



Documentation

Documentation Home	6
Unify UI	7
Audience Pulse	8
Households and Individuals	9
Devices and Audiences	10
Platform Mix	11
Most and Least Connected Platforms	12
Revenue Opportunity	13
Platform IDs	14
Dashboard Filters, Devices, and Data	15
How Confidence Scores Affect Report Results	16
On-site and Off-site Data: About	19
Counting on-site and off-site devices	20
Campaign Pulse	23
Delivery	24
Cross Platform Lift	26
Reach and Frequency	27
Pathing	28
Path Splitting Rules for Campaign Pulse	30
Path Truncation Rules for Campaign Pulse	32
Audience Insight	36
CPC and CPCV Audience Score Equations and Variables	39
CPA Audience Score Equations and Variables	41
Third-party Ad Server (3PAS) Attribution	44
Creative Management	46
Dataflow Management	47
Create a Data Flow	48
Select Data to Include	49
Filter By Synced Devices	51
Dataflow Metrics	52
Primary ID Analysis	53
Connections Analysis: Cookie IDs	55
Connections Analysis: Hardware IDs	56
Related ID Analysis	57
Supplier ID Analysis	58
ID Syncs Management	59
Supplier Health Dashboard	62
Data Visualization in the Supplier Health Dashboard	63
Supplier Health Dashboard: Available Data	64
Budgets, Spending, and Pacing	65
Understanding Budget Distribution and Update Cycles	66
Budget Distribution Between and Within Data Centers	68
How the Budget Update Process Can Delay Tactic Delivery	69
Pacing: Everything You Wanted to Know but Were Afraid to Ask	70
Marketplace Buyer	73
Programmatic Real Time Bidding Access	74
Programmatic RTB Access Introduction	75
Tapad's Device Graph	76
RTB Access Integration Process	77
Technical Specifications for Programmatic RTB Access	79
Bid Request Object	80
Bid Request Format Example	92
Bid Response Object	95
Bid Response Format Example	101
Creative Formats and Creative Approval API	102
Creative Approval API V1.2	103
Programmatic RTB Access Version and Revisions	114
Data Integrations	115
Analytics Plug-in	116
Tapad - Adobe Analytics (SiteCatalyst) Data Integration	117
Tapad - Adobe Analytics Data Integration: About	118
Requirements and Implementation Questionnaire for Adobe Analytics	119
Tapad Parameters Available in Adobe Analytics	120
Deactivate a Tapad - Adobe Data Connection	121
Integration Code Sample for Adobe Analytics	122
Tapad and Adobe Analytics Integration Guides	124
Tapad - Google Analytics Data Integration	126
Tapad - Google Analytics Integration: About	127
Data Collection With Google's Classic Analytics	128
Data Collection With Google's Universal Analytics	131
Tapad Parameters in Google Analytics: Definitions and Syntax	134
Viewing Report Data in Google Analytics	135

Tapad - Google Analytics Integration Guide	136
Analytics Plug-in SDKs	137
Android Plug-in Guide for the Analytics SDK	138
iOS Plug-in Guide for the Analytics SDK	141
Analytics Plug-in PDF	145
Analytics Plug-in Metrics Defined	146
Visited Platforms	147
Platforms Associated	149
Platform Types	152
Platforms Associated and Visited Platforms Compared	154
First Visited Platform	155
Most Often Visited	156
Most Recent Visited	158
Data Flow: Send Audiences To Tapad	159
Audience Data File	160
JSON Format for Audience Data Files	161
Name and Update Your Audience Data Files	162
Text Format for Audience Data Files (Not Used)	163
Audience Taxonomy File	164
JSON Format for Audience Taxonomy Files	165
Name and Update Your Audience Taxonomy File	166
Text Format for Audience Taxonomy Files (Not Used)	167
Delivery Methods for Audience Data and Taxonomy Files	168
Audience Data File PDF	169
Data Flow: Receive Connected IDs From Tapad	170
JSON Data Files: Syntax and Content Options	171
Basic JSON Data	172
JSON Data With Platform Types	173
JSON Data With Scores	174
Text Data Files: Syntax and Content Options	175
Basic Text Data File	176
Text Data File With Platform Types	177
Text Data File With Scores	179
Content Separators and Delimiters for Text Data Files	181
Data File Delivery Methods and Filter Options	183
Standard Delivery Methods	184
Enhanced Delivery Methods	185
Filter Options	186
Data Flow Appendix	188
Data Flow PDFs	189
Mapping Primary and Related IDs	195
Mapping Primary and Related IDs PDF	198
Understanding the ID Mapping File	199
ID Mapping File Metrics Defined	200
Sample ID Mapping File	201
The Tapad User Opt-out Process: About	202
Tapad Opt-out Overview	203
Batch File Opt-out Process: About	204
Real-time Opt-out Process: About	205
Opt-out ID types Defined	206
Opt-out PDFs	207
Technical Specifications for the Impression Log File	208
Impression Log File PDF	210
Audience and User ID Synchronization	211
Partner Initiated ID Synchronization Code: Syntax, Parameters, and Sample Request	212
Tapad Initiated ID Synchronization Code: Syntax, Parameters, and Sample Request	214
User ID Synchronization Methods	215
Audience Metadata File	217
Getting Started Questionnaire	218
Using Tapad as a DMP	219
Tapad as a DMP: Overview and Requirements	220
DMP Code Placement and Configuration	221
Tapad DMP Code Sample	222
AudienceManager Integration Guide	223
Tapad - Adobe AudienceManager Integration Details	224
AudienceManager Configuration Process	227
Send AudienceManager segments to Tapad	228
Set up Tapad as an FTP destination in AudienceManager	229
Set up Tapad as a real-time HTTP destination in AudienceManager	231
BlueKai Integration Guide	233
Overview and Getting Started	234
Debugging	235

BlueKai Integration Details	236
Lotame Data Integration Guide	238
Lotame Integration Details	239
Turn Integration Overview	241
Technical Specifications for ID Bridging	242
Web ID Bridging	243
Batch File ID Bridging	244
Mobile SDK ID Bridging	245
Audience Amplify	246
Audience Amplify: About	247
Data File Formats for Audience Amplify	248
Standard Format for Audience Amplify Data Files	249
AppNexus DSP Format for Audience Amplify Data Files	250
Delivery Methods	251
Audience Amplify Appendix	252
Device ID Types	253
Platform Types List	254
Feature Phones	255
Audience Amplify PDF	256
Cross-plan Analytics	257
Cross-plan Analytics: About	258
Cross-plan Analytics: Available Reports	259
Set Up Cross-plan Analytics	261
Set up the Universal SDK	262
Set up conversion tracking	263
Mobile ads: Set up impression and click tracking	264
Mobile video: Set up impression, quartile, and click tracking	265
Device Graph	266
Device Graph Basics	267
Household Scores	269
Individual Scores	271
Understanding Depth	274
Merge Logic	275
Merging Conditions	276
Non-merging Conditions	278
Replacing and Appending IDs	280
Device ID Superclusters	283
Sources of ID Superclusters	284
Filtering ID Superclusters	285
A Field Guide to ID Supercluster Structures	286
Device Identifiers	288
IDs Used for Attribution Between Devices and Apps	290
Device Graph Methodology	291
Device Graph Data Structures	300
SDK and Server-to-Server Data Collection	309
SDK Data Collection	310
Standard SDK: About	311
Universal SDK: About	312
Set up the Android SDK	314
Set up the iOS SDK	316
Set Up Impression and Click Tracking	318
Endpoint Configuration File: Requirements	319
Server-to-server data collection	320
Server-to-Server Data Transfers: Technical Specifications	321
Device ID Collection with typed_device_id	322
Android and iOS Device IDs	323
Reference	324
Device IDs	325
Glossary	327
A	330
B	332
C	333
D	335
E	337
F	338
G	339
H	340
I	341
K	342
L	343
M	344
N	345

O	346
P	347
R	349
S	350
T	351
U	352
V	353
W	354
Z	355
IAB Real-time Bidding Specifications	356
JIRA - Confluence Linking	357
Add Confluence Page Links to a JIRA Issue	358
Add JIRA Issues to a Confluence Page	359
Linked JIRA Issues: Examples	361
Key-Value Pairs: About	362
Mobile Device Bid Request	363
Publisher ID: How to find it	367
Request a Deal ID	368
Tapad Rate Cards	369
Tapad Rate Card I Q3 2014	370
Tapad Rate Card I Q2 2014	374
Tapad Rate Card I Q1 2014	378
Tapad Rate Card I Q4 2013	382
Understanding User Agents	383
Whitelisting Steps and Site/App Evaluation Guide	385
How to Whitelist a Website or App	386
Whitelist/Blacklist Evaluation Criteria for Sites and Apps	388
Maturity Rating	391
Index	392

Documentation Home

Tapad documentation

Documentation contains concept, task, and reference material on how Tapad products, systems, and services work or should be used. Search or browse the contents of this space for help with concepts, tasks, and reference material for Backoffice, Unify, the Tapestry or TAJ APIs, data integrations, and related material. For additional information about our products and services visit the [website](#) or browse the [other Confluence spaces](#).

Contents

- [APIs](#) — Descriptions, methods, and code samples for Tapestry, TAJ, and the UI APIs along with SDK code for Android and iOS devices.
- [Backoffice UI](#) — Concepts, tasks, and reference material for our internal, managed service UI.
- [Unify UI](#) — Concepts, tasks, and reference material for Unify UI features, tools, and processes.
- [Budgets, Spending, and Pacing](#) — A detailed examination of how Tapad systems and software manage budgets, spending, and pacing processes.
- [Marketplace Buyer](#) — Products, features, and related material.
- [Data Integrations](#) — Send data to Tapad from other products for cross-device targeting and retargeting.
- [Cross-plan Analytics](#) — Overview, reports, and set up instructions.
- [Device Graph](#) — Overview and functionality.
- [SDK and Server-to-Server Data Collection](#) — Overview and set up guides for collecting data with our software development kit (SDK) and server-to-server data transfers.
- [Reference](#) — General information, terms, useful facts, tables, and other data.
- [Index](#) — An alphabetical list of articles by title.

Search

New and revised

Featured

- | | |
|---|---|
|  Delivery Methods
• updated yesterday at 9:04 PM
<small>↳ Returns search results for the Documentation space only.
View change</small> |
• zz_inclusions library (Documentation) |
|  Delivery Methods
• updated Mar 29, 2015 • view change | |
|  Standard Format for Audience Amplify Data Files
• updated Mar 29, 2015 • view change | |
|  Audience Amplify: About
• updated Mar 29, 2015 • view change | |
|  Reallocate Budgets to Efficient Tactics
• updated Mar 27, 2015 • view change | |

Unify UI

Concepts, tasks, and reference material for Unify UI features, tools, and processes.

- **Audience Pulse** — Audience Pulse is an analytics dashboard that provides insights about data overlap, the size of household and individual audiences, connectivity, reach, revenue, and other data. The dashboard uses unique data visualization techniques that help you spot trends across different combinations of metrics.
- **Campaign Pulse** — Campaign Pulse is an analytics dashboard that provides insights derived from media delivery and the Device Graph. Rather than returning numeric data in columns and rows, this dashboard uses unique visualization techniques that help you spot trends across combinations of audiences, devices, and media delivery. Dashboard tiles return data about cross-device ad delivery, lift, reach/frequency, conversion paths, and improved impression/conversion attribution reporting for devices that don't use or a
- **Creative Management** — All Unify docs about creatives have moved to the Creative Management section in the Self-Service DSP space.
- **Dataflow Management** — Dataflow Management provides features and workflows that let you set up and manage your data flows. Refer to this section for help with tasks like creating data flows and for feature settings, descriptions, and definitions.
- **ID Syncs Management** — ID Syncs Management provides settings that let you start ID syncs, configure storage, set up redirects, specify endpoint URLs, and other options. To manage ID synchronization, go to ID Syncs Management, use the search feature to find your advertiser, and configure required settings in the following sections.
- **Supplier Health Dashboard** — Monitor data for bids, impressions, CPM, CTR, margin %, and the % of bids with a device ID. The Dashboard displays data in near real time and generates an alert if performance falls below or above a predicted range for the supplier.

Audience Pulse

DRAFT

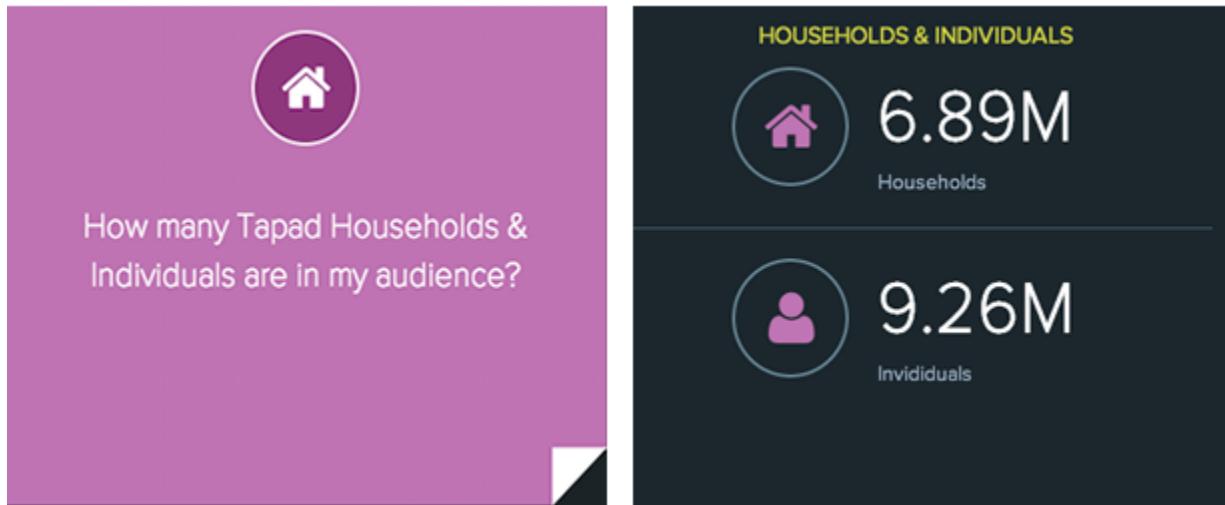
Audience Pulse is an analytics dashboard that provides insights about data overlap, the size of household and individual audiences, connectivity, reach, revenue, and other data. The dashboard uses unique data visualization techniques that help you spot trends across different combinations of metrics.

The Audience Pulse dashboard includes the following tiles.

- **Households and Individuals** — Click to see the number of household and individual groups for your selected audience and confidence score. A single household or individual group can contain 1 (or more) devices depending on how you limit data with the Audience and Confidence Score filters. To help display large numbers, this tile truncates billions with a B and millions with an M.
- **Devices and Audiences** — Click to see how many devices Tapad can add to an on-site or off-site audience at a selected confidence score. This tile shows device % and totals in a doughnut chart. Device types include mobile, tablets, desktops. To help display large numbers, this tile truncates billions with a B and millions with an M.
- **Platform Mix** — Click to see the % of devices and device combinations for the selected audience or confidence score. This tile displays results in a Venn Diagram. The overlap between circles shows you how many household or individual groups share connections between multiple devices. Devices include mobile, tablet, and desktop. Note, the Venn Diagram is not drawn to scale.
- **Most and Least Connected Platforms** — Click to see the device types that have the most and least connections for your selected audience and confidence score.
- **Revenue Opportunity** — Click to see potential revenue (before and after margin) for an on-site or off-site audience. This report lets you adjust CPM, unique impressions, frequency caps, and margin % to help you discover the optimum revenue settings. Calculations are based on total number of devices in the Devices and Audiences tile. To help display large numbers, this tile truncates billions with a B and millions with an M.
- **Platform IDs** — Click to see audience composition by device hardware or cookie type for the selected platform. Platform options include mobile, desktop, tablet, or other devices.
- **Dashboard Filters, Devices, and Data** — Audience Pulse dashboard filters let you limit results by audience, Confidence Score, and device types.
- **How Confidence Scores Affect Report Results** — Most devices in the Device Graph share a "connection" with another device. The data linking groups of devices together is called a connected component. Connected component data includes a household score (H) and an individual score (I). Collectively, these values are known as confidence scores. Confidence scores define the strength of the relationship between each device. Confidence score filtering is a common component of many Unify features (e.g., Tapad Pulse). When you raise or lower the val
- **On-site and Off-site Data: About** — Definitions and examples of on-site and off-site data and how Tapad counts devices in these categories.
- **Counting on-site and off-site devices** — To help you understand how Tapad counts on and off-site devices let's look at 2 use cases. In these examples, we'll use a hypothetical automotive website seen by a visitor who has 3 devices (A, B, and C) linked together in the Device Graph. The important thing to note is how the on-site and off-site counts change as this user visits the automotive website with different devices.

Households and Individuals

Click to see the number of household and individual *groups* for your selected audience and confidence score. A single household or individual group can contain 1 (or more) devices depending on how you limit data with the **Audience** and **Confidence Score** filters. To help display large numbers, this tile truncates billions with a B and millions with an M.



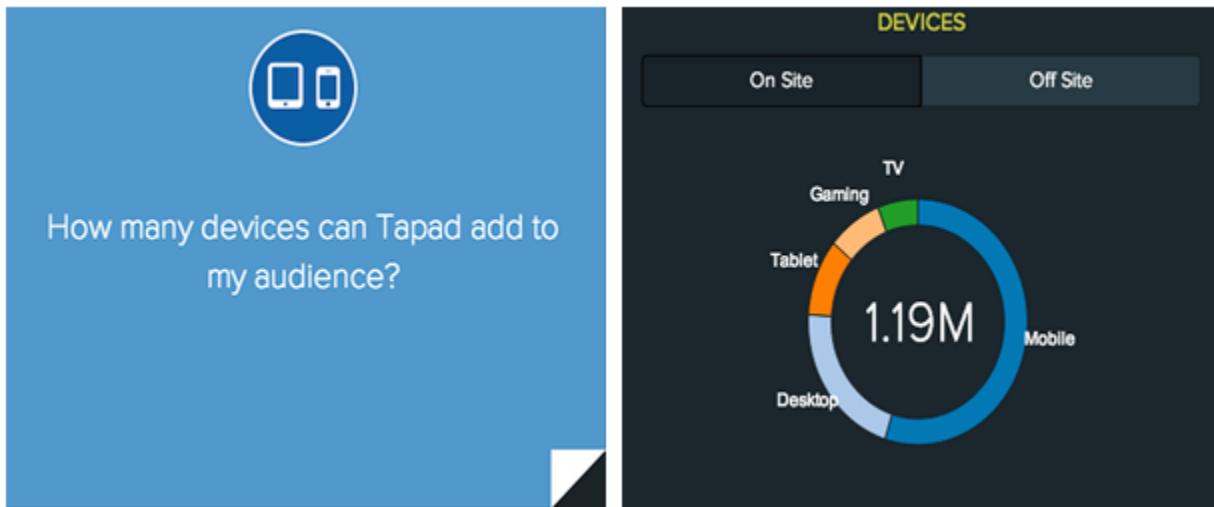
Tile results and confidence scores

See [How Confidence Scores Affect Report Results](#) for more information.

Devices and Audiences

DRAFT

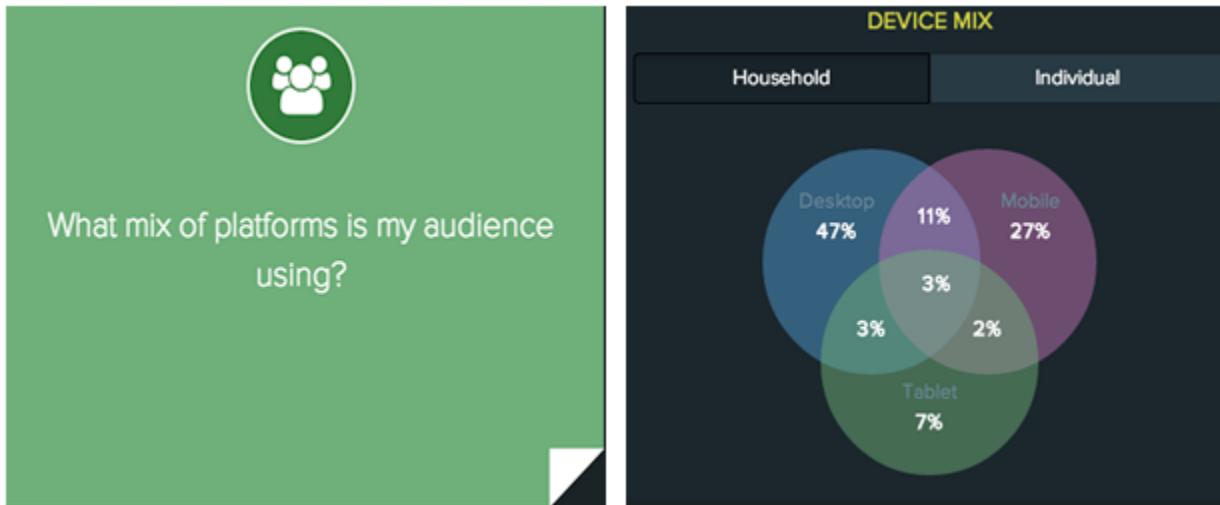
Click to see how many devices Tapad can add to an on-site or off-site audience at a selected confidence score. This tile shows device % and totals in a doughnut chart. Device types include mobile, tablets, desktops. To help display large numbers, this tile truncates billions with a B and millions with an M.



