickthrough.url element. Unless lead-in text is desired or appropriate for the page on which the ad will be displayed, it can be safely ignored.

As an aside, understand that, in general, the browser should never cache either the tracking or the banner images. Informing the browser of this requirement is the responsibility of the ad network hosting these images through expiration\- or cache-related HTTP headers or “cache busting” values in the URLs. Browser caching of these images can lead to impression underreporting.

Mobile Client Applications

At present, there are no firm guidelines for display advertisements in mobile client applications. To a large degree, the capabilities, user interface conventions, and user expectations of a given mobile platform (e.g., iPhone, BlackBerry, Windows Mobile, etc.) will dictate how ads should be presented.

However, there are a few guidelines with respect to AdCel integration within mobile client applications:

1. The application must not cache the media.image_url nor tracking.tracking_image_url images. These images must be downloaded anew each time the ad is displayed.
2. The tracking.tracking_image_url image does not need to be displayed; the only requirement for it is that the application issue a GET request for the image at the time the advertisement is displayed. The image returned can be safely discarded.
3. Where possible, the application should render the ad using a platform facility that provides full Web browser capabilities (e.g., the iPhone’s “webview”). Though it is acceptable to display the banner image “manually,” using normal browser rendering is a more robust solution: it avoids problems with unexpected image formats such as animated GIFs and allows more flexibility for future ad units.
4. The raw HTML (rawResponse element) portion of the AdCel response does not need to be implemented in mobile client applications.
5. The response URL (clickthrough.url element) should be handled in such a fashion that allows mobile platform-specific URLs to behave as intended. For example, the iPhone allows URLs to link to specific applications within the iPhone App Store; responses for such URLs should take the user to the App Store. Likewise, the application should properly handle URLs that, e.g., open a mapping application, dial a number, or send an SMS. (One complication with this requirement is that the clickthrough.url value is typically not the final destination. It is common for response URLs to be proxied (using 302 redirects) in order to count ad responses.)
6. It is never acceptable to delay the display of a user-requested page of content while waiting on an advertisement. Any failure or delay on the part of AdCel to respond to an ad request must not prevent or delay content display.
7. It is never acceptable to delay the display of a user-requested page of content while waiting on an advertisement. Any failure or delay on the part of AdCel to respond to an ad request must not prevent or delay content display.
8. The page layout and rendering must accommodate the case when AdCel does not return an ad. This may be done by simply not displaying the ad unit or by displaying a placeholder (house) ad, but it is not acceptable to display a blank ad unit.
9. At present, AdCel only supports “online” advertisement display, meaning the client application must have an active network connection in order to make ad requests at the time the ad is to be displayed. The client application must not pre-fetch or repeat advertisements for display in an offline scenario. When the application is offline, it should either display no advertisements or display placeholder (house) ads. No impression reporting facilities are available for counting offline ad placement, though full offline advertising support is planned for future release.

Appendices

Appendix A: Content Categories

ID	Category
1	Automotive
15	Careers
22	Business
24	Arts and Entertainment
32	Arts and Entertainment::Television
34	Family and Parenting
37	Events
40	Movies
44	Hobbies and Interests
52	Food and Drink

77	Music
83	News and Information::International News
84	News and Information::Local
86	News and Information::National
87	News and Information::Regional
93	Style and Fashion
97	News and Information
110	Sports
117	Technology and Computing
122	Travel
133	Weather
134	Real Estate
135	Traffic
136	Front Page
137	Article Listing Page
888	Home Page
999	Unknown/Uncategorizable
1000	Arts and Entertainment::Books and Literature
1001	Arts and Entertainment::Celebrity Fan/Gossip
1002	Arts and Entertainment::Fine Art
1003	Arts and Entertainment::Humor
1004	Arts and Entertainment::Music
1005	Automotive::Auto Parts
1006	Automotive::Auto Repair
1007	Automotive::Buying/Selling Cars
1008	Automotive::Car Culture
1009	Automotive::Certified Pre-Owned
1010	Automotive::Convertible
1011	Automotive::Coupe
?1012	Automotive::Crossover
1013	Automotive::Diesel
1014	Automotive::Electric Vehicle
1015	Automotive::Hatchback
1016	Automotive::Hybrid
1017	Automotive::Luxury
1018	Automotive::Mini Van
1019	Automotive::Motorcycles

