

Google Analytics for the Kaltura Player

Overview

Google Analytics (GA) is a web analytics service that uses a combination of cookies and APIs to enter events into a tracking site. Kaltura player includes a plugin to easily translate player events into GA beacons. The collection of beacons can then provide an analysis of player actions.

✓ Google has announced that version 3 of Google Analytics will sunset on July 1, 2023, and will be replaced with version 4. See reference [here](#). Kaltura V7 player is fully supporting Google Analytics 4 (GA4). See Github repo's <https://github.com/kaltura/playkit-js-google-analytics/blob/master/docs/configuration.md>.

For the updated article, visit [Google Analytics 4 for the Kaltura Video Portal](#).

Configure

To use this plugin, you must first have a Google Analytics account (<http://www.google.com/analytics/>).

After signing up, you'll receive an urchin ID (aka Tracking ID), appearing similar to UA-XXXXXXXX-1. This ID is the link between your player and the GA servers. After you install and configure Google Analytics in the player, events may be reported to your Google Analytics account and will be collected under the Content->Events page.

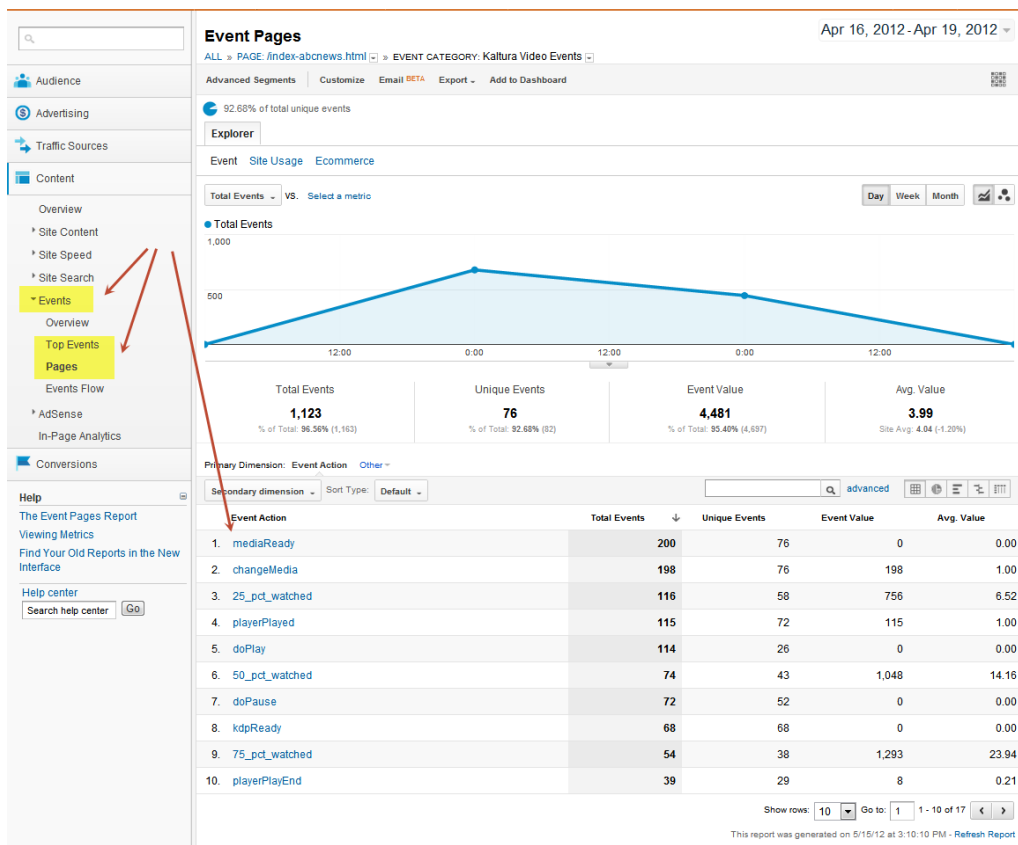
To drill down into events: Go to Content->Events->Pages.

Under each page, the Kaltura Events category is displayed with the relevant events. Under each event, the label is in the form:

Clip title | entry Id | widget Id

where each line shows the number of times the event was received, with the exception of progress events (xx_pct_watched).

See [Implementation](#). The following is a screen capture of the Google Analytics Events Page.



Demo Page

A player with the Google Analytics plugin installed is showcased on the following page:
<http://player.kaltura.com/docs/GoogleAnalytics>

You can use Fiddler or the onPage tracking tool, to track the events sent to Google.