For information about the **On Site** and **Off Site** options, see [Understanding On-Site and Off-Site Data](#).

Platform Mix

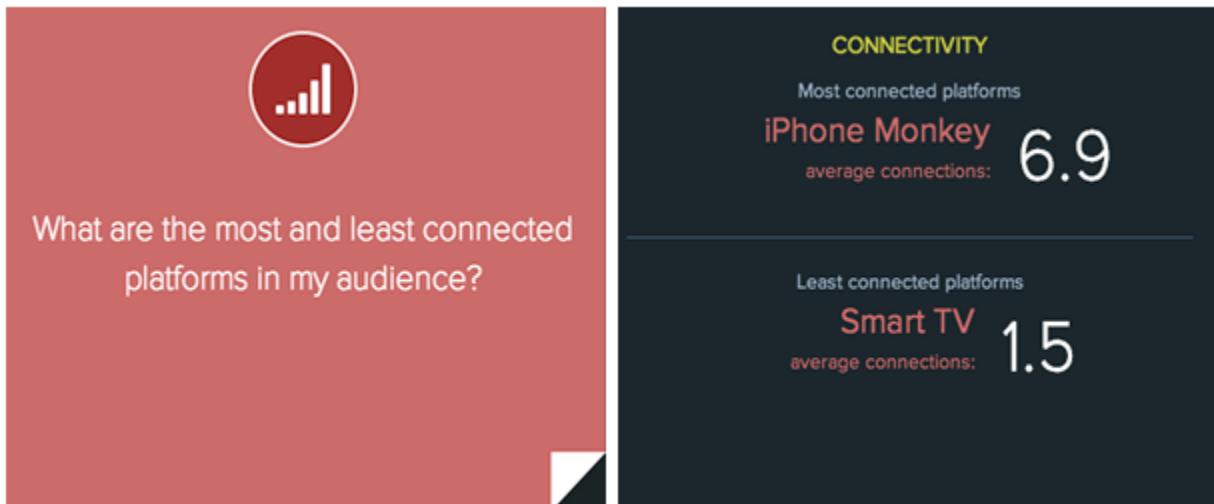
Click to see the % of devices and device combinations for the selected audience or confidence score. This tile displays results in a Venn Diagram. The overlap between circles shows you how many household or individual groups share connections between multiple devices. Devices include mobile, tablet, and desktop. Note, the Venn Diagram is not drawn to scale.



For more information about the **Household** and **Individual** filters, see [Households and Individuals](#).

Most and Least Connected Platforms

Click to see the device types that have the most and least connections for your selected audience and confidence score.

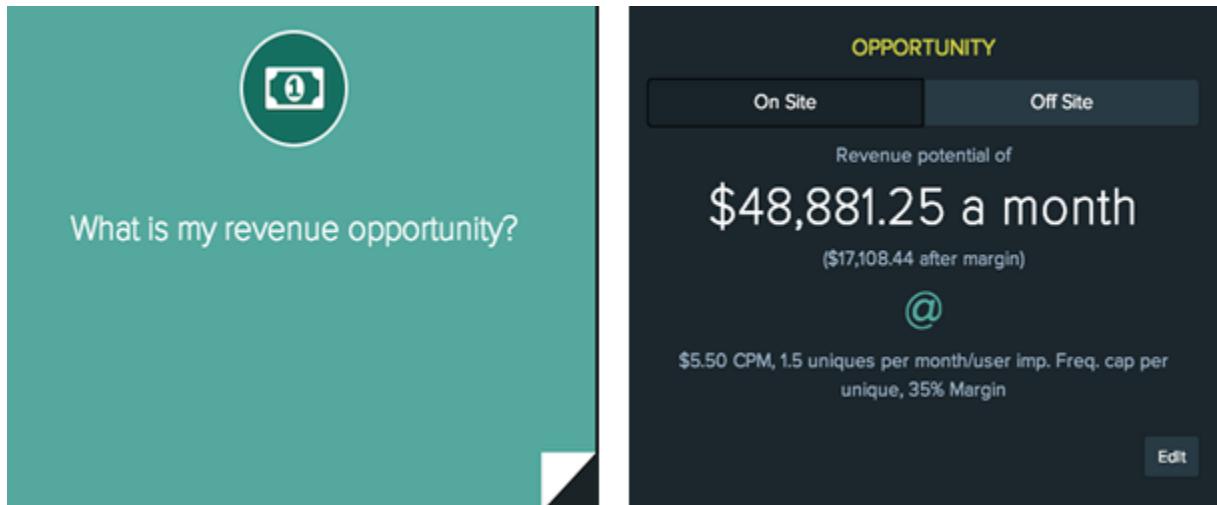


Tile results and confidence scores

See [How Confidence Scores Affect Report Results](#) for more information.

Revenue Opportunity

Click to see potential revenue (before and after margin) for an on-site or off-site audience. This report lets you adjust CPM, unique impressions, frequency caps, and margin % to help you discover the optimum revenue settings. Calculations are based on total number of devices in the Devices and Audiences tile. To help display large numbers, this tile truncates billions with a B and millions with an M.



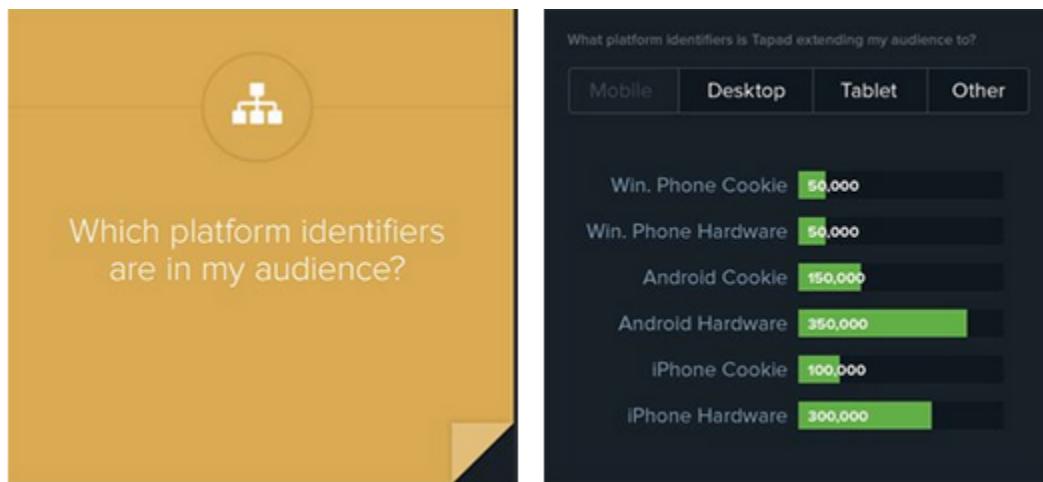
For information about the **On Site** and **Off Site** options, see [Understanding On-Site and Off-Site Data](#).

Platform IDs

Click to see audience composition by device hardware or cookie type for the selected platform. Platform options include mobile, desktop, tablet, or other devices.

Tile details

This illustration shows the Audience Platform tile in the Ask Tapad dashboard.



Platform, hardware, and cookie types

Platform	Returns hardware or cookie data for
Computer	Desktop computers
Mobile	<ul style="list-style-type: none">Android phonesiPhonesOther phones (Blackberry, Symbian, etc.)Windows phones
Other devices	<ul style="list-style-type: none">PlaystationSmart TVsWiiXbox
Tablet	<ul style="list-style-type: none">Android tabletsiPadsOther tablets (Kindle, Kindle Fire, etc.)Windows tablets

Dashboard Filters, Devices, and Data

Audience Pulse dashboard filters let you limit results by audience, Confidence Score, and device types.

Data filters

Dashboard filters let you limit results with the **Audience** and **Confidence Score** options. These filters help you control the amount of information returned by each tile. Use the:

- **Audience filter** to return data in the all audiences group or by selected audiences. An audience is a group of individuals or households that share common attributes. By default, all of your devices belong to the all audiences group. This group is used by on-site filters in other dashboard tiles.
- **Confidence Score filter** to return data based on the strength of the relationship between connected devices in the Device Graph. Higher scores suggest that devices are more likely to be related to each other than devices with lower scores. Confidence scores range from 1-10. The dashboard returns data for values \geq the selected score.

Platforms, devices, and on-site/off-site data

In addition to the audience and Confidence Score filters, dashboard tiles can contain other device or data filters. These include:

- **Platforms:** High level hardware categories like mobile, tablet, desktop, smart TV, game console, etc.
- **Devices:** Specific types of platforms like iPhones, iPads, Android devices, Xbox, Wii, etc.
- **On-site/off-site data:** See [Understanding On-Site and Off-Site Data](#).

How Confidence Scores Affect Report Results

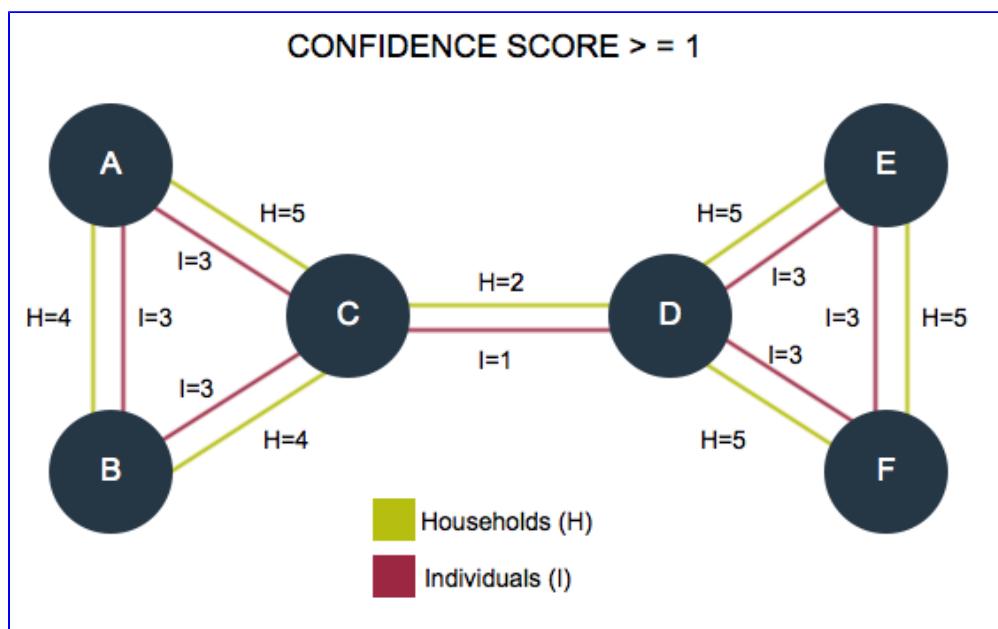
Most devices in the Device Graph share a "connection" with another device. The data linking groups of devices together is called a *connected component*. Connected component data includes a household score (H) and an individual score (I). Collectively, these values are known as *confidence scores*. Confidence scores define the strength of the relationship between each device. Confidence score filtering is a common component of many Unify features (e.g., Tapad Pulse). When you raise or lower the value in a Confidence Score option, this action increases or decreases the results returned by that feature. Keep in mind, however, that these scores and connections are abstract representations of different data points. They are not physical links.

How confidence scores work

To help you understand how these scores affect your results, let's take a look at a few examples using the Households and Individuals tile in the Tapad Pulse dashboard. The following diagrams represent connected devices in the Device Graph. In each graph, a circle represents a device and the lines between each device represent the household or individual connection between them.

Example: Confidence score 1

For our first example, let's set the confidence score = > 1 . In the graph, all of the connected components match or exceed this value. As a result, all 6 devices are in a single "household" and "individual" group for this score and audience.

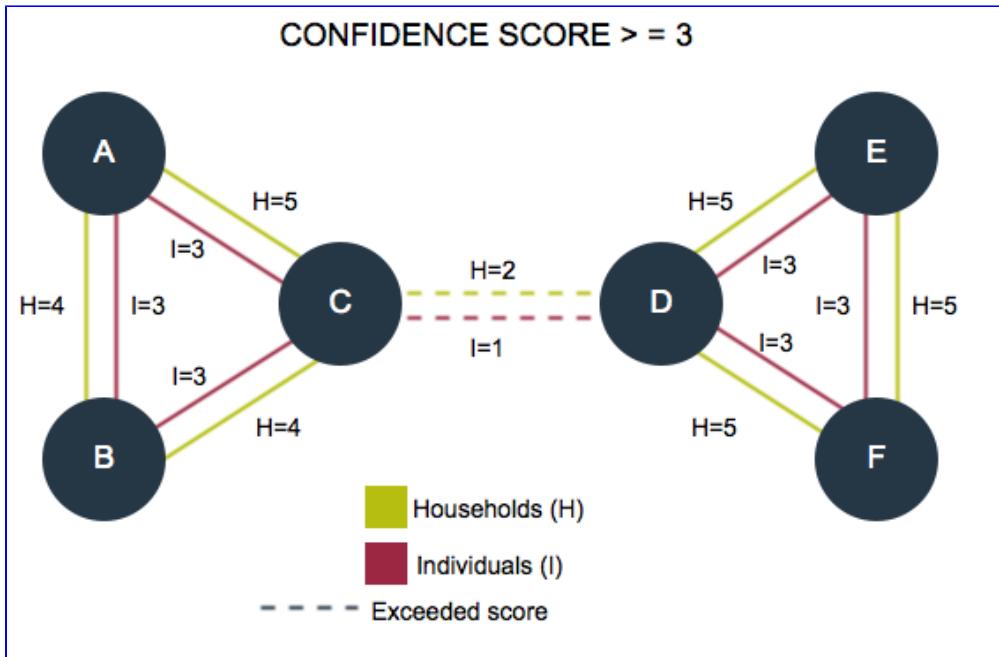


With this confidence score the Households and Individuals tile returns a count of 1 household and 1 individual group.



Example: Confidence score 3

Next, let's set the confidence score = > 3 . In the graph, the selected value is greater than the scores set on the connected component between devices C and D. The dashed line indicates the relationship between these devices is no longer valid at this confidence score. As a result, we now have 2 household and 2 individual groups for this score and audience.



When clicked, the Households and Individuals tile returns a count of 2 households and 2 individual groups.

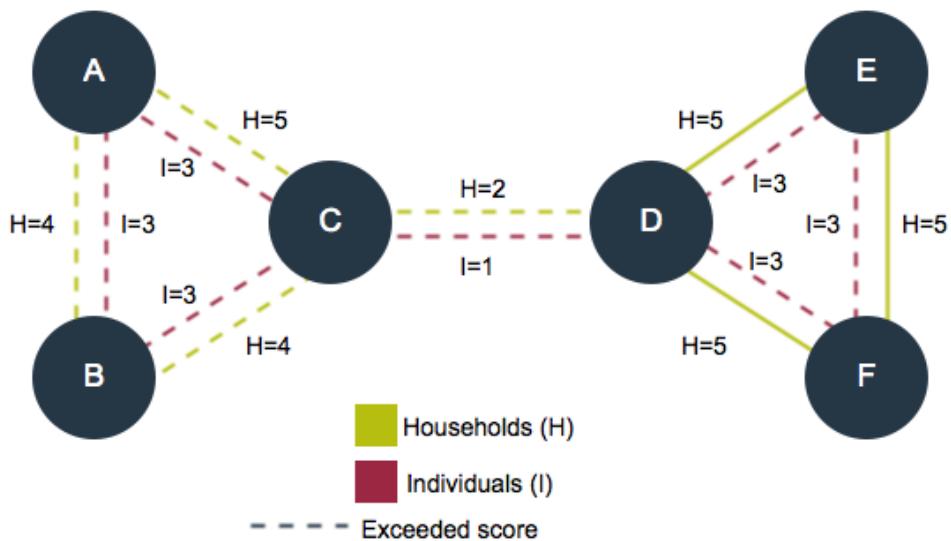


Example: Confidence score 5

As you raise the confidence score, and exceed the values set on each connected component, the number of households and/or individual groups increases. For example, let's look at what happens when the confidence score $>= 5$. This value is greater than most household and individual scores in the graph sample graph below. Only devices D, E, F remain connected at the household level. All other devices get counted as separate groups. These results give us:

- 4 household groups: A, B, C (each is a separate group), and D, E, F (counted as 1 group)
- 6 individual groups for this score and audience: A, B, C, D, E, F (each is a separate group)

CONFIDENCE SCORE ≥ 5



When clicked, the tile returns a count of 4 households and 6 individual groups.



On-site and Off-site Data: About

Definitions and examples of on-site and off-site data and how Tapad counts devices in these categories.

Some Ask Tapad dashboard tiles let you view results for on-site or off-site devices. Tapad defines these devices as follows:

- **On-site:** Devices that *have* visited your site. Audience data for an on-site device is known as direct fire data.
- **Off-site:** Devices that *have not* visited your site. When connected to an on-site device in the Device Graph, an off-site device inherits audience membership from a qualified on-site device. Audience data for an off-site device is known as indirect fire data.

Advantages/benefits

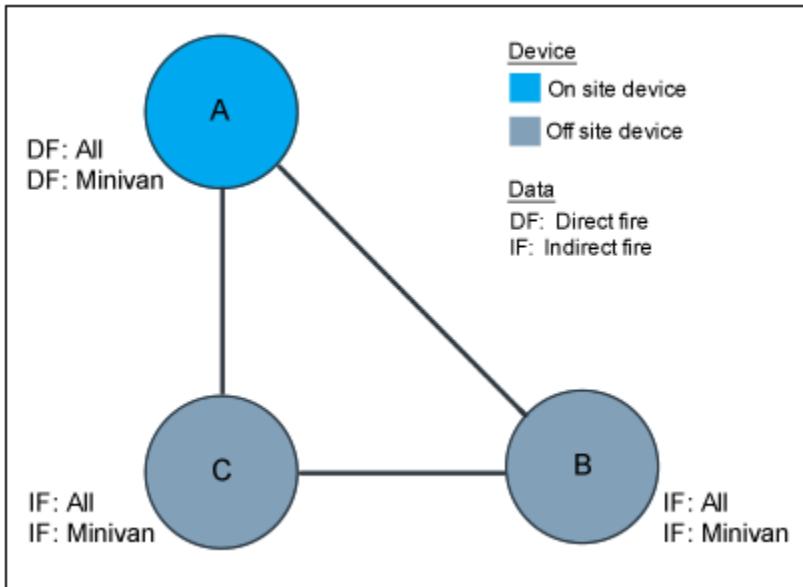
Aside from being a nice way to classify data, [counting on-site/off-site devices](#) provides real information about the size of your audiences and helps extend audience reach. For example, let's say you already have audience data on 100 devices that have visited your site. If these devices are connected to 50 more devices in the Device Graph, then Tapad can pass audience data from the devices you know about to those you've never seen. This is a 50% increase in your audience population, all *without* a site visit to your page. And, we're not just qualifying any random device either. Because of these devices are connected in the Device Graph, you're reaching related mobile phones, tablets, laptops, etc. that most likely belong to a specific household or person.

Counting on-site and off-site devices

To help you understand how Tapad counts on and off-site devices let's look at 2 use cases. In these examples, we'll use a hypothetical automotive website seen by a visitor who has 3 devices (A, B, and C) linked together in the Device Graph. The important thing to note is how the on-site and off-site counts change as this user visits the automotive website with different devices.

First visit using device A

In this example, the site visitor lands on the homepage of our automotive website using device A. Device A becomes an on-site device and gets put into a Tapad-created audience called "all audiences." All audiences helps Tapad determine if a device is an on-site or off-site device. Next, this visitor searches for a minivan and clicks on a minivan page. This search activity puts device A in the minivan audience. But this isn't the end of the process. Remember, device A is connected to B and C in the Device Graph. As shown in the illustration below, devices B and C are off-site devices and inherit audience membership (as indirect-fire data) from device A.