1020	Automotive::Off-Road Vehicles
1021	Automotive::Performance Vehicles
1022	Automotive::Pickup
1023	Automotive::Road-Side Assistance
1024	Automotive::Sedan
1025	Automotive::Trucks Accessories
1026	Automotive::Vintage Cars
1027	Automotive::Wagon
1028	Business::Advertising
1029	Business::Agriculture
1030	Business::Biotech/Biomedical
1031	Business::Business Software
1032	Business::Construction
1033	Business::Forestry
1034	Business::Government
1035	Business::Green Solutions
1036	Business::Human Resources
1037	Business::Logistics
1038	Business::Marketing
1039	Business::Metals
1040	Careers::Career Planning
1041	Careers::College
1042	Careers::Financial Aid
1043	Careers::Job Fairs
1044	Careers::Job Search
1045	Careers::Resume Writing/Advice
1046	Careers::Nursing
1047	Careers::Scholarships
1048	Careers::Telecommuting
1049	Careers::U.S. Military
1050	Careers::Career Advice
1051	Education
1052	Education::7-12 Education
1053	Education::Adult Education
1054	Education::Art History
1055	Education::College Administration
1056	Education::College Life

1057	Education::Distance Learning
1058	Education::English as a 2nd Language
1059	Education::Language Learning
1060	Education::Graduate School
1061	Education::Homeschooling
1062	Education::Homework/Study Tips
1063	Education::K-6 Educators
1064	Education::Private School
1065	Education::Special Education
1066	Education::Studying Business
1067	Family and Parenting::Adoption
1068	Family and Parenting::Babies and Toddlers
1069	Family and Parenting::Daycare/Pre School
1070	Family and Parenting::Family Internet
1071	Family and Parenting::Parenting - K-6 Kids
1072	Family and Parenting::Parenting Teens
1073	Family and Parenting::Pregnancy
1074	Family and Parenting::Special Needs Kids
1075	Family and Parenting::Eldercare
1076	Health and Fitness
1077	Health and Fitness::Exercise
1078	Health and Fitness::A.D.D.
1079	Health and Fitness::AIDS/HIV
1080	Health and Fitness::Allergies
1081	Health and Fitness::Alternative Medicine
1082	Health and Fitness::Arthritis
1083	Health and Fitness::Asthma
1084	Health and Fitness::Autism/PDD
1085	Health and Fitness::Bipolar Disorder
1086	Health and Fitness::Brain Tumor
1087	Health and Fitness::Cancer
1088	Health and Fitness::Cholesterol
1089	Health and Fitness::Chronic Fatigue Syndrome
1090	Health and Fitness::Chronic Pain
1091	Health and Fitness::Cold and Flu
1092	Health and Fitness::Deafness
1093	Health and Fitness::Dental Care

1094	Health and Fitness::Depression
1095	Health and Fitness::Dermatology
1096	Health and Fitness::Diabetes
1097	Health and Fitness::Epilepsy
1098	Health and Fitness::GERD/Acid Reflux
1099	Health and Fitness::Headaches/Migraines
1100	Health and Fitness::Heart Disease
1101	Health and Fitness::Herbs for Health
1102	Health and Fitness::Holistic Healing
1103	Health and Fitness::IBS/Crohn's Disease
1104	Health and Fitness::Incest/Abuse Support
1105	Health and Fitness::Incontinence
1106	Health and Fitness::Infertility
1107	Health and Fitness::Men's Health
1108	Health and Fitness::Nutrition
1109	Health and Fitness::Orthopedics
1110	Health and Fitness::Panic/Anxiety Disorders
1111	Health and Fitness::Pediatrics
1112	Health and Fitness::Physical Therapy
1113	Health and Fitness::Psychology/Psychiatry
1114	Health and Fitness::Senior Health
1115	Health and Fitness::Sexuality
1116	Health and Fitness::Sleep Disorders
1117	Health and Fitness::Smoking Cessation
1118	Health and Fitness::Substance Abuse
1119	Health and Fitness::Thyroid Disease
1120	Health and Fitness::Weight Loss
1121	Health and Fitness::Women's Health
1122	Food and Drink::American Cuisine
1123	Food and Drink::Barbecues and Grilling
1124	Food and Drink::Cajun/Creole
1125	Food and Drink::Chinese Cuisine
1126	Food and Drink::Cocktails/Beer
1127	Food and Drink::Coffee/Tea
1128	Food and Drink::Cuisine-Specific
1129	Food and Drink::Desserts and Baking
1130	Food and Drink::Dining Out