Parameter	Value
utmwv	4.3as
utmn	609571541
utmhn	cdnbakmi.kaltura.com
utmt	event
utme	5(Kaltura Video Events*playerPlayed*Sample Big Buck Bunny Trailer (HD'1 0_p1s6zc0m _626742)(1)
utmcs	UTF-8
utmsr	1280x800
utmsc	24-bit
utmul	en-us
utmje	0
utmfl	11.0 r1
utmhid	1617619397
utmr	-
utmp	/nu/wwwGAanalytics/embedCode.html
utmac	UA-30149691-1
utmcc	__utma=86858442.1890899863462500600.1332371291.1332377425.1332485054.12;+__utmz=86858442....

Contents are broken up as:

(["Kaltura Video Events"])[type of event]["Clip title | entry Id | widget Id"])([value])

Technical Details

Compiled against: KDP 3.5.43	
Release file size(s): 49kb	
UIConf path: /content/uiconf/google_analytics/kdp3.5.43/plugins/googleAnalyticsPlugin2.swf	
Code paths: kdp/trunk/plugins/googleAnalyticsPlugin	

Google Analytics Information

Google documentation can be found at:

<http://code.google.com/apis/analytics/docs/tracking/eventTrackerGuide.html#Anatomy>

A Google Analytics event has the following specifications:

- Category – in our case – “Kaltura Video Events” (hard coded)
- Action – the event name
- Label (optional) – dimension for the action
- Value (optional) – numerical data

Event Details

Data is recorded as listed:

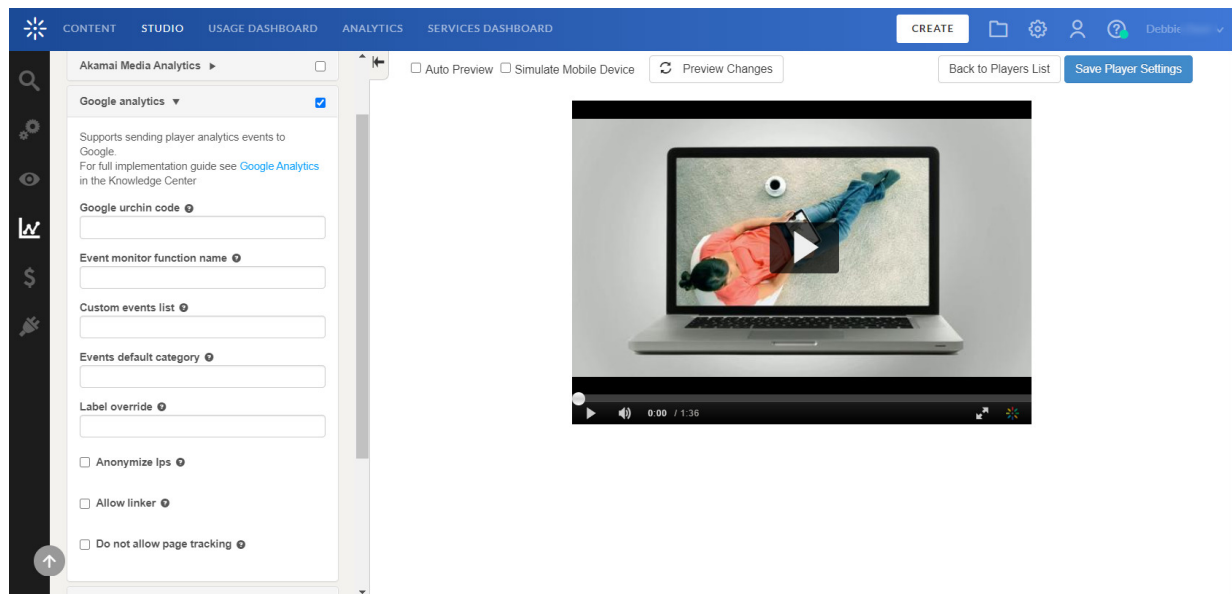
- Label – is always the title of the video, coupled with the entry ID and player ID:
video title | entryID | widgetId (also known as the partner id)
The use of the vertical separator “|” is for easy readability.
- Category – is always “Kaltura Events”
- Action – the event type (as in the player event name)
- Value – usually 1, for events where it makes sense to count the aggregate amount of times the event was triggered.

Add Google Analytics with the KMC

You can add Google Analytics using the KMC Studio, by selecting the Google Analytics plugin within the KMC.

1. Select the Studio tab and then select the Universal Studio menu.
2. Select a Player.

3. Select the Analytics icon and then Select Google Analytics. For more information see the [Universal Studio Information Guide](#).