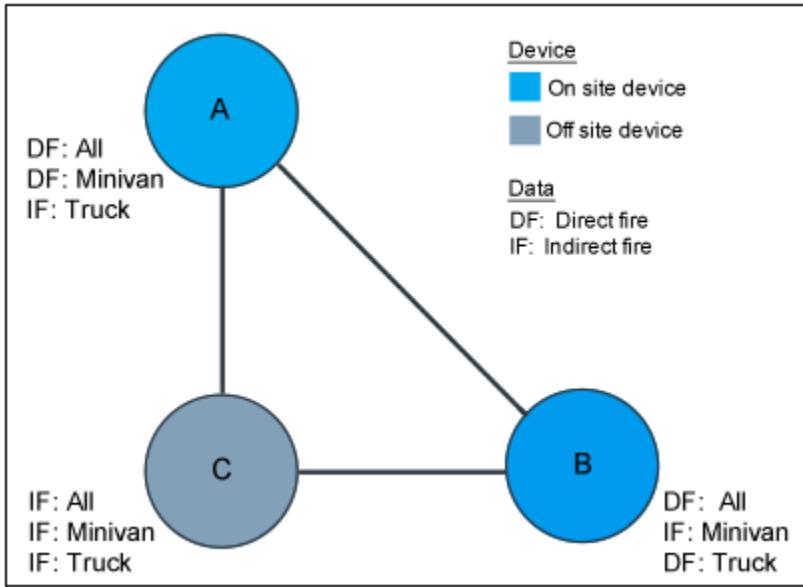


Based on this example, Tapad counts on and off-site audiences as shown in the following table. The results include the audience count and the device related to that audience. Note, the report only counts devices/audience combinations you do not know about. As a result, the on-site/all-audiences group never returns results because this is data you should already have from your own site metrics.

Audience	On-site count (and device)	Off-site count (and device)
All	n/a	2 (B, C)
Minivan	1 (A)	2 (B, C)

Second visit with device B

Let's take a look at how on-site and off-site numbers can change. In this example, our visitor returns to the automotive website using device B. Device B becomes an on-site device and its all audience status changes to direct fire (from indirect fire). Next, during this visit our user searches for a truck. Clicking a "truck" search result puts device B in the truck audience. Because device B is now an on-site device, the truck audience, and any other new audience data, is direct fire data. Older, indirect-fire data does not change (except for all audiences). As in the previous example, this isn't the end of the process. Remember, A and C are connected to B so they inherit the truck audience. However, these devices are off-site devices for this visit so the truck audience is classified as indirect-fire data for A and C.

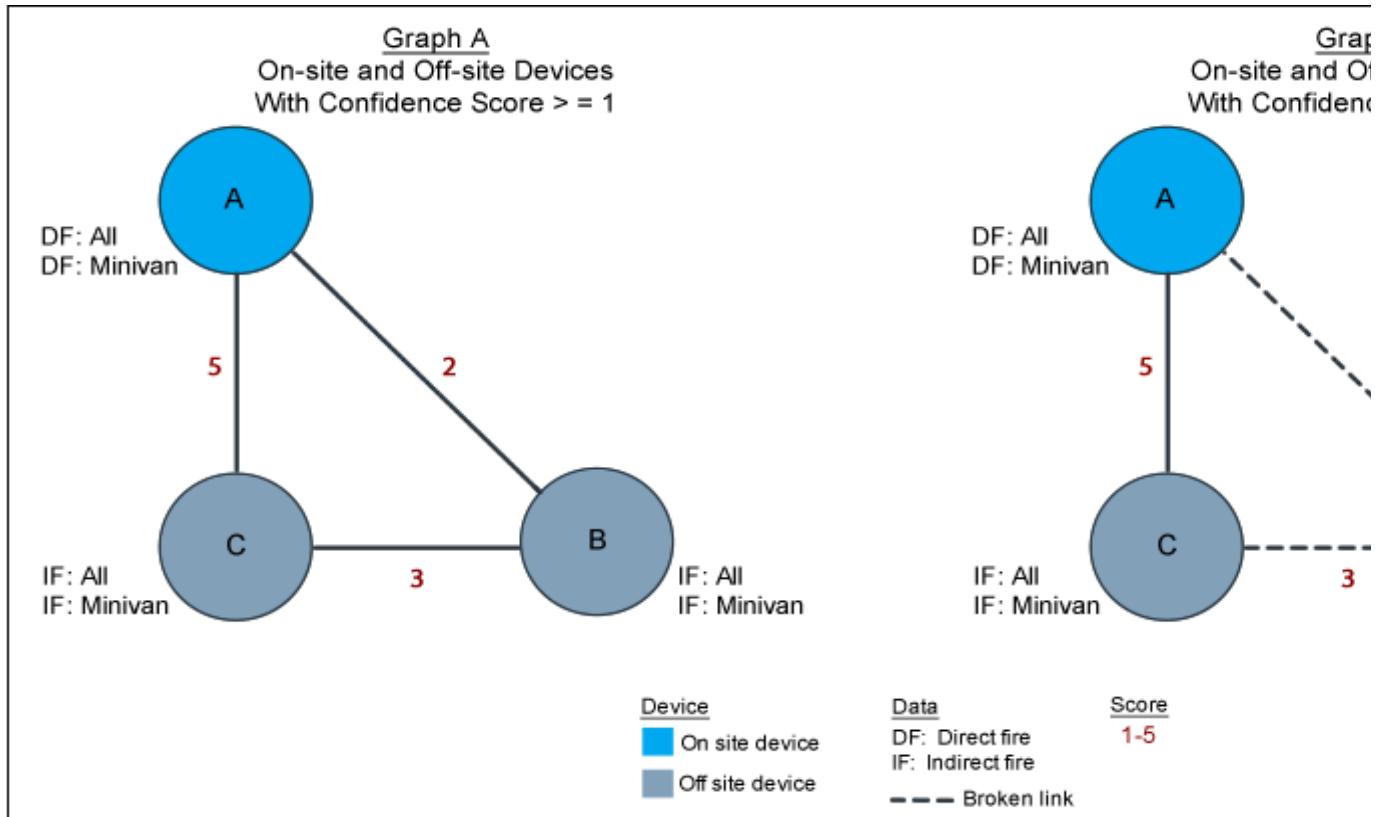


Based on this example, Tapad counts on and off site audiences as shown in the following table. The results include the audience count and the device related to that audience. Note, the report only counts devices/audience combinations you do not know about. As a result, the on-site/all-audiences group never returns results because this is data you should already have from your own site metrics.

Audience	On-site count (and device)	Off-site count (and device)
All	n/a	1 (C)
Minivan	1 (B)	2 (B, C)
Truck	1 (A)	2 (A, C)

Confidence scores affect audience counts

Because most devices in the Device Graph share a "connection" with another device, Tapad uses a value called the *confidence score* to help define the strength of the relationship between each connected device. For dashboard features such as [Devices and Audiences](#) and [Revenue Opportunity](#), raising or lowering the confidence score can increase or decrease the on-site and off-site audience count. Let's use the illustration below to discuss how this works.



In Graph A, let's see what happens when we filter data with the confidence score set to 1. In this case, all the devices in Graph A can be included in on-site or off-site results because their confidence scores are ≥ 1 . In fact, with the confidence score set to 1, this gives us the same results shown in the [first visit with device A](#) above. Next, let's see what happens when we filter Graph B with the confidence score set to 4. Increasing the score to 4 exceeds the values set on the connections between device B and the rest of the graph. This excludes device B from the results. With a confidence score of 4, we can only count the data on devices A and C as shown in the table below.

Audience	On-site count (and device)	Off-site count (and device)
All	n/a	1 (C)
Minivan	1 (A)	1 (C)

Campaign Pulse

Campaign Pulse is an analytics dashboard that provides insights derived from media delivery and the Device Graph. Rather than returning numeric data in columns and rows, this dashboard uses unique visualization techniques that help you spot trends across combinations of audiences, devices, and media delivery. Dashboard tiles return data about cross-device ad delivery, lift, reach/frequency, conversion paths, and improved impression/conversion attribution reporting for devices that don't use or accept cookies. Information presented in the Campaign Pulse dashboard can help you make smart decisions about how to reach new audiences, optimize media delivery, or invest in platform and conversion path combinations that lead to improvements in consumer response.

True story

It can take up to 48-hours for new data to appear in the Campaign Pulse dashboard.

Campaign Pulse returns data about:

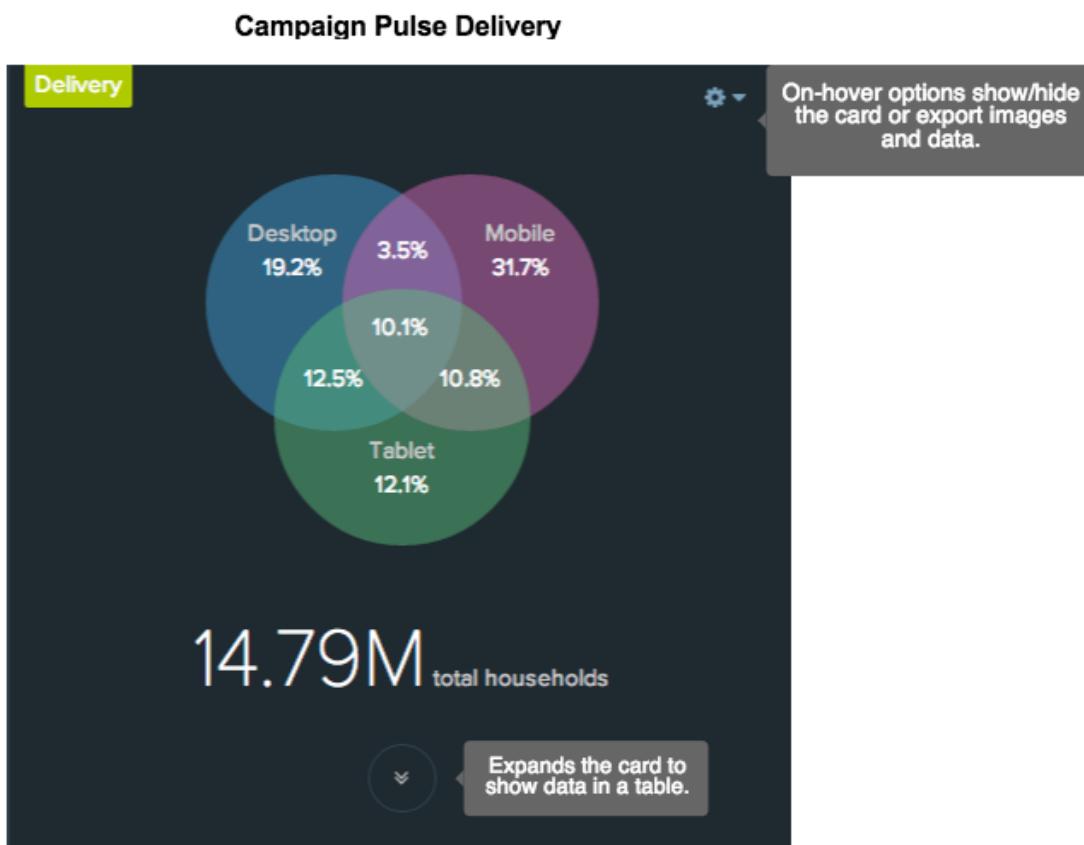
- **Delivery** — The Delivery tile returns household impression data by device in a Venn diagram. The overlap between circles in the diagram show you what percentage of those impressions were delivered to consumers across multiple devices. Delivery tile platforms include desktop, mobile, and tablet devices. Delivery tile data is available for all goal types.
- **Cross Platform Lift** — When you select a CPC, CPA, or CPCV goal, the Cross Platform Lift tile compares response rates when a consumer is exposed to an ad across 1, 2, or 3 platforms. Lift is an important measurement because it helps you determine if media is more effective when ads are shown across multiple platforms. Looking at lift helps advertisers find the most responsive and effective cross-device delivery combination for a campaign. However, it's also important to remember that the amount of lift shown on the re
- **Reach and Frequency** — Reach is the number of unique platforms that have seen an ad for your brand, product, or service. Frequency describes how many times an ad was delivered to those platforms. The Reach and Frequency tile summarizes unique impressions for your selected campaign at the household and device level. The Reach and Frequency tile is designed to help you understand cross-platform exposure based on your selected filters. Reach and Frequency tile data is available for all goal types.
- **Pathing** — The Pathing (conversion path) tile uses a starburst graph composed of concentric rings to show you the devices used to view an ad, from impression to conversion. By default, the interaction path starts with an impression in the innermost ring. Intermediate rings represent subsequent user interactions with that ad across different devices. The outermost rings show you which devices were used during a conversion event. Mapping the device path from first impression to final action provides insight
- **Audience Insight** — When you select a CPC, CPA, or CPCV goal in Campaign Pulse, the tile returns data in a scatterplot about the other audiences you have access to by size, audience score, and audience type. The tile displays first-party data, which is your own data, and any third-party data from your data partner that you have permissions to access through Tapad. Use the Audience Insight tile to help you find and target new audiences by size and receptivity to the campaign and/or line item combination in your sel
- **Third-party Ad Server (3PAS) Attribution** — Reporting discrepancies between different ad serving systems are common in digital advertising. Mobile devices and apps contribute to the discrepancy problem because these devices don't accept cookies, which ad servers use for conversion and attribution reporting. As a result, ad servers can miss conversions between mobile devices, between display and mobile devices, or from in-app conversions. However, because of Tapad's Device Graph technology, the 3PAS tile can track and attribute conversions

Delivery

The Delivery tile returns household impression data by device in a Venn diagram. The overlap between circles in the diagram show you what percentage of those impressions were delivered to consumers across multiple devices. Delivery tile platforms include desktop, mobile, and tablet devices. Delivery tile data is available for all goal types.

Delivery tile data helps you:

- Make mid-campaign course corrections based on platform combinations responsible the most and least impressions. For platform sets that show a low overlap (< 3%), consider changing your tactics or re-optimizing the campaign to try and push more consumers to cross-platform sets that generate more ad exposure. For devices that show a lot of overlap (> 5%), well, pat yourself on the back for a job well done.
- Determine if data in the [cross-platform lift](#) is significant. For example, let's say the Delivery tile shows that desktop devices and tablets account for < 3% of impressions. This indicates the 2 platform results in Cross-platform Lift tile may not be conclusive. In this case, device overlap > 5% for 2 or 3 platforms suggests that the lift data is useful and significant.



Tile results: Percentages and total impressions

- **Venn diagram %:** Sometimes the results don't always equal 100%. This is because the diagram shows data for mobile, tablet, and desktop devices only. It *excludes* data for game consoles and connected TVs, which is acceptable because these devices typically account for a small proportion of all our impressions.
- **Impression totals:** Total impression data *includes* results for all device types. Even though the Venn diagram doesn't always equal 100%, the expandable table always shows accurate count of total impressions for your selected filters.

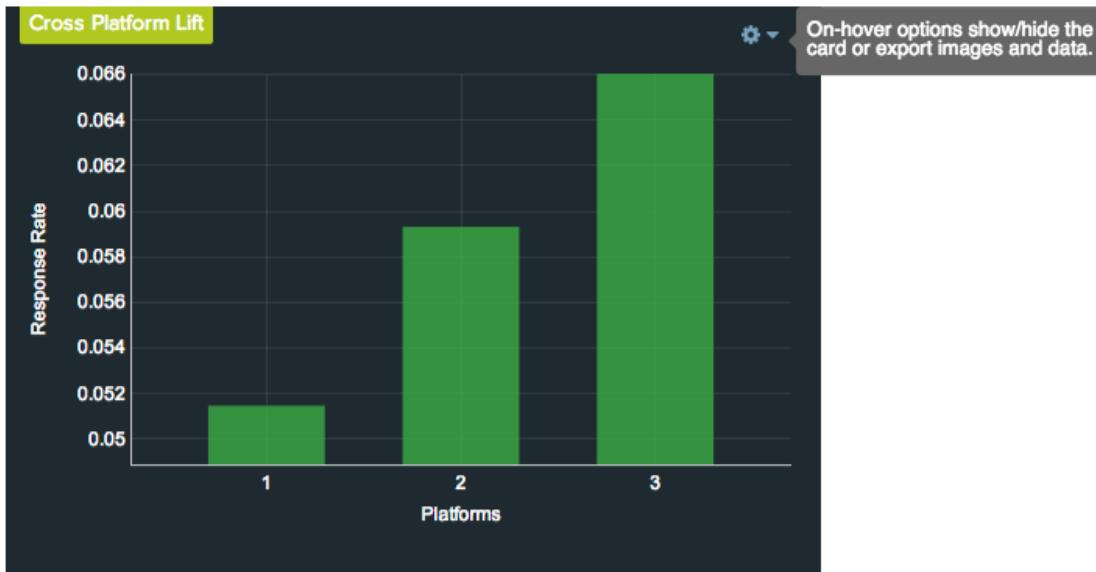
Recommended comparisons

You can draw conclusions about cross-device delivery independently from other tiles on the dashboard. However, it can be useful to compare delivery data in combination with [Reach and Frequency](#) data. Looking at reach/frequency data with impression data provides insight about total impressions used to reach the number of devices, households, or individuals in the Reach and Frequency tile. These tiles also share a relationship because you can multiply the household, individual, or device reach and frequency results to get the total impressions shown in the Delivery tile. Delivery data is also useful when you want to evaluate the significance of the results in the cross-platform lift tile (see [Cross Platform Lift](#) for more information).

Cross Platform Lift

When you select a CPC, CPA, or CPCV goal, the Cross Platform Lift tile compares response rates when a consumer is exposed to an ad across 1, 2, or 3 platforms. Lift is an important measurement because it helps you determine if media is more effective when ads are shown across multiple platforms. Looking at lift helps advertisers find the most responsive and effective cross-device delivery combination for a campaign. However, it's also important to remember that the amount of lift shown on the response axis depends on your campaign goals the dashboard filters you select.

Campaign Pulse Cross Platform Lift



The % lift between platforms is determined as:

LaTeX Math rendering error ([show error message](#)).

Recommended comparisons

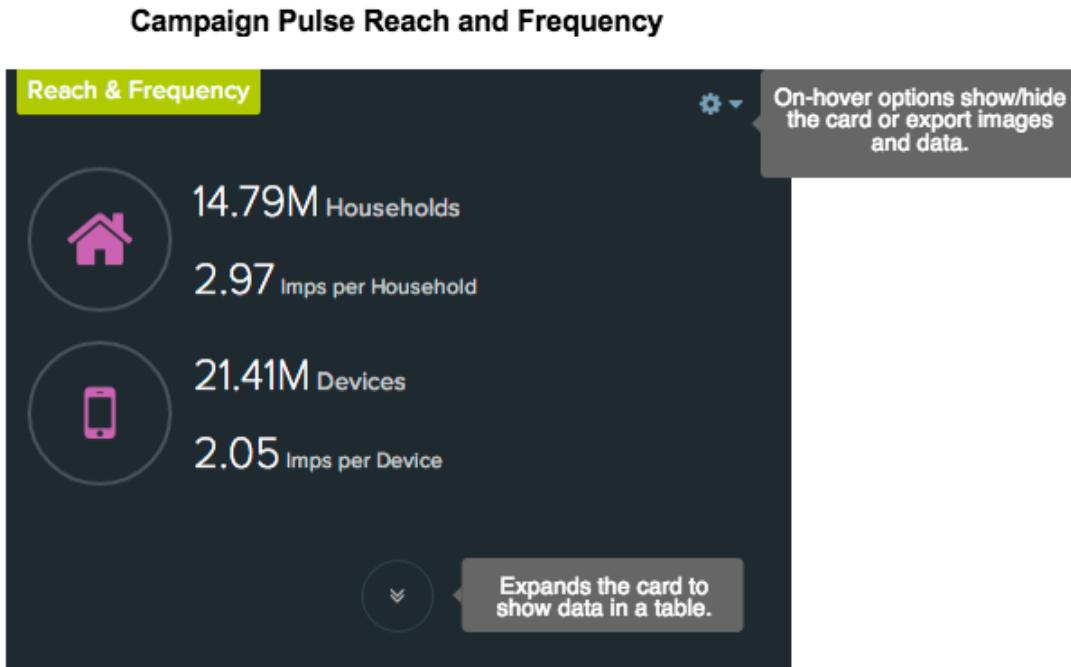
You can draw conclusions about lift independently from other tiles on the dashboard. However, to test the significance of your lift results, consider evaluating this data with information returned by the [Delivery](#) tile. When you compare lift with delivery, look at the % of impressions that appear on multiple devices in the Delivery tile Venn diagram (i.e., where the platform circles overlap). For example, a delivery overlap of < 3% for 2 or 3 platforms suggests your cross-platform lift may not be significant. By contrast, a delivery overlap of > 5% for 2 or 3 platforms suggests your cross-platform lift results are significant and valuable.

Related concepts

For more information about lift, see [Surveys: About](#).

Reach and Frequency

Reach is the number of unique platforms that have seen an ad for your brand, product, or service. Frequency describes how many times an ad was delivered to those platforms. The Reach and Frequency tile summarizes unique impressions for your selected campaign at the household and device level. The Reach and Frequency tile is designed to help you understand cross-platform exposure based on your selected filters. Reach and Frequency tile data is available for all goal types.