1131	Food and Drink::Food Allergies
1132	Food and Drink::French Cuislne
1133	Food and Drink::Health/LowFat Cooking
1134	Food and Drink::Italian Cuisine
1135	Food and Drink::Japanese Cuisine
1136	Food and Drink::Mexican Cuisine
1137	Food and Drink::Vegan
1138	Food and Drink::Vegetarian
1139	Food and Drink::Wine
1140	Hobbies and Interests::Arts/Technology
1141	Hobbies and Interests::Arts and Crafts
1142	Hobbies and Interests::Beadwork
1143	Hobbies and Interests::Birdwatching
1144	Hobbies and Interests::Board Games/Puzzles
1145	Hobbies and Interests::Candle and Soap Making
1146	Hobbies and Interests::Card Games
1147	Hobbies and Interests::Chess
1148	Hobbies and Interests::Cigars
1149	Hobbies and Interests::Collecting
1150	Hobbies and Interests::Comic Books
1151	Hobbies and Interests::Drawing/Sketching
1152	Hobbies and Interests::Freelance Writing
1153	Hobbies and Interests::Genealogy
1154	Hobbies and Interests::Getting Publishing
1155	Hobbies and Interests::Guitar
1156	Hobbies and Interests::Home Recording
1157	Hobbies and Interests::Investors and Patents
1158	Hobbies and Interests::Jewelry Making
1159	Hobbies and Interests::Magic and Illusion
1160	Hobbies and Interests::Needlework
1161	Hobbies and Interests::Painting
1162	Hobbies and Interests::Photography
1163	Hobbies and Interests::Radio
1164	Hobbies and Interests::Roleplaying Games
1165	Hobbies and Interests::Sci-Fi and Fantasy
1166	Hobbies and Interests::Scrapbooking
1167	Hobbies and Interests::Screenwriting

1168	Hobbies and Interests::Stamps and Coins
1169	Hobbies and Interests::Video and Computer Games
1170	Hobbies and Interests::Woodworking
1171	Home and Garden
1172	Home and Garden::Appliances
1173	Home and Garden::Entertaining
1174	Home and Garden::Environmental Safety
1175	Home and Garden::Gardening
1176	Home and Garden::Home Repair
1177	Home and Garden::Home Theater
1178	Home and Garden::Interior Decorating
1179	Home and Garden::Landscaping
1180	Home and Garden::Remodeling and Construction
1181	Law Gov't and Politics
1182	Law Gov't and Politics::Immigration
1183	Law Gov't and Politics::Legal Issues
1184	Law Gov't and Politics::U.S. Government Resources
1185	Law Gov't and Politics::Politics
1186	Law Gov't and Politics::Commentary
1187	Personal Finance
1188	Personal Finance::Beginning Investing
1189	Personal Finance::Credit/Debt and Loans
1190	Personal Finance::Financial News
1191	Personal Finance::Financial Planning
1192	Personal Finance::Hedge Fund
1193	Personal Finance::Insurance
1194	Personal Finance::Investing
1195	Personal Finance::Mutual Funds
1196	Personal Finance::Options
1197	Personal Finance::Retirement Planning
1198	Personal Finance::Stocks
1199	Personal Finance::Tax Planning
1200	Society
1201	Society::Dating
1202	Society::Divorce Support
1203	Society::Gay Life
1204	Society::Marriage

1205	Society::Senior Living
1206	Society::Teens
1207	Society::Weddings
1208	Society::Ethnic Specific
1209	Science
1210	Science::Astrology
1211	Science::Biology
1212	Science::Chemistry
1213	Science::Geology
1214	Science::Paranormal Phenomena
1215	Science::Physics
1216	Science::Space/Astronomy
1217	Science::Geography
1218	Science::Botany
1219	Pets
1220	Pets::Aquariums
1221	Pets::Birds
1222	Pets::Cats
1223	Pets::Dogs
1224	Pets::Large Animals
1225	Pets::Reptiles
1226	Pets::Veterinary Medicine
1227	Sports::Auto Racing
1228	Sports::Baseball
1229	Sports::Bicycling
1230	Sports::Bodybuilding
1231	Sports::Boxing
1232	Sports::Canoeing/Kayaking
1233	Sports::Cheerleading
1234	Sports::Climbing
1235	Sports::Cricket
1236	Sports::Figure Skating
1237	Sports::Fly Fishing
1238	Sports::Football
1239	Sports::Freshwater Fishing
1240	Sports::Game and Fish
1241	Sports::Horse Racing