Plugin Definition

The following is a sample plugin configuration that can be copied to your player's uiConf. Change the values in blue to accommodate the events to your own setup.

```
<Plugin id="googleAnalytics" visualDebug="false" width="0%" height="0%"
loadingPolicy="wait" urchinCode="UA-XXXXXXX-1"/>
```

This configuration beams the following events:

Action	Value (integer)	Comments
Play	1	Actually viewing the file
OpenedFullScreen	1	Indicates more interest
CloseFullScreen	1	
kdpReady		
playerEmpty		For cases like MediaSpace where the media is loaded after the player by external javascript
mediaReady		Google time stamp will allow detecting problems (slow load)
changeMedia	1	Track “related” and similar video changes.
doDownload	1	
doGigya	1	

25_pct_watched	(current clip time)	Dispatch after the player reaches the 25% mark of the clip.
50_pct_watched	(current clip time)	Dispatch after the player reaches the 50% mark of the clip.
75_pct_watched	(current clip time)	Dispatch after the player reaches the 75% mark of the clip.
100_pct_watched	(current clip time)	Dispatch after the player reaches the 100% mark of the clip.
doPause		

Advanced Implementation – Custom Events

Example 1

You can add the ability to listen for additional events that are not listed in the table. You can also specify your own values in the beacon string:

`(["Kaltura Video Events"][type of event]["Clip title | entry Id | widget Id"])([value])`

where:

<code>["Kaltura Video Events"]</code>	Category
<code>[type of event]</code>	Action
<code>["Clip title entry Id widget Id"]</code>	Label
<code>[value]</code>	Value

To add additional events you will need to identify the event and set each definition. Omitting any of the definitions will result in the plugin's default settings. You can use

this capability to listen for events triggered by other plugins, and beam those events to Google Analytics.

Example 2

The following is a configuration example that listens for additional events such as “doPlay”.

```
<Plugin id="googleAnalytics" visualDebug="false" path="googleAnalyticsPlugin.swf"
customEvent="doPlay"
doPlayCategory="My Custom event"
doPlayAction="player is playing"
doPlayValue="1"
width="0%" height="0%" loadingPolicy="wait" urchinCode="UA-XXXXXXX-1"/>
```

The configuration above sends doPlay events each time a play event occurs. A play event happens after each click of the Play button. Notice the doPlayLabel parameter was left out. Leaving the doPlayLabel parameter out will default to the plugin's own settings and change the other parameters. The following is what each sent beacon will look like after each play event.

```
("My Custom event" "player is playing" ["Clip title | entry Id | widget Id"])(“1”)
```

["Clip title | entry Id | widget Id"] will display the Clip's title, entry Id, and widget Id in the above beacon.

Example 3

Users can omit a Category, Action, Value, and Label and only declare the event in “customEvents” as in the following example.

```
<Plugin id="googleAnalytics" visualDebug="false" path="googleAnalyticsPlugin.swf"
customEvent="doPlay" width="0%" height="0%" loadingPolicy="wait"
urchinCode="UA-30149691-1"/>
```

This custom event sends all the plugin's default settings each time the doPlay action is triggered. Multiple custom events are separated by commas as in the following:

```
<Plugin id="googleAnalytics" visualDebug="false" path="googleAnalyticsPlugin.swf"
customEvent="doPlay,playerStateChange,addThis"
```

```
addThisCategory="My AddThis Category"
```

```
addThisAction="My AddThis Action"
```

```
addThisLabel="My AddThis Label"
```

```
addThisValue="1"
```

```
width="0%" height="0%" loadingPolicy="wait" urchinCode="UA-30149691-1"/>
```