Recommended comparisons

You can draw conclusions about reach and frequency independently from other tiles on the dashboard. However, it can be useful to compare results from this tile with other tiles on the dashboard. For example, perhaps the [Delivery](#) and [Cross Platform Lift](#) tiles suggest there's not a lot of delivery or improvement in response rates across multiple devices. Ideally, you'd want a lot of cross-device delivery and strong lift results, but even these metrics are soft, reach and frequency results can still be substantial. This indicates that even though users aren't viewing or acting on ads, you're still exposing creative content to a lot of households, individuals, and devices. Reach and Frequency can also be used as a check on the total number of impressions returned by the Delivery tile. The sum of household, individual, and device impressions should equal the total number of impressions shown in the Delivery tile.

Pathing

DRAFT

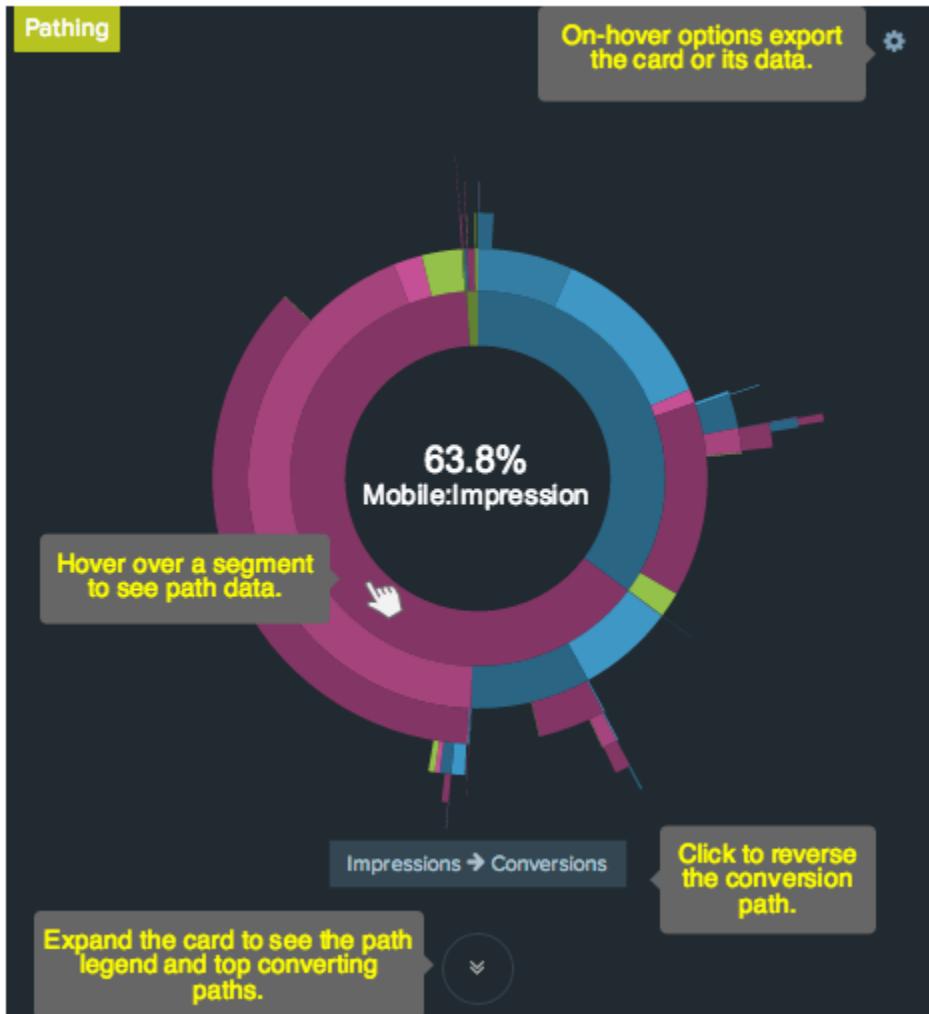
REVISED

Reference is

DOC-253 - Pathing revisions RESOLVED

The Pathing (conversion path) tile uses a starburst graph composed of concentric rings to show you the devices used to view an ad, from impression to conversion. By default, the interaction path starts with an impression in the innermost ring. Intermediate rings represent subsequent user interactions with that ad across different devices. The outermost rings show you which devices were used during a conversion event. Mapping the device path from first impression to final action provides insight about the effectiveness of a campaign or line item across different devices. Understanding which devices and ads work together well or poorly can help you optimize campaign performance. The Pathing tile returns data for CPA goals only.

Campaign Pulse Pathing Features



Pathing value: Identify the best and worst conversion sequences

Because the Pathing tile maps all of the events for your selected campaign, you can use this tile to help determine which platforms and ad types work well together and which ones do not. To create these paths, the tile follows a rule set that determines how to shorten long paths accurately and split them into understandable sequences. When looking for an effective conversion path, it's helpful to look at the outermost (conversion) segments.

- Broad end segments, suggest your campaign is using an effective mix of ad content and platform types.
- Narrow end segments, suggest your campaign is not reaching an effective mix of ad content and platform types.

Path and grouping options

The Pathing tile provides with additional display options that let you view paths:

- From impressions to conversions or in reverse, from conversions to impressions.
- By engagements only. An engagement is any subsequent action after the first impression. Note, this feature is available only when you click to view a path from impressions to conversions.

Recommended comparisons

You can draw conclusions about the conversion path independently from other tiles in the on the dashboard. However, it can be useful to compare path results with data returned by the [Ad Server Attribution](#) tile. The Attribution tile shows results for the last touch point responsible for a click, conversion, or other desired action from your first and third-party ad servers. Comparing these 2 tiles helps advance your understanding beyond simple last touch attribution because you get a complete overview of how users reached an conversion, including conversion data from mobile browsers and apps, which typical ad servers cannot report on because those platforms and software packages do not use cookies.

Understanding path splitting and truncation rules

Sometimes conversion paths are too long and cannot be displayed in the starburst graph. To shorten long paths and still retain accuracy, the Pathing tile splits and truncates conversion paths. For more information, see:

- [Path Splitting Rules for Campaign Pulse](#)
- [Path Truncation Rules for Campaign Pulse](#)

Path Splitting Rules for Campaign Pulse

In [Campaign Pulse](#), path splitting rules help the [Pathing tile](#) manage and display complex conversion strings. This logic provides rules that filter by selected conversions, for merging events, and splitting the conversion path. After a path is split, it is subject to additional processes governed by the [path truncation rules](#).

Pathing overview

The Pathing tile lets you follow an ad on each device and placement type, from first impression to final conversion. It displays this data in concentric rings in a [sunburst chart](#) or graph. Frequently, a complete conversion path contains so many events, platforms, and conversions that it's not practical to try and force all that data in to each concentric segment on the chart. As a result, the Pathing tile doesn't show the entire conversion path. Instead, it uses a few rules to identify individual conversions from within long paths, break them into separate paths, qualify paths for inclusion in the chart, and reduce the path size. These processes help the Pathing tile display long, complex events chains in a concise and accurate manner.

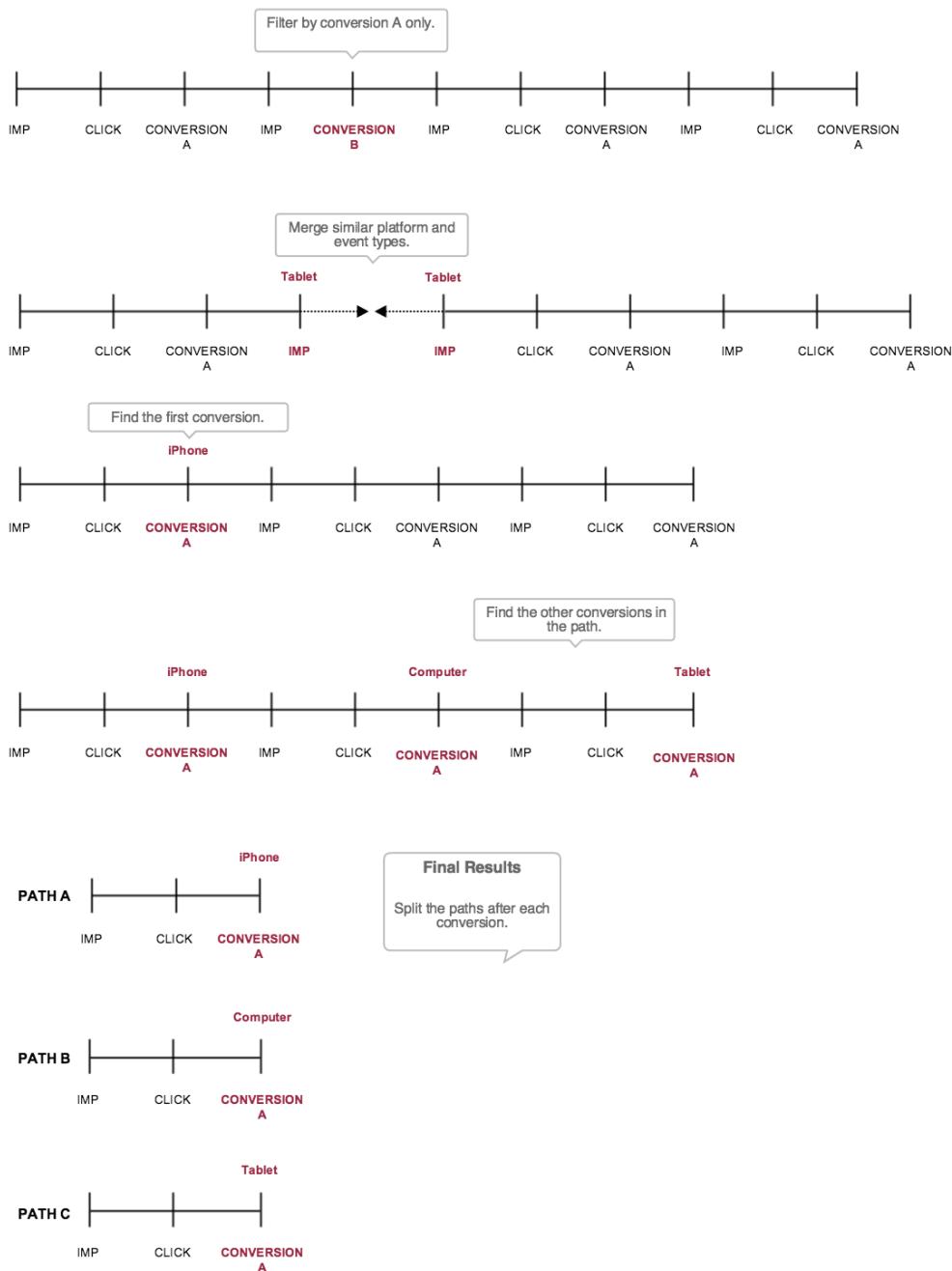
Path splitting rules

In this example, let's say a user goes into Campaign Pulse and selects a campaign, CPA goal, and Conversion A from the Events menu. To display data for these filters, the Pathing tile works through the following steps to identify and split complex conversion paths:

1. Filter conversions. Because we're interested in Conversion A only, this user-selection removes any other conversions from the conversion path.
2. Merge similar path events and platforms. Note, we still keep an accurate count of events and platforms, but will merge them to reduce path size.
3. Find the first conversion in the path.
4. Find the other conversions in the path
5. Split the path after each conversion event.

Refer to the illustration for an example of how it this process works.

Path Splitting Rules



Related concepts

[Path Truncation Rules for Campaign Pulse](#)

Path Truncation Rules for Campaign Pulse

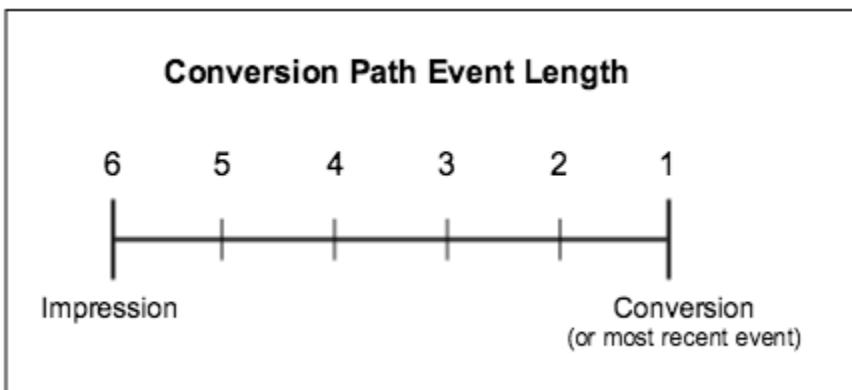
In **Campaign Pulse**, the **Pathing** tile follows several rules that define the path length, qualifying initial events, and how to truncate the path and maintain accuracy. These rules also define what to do when a path does not contain any impressions or conversions. However, *before* truncating a path, they're processed by the [path splitting rules](#).

Pathing overview

The Pathing tile lets you follow an ad on each device and placement type, from first impression to final conversion. It displays this data in concentric rings in a [sunburst chart](#) or graph. Frequently, a complete conversion path contains so many events, platforms, and conversions that it's not practical to try and force all that data in to each concentric segment on the chart. As a result, the Pathing tile doesn't show the entire conversion path. Instead, it uses a few rules to identify individual conversions from within long paths, break them into separate paths, qualify paths for inclusion in the chart, and reduce the path size. These processes help the Pathing tile display long, complex events chains in a concise and accurate manner.

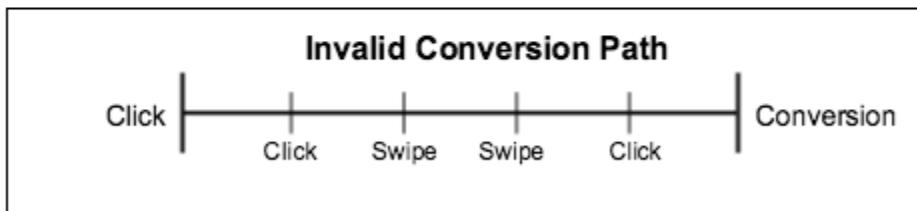
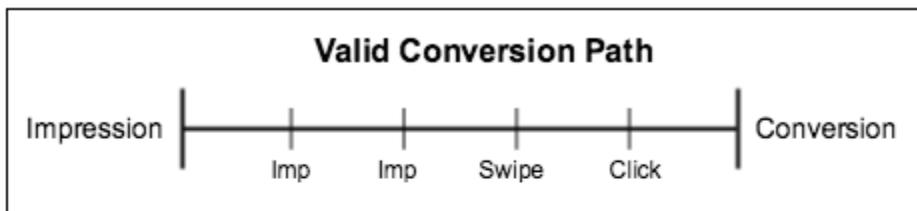
Path length

The length of a path must be *equal to* or *less than* 6 steps from the most recent event, which may or may not be a conversion (paths can end on impressions, clicks, or swipes). As a method of identification, you can number the path steps in reverse order from 1 to 6. The most recent event is position 1. From that point, number each step backwards in ascending order to the initial impression at position 6.



Initial event

Every path *must* start with an impression. A path without an impression event is excluded from the graph.



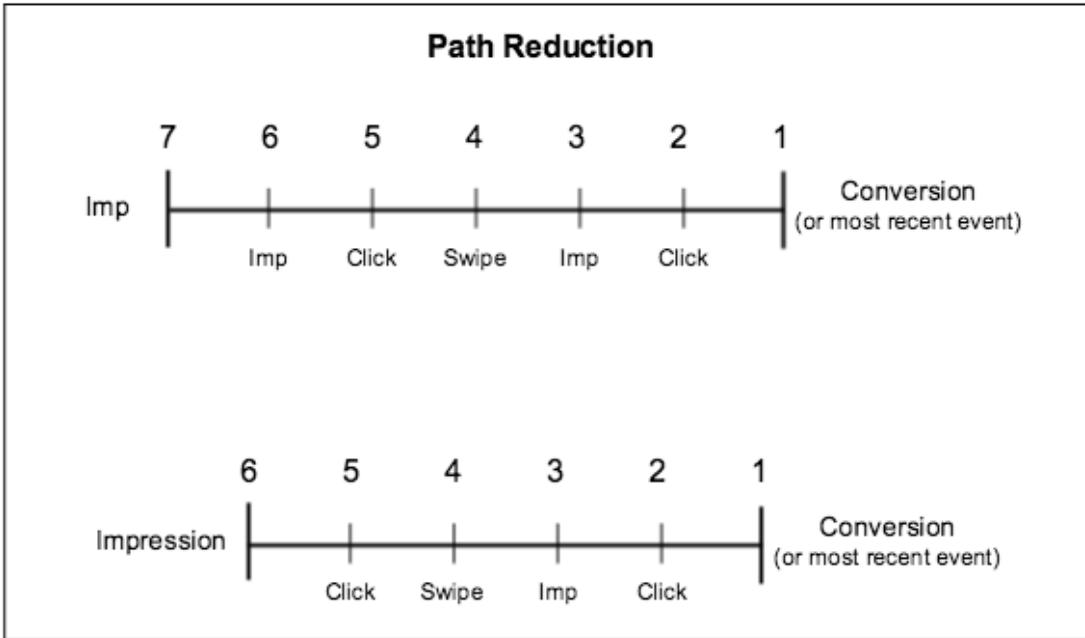
Path length reduction and truncation

Frequently, conversion paths don't have impressions within 6 steps of a conversion or the most recent action. When a conversion path *does not* st

art with an impression that is 6 steps from the final event, path event logic attempts to find an impression by going forwards or backwards from the 6th event. The position of the impression determines how the truncation algorithm prunes the conversion path.

Path reduction

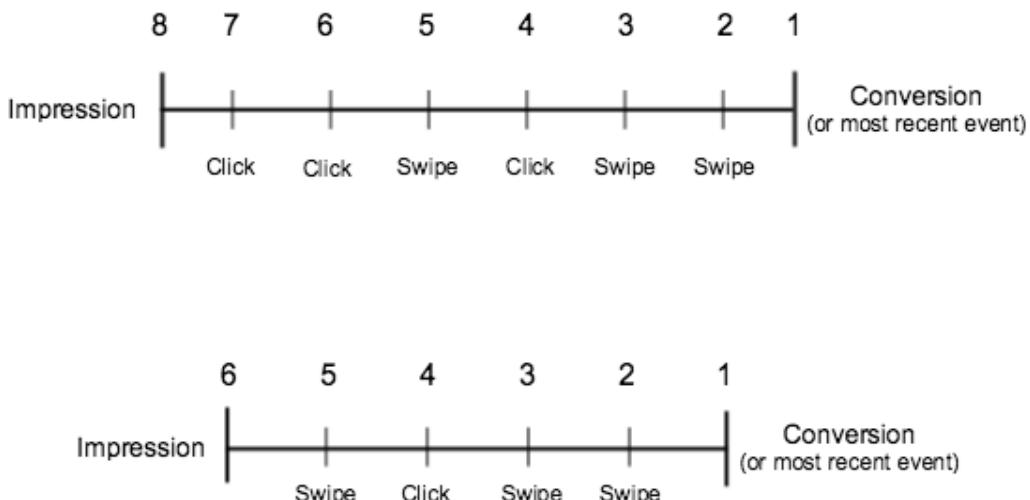
If an impression is $<= 6$ events from the conversion, cut every other event from the path until it starts with the closest impression. In the following example, the reduction process cuts step 7 (a swipe) from the path. This makes the path legitimate because it starts with an impression and is 6 steps long.



Path truncation

If an impression is > 6 events from the conversion, remove intermediate events (truncation), and start the path 6 steps from the conversion. In the following example, the truncation process removes steps 6 and 7 (both clicks). This moves the impression to step 6 (from step 8). This makes the path legitimate because it starts with an impression and is 6 steps long.

Path Truncation



No impressions or conversions

When an event path does not contain any impressions or conversions, the path logic responds as follows:

- **No impressions:** Exclude the entire path from the chart.
- **No conversions:** Keep the data and include the path in the chart. A path must start with an impression, but it doesn't have to end with a conversion (so we want to keep the data).

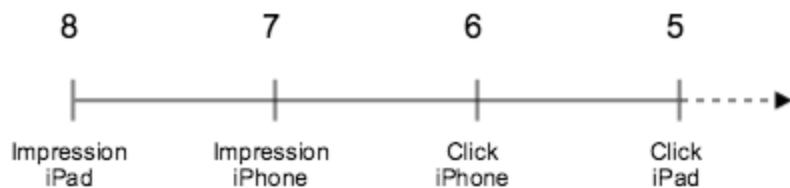
Truncation can create unusual path steps

Each event in a conversion path is associated with a device, but the previous examples don't show them. This is deliberate because it helps keep the paths simple and easy to understand. But, because paths contain devices *and* events, sometimes you may see odd device/event steps in an individual path. For example, let's say you see a path that contains an iPhone impression prior to an iPad click. In reality, this is impossible because you can't go from an iPhone impression to an iPad click. This sequence requires an iPad impression prior to the iPad click. These types of unlikely event/device steps are simply a function of how the truncation rules create short paths.