1242	Sports::Horses
1243	Sports::Hunting/Shooting
1244	Sports::Inline Skating
1245	Sports::Martial Arts
1246	Sports::Mountain Biking
1247	Sports::NASCAR Racing
1248	Sports::Olympics
1249	Sports::Paintball
1250	Sports::Power and Motorcycles
1251	Sports::Pro Basketball
1252	Sports::Pro Ice Hockey
1253	Sports::Rodeo
1254	Sports::Rugby
1255	Sports::Running/Jogging
1256	Sports::Sailing
1257	Sports::Saltwater Fishing
1258	Sports::Scuba Diving
1259	Sports::Skateboarding
1260	Sports::Skiing
1261	Sports::Snowboarding
1262	Sports::Surfing/Bodyboarding
1263	Sports::Swimming
1264	Sports::Table Tennis/Ping Pong
1265	Sports::Tennis
1266	Sports::Volleyball
1267	Sports::Walking
1268	Sports::Waterski/Wakeboard
1269	Sports::World Soccer
1270	Style and Fashion::Beauty
1271	Style and Fashion::Body Art
1272	Style and Fashion::Fashion
1273	Style and Fashion::Jewelry
1274	Style and Fashion::Clothing
1275	Style and Fashion::Accessories
1276	Technology and Computing::3-D Graphics
1277	Technology and Computing::Animation
1278	Technology and Computing::Antivius Software

1279	Technology and Computing::C/C++
1280	Technology and Computing::Cameras and Camcorders
1281	Technology and Computing::Cell Phones
1282	Technology and Computing::Computer Certification
1283	Technology and Computing::Computer Networking
1284	Technology and Computing::Computer Peripherals
1285	Technology and Computing::Computer Reviews
1286	Technology and Computing::Data Centers
1287	Technology and Computing::Databases
1288	Technology and Computing::Desktop Publishing
1289	Technology and Computing::Desktop Video
1290	Technology and Computing::Email
1291	Technology and Computing::Graphics Software
1292	Technology and Computing::Home Video/DVD
1293	Technology and Computing::Internet Technology
1294	Technology and Computing::Java
1295	Technology and Computing::JavaScript
1296	Technology and Computing::Linux
1297	Technology and Computing::Mac OS
1298	Technology and Computing::Mac Support
1299	Technology and Computing::MP3/MIDI
1300	Technology and Computing::Net Conferencing
1301	Technology and Computing::Net for Beginners
1302	Technology and Computing::Network Security
1303	Technology and Computing::Palmtops/PDAs
1304	Technology and Computing::PC Support
1305	Technology and Computing::Portable Entertainment
1306	Technology and Computing::Shareware/Freeware
1307	Technology and Computing::Unix
1308	Technology and Computing::Visual Basic
1309	Technology and Computing::Web Clip Art
1310	Technology and Computing::Web Design/HTML
1311	Technology and Computing::Web Search
1312	Technology and Computing::Windows
1313	Travel::Adventure Travel
1314	Travel::Africa
1315	Travel::Air Travel

1316	Travel::Australia and New Zealand
1317	Travel::Bed and Breakfast
1318	Travel::Budget Travel
1319	Travel::Business Travel
1320	Travel::By US Locale
1321	Travel::Camping
1322	Travel::Canada
1323	Travel::Caribbean
1324	Travel::Cruises
1325	Travel::Eastern Europe
1326	Travel::Europe
1327	Travel::France
1328	Travel::Greece
1329	Travel::Honeymoons/Getaways
1330	Travel::Hotels
1331	Travel::Japan
1332	Travel::Mexico and Central America
1333	Travel::National Parks
1334	Travel::South America
1335	Travel::Spas
1336	Travel::Theme Parks
1337	Travel::Traveling with Kids
1338	Travel::United Kingdom
1339	Real Estate::Apartments
1340	Real Estate::Architects
1341	Real Estate::Buying/Selling Homes
1342	Shopping
1343	Shopping::Contests and Freebies
1344	Shopping::Couponing
1345	Shopping::Comparison Engines
1346	Religion and Spirituality
1347	Religion and Spirituality::Alternative Religions
1348	Religion and Spirituality::Atheism/Agnosticism
1349	Religion and Spirituality::Buddhism
1350	Religion and Spirituality::Catholicism
1351	Religion and Spirituality::Christianity
1352	Religion and Spirituality::Hinduism