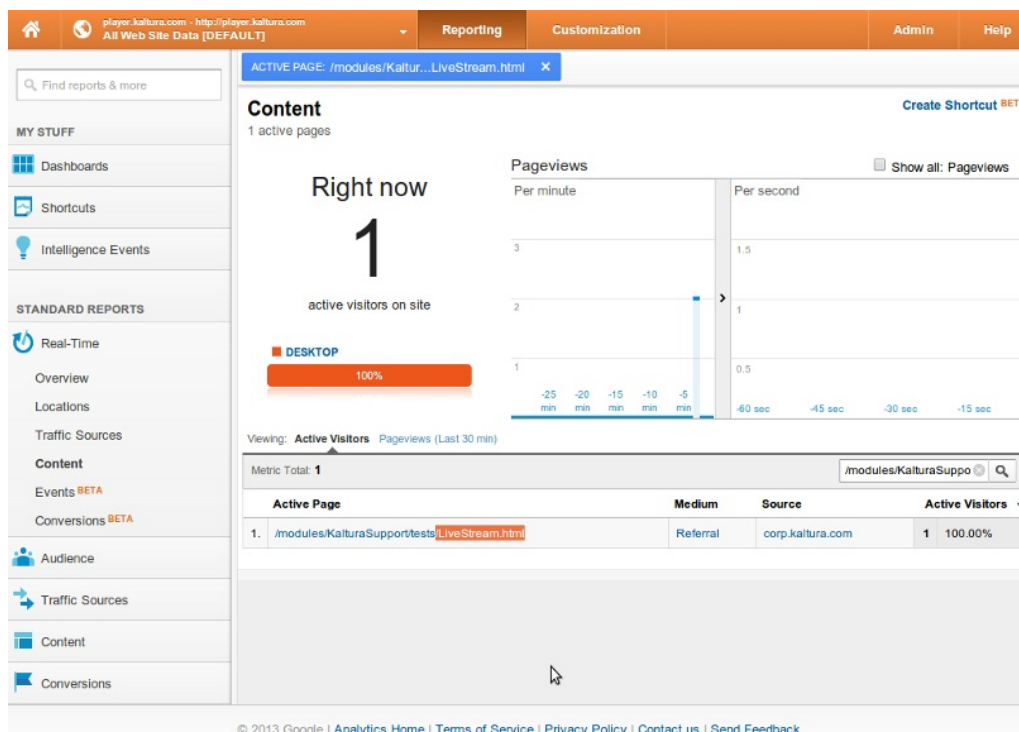
Using the Google Analytics Interface

Live Reports on Active Viewers

Google has added support for realtime analytics, enabling you to list the active users on a page. This can be used for real-time estimates of live stream viewers.

1. Configure Google analytics for your player.
2. Embed your live entry with your GA enabled player.
3. Navigate to the "real-time" section within your Google Analytics console.
4. Locate the content page with the live stream.

You can now monitor real-time visitors for your live stream video.

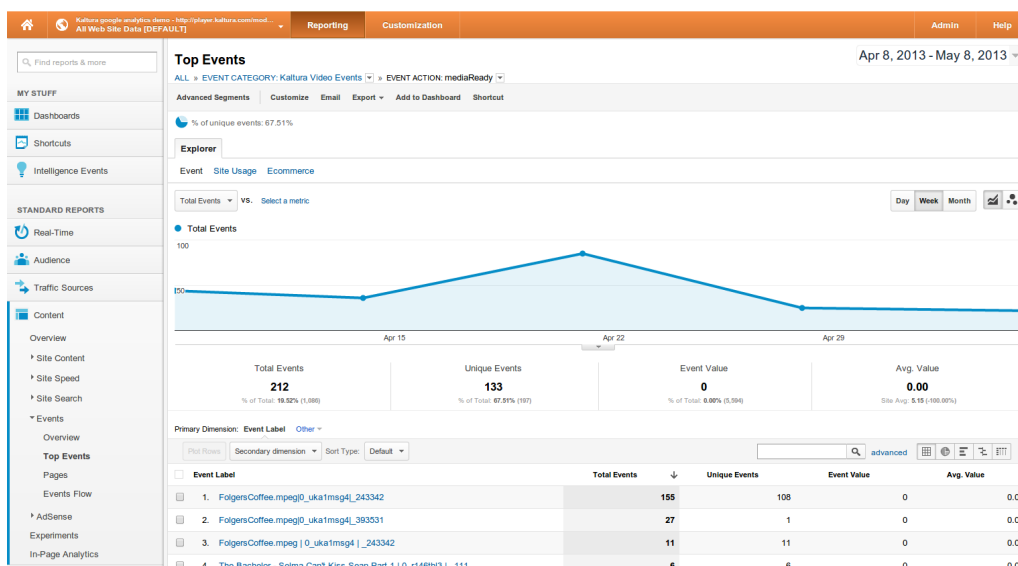


Accessing Per Content Entry Reports

Google Analytics collects details reports on a per entry basis. This enables you to create

segment views per content per event. For example if you wanted to see how many users reached the "playerPlayEnd" event (i.e completed a video view):

1. Configure and embed a player with google analytics.
2. Within the google console select "Content"
3. Select the "Events" pull down
4. Click on "Top Events" then "Kaltura Video Events" (for the default kaltura player events)
5. Click on the "Event Action" your interested in i.e "mediaReady" or "75_pct_watched"
6. Enter the **entryId** your interested in tracking within the search field.
7. Here we select mediaReady, we then get the number of times the mediaReady event was called for that entry.



Accessing Per Live Content Entry Reports

Google analytics content reports for live entries can be handled very similar to VOD entries. Just be sure to make note of the specific **entryId** that your interested in tracking.

1. Configure and embed a player with Google Analytics.
2. Within the google console select "Content"
3. Select the "Events" pull down
4. Click on "Top Events" then "Kaltura Video Events" (for the default kaltura player events)
5. Click on the "Event Action" your interested in i.e "mediaReady" or "75_pct_watched"
6. Enter the **live entryId** in the search field.
7. Here we select mediaReady, we then get segments by media labels. This gives details on how many player impressions where rendered to clients. The mediaReady

event is fired once the player is displayed on page.

```
[template("cat-subscribe")]
```