As another example, refer to the illustrations below. The first example shows a normal conversion path. However, this path is too long and needs to start with an impression. Given these conditions, the path truncation rules:

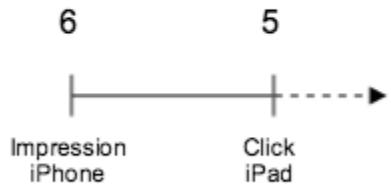
- Remove the iPhone click event from position 6.
- Move the iPhone impression event to position 6.
- Drop everything else in the path prior to iPhone impression.

Conversion Path Before Truncation



This gives us a new, initial path that contains the initial steps show below.

Conversion Path After Truncation



Related concepts

[Path Splitting Rules for Campaign Pulse](#)

Audience Insight

DRAFT

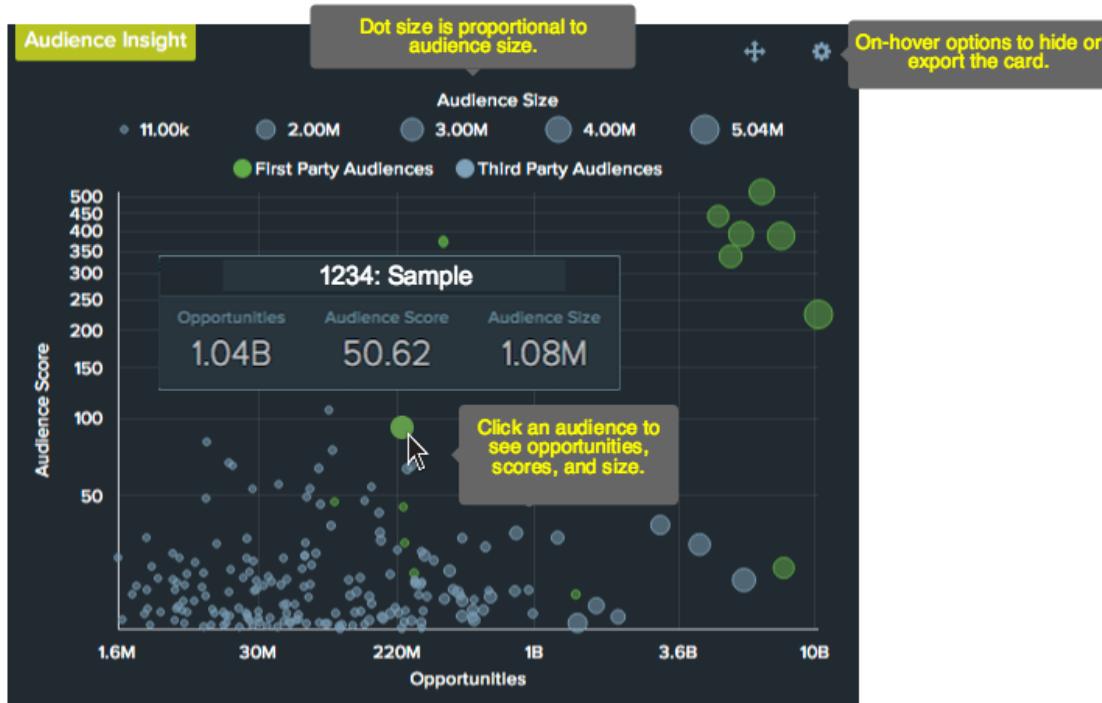
REVISED

Reference is

DOC-246 - Campaign Pulse Revisions: Audience Scores RESOLVED

When you select a CPC, CPA, or CPCV goal in Campaign Pulse, the Audience Insight tile returns data in a scatterplot about the other audiences you have access to by size, audience score, and audience type. The tile displays first-party data, which is your own data, and any third-party data from your data partner that you have permissions to access through Tapad. Use the Audience Insight tile to help you find and target new audiences by size and receptivity to the campaign and/or line item combination in your selected Campaign Pulse filters.

Campaign Pulse Audience Insight Features



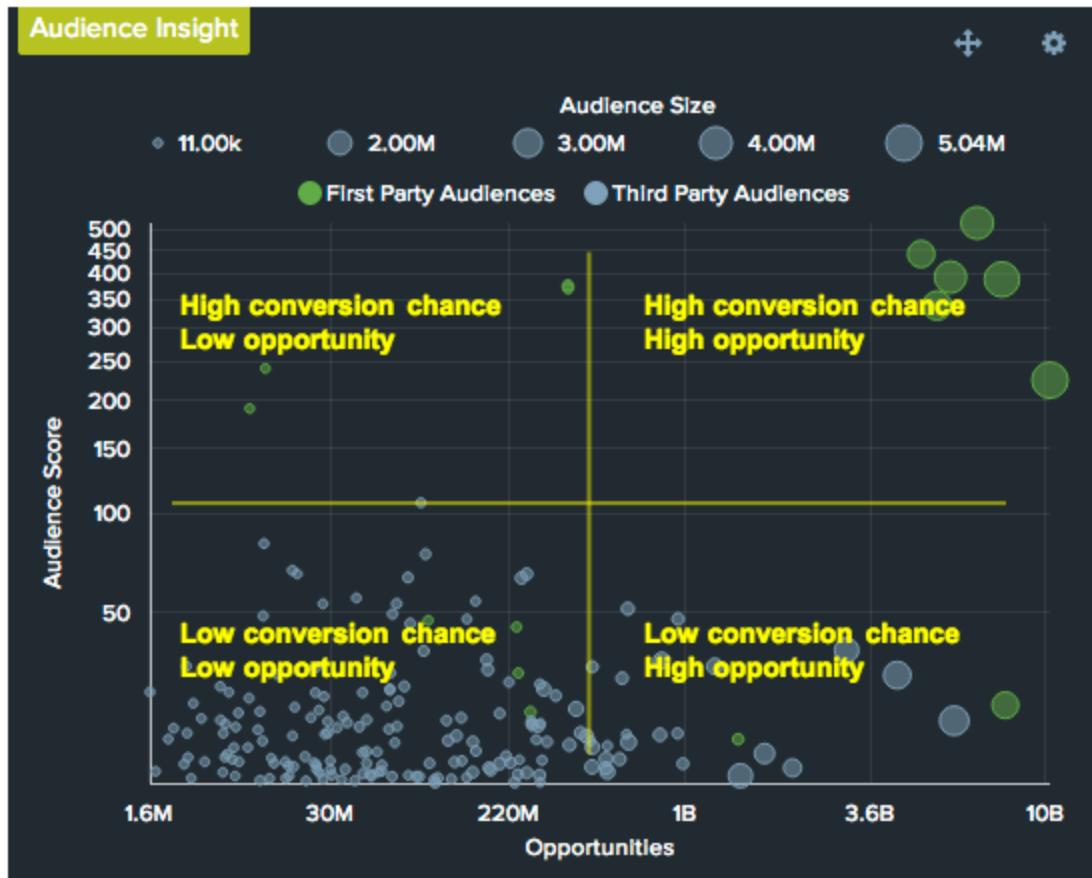
Find value in unexpected audiences

Audience Insight provides value because it helps you find other audiences that your campaign did not target, but that may have an affinity for converting on your selected campaign or line item. When evaluating the value of your audience data, look for the right combinations of:

- **Audience scores:** These measure likelihood of an un-targeted audience converting on your campaign. On the graph, audiences higher up on the score axis are more likely to convert relative to other audiences in a lower position.
- **Audience size:** The size of each point on the graph is directly proportional to the size of the audience it represents — large points represent big audiences and small points represent small audiences.
- **Opportunities:** The number of bid requests received by each audience.

Audiences with high scores, large sizes, and a lot of bid requests are among the most valuable data points returned by the graph. These appear in the upper right section of the Insight Tile scatterplot.

Finding Value on the Audience Insight Tile



Understanding audience scores

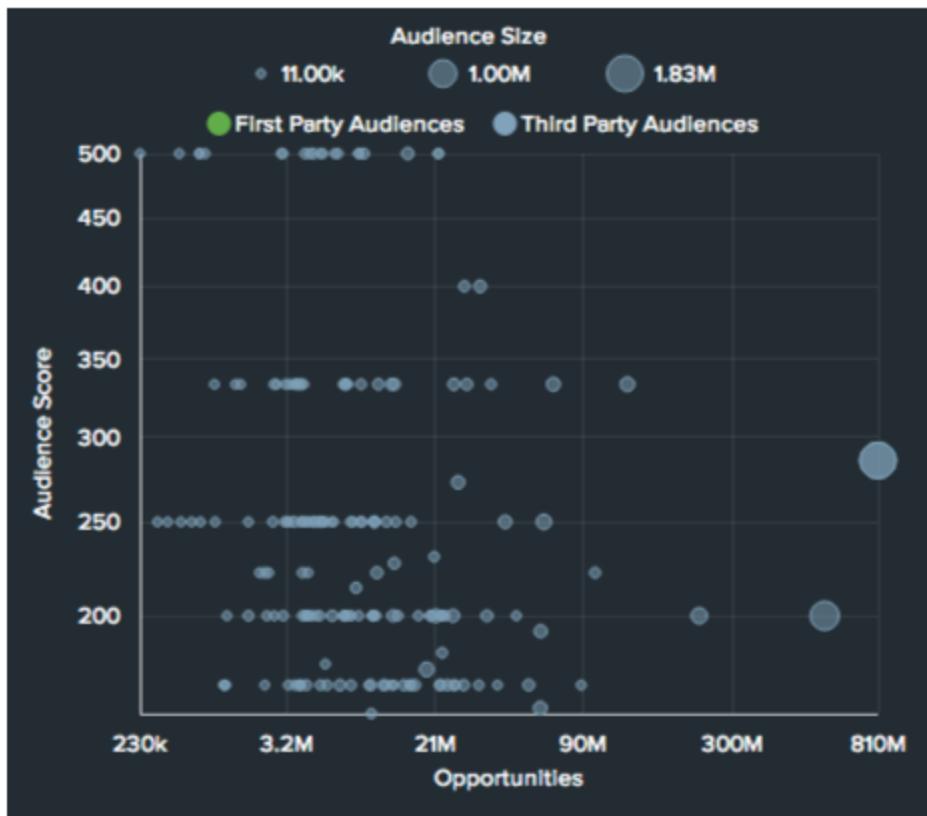
Audience scores are derived from a pool of all the devices shown an impression for a campaign or campaign/line item combination. If enough of these devices belong to various other audiences, we can create a quantitative measurement that suggests how well they would perform if targeted by your selected campaign or line item. For more information about audience scores, see:

- CPA Audience Score Equations and Variables
- CPC and CPCV Audience Score Equations and Variables

Horizontal rows of audiences indicate insufficient data

The Audience Insight tile displays horizontally aligned audience points when it doesn't have enough data to create reliable audience scores. In general, you should never see this kind of data "striping" but it can happen. These results can mean that your audiences aren't very large. Be sure tell the Product team if you ever see data similar the following example.

Striped Data in the Audience Insight Tile



CPC and CPCV Audience Score Equations and Variables

DRAFT

Reference is  [DOC-246](#) - Campaign Pulse Revisions: Audience Scores RESOLVED

An overview of the cost-per-click (CPC) and cost-per-completed-view (CPCV) formulas and variables used by the Audience Insight tile to calculate audience scores.

Audience score overview

In Campaign Pulse, the Audience Insight tile displays data that includes an *audience score*. The audience score measures the likelihood of users in other audiences to convert on the campaign and/or line item you've selected on the dashboard. The data used to calculate the score is taken from all of the first and third-party audiences that your account is authorized to see. However, because of space limits, the Audience Insight tile can only display a maximum of 1000 audiences and scores. Also, keep in mind that raw audience scores are often small, fractional values. To generate a meaningful score, the Audience Insight tile applies a multiplier to these raw values and then displays the score when you hover over an audience in the scatterplot. This section explains how the Audience Insight tile calculates audience scores when you select a CPC or CPCV goal.

For an explanation of score calculations with cost-per-acquisition (CPA) goals, see [CPA Audience Score Equations and Variables](#).

CPC and CPCV audience score equations

The Campaign Pulse dashboard provides filters that let you return data for a specific campaign or for a campaign and line item combination. These filters affect how the Audience Insight tile calculates the audience score for your selected campaign or line item. The following table lists the formulas used by the Audience Insight tile to calculate these audience scores when you're using a CPC or CPCV goal filter.

Goal type	Equations by filter selection
CPC	Campaign only: LaTeX Math rendering error (show error message).
	Campaign and line item: LaTeX Math rendering error (show error message).
CPCV	Campaign only: LaTeX Math rendering error (show error message).
	Campaign and line item: LaTeX Math rendering error (show error message).

CPC and CPCV variables: Total campaign or line item audience and max totals

Each of these formulas contain similarly named variables and perform identical operations. However, they each draw upon different data sets based on your goal selection. These include:

- **Total clicks or completed views for a campaign and audience:** In the numerator, the campaign audience variable is the sum of all the clicks or completed views for the devices in a given audience that were exposed to an impression served by a campaign. Campaign audience totals vary because each device in the Device Graph can belong to multiple audiences and may or may not have clicked on an impression or completed a view.
- **Max totals:** In the denominator, these variables use the `max` function to let the equation use whichever data set contains the most data for your campaign or line item. For example, if you filter by CPC goal, the equation will use either total clicks or total impressions (whichever is larger) when calculating an audience score. Because you're working with totals for a selected campaign or line item, the values in the numerator do not change when the Audience Insight tile generates scores for other audiences. This is different than the numerator values for CPA audience score calculations because those formulas use audiences of different sizes.

Sample score calculation

Let's take a look at how this works with a simple CPC campaign called Campaign Q. In this example:

- Campaign Q served impressions to Device Alpha and Device Beta.
- Campaign Q has recorded 100 impressions and 5 clicks total (2 on Device Alpha and 3 on Device Beta).
- Each device is already in other audiences labelled A, B, C, D, and E.

As shown below, these conditions give us two devices that belong to 3 different audiences each.

Device Alpha		Device Beta	
Clicks	Audiences	Clicks	Audiences

2	A	3	A
	B		C
	E		D

Next, we need to calculate amounts for the variable Total Clicks for a Campaign and Audience. Using audience A as an example, the process works like this:

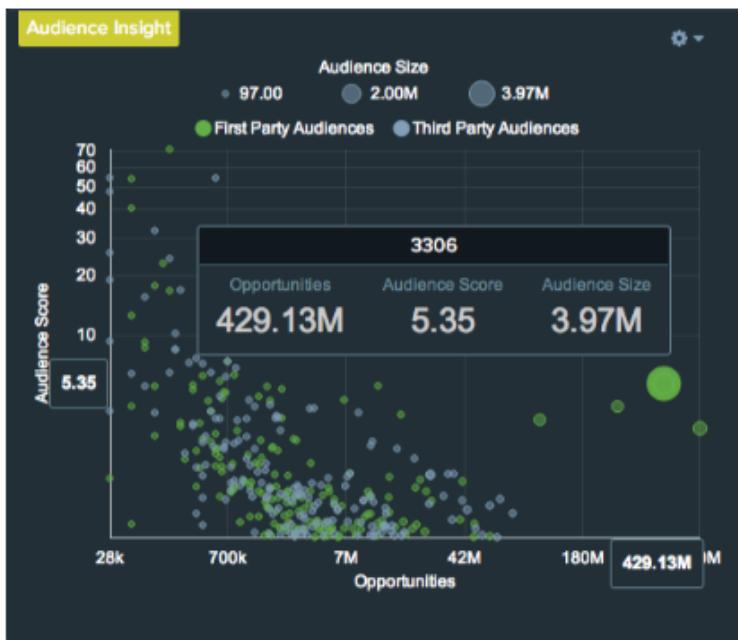
- Device Alpha: 2 clicks
- Device Beta : 3 clicks
- Total: 5 clicks for audience A ($2 + 3 = 5$)

Given this data, we can now determine the campaign and audience click totals, put these results over the total number of impressions for Campaign Q ($n = 100$), and create a raw audience score.

Audience	Total clicks for campaign and audience	Completed equation	Raw audience score
A	5	5/100	0.005
B	2	2/100	0.002
C	3	3/100	0.003
D	3	3/100	0.003
E	2	2/100	0.002

Because the raw scores are so small, the Audience Insight tile uses a multiplier to convert them into larger, whole numbers. These converted values are the CPC or CPCV audience scores you see for each point on the Audience Insight graph.

Insight Tile With Sample Audience Score



Related reference

[Common Price Types: Formulas and Examples](#)

CPA Audience Score Equations and Variables

DRAFT

Reference is  [DOC-246](#) - Campaign Pulse Revisions: Audience Scores RESOLVED

An overview of the cost-per-acquisition (CPA) formulas and variables used by the Audience Insight tile to calculate audience scores.

Audience score overview

In [Campaign Pulse](#), the [Audience Insight](#) tile displays data that includes an *audience score*. The audience score measures the likelihood of users in other audiences to convert on the campaign and/or line item you've selected on the dashboard. The data used to calculate the score is taken from all of the first and third-party audiences that your account is authorized to see. However, because of space limits, the Audience Insight tile can only display a maximum of 1000 audiences and scores. Additionally, a CPA data point needs to record 10,000 pixel fires to be eligible for scoring. Also, keep in mind that raw audience scores are often small, fractional values. To generate a meaningful score, the Audience Insight tile applies a multiplier to these raw values and then displays the score when you hover over an audience in the scatterplot. This section explains how the Audience Insight tile calculates audience scores when you select a CPA goal.

For an explanation of score calculations with cost-per-click (CPC) or cost-per-completed-view (CPCV) goals, see [CPC and CPCV Audience Score Equations and Variables](#).

CPA audience score equations

The Campaign Pulse dashboard provides filters that let you return data for a specific campaign or for a campaign and line item combination. These filters affect how the Audience Insight tile calculates the [audience score](#) for your selected campaign or campaign/line item choice. The following table lists the formulas used by the Audience Insight tile to calculate these audience scores when you're using a CPA goal filter.

CPA filter selection	Equation
Campaign only	LaTeX Math rendering error (show error message).
Campaign and line item	LaTeX Math rendering error (show error message).

CPA variables: Total conversion events and total audience size

Each of these formulas contain similarly named variables and perform identical operations. However, they each draw upon different data sets based on your campaign or line item selection. These formulas include:

- **Total conversion events for a campaign and audience:** In the numerator, the campaign audience variable is the sum of all the [events](#) for the devices in a given audience that were exposed to an impression served by a campaign. Campaign audience totals vary because each device in the Device Graph can belong to multiple audiences and may or may not have clicked on an impression or completed a view.
- **Total audience size:** In the denominator, these variables represent audience size. Because you're working with different audiences, the values in the numerator vary. This is different than the numerator values for CPC or CPCV audience score calculations because those formulas look at the totals for one selected audience only.

Sample score calculation

Let's take a look at how this works with a simple CPA campaign called Campaign Q. In this example:

- Campaign Q served impressions to Device Alpha and Device Beta.
- Campaign Q has recorded 5 conversions total (2 on Device Alpha and 3 on Device Beta).
- Each device is already in other audiences labelled A, B, C, D, and E.
- Each audience is a different size:
 - A = 182
 - B = 500
 - C = 620
 - D = 1400
 - E = 900

As shown below, these conditions give us two devices that belong to 3 different audiences each.

Device Alpha		Device Beta	
Conversions	Audiences	Conversions	Audiences
2	A	3	A
	B		C
	E		D

Next, we need to calculate amounts for the variable Total Conversion Events for a Campaign and Audience. Using audience A as an example, the process works like this:

- Device Alpha: 2 conversions
- Device Beta: 3 conversions
- Total: 5 conversions for audience A ($2 + 3 = 5$)

Given this data, we can now determine the campaign and audience conversions, put these results over the audience sizes, and create a raw audience score.

Audience	Total conversion events for campaign and audience	Completed equation	Raw audience score
A	5	5/182	0.027
B	2	2/500	0.004
C	3	3/620	0.004
D	3	3/1400	0.002
E	2	2/900	0.002

Because the raw scores are so small, the Audience Insight tile uses a multiplier to convert them into larger, whole numbers. These converted values are the CPA audience scores you see for each point on the Audience Insight graph.