1353	Religion and Spirituality::Islam
1354	Religion and Spirituality::Judaism
1355	Religion and Spirituality::Latter-Day Saints
1356	Religion and Spirituality::Pagan/Wiccan

Appendix B: Sample Location Encoding

Following is a sample implementation of the method, in C/Objective-C. Note that the decoding functionality is also included for testing and verification purposes.

```
unsigned char translateEncodedChar(unsigned char c) {
        if (c >= '?') {
        switch (c) {
        case '{':
        return ('0');
        case '}':
        return ('1');
        case '|':
        return ('2');
        case '\\':
        return ('3');
        case '^':
        return ('4');
        case '~':
        return ('5');
        case '[':
        return ('6');
        case ']':
        return ('7');
        case '`':
        return ('8');
        case '@':
        return ('9');
        default:
        return (c);
        }
        }

        #ifdef DEBUG
        switch (c) {
        case '0':
        return ('{');
        case '1':
        return ('}');
        case '2':
        return ('|');
        case '3':
        return ('\\');
        case '4':
        return ('^');
        case '5':
```

```
        return ('~');
        case '6':
        return ('[');
        case '7':
        return (']');
        case '8':
        return ('`');
        case '9':
        return ('@');
        default:
        return (c);
        }
        #else
        return (c);
        #endif
        }
        #ifdef DEBUG

        /*
        * Decode coordinates (testing only).
        */
        + (CLLocation *)decodeLocation:(NSString *)enc {
        int lat = 0;
        int lon = 0;
        const char *cenc = [enc
cStringUsingEncoding:NSASCIIStringEncoding];
        int b;
        int i = 0;
        int shift = 0;
        int result = 0;

        do {
        b = translateEncodedChar(cenc[i++]) - '?';
        result |= (b & 0x1f) << shift;
        shift += 5;
        } while (b >= 0x20);

        lat = (((result & 1) > 0) ? ~(result >> 1) : (result >> 1));

        shift = result = 0;

        do {
        b = translateEncodedChar(cenc[i++]) - '?';
        result |= (b & 0x1f) << shift;
        shift += 5;
        } while (b >= 0x20);

        lon = (((result & 1) > 0) ? ~(result >> 1) : (result >> 1));

        CLLocation *tmp = [[CLLocation alloc]
        initWithLatitude:(lat * 1e-5) longitude:(lon * 1e-5)];

        return ([tmp autorelease]);
        }
        #endif

        /*
        * Encode coordinates roughly according to an algorithm used for
Google Maps.
        */
        + (NSString *)encodeLocation:(CLLocation *)loc {
        if (!loc) return (nil);

        /* Round to 5 decimal places and drop the decimal. */
        int lat = (int)(loc.coordinate.latitude * 1e5);
        int lon = (int)(loc.coordinate.longitude * 1e5);

        /* Shift binary value & invert negatives. */
        if (lat < 0) { lat <<= 1; lat = ~(lat); }
        else lat <<= 1;

        if (lon < 0) { lon <<= 1; lon = ~(lon); }
        else lon <<= 1;

        NSMutableString *tmp = [NSMutableString string];

        while (lat >= 0x20) {
        [tmp appendFormat:@"%c", translateEncodedChar((0x20 | (lat & 0x1f))
+ '?')];
        lat >>= 5;
        }

        [tmp appendFormat:@"%c", translateEncodedChar(lat + '?')];

        while (lon >= 0x20) {
        [tmp appendFormat:@"%c", translateEncodedChar((0x20 | (lon & 0x1f))
+ '?')];
        lon >>= 5;
        }

        [tmp appendFormat:@"%c", translateEncodedChar(lon + '?')];
```

```
           return (tmp);
           }
```