Insight Tile With Sample Audience Score



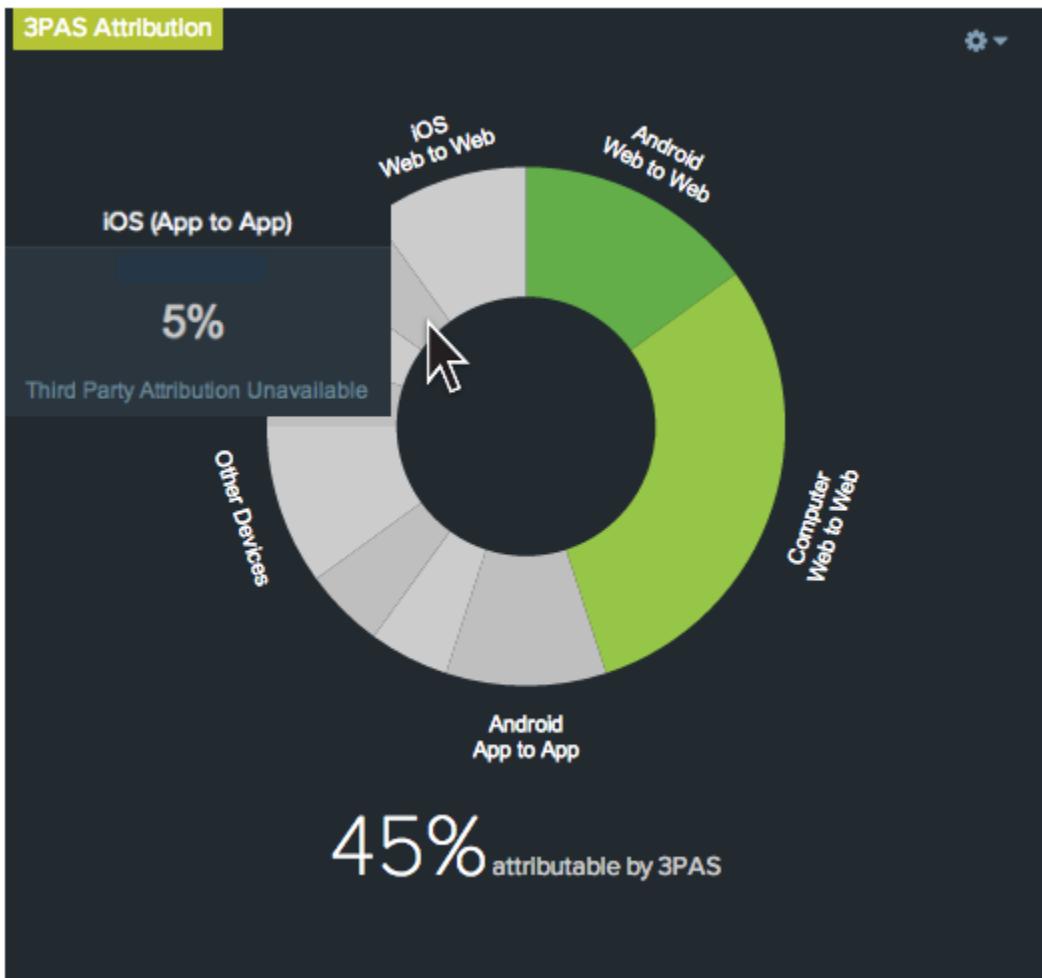
Related reference

[Common Price Types: Formulas and Examples](#)

Third-party Ad Server (3PAS) Attribution

Reporting discrepancies between different ad serving systems are common in digital advertising. Mobile devices and apps contribute to the discrepancy problem because these devices don't accept cookies, which ad servers use for conversion and attribution reporting. As a result, ad servers can miss conversions between mobile devices, between display and mobile devices, or from in-app conversions. However, because of Tapad's Device Graph technology, the 3PAS tile can track and attribute conversions from mobile devices and their apps. The 3PAS tile gives you a comprehensive view of *all* your conversions, including those ad servers typically report on and from Android, iOS, or other devices these systems cannot see. The 3PAS Attribution tile returns data for CPA goals only.

The illustration shows the Ad Server Attribution tile in the Campaign Pulse dashboard. In this example, third-party ad servers can report on only 45% of all these campaign conversions. Android, iOS, and other platforms account for the remaining 55% of attributable conversions. Without the 3PAS data, you would miss all these other conversions. Hover over a ring segment to see attribution details.



Reading the attribution pop

Moving the pointer over the graph lets you drill down into attribution details for each segment. The example above shows you that iOS in-app activities account for 5% of all conversions returned by the 3PAS tile. The phrase "Third Party Attribution Unavailable" indicates this data is not available to typical ad server systems. However, Tapad can report on these conversions because of our Device Graph technology. The pop-up header "iOS (App to App)" tells you where the impression was first served (in an iOS app) and where the subsequent conversion took place (also in an iOS app). The 3PAS tile tracks other impression to conversion combinations for Android and iOS mobile devices including:

- Web to web
- Web to app
- App to web
- App to app

The category "Other Devices" includes platforms like connected TVs and game consoles.

Recommended comparisons

You can draw conclusions about attribution independently from other tiles on the dashboard. However, it can be useful to compare attribution data with data returned by the [Conversion Path](#) tile. This is a helpful comparison because the attribution tile is most useful when trying to understand the end of a conversion path. Evaluating attribution data with path data helps create a more complete picture about what where consumers first view an ad, subsequent interactions on other devices, and which devices account for the final conversion. Comparisons between the Conversion Path tile and the 3PAS tile give you an overview of a creative's lifecycle, from first impression to final conversion.

Creative Management

All Unify docs about creatives have moved to the [Creative Management](#) section in the [Self-Service DSP space](#).

Dataflow Management

Dataflow Management provides features and workflows that let you set up and manage your data flows. Refer to this section for help with tasks like creating data flows and for feature settings, descriptions, and definitions.

A dataflow is part of an ID synchronization process that takes device IDs you've sent to Tapad and matches them to other device IDs. As determined by our Device Graph, these new IDs represent devices that share a connection with your original ID. When the matching process is complete, Tapad returns this data to you as a JSON object or in a flat file. You can use this data to target your customers across all their known devices, deliver personalized content, and use it with your own analytics systems or internal processes. Because partner systems and data requirements vary, Tapad can structure and deliver this information to you using the file format, content, and delivery options described in this material.

- **Create a Data Flow** — When you add a new data flow most of the options you need are already set up by default. Follow these instructions to set up or review a new dataflow.
- **Select Data to Include** — When creating a data flow, the Select Data to Include section provides settings that give you control over the contents of your dataflow file. For example, you can limit results by known or unknown devices, by hardware platforms, household scores, and other metrics. Many of these options are selected by default when you create a new data flow. Generally, all you need to do is review these settings and change the options relevant to the type of data you want receive in the file.
- **Filter By Synced Devices** — By default, Tapad returns all synchronized data for every device mapped to the IDs you send us. This includes data for devices and IDs that your systems do not know about. These "unknown" devices get included because they're connected to your known IDs in the Device Graph. To exclude unknown devices click Only Synced Devices.
- **Dataflow Metrics** — Evaluate your data flows with these dashboard tiles that return information about devices and their related levels of connectivity.

Create a Data Flow

When you add a new data flow most of the options you need are already set up by default. Follow these instructions to set up or review a new dataflow.

To set up a new data flow, go to Dataflow Management, click **Add new dataflow**, and review the following section preferences.

Set Up Dataflow

1. Name the data flow.
2. Provide your partner name. For example, if your company is called Big Data Company you would put this in the **Partner Name** field.
3. Select a market. Markets include **North America** (default) and **European Union**.
4. Select a delivery status. Options include:
 - **Active:** A data flow is operational and sending data.
 - **Paused** (default): Suspends sending the data flow file.
 - **Forecasting:** Returns data in the Metrics tab, but does not send you a data file.

Set Partner

Substitute the Tapad-provided cookie ID with the name of your data partner and an ID used by your systems.

1. Click the partner **Name** field and select your data partner from the options menu.
2. Provide the code or other ID your systems use to identify data from the selected partner.
3. *(Optional)* Click **Add Partner** to add additional records.

Set Format Options

Choose a format option to control file and content formatting. Format options include:

- **File format:** Receive data in a JSON object or a in a flat file (.txt).
- **Compression:** Files can contain a lot of data. You can choose to reduce file sizes with .zip, .gz, or .bz2 compression formats.
- **ID Delimiter:** Choose a character that separates individual key-value pairs (e.g., key1=value1 <tab> key2=value2).
- **Type Delimiter:** Choose a character that defines key-value data pairs (e.g., key1=value1, key1:value1, etc.).
- **Filename Macros:** Tapad names your data file by default with the partner name, type (full is the only option at this time) and a date. We recommend you use the default naming conventions.
- **Split File:** Select to break a single, large data file into multiple smaller files. Upon selection, you can specify the maximum file size (in GB).

Select Data to Include

The Select Data options form the core of the Tapad dataflow feature. By default, these settings create reports that include:

- Cookie IDs from your primary data partner.
- All primary and related hardware platforms and corresponding IDs.

As this part is bit more complicated than the others, please see [Select Data to Include](#) and [Synced Devices](#).

Set Up Server

Tapad sends data flow files to you by FTP or secure FTP (SFTP) file transfers. Add your FTP address and log on information in this section if you want to receive the data flow file.

Heads up!

Your data flow must be active to see the FTP options. Click **Active** in the Set Up Dataflow section to activate your data flow. FTP features are not available if your data flow is paused or in forecasting mode.

Select Data to Include

DRAFT Reference is [DOC-219 - Update dataflow reference re: Primary and related IDs](#) CLOSED and
[MTP-1115 - Document UI change](#) CLOSED

When creating a data flow, the Select Data to Include section provides settings that give you control over the contents of your dataflow file. For example, you can limit results by known or unknown devices, by hardware platforms, [household scores](#), and other metrics. Many of these options are selected by default when you create a new data flow. Generally, all you need to do is review these settings and change the options relevant to the type of data you want receive in the file.

The screenshot shows the 'Select Data To Include' configuration interface. At the top, there are several settings:

- Synced Devices:** Radio buttons for "All Devices (Include unsynced devices)" (selected) and "Only Synced Devices".
- Include Orphans:** A checkbox that is unchecked.
- Household Score:** A slider set to "1+".
- Optional Fields:** Checkboxes for "Display Platform Type" and "Display Household Score", both of which are unchecked.

Below these are two tabs: **PRIMARY IDS** and **RELATED IDS**. Under **RELATED IDS**, the following options are listed:

- Include Primary Tapad Cookie IDs:** Unchecked checkbox.
- Include Primary Partner Cookie IDs:** Checked checkbox.
- Include Primary Supplier IDs:** A checkbox next to "AppNexus" which is unchecked.
- Select Primary Platforms:** A dropdown menu showing "All" with a checked checkmark.
- Select Primary Hardware IDs:** A dropdown menu showing "All" with a checked checkmark.

The table below defines the available elements in the Select Data to Include section.

Control	Description
Synced Devices	Lets you control the type of data contained in the file. Options include: <ul style="list-style-type: none">All devices: (Default) Returns data about IDs that you already know about and new information about connections to unknown IDs. These "unknown" IDs get included because they're connected to IDs in the Device Graph that you already know about.Only Synced Devices: This option returns synchronized data for only those IDs that you know about and have sent to Tapad previously. See Filter By Synced Devices .
Include Orphans	Orphans are device IDs without connections to other devices.
Household Score	Scores that quantify the relationships between connected nodes in the Device Graph. Scores range from 1 (weakest connections) to 10 (strongest connections). Choosing a score limits the data to devices that are connected at the selected level or greater. See Household Scores .

Optional Fields	Selecting Display Platform Type and Display Strength Score includes that information in the data flow file. Leave these blank to return data by platform type and strength score <i>without</i> including those values in the data file. This helps keep your data files simple and easier to read. JSON and text file data <i>with</i> platform types and strength scores could look similar to the following examples: <ul style="list-style-type: none"> • JSON data: { "id_type": "HARDWARE_IDFA", "id": "fca73806-3426", "platform": "ipad", "strength": "2" } • Text file data: HARDWARE_IDFA=fca73806-3426=ipad=2 See JSON Data With Scores , and Text Data File With Scores .
Include Partner Cookie IDs	In general, you should always leave this option selected. Clear the check box only when you do not have cookie IDs to sync with Tapad.
Platforms	By default, Tapad returns data about all your platform IDs. If this is too much information, limit the amount of data returned in the file by selecting only those platforms that you're interested in. For example, selecting Computer , iPad , and Kindle would give you synched IDs for those platforms only.
Hardware IDs	By default, Tapad returns data about all your hardware IDs. If this is too much information, limit the amount of data returned in the file by selecting only those hardware IDs that you're interested in. For example, selecting Android IDs returns data for Android devices only.
Primary IDs	Options that let you control what type of primary ID data you want to receive from Tapad.
Related IDs	Options that let you control what type of related ID data you want to receive from Tapad.

Related tasks

[Create a Data Flow](#)

Filter By Synced Devices

By default, Tapad returns all synchronized data for every device mapped to the IDs you send us. This includes data for devices and IDs that your systems do not know about. These "unknown" devices get included because they're connected to your known IDs in the Device Graph. To exclude unknown devices click **Only Synced Devices**.

All devices

The **All Devices** option include IDs:

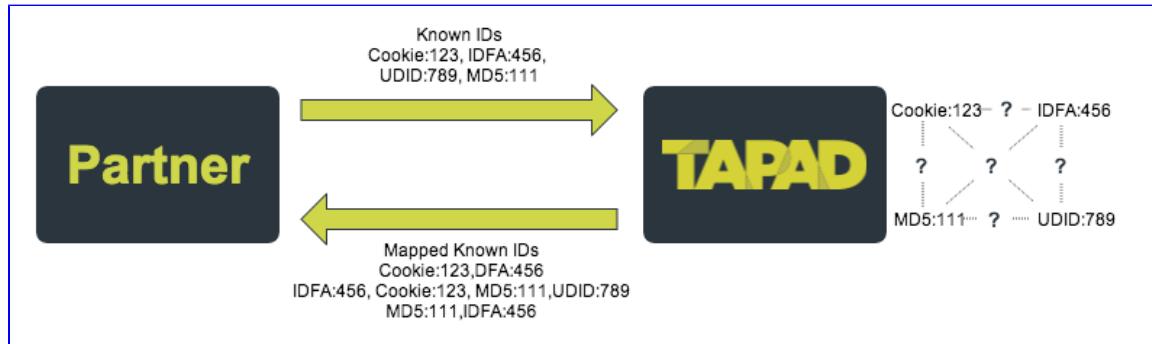
- Sent to Tapad *from* a data partner.
- Sent *from* Tapad *to* a data partner.
- Associated with a particular audience.
- Passed in on API or other data synchronization calls.

Synced devices

Sometimes you may not want to receive data for every device mapped to an ID you send us, particularly when the device is unknown to you. Tapad can exclude unknown devices from the data file and send you results for your known devices only. For example, this filter may be useful when you want to know:

- How Tapad has mapped your IDs to known IDs from other data partners you work with.
- How Your devices are connected to each other in the Tapad Device Graph.

The following diagram illustrates how the "synced devices" filter works.



Dataflow Metrics

Evaluate your data flows with these dashboard tiles that return information about devices and their related levels of connectivity.

- **Primary ID Analysis** — For all the IDs you send to Tapad, the Primary ID Analysis shows how many of them have at least 1 connection to another ID. This report returns data as the total % connected, total number of IDs sent, total count by hardware and cookie IDs, and a bar graph that displays connections by platform. You can use this data to determine the levels of connectivity throughout your entire user population and to see how many devices are available for cross-platform targeting. A good connectivity range is so
- **Connections Analysis: Cookie IDs** — A Cookie ID Connections Analysis takes the total number of connected IDs returned by the Primary ID Analysis tile and shows you how many of those have at least 1 connection to another cookie ID. This report returns the average number of connections, the total number of connections, and a bar graph that displays cookie connections by platform. You can use this data to evaluate the degree of cookie-level connectivity throughout your connected user population and to see how many devices are available
- **Connections Analysis: Hardware IDs** — A Hardware ID Connections Analysis takes the total number of connected IDs returned by the Primary ID Analysis tile <https://confluence.tapad.com/display/DOC/Primary+ID+Analysis> and shows you how many of those have at least 1 connection to another hardware ID. This report returns the average number of connections, the total number of connections, and a bar graph that displays hardware connections by platform. You can use this data to evaluate the degree of hardware-level connectivity throughout y
- **Related ID Analysis** — The Related ID Analysis tile returns a count of all the related IDs that Tapad has matched to the IDs you send us. This tile returns data by the total number of Android devices, iOS devices, and cookies that have a connection to a related ID. In addition to these hardware and cookie totals, the Related ID analysis uses a bar graph to breakdown results by hardware ID (e.g., IDFA, Android ID, Google Ad ID, etc.) and by platform type (e.g., computer, iPad, Kindle, etc.). You can use this data to de
- **Supplier ID Analysis** — The Supplier ID Analysis tile shows you how many supplier cookie IDs have connections to other supplier IDs from all of our supplier providers. It shows you the average number of connections between supplier IDs (how many are connected to each other), the total number of connected IDs, how many IDs are connected to primary IDs, and how many are connected to related IDs. Compare these results with the number of IDs in your system to see how many more connected devices you could reach when working

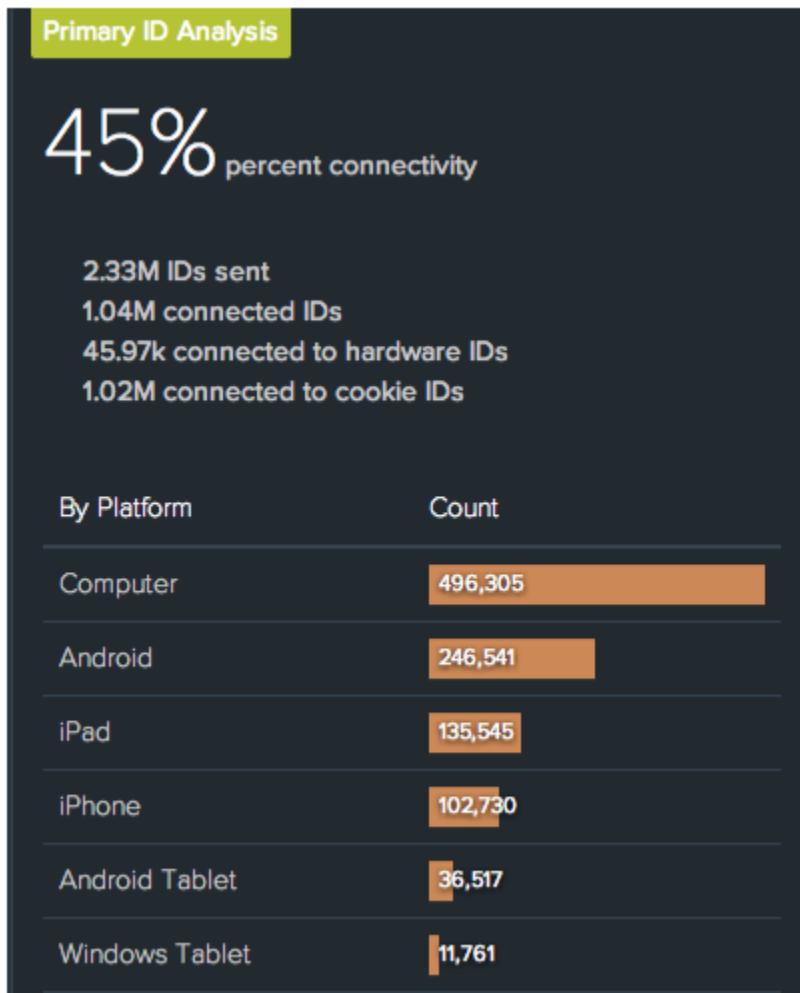
Primary ID Analysis

For all the IDs you send to Tapad, the Primary ID Analysis shows how many of them have at least 1 connection to another ID. This report returns data as the total % connected, total number of IDs sent, total count by hardware and cookie IDs, and a bar graph that displays connections by platform. You can use this data to determine the levels of connectivity throughout your entire user population and to see how many devices are available for cross-platform targeting. A good connectivity range is somewhere between 20% - 30%.

Report details

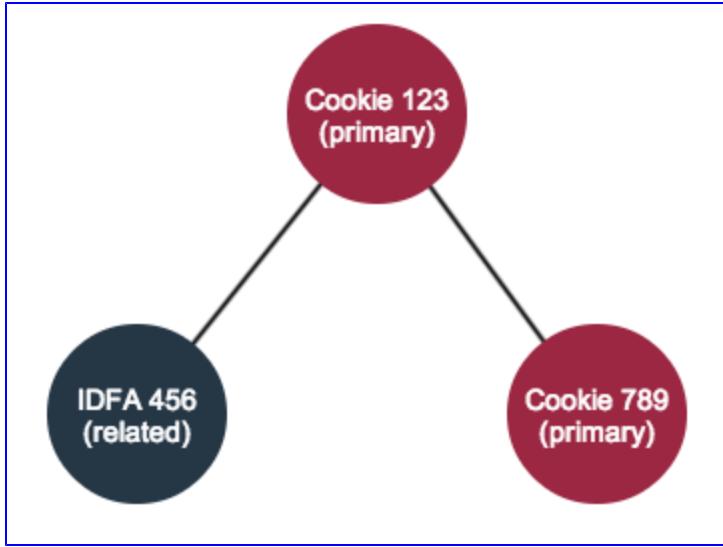
The illustration below is a shortened sample from the Primary ID Analysis report. The data in this report shows:

- 45% of the 2.33 million IDs sent to Tapad share at least 1 connection with a hardware or cookie ID.
- Almost 46 thousand primary IDs are connected to a hardware ID.
- Slightly more than 1 million primary IDs are connected to a cookie ID.
- A bar graph with connection totals by hardware platform



Hardware and cookie ID totals don't always match the total sent

In a Primary ID Analysis report, the sum of all your cookie and hardware IDs are not always equal to the total number of connected IDs. This happens because the report counts connections to primary IDs only. To help you understand how this works, take a look at the following simple device map and table. In this example, we have connections between 2 cookie IDs (primary IDs) and one IDFA hardware ID (a related ID). For more information about primary and related IDs, see [Mapping Primary and Related IDs](#).



Next, let's organize this information in a table so we can get a proper count of all the IDs.

Primary IDs	Connected IDs	Results
Cookie 123	IDFA 456, Cookie 789	<ul style="list-style-type: none"> • 1 hardware connection (Cookie 123 to IDFA 456) • 1 cookie connection (Cookie 123 to Cookie 789)
Cookie 789	Cookie 123	<ul style="list-style-type: none"> • 1 cookie connection (Cookie 789 to Cookie 123)

Because the Primary ID Analysis counts connections from primary IDs only, you would see the following results in this report:

- 2 primary IDs connected
- 1 connected hardware ID
- 2 connected cookie IDs

This gives you a total of 3 connected IDs (1 hardware + 2 cookies).

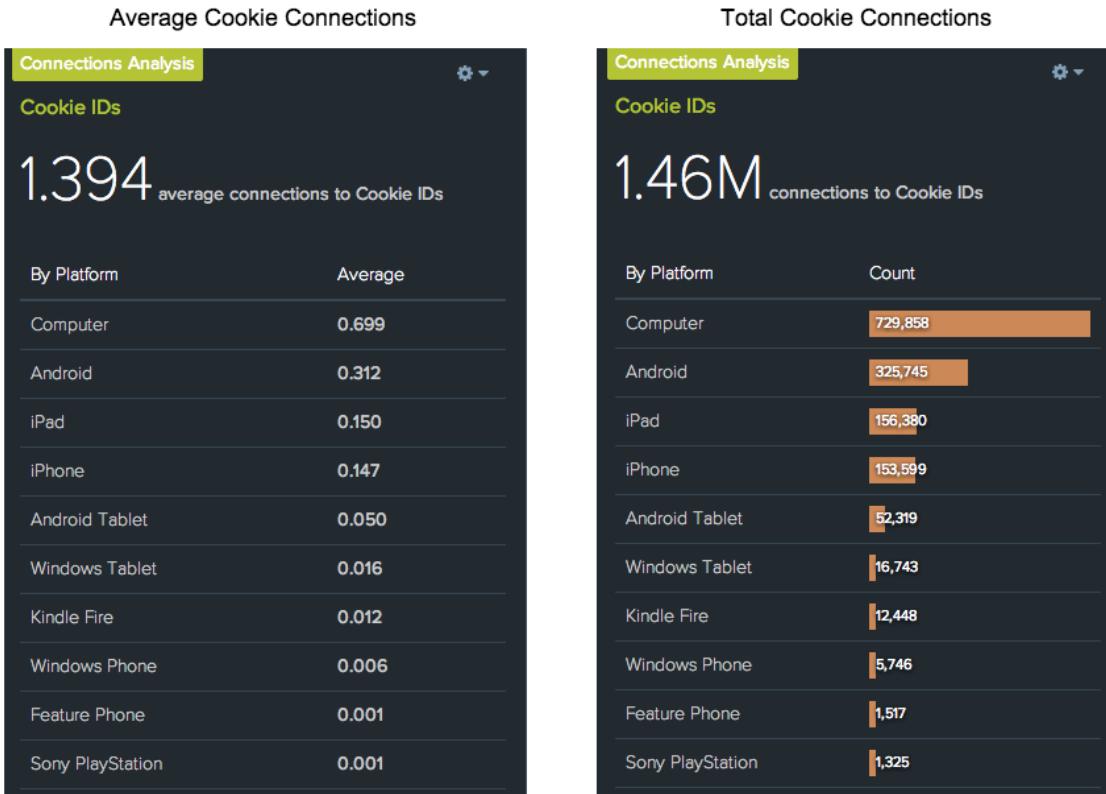
Connections Analysis: Cookie IDs

DRAFT

A Cookie ID Connections Analysis takes the total number of connected IDs returned by the [Primary ID Analysis tile](#) and shows you how many of those have at least 1 connection to another cookie ID. This report returns the average number of connections, the total number of connections, and a bar graph that displays cookie connections by platform. You can use this data to evaluate the degree of cookie-level connectivity throughout your connected user population and to see how many devices are available for cross-platform retargeting by cookies. A good, average connectivity level is > 1.x devices/cookie. Finally, when looking at a Connections Analysis report, it's important to remember this tile counts connections (or edges) between IDs *not* the individual devices (or nodes) themselves.

Report details

The illustrations below are shortened samples from the Connections Analysis report for cookie IDs. By default, this report returns the *average* number of connections between IDs. To see raw totals, hover over the top right corner and select **Count** from the options menu.



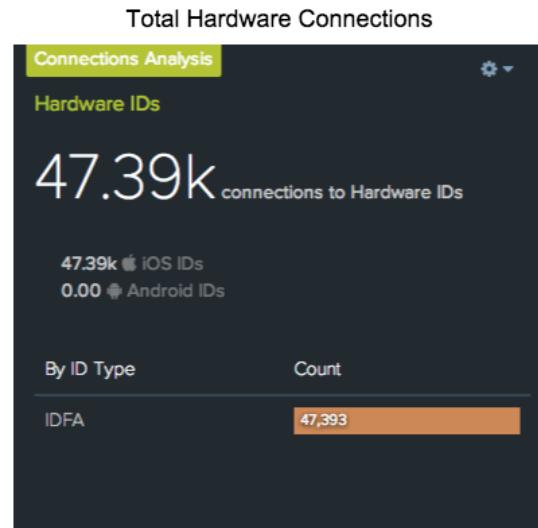
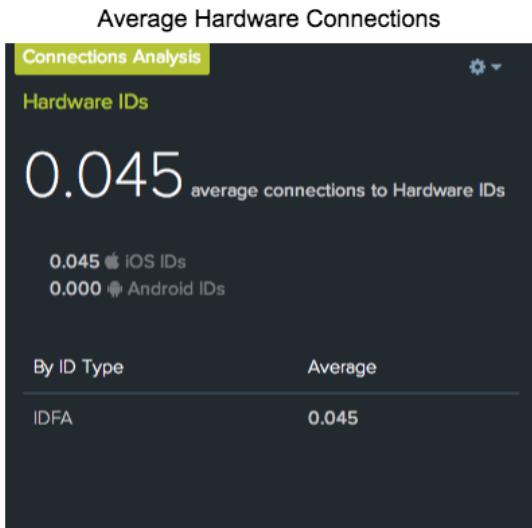
Connections Analysis: Hardware IDs

DRAFT

A Hardware ID Connections Analysis takes the total number of connected IDs returned by the [Primary ID Analysis tile](#) and shows you how many of those have at least 1 connection to another hardware ID. This report returns the average number of connections, the total number of connections, and a bar graph that displays hardware connections by platform. You can use this data to evaluate the degree of hardware-level connectivity throughout your connected user population and to see how many devices are available for cross-platform retargeting by platform. A good, average connectivity level is > 1.x connection for each hardware ID. Finally, when looking at a Connections Analysis report, it's important to remember this tile counts connections (or edges) between IDs *not* the individual devices (or nodes) themselves.

Report details

The illustrations below are shortened samples from the Connections Analysis report for hardware IDs. By default, this report returns the *average* number of connections between IDs. To see raw totals, hover over the top right corner and select **Count** from the options menu.



Related ID Analysis

The Related ID Analysis tile returns a count of all the related IDs that Tapad has matched to the IDs you send us. This tile returns data by the total number of Android devices, iOS devices, and cookies that have a connection to a related ID. In addition to these hardware and cookie totals, the Related ID analysis uses a bar graph to breakdown results by hardware ID (e.g., IDFA, Android ID, Google Ad ID, etc.) and by platform type (e.g., computer, iPad, Kindle, etc.). You can use this data to determine the levels of connectivity throughout your entire user population and to see how many related IDs are available for cross-platform targeting.

Report details

The illustrations below show sample results from the Related ID Analysis report. This report counts hardware connections (iOS and Android) by default. To see a count of cookie connections, hover over the top right corner and select **Cookie** from options menu. The sample reports show that Tapad has matched your primary IDs to approximately 33 million related IDs. Of that total:

- 12.5 million IDs are iOS IDs.
- 6.6 million IDs are Android IDs.
- 14 million IDs are cookie IDs.

Hardware Connections to Related IDs



Cookie Connections to Related IDs



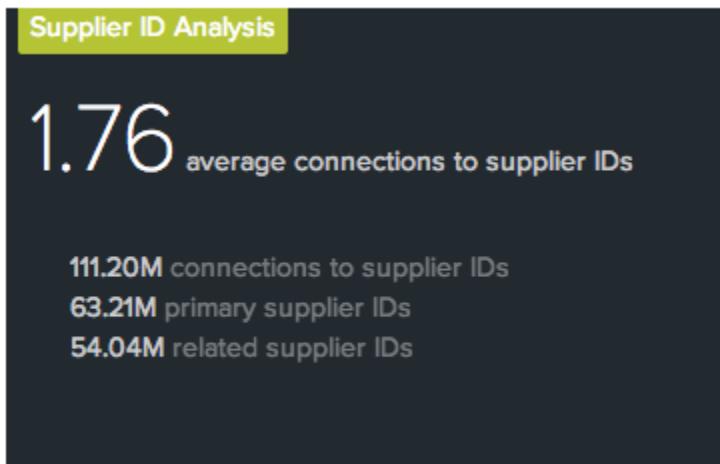
Supplier ID Analysis

DRAFT

The Supplier ID Analysis tile shows you how many supplier cookie IDs have connections to other supplier IDs from all of our supplier providers. It shows you the average number of connections between supplier IDs (how many are connected to each other), the total number of connected IDs, how many IDs are connected to primary IDs, and how many are connected to related IDs. Compare these results with the number of IDs in your system to see how many more connected devices you could reach when working with Tapad.

Report details

The illustration below shows a sample Supplier ID Analysis report. For more information about primary and related (connected) IDs, see [this section](#) on counting IDs.



ID Syncs Management

DRAFT

Reference is [SM-1058](#) - Add to ID Sync UI set up docs RESOLVED

ID Syncs Management provides settings that let you start ID syncs, configure storage, set up redirects, specify endpoint URLs, and other options. To manage ID synchronization, go to ID Syncs Management, use the search feature to find your advertiser, and configure required settings in the following sections.

Prerequisites: Before you can configure ID synchronization, your client's advertiser must be set up in Backoffice. See [Create an Advertiser](#) to add a new advertiser.

General settings

All of the General settings are required, except for Overwrite Generated Endpoints.

The screenshot shows a dark-themed configuration interface for 'General' settings. At the top, there is a 'SETTINGS' header. Below it, the 'General' section is selected. The interface contains several input fields with dropdown menus:

- Advertiser**:
 - Overwrite Generated Endpoints**: Options: Yes (selected), No
 - Is Realtime ***: Options: Yes (selected), No
 - Sync Initiation ***: Options: Tapad, Client, Both
 - Mapping Storage ***: Options: Tapad, Client, Both
 - Supports Redirect ***: Options: Yes (selected), No
 - Redirect Explicit or Internal ***: Options: Explicit (selected), Internal
- Client Macro**: Value: {PARTNER-COOKIE-ID}

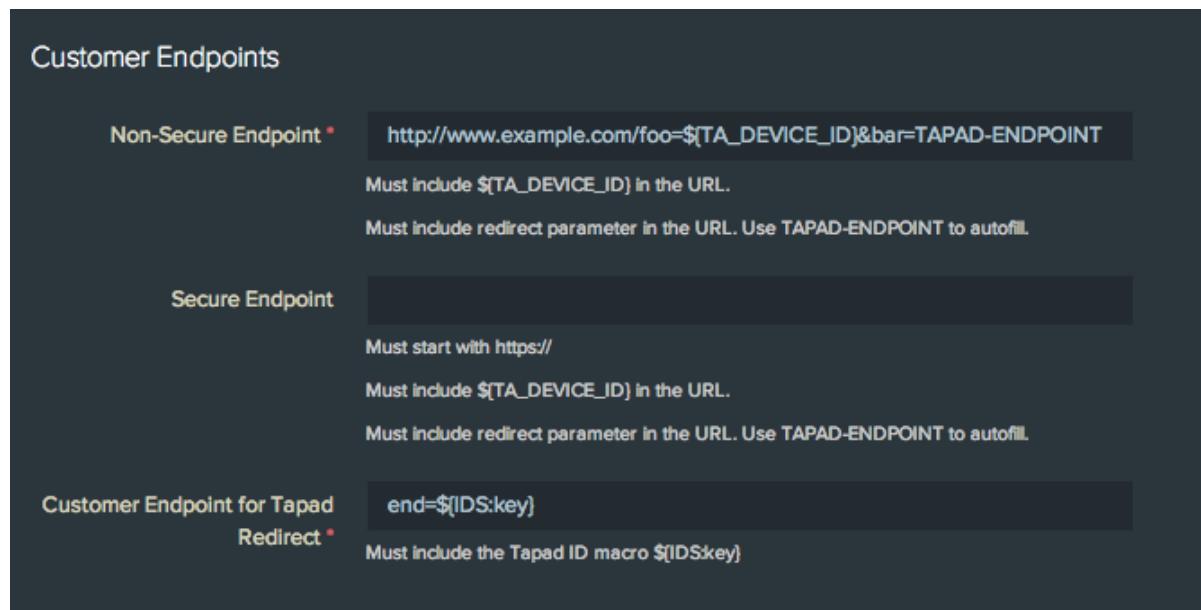
The table below describes the general settings you need to set up an ID sync.

Setting	Description
Is Realtime	Options include: <ul style="list-style-type: none">Yes: Synchronizes IDs in real time via direct, server-to-server data transfers.No: Synchronizes IDs using batch files that transmit data at set intervals.

Sync Initiation	Specify who starts the ID synchronization process, Tapad or the advertiser. Tapad recommends that <i>both</i> parties initiate ID synchronizations. Options include: <ul style="list-style-type: none"> • Tapad: Tapad initiates the ID synchronization process. • Client: The client initiates the ID synchronization process. • Both: (<i>Recommended</i>) Both parties initiate an ID sync. This option helps ensure the maximum amount of data gets exchanged during the ID sync.
Mapping Storage	Specify who stores synchronized data, Tapad or the advertiser. Tapad recommends that <i>both</i> parties store the synchronized data. Options include: <ul style="list-style-type: none"> • Tapad: Tapad stores the data. • Client: The client stores the data. • Both: (<i>Recommended</i>) Both parties store the data.
Supports Redirect	Confirms that the client can do an ID redirect. Select Yes if Tapad initiates the ID sync and stores the ID mapping.
Redirect Explicit or Internal	Specifies how a redirect works. Options include: <ul style="list-style-type: none"> • Explicit: Used when the redirect is contained in a URL string. • Internal: Used when a web server takes care of the redirect on the backend.
Client Macro	Type the client ID. This value appears in the Tapad endpoints URL field.

Customer endpoints

The endpoint fields let you specify where to send synchronized ID data. These fields are dynamic and change based on the sync initiation and mapping storage choices you make in the General section.



The table below describes the customer endpoint fields and macros.

Endpoint type	Description
Non-secure	A standard http URL endpoint for receiving ID sync data. A non-secure endpoint URL must include a placeholder variable for: <ul style="list-style-type: none"> • The macro \${TA_DEVICE_ID}. This automatically inserts the Tapad cookie ID into the endpoint URL. • The Tapad redirect parameter TAPAD-ENDPOINT. This expands in the endpoint URL fields to include the Tapad redirect URL in your endpoint. Example: <code>http://www.sample.com/foo=\${TA_DEVICE_ID}&bar=TAPAD-ENDPOINT</code>

Secure	<p>A secure <code>https</code> URL endpoint for receiving ID sync data. A secure endpoint URL must include placeholder variables for:</p> <ul style="list-style-type: none">• The macro <code> \${TA_DEVICE_ID}</code>. This automatically inserts the Tapad cookie ID into the endpoint URL.• The Tapad redirect parameter <code>TAPAD-ENDPOINT</code>. This expands in the endpoint URL fields include the Tapad redirect URL in your endpoint. <p>Example:</p> <pre>https://www.sample.com/foo=\${TA_DEVICE_ID}&bar=TAPAD-ENDPOINT</pre>
Customer endpoint for Tapad redirect	<p>The endpoint Tapad should use when redirecting ID sync data back to the client. A customer endpoint URL must include a placeholder variable for the ID sync macro <code> \${IDS:key}</code>.</p> <p>Example:</p> <pre>var=\${IDS:key}</pre> <p>See Tapestry \${DATA} or \${IDS} Macro for more information.</p>

Endpoint URLs

Endpoint URLs, including the expanded macros, appear in the Customer Endpoints and Tapad Endpoints fields as shown below.

- **Customer Endpoints:** The customer endpoint URL automatically configures Tapad systems to send synchronized ID data to customer based on the previous settings.
 - **Tapad Endpoints:** Send the Tapad endpoint URL to the customer so they can send synchronized ID data to Tapad.

Customer Endpoints	
	NON-SECURE SECURE
1	<code>http://www.example.com/foo=\${TA_DEVICE_ID}&bar=http%3A%2F%2Ftapestry.tapad.com%2Ftapestry%2F1%3Fta_partner_did%3D1234%26ta_format%3Dpng</code>
Tapad Endpoints	
	NON-SECURE SECURE
1	<code>http://tapestry.tapad.com/tapestry/1?ta_partner_did=1234&ta_redirect=http%3A%2F%2Fwww.example.com%2Ffoo%3D%24%7BIDS%3Akey%7D</code>

Supplier Health Dashboard

Monitor data for bids, impressions, CPM, CTR, margin %, and the % of bids with a device ID. The Dashboard displays data in near real time and generates an alert if performance falls below or above a predicted range for the supplier.

Purpose

The Supplier Health Dashboard helps you understand how all your assets are performing across different *supply side platforms* (SSPs). By checking the dashboard you can:

- Monitor supplier health on a near real-time basis. Data is refreshed at 1-hour intervals.
 - Identify and respond to problems quickly with alerts. The dashboard uses unique data visualization techniques to flag supplier metrics that return data outside of a predicted range of expected behavior.
 - Develop insight on supplier activity without creating and analyzing complex reports.
 - Help improve margin through better supply evaluation.
-
- **Data Visualization in the Supplier Health Dashboard** — The Supplier Health Dashboard displays supply partner data with sparklines and as percentages. Scroll through the list for an at-a-glance overview of supplier performance or click on each data point to view detailed results in a pop-up window. Data is updated hourly. For more information about each metric, see [Supplier Health Dashboard: Available Data](#).
 - **Supplier Health Dashboard: Available Data** — Lists and describes the data returned by the Supplier Health Dashboard. Data is updated hourly.

Data Visualization in the Supplier Health Dashboard

The Supplier Health Dashboard displays supply partner data with sparklines and as percentages. Scroll through the list for an at-a-glance overview of supplier performance or click on each data point to view detailed results in a pop-up window. Data is updated hourly. For more information about each metric, see [Supplier Health Dashboard: Available Data](#).

Sparklines

Sparklines are small line graphs. Generally, they show data inline with text and don't display information in relation to an x or y-axis. The Supplier Health Dashboard uses sparklines to display data for:

- Bid requests
- Impression counts
- CPM data
- CTR data

Additionally, shaded areas inside the sparklines define the predicted upper and lower boundaries for each metric. Click a sparkline to open a pop-up window that displays detailed data for the selected metric.