Appendix C: VAST event tracking

When adcel delivers a vastlinear adunit using the /vast endpoint, it will add the following "events" to the VAST creative:

Event tracker urls with the event name ('e') set to the following are added to 'Tracking' elements within the xpath: //Creatives/Creative/Linear/TrackingEvents

Vast event attribute	Event Service event name ('e' value)
start	video0
firstQuartile	video25
midpoint	video50
thirdQuartile	video75
complete	video100
pause	pause
resume	resume
mute	mute
unmute	unmute
acceptInvitation	acceptInvitation
close	close
skip	skip
collapse	collapse
expand	expand
fullscreen	fullscreen
rewind	rewind

An event tracker url with the event name ('imp') is added to the document at the location //Impression

An event tracker url with the event name ('click') is added to the document at the location //Linear/VideoClicks as the value of a new ClickTracking element.

If the ad is retrieved from the DFP Ad Network (currently, the only Ad Network available for vast requests), the "google-delayed-impression" url is added to the document at the location //Impression.

example response:

```
<?xml version="1.0" encoding="UTF-8"?>
<VAST version="2.0">
   <Ad id="49">
      <Wrapper>

<Impression><![CDATA[https://go.vrvm.com/t?e=imp&amp;r=0000012b59c97967dd5
60e1565ede7c4&amp;adnet=4&amp;paid=-10&amp;poid=-5&amp;ui=-555&amp;uis=I&a
```

```
mp;preview=true]]></Impression>
         <Impression><![CDATA[https://go.vrvm.com/h?gdi]]></Impression>
         <AdSystem>Verve AdServer</AdSystem>

<VASTAdTagURI><![CDATA[http://ds.serving-sys.com/BurstingRes/Site-44744/Ty
pe-16/3fe9efcf-1765-462f-b66d-5709b601c520.mp4]]></VASTAdTagURI>
         <Creatives>
            <Creative AdID="49">
               <Linear>
                  <Duration>00:00:30</Duration>
                  <TrackingEvents>
                     <Tracking
event="resume"><![CDATA[https://go.vrvm.com/t?r=0000012b59c97967dd560e1565
ede7c4&amp;e=resume&amp;adnet=4&amp;paid=-10&amp;poid=-5&amp;ui=-555&amp;u
is=I&amp;preview=true]]></Tracking>
                     <Tracking
event="acceptInvitation"><![CDATA[https://go.vrvm.com/t?r=0000012b59c97967
dd560e1565ede7c4&amp;e=acceptInvitation&amp;adnet=4&amp;paid=-10&amp;poid=
-5&amp;ui=-555&amp;uis=I&amp;preview=true]]></Tracking>
                     <Tracking
event="start"><![CDATA[https://go.vrvm.com/t?r=0000012b59c97967dd560e1565e
de7c4&amp;e=video0&amp;adnet=4&amp;paid=-10&amp;poid=-5&amp;ui=-555&amp;ui
s=I&amp;preview=true]]></Tracking>
                     <Tracking
event="thirdQuartile"><![CDATA[https://go.vrvm.com/t?r=0000012b59c97967dd5
60e1565ede7c4&amp;e=video75&amp;adnet=4&amp;paid=-10&amp;poid=-5&amp;ui=-5
55&amp;uis=I&amp;preview=true]]></Tracking>
                     <Tracking
event="mute"><![CDATA[https://go.vrvm.com/t?r=0000012b59c97967dd560e1565ed
e7c4&amp;e=mute&amp;adnet=4&amp;paid=-10&amp;poid=-5&amp;ui=-555&amp;uis=I
&amp;preview=true]]></Tracking>
                     <Tracking
event="skip"><![CDATA[https://go.vrvm.com/t?r=0000012b59c97967dd560e1565ed
e7c4&amp;e=skip&amp;adnet=4&amp;paid=-10&amp;poid=-5&amp;ui=-555&amp;uis=I
&amp;preview=true]]></Tracking>
                     <Tracking
event="pause"><![CDATA[https://go.vrvm.com/t?r=0000012b59c97967dd560e1565e
de7c4&amp;e=pause&amp;adnet=4&amp;paid=-10&amp;poid=-5&amp;ui=-555&amp;uis
=I&amp;preview=true]]></Tracking>
                     <Tracking
event="expand"><![CDATA[https://go.vrvm.com/t?r=0000012b59c97967dd560e1565
ede7c4&amp;e=expand&amp;adnet=4&amp;paid=-10&amp;poid=-5&amp;ui=-555&amp;u
is=I&amp;preview=true]]></Tracking>
                     <Tracking
event="rewind"><![CDATA[https://go.vrvm.com/t?r=0000012b59c97967dd560e1565
ede7c4&amp;e=rewind&amp;adnet=4&amp;paid=-10&amp;poid=-5&amp;ui=-555&amp;u
is=I&amp;preview=true]]></Tracking>
                     <Tracking
event="fullscreen"><![CDATA[https://go.vrvm.com/t?r=0000012b59c97967dd560e
1565ede7c4&amp;e=fullscreen&amp;adnet=4&amp;paid=-10&amp;poid=-5&amp;ui=-5
55&amp;uis=I&amp;preview=true]]></Tracking>
                     <Tracking
event="firstQuartile"><![CDATA[https://go.vrvm.com/t?r=0000012b59c97967dd5
```

```
60e1565ede7c4&amp;e=video25&amp;adnet=4&amp;paid=-10&amp;poid=-5&amp;ui=-5
55&amp;uis=I&amp;preview=true]]></Tracking>
                     <Tracking
event="midpoint"><![CDATA[https://go.vrvm.com/t?r=0000012b59c97967dd560e15
65ede7c4&amp;e=video50&amp;adnet=4&amp;paid=-10&amp;poid=-5&amp;ui=-555&am
p;uis=I&amp;preview=true]]></Tracking>
                     <Tracking
event="unmute"><![CDATA[https://go.vrvm.com/t?r=0000012b59c97967dd560e1565
ede7c4&amp;e=unmute&amp;adnet=4&amp;paid=-10&amp;poid=-5&amp;ui=-555&amp;u
is=I&amp;preview=true]]></Tracking>
                     <Tracking
event="complete"><![CDATA[https://go.vrvm.com/t?r=0000012b59c97967dd560e15
65ede7c4&amp;e=video100&amp;adnet=4&amp;paid=-10&amp;poid=-5&amp;ui=-555&a
mp;uis=I&amp;preview=true]]></Tracking>
                     <Tracking
event="close"><![CDATA[https://go.vrvm.com/t?r=0000012b59c97967dd560e1565e
de7c4&amp;e=close&amp;adnet=4&amp;paid=-10&amp;poid=-5&amp;ui=-555&amp;uis
=I&amp;preview=true]]></Tracking>
                     <Tracking
event="collapse"><![CDATA[https://go.vrvm.com/t?r=0000012b59c97967dd560e15
65ede7c4&amp;e=collapse&amp;adnet=4&amp;paid=-10&amp;poid=-5&amp;ui=-555&a
mp;uis=I&amp;preview=true]]></Tracking>
                  </TrackingEvents>
                  <VideoClicks>

<ClickTracking><![CDATA[https://adclick.g.doubleclick.net/aclk?sa]]></Clic
kTracking>

<ClickTracking><![CDATA[https://go.vrvm.com/t?e=click&amp;r=0000012b59c979
67dd560e1565ede7c4&amp;adnet=4&amp;paid=-10&amp;poid=-5&amp;ui=-555&amp;ui
s=I&amp;preview=true]]></ClickTracking>
                  </VideoClicks>
               </Linear>
            </Creative>
         </Creatives>
```

```
      </Wrapper>
   </Ad>
</VAST>
```

this example response is based upon the input creative:

```
<VAST version="2.0">
  <Ad id="49">
    <Wrapper>
      <AdSystem>Verve AdServer</AdSystem>

<VASTAdTagURI><![CDATA[http://ds.serving-sys.com/BurstingRes/Site-44744/Ty
pe-16/3fe9efcf-1765-462f-b66d-5709b601c520.mp4]]></VASTAdTagURI>
      <Creatives>
        <Creative AdID="49">
          <Linear>
            <Duration>00:00:30</Duration>
            <TrackingEvents/>
            <VideoClicks>

<ClickTracking><![CDATA[%%CLICK_URL_UNESC%%]]></ClickTracking>
            </VideoClicks>
          </Linear>
        </Creative>
      </Creatives>
    </Wrapper>
  </Ad>
</VAST>
```