Percentages

The Supplier Health Dashboard results as a percent for:

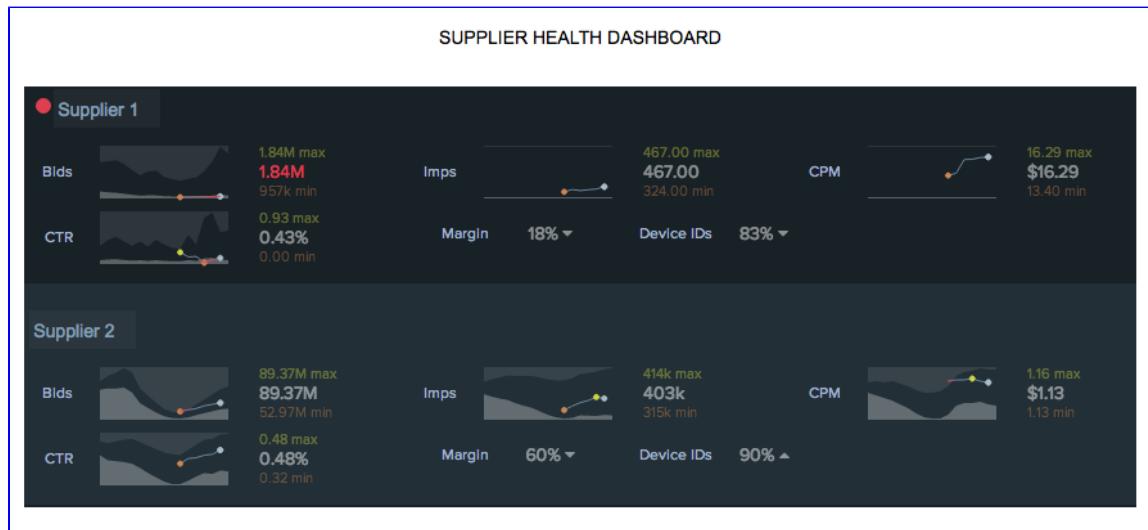
- Margin
- Device IDs

Click margin or device ID results to open a pop-up window that displays detailed data for the selected metric.

Colors

Dashboard data is color coded to provide visual performance cues.

- Normal and expected results are gray.
- Red indicates an error.
- Green and brown indicate maximum and minimum predicted values respectively.



Supplier Health Dashboard: Available Data

Lists and describes the data returned by the Supplier Health Dashboard. Data is updated hourly.

Supplier Health Dashboard displays data for the following metrics.

Report	Description	Why is this useful?
Bids	Shows bid requests from monitored suppliers.	<p>Monitoring bid requests helps ensure that data integrations are working properly.</p> <ul style="list-style-type: none"> A decrease in bid requests could indicate an integration problem. An increase in bid requests should prompt you to check with the SysOps team to make sure we can handle the additional traffic. <p>The dashboard displays an alert when bid requests fall outside of the predicted range for that supplier.</p>
Impressions	Number of times an ad has been served.	Impression tracking is crucial for accurate billing on CPM campaigns. The dashboard displays an alert when impression amounts fall outside of the predicted range for that supplier.
CPM	Calculated as: $CPM \text{ price} = \text{Revenue} / (\text{Impressions} / 1000)$	Most Tapad campaigns are flat-rate CPM campaigns. As a result, what we bill clients does not change. However, if suppliers increase their CPM, our margins decrease and we could lose money. This metric monitors suppliers CPMs to make sure they remain consistent. The dashboard displays an alert when CPM amounts fall outside of the predicted range for that supplier.
CTR	Calculated as: $\text{CTR \%} = (\text{Clicks} / \text{Impressions}) \times 100$	Clients use CTR as a key performance evaluation metric. It's important to accurately track clicks across all of our suppliers because CTR amounts outside of a predicted range could indicate a technical problem with our click-trackers or click fraud. The dashboard displays an alert when CTR amounts fall outside of a predicted range for that supplier.
Margins	Calculated as: $\text{Margin \%} = ((\text{Revenue} - \text{Cost}) / \text{Revenue}) \times 100$	Monitoring margin helps ensure that a campaign does not lose money because of increased supplier costs. The dashboard displays an alert when margin amounts fall outside of a predicted range for that supplier.
Device IDs	% of bid requests that contain a device ID.	<p>Device IDs are required for attribution. Monitoring the device ID % can tell us if something is wrong with an integration or a supplier. Attribution suffers if the device ID % declines. A device ID includes identification mechanisms such as:</p> <ul style="list-style-type: none"> Tapad cookies Supplier cookies Hardware IDs and user agents <p>The dashboard displays an alert when the device ID % falls outside of a predicted range for that supplier.</p>

Related concepts

[Data Visualization in the Supplier Health Dashboard](#)

Budgets, Spending, and Pacing

A detailed examination of how Tapad systems and software manage budgets, spending, and pacing processes.

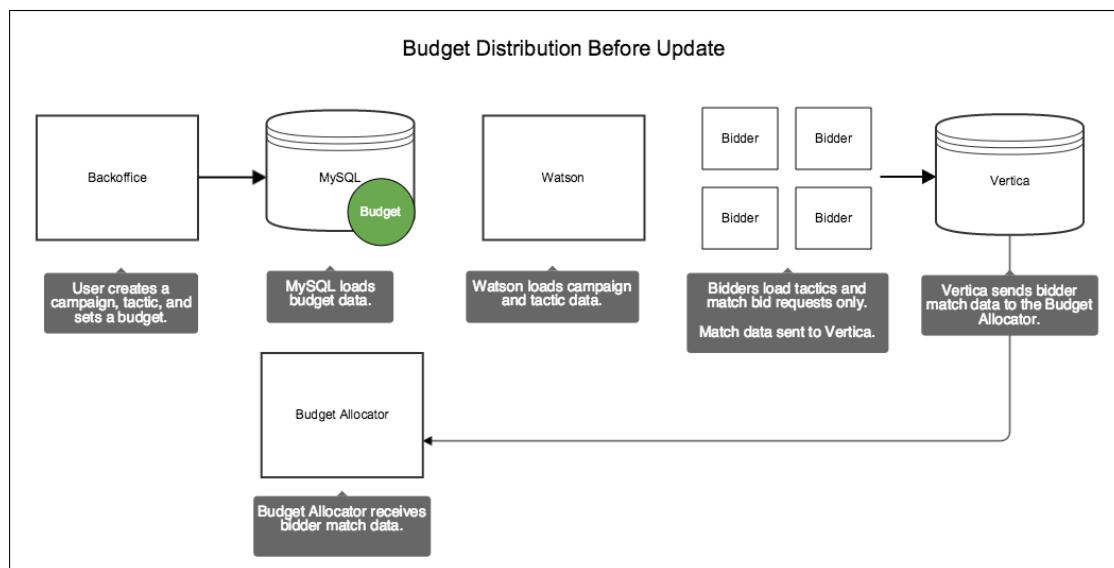
- **Understanding Budget Distribution and Update Cycles** — The budget allocator, MySQL, Watson, Vertica, and the bidders are important components that manage and distribute budgets. These systems work together, but the budget allocator and the bidders update budget data on separate schedules. Let's take a look at how their independent operations affect budget data updates and distribution.
- **Budget Distribution Between and Within Data Centers** — Each Tapad data center receives campaign budget in direct proportion to the bid requests matched by their bidders. Within a data center, hardware differences between servers affects how much budget each bidder receives.
- **How the Budget Update Process Can Delay Tactic Delivery** — Asynchronous budget updates can delay tactic delivery from 10 minutes to an hour. However, you can force a budget update if you change and save any setting for a campaign or tactic.
- **Pacing: Everything You Wanted to Know but Were Afraid to Ask** — An overview of how Tapad manages pacing to help ensure smooth, even spending and delivery during the lifetime of a campaign.

Understanding Budget Distribution and Update Cycles

The budget allocator, MySQL, Watson, Vertica, and the bidders are important components that manage and distribute budgets. These systems work together, but the budget allocator and the bidders update budget data on separate schedules. Let's take a look at how their independent operations affect budget data updates and distribution.

Budget distribution before updates

In Backoffice, a MySQL database stores budget cap information for each tactic. However, budgets aren't immediately available to the bidders. Another system, the budget allocator, needs to run first and send bidder match rate data to Watson. Match rates let Watson assign budget amounts to each bidder in proportion to how many bids they've matched given the constraints set by a tactic. If the bidders don't have a budget, they'll just send match rate data to Vertica where it's passed on to the budget allocator. In the illustration, the green circle shows you how budget information is distributed before the budget allocator or the bidders update themselves.

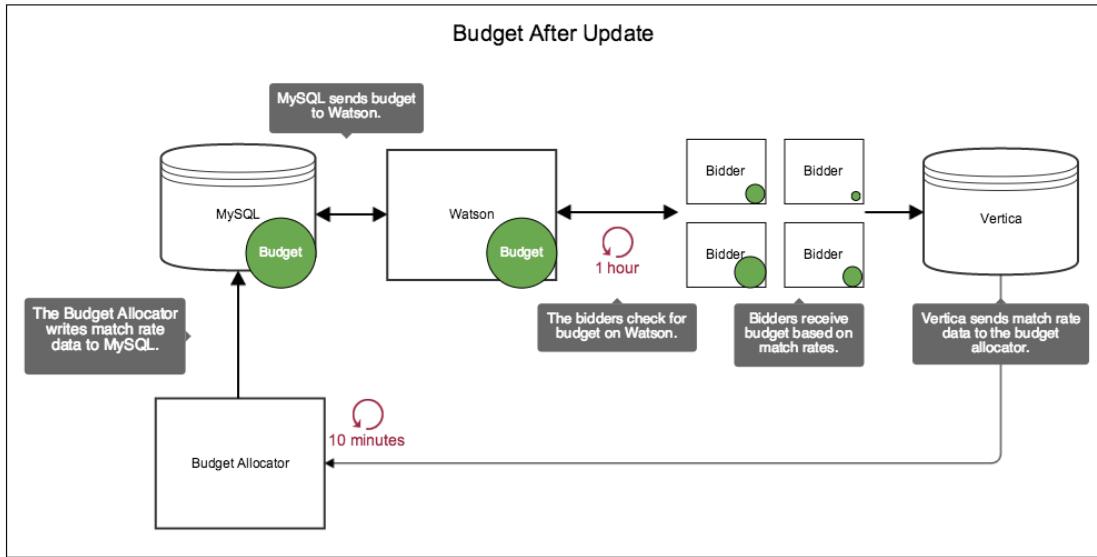


Post-update budget distribution

Now let's take a look at how budget data through our technology stack *after* an update.

System	Description
Budget allocator	At 10-minute intervals, the budget allocator checks Vertica for bidder match rate data. It applies a factor to this data, which sets the budget for each bidder based on their performance. This factor can be expressed as: LaTeX Math rendering error (show error message). The budget allocator sends this information to MySQL where it's passed on to Watson.
Watson	Watson is the authoritative source of budget information for the bidders. In this system, Watson: <ul style="list-style-type: none">Allocates budgets to the bidders.Tracks how much they spend.Stops the bidders when they've exhausted their budget. For more information about how Watson manages budgets, see Pacing: Everything You Wanted to Know but Were Afraid to Ask .
Bidders	At 1-hour intervals, the bidders load/reload all the available campaign data from Watson. This ensures they have the latest budget and campaign data. Also, this update takes bidder performance into account. As a result, the bidders receive budget amounts in proportion to their match rates. Bidders that match a lot of requests receive more budget than those with fewer bid requests. Finally, the bidders continue to send match rate data to Vertica, which is passed on to the budget allocator.

In the illustration, the green circle shows you how budget information is distributed after the budget allocator and the bidders update themselves.



Note that these separate update intervals can prevent a tactic from serving from 10 minutes up to an hour. For more information about this delay and how to force an update, see [How the Budget Update Process Can Delay Tactic Delivery](#).

Budget Distribution Between and Within Data Centers

Each Tapad data center receives campaign budget in direct proportion to the bid requests matched by their bidders. Within a data center, hardware differences between servers affects how much budget each bidder receives.

Match rates determine budget distribution between data centers

Budget management is easy when you have only one data center. In this scenario, the bidders receive all the available budget. But Tapad has several regional data centers, not just one. This design makes systems fault tolerant and responsive to bid requests, but also exposes the limitations of even budget distribution.

Budget management is a challenge when you have multiple data centers. In this scenario, even budget distribution is inefficient because one data center may receive more bid requests than the others. Also, even distribution strands budget in data centers that don't get many (or any) bid requests. To solve this problem, Tapad allocates budgets to each data center in direct proportion to their *match rate*. The match rate measures how often the bidders in a data center can fulfill a bid request given the constraints set by each tactic. If a data center and its bidders match more bids, they receive a larger share of the available budget. This is a dynamic method of budget allocation that can shift resources between data centers and bidders in response to demand.

In the Tapad technology stack, the budget allocator calculates the match rate and Watson manages and distributes budgets to the bidders. For more information about how these systems work, see [Understanding Budget Distribution and Update Cycles](#).

Server hardware affects budget distribution between bidders

Hardware also affects budget distribution within a data center. For example, bidders running on more powerful servers have better match rates than bidders running on slower machines. As a result, bidders on more powerful machines receive a larger share of the budget than those running on slower servers.

How the Budget Update Process Can Delay Tactic Delivery

Asynchronous budget updates can delay tactic delivery from 10 minutes to an hour. However, you can force a budget update if you change and save any setting for a campaign or tactic.

Budget allocator and bidder update cycles are not synchronized

Budget allocators and bidders update budget information at different times. The budget allocators check for budget at 10-minute intervals. Also, the allocators distribute budgets to the bidders indirectly through a MySQL database and another system called [Watson](#). The bidders check Watson for budget at 1-hour intervals. A new tactic starts delivering ads once the bidders have a budget. However, ad delivery depends on where the allocators and bidders are in their respective update cycles. These separate update cycles can delay ad delivery from 10 minutes to an hour.

If the bidders check for a budget amount:

- *Before* the budget allocator, the tactics *may not* deliver ads. In this case, the budget allocator might not have had enough time to update and distribute budget before the bidder update cycle runs.
- *After* the budget allocator, the tactics *may* deliver ads. In this case, the budget allocator might have had enough time to update and distribute budget before the bidder update cycle runs.

For details about budget updates work, see [Understanding Budget Distribution and Update Cycles](#).

Force a budget update

You can force an out-of-cycle budget update if you need a tactic to start bidding right away. To do this, simply make *any* change to the campaign or tactic and save that change. For example, you could add a space to a campaign's description field and save it. Saved changes prompt budget allocators and the bidders to check for new or changed budgets outside of their scheduled update intervals.

Pacing: Everything You Wanted to Know but Were Afraid to Ask

An overview of how Tapad manages pacing to help ensure smooth, even spending and delivery during the lifetime of a campaign.

Overview

Pacing is the rate at which a campaign spends budget during its lifetime. Tapad controls pacing at the tactic level. Campaigns and line items may affect pacing, but only when their budgets or flight dates act as a cap on the tactic (see [Budget Caps and Flight Dates for Campaigns, Line Items, and Tactics](#)). It is challenging to manage pacing so tactics spend budget in a consistent manner instead of all at once or at random. To keep pacing even, Tapad divides the flight time of a tactic in to 5-minute segments. Each segment gets an equal budget amount. Uniform budget distribution creates even pacing, which helps tactics bid and win auctions throughout their designated lifetime.

Types of pacing

Although our bidding system tries to spend efficiently, factors like small budgets, constrained inventory, limited audience size, restrictive targeting, or deliberate business strategies could cause a tactic to deviate from an even pacing model. There are 3 main types of pacing behavior given these conditions:

- **Even pacing:** With even pacing, a tactic spends budget at a consistent rate. As a result, it runs out of budget and stops bidding as expected.
- **Over pacing:** With over pacing, a tactic spends budget too quickly. As a result, it runs out of budget early and stops bidding before the last day of its scheduled flight.
- **Under pacing:** With under pacing, a tactic spends budget too slowly. As a result, it bids less often, delivers fewer impressions, and has budget remaining on the last day of its scheduled flight.

The following graphs illustrate these different pacing conditions. In each, the red line represents how a tactic should pace to spend budget evenly during its lifetime. The green line represents actual spending. To pace evenly, the green line should closely follow the slope of the red line. Deviations above or below red line indicate over or under pacing conditions respectively.



System-level pacing management

The following table describes the systems (software) that help manage pacing and keep delivery smooth over the lifetime of a campaign and tactic.

System	Description
Bidders	Bidders respond to bid requests and spend budget. They are geographically split and load-balanced between the east and west coasts. They receive budget amounts from the budget allocators and report spending to Watson. For more information about the bidders, see Understanding Budget Distribution and Update Cycles .
Budget allocators	The budget allocators distribute budget to the bidders. This system can assign budget to the bidders proportionally in response to geographic demands or match rates on the bidders. Basically, this system makes sure that bidders that see a lot of requests get more budget than bidders with fewer matches. This gives the active bidders a chance to respond effectively to bid requests and prevents budgets from being assigned to bidders that are not active. For more information about the budget allocators, see Understanding Budget Distribution and Update Cycles .
Watson	Watson is the source of truth regarding available budget for the bidders. The bidders send spending data to Watson. Watson tracks spending and tells the bidders when to stop because they've used up their allocated budget. See also, Watson .

Budget-level pacing management

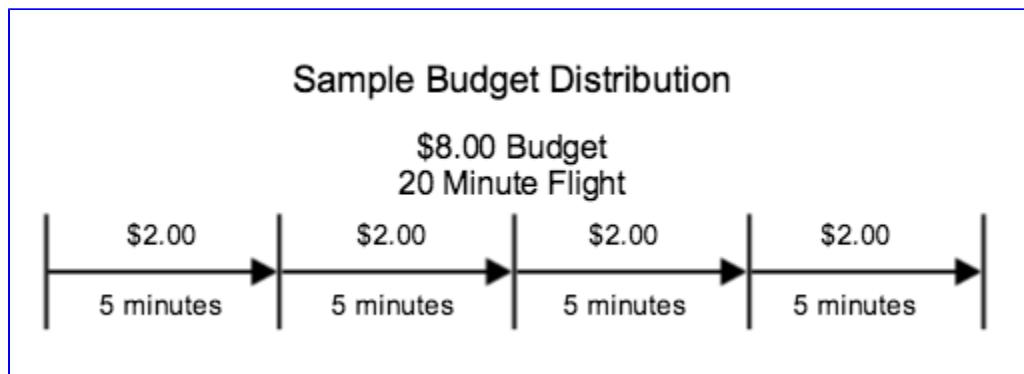
In addition to system-level components like Watson and the budget allocators, budget management and distribution logic also helps maintain even pacing during the lifetime of a campaign. There are 3 main budget management processes that help keep pacing even. These include:

- **Initial distribution:** In each campaign, Tapad allocates budgets evenly in 5-minute increments across the flight dates of a tactic. These 5-minute increments are known as *bid segments*.
- **Recalculate and redistribute:** After a bid segment ends, the left over budget is recalculated and distributed evenly across the remaining bid segments for that tactic. To recalculate the budget, we use the formula $\text{budget} - \text{spend} / \text{remaining bid segments}$.
- **Determine bid rate:** The bid rate is determined by looking at the number of bid requests a tactic received, the number of times it responded, and how many times it won. The bid rate is important because it is used by the tactic in subsequent bid segments to help it spend budget evenly during that 5-minute interval. This is less significant for flat, CPM pricing, but is key to other bid processes such as [Tiger Pricing](#).

Let's take a look at how this works in the following examples.

Distribute budget

Before we can spend budget, it has to get loaded into the bidders. This process divides a tactic's start and end times into 5-minute bid segments and distributes the available budget among them. As an example, let's use a [model](#) with an \$8.00 budget that's set to run for 20 minutes. This gives us four 5-minute segments, each with a \$2.00 budget. At the start of this campaign, before any bidding or spending, our budget is evenly distributed as shown below.



Spend, recalculate, and redistribute budget

After the budget is loaded and distributed, the campaign goes live and starts bidding. However, sometimes the bidders can't spend all of their allocated budget within a 5-minute bid segment. This could happen because of restrictive targeting, scarce inventory, or small audiences and budgets. Regardless of the cause, we have to do something with this unused budget. To manage this extra money, at the end of each 5-minute interval, the bidders recalculate the budget and redistribute it equally across the remaining bid segments. As an example, let's take the previous campaign and say it spent \$1.50 in the first 5-minute segment. The following table shows how the new budget is calculated.

Step	Process
Calculate new budget	<ul style="list-style-type: none">• Budget - Spending• $\\$8.00 - \\$1.50 = \\$6.50$
Redistribute budget	<ul style="list-style-type: none">• Budget / # of 5 minute increments• $\\$6.50 / 3 = \\2.16

Each of the three remaining 5-minute bid segments get \$2.16 budget as shown below.

Sample Budget Redistribution

\$6.50 Recalculated Budget
15 Minutes Remaining in Flight



Equal budget distribution and redistribution helps keep pacing smooth and even throughout the campaign's lifetime.

Sources of underpacing

As shown in the [pacing examples](#), underpacing is when a campaign performs below expectations and ends with surplus budget. Underpacing can be worse than overpacing because if campaigns can't bid, they miss opportunities to reach the desired audience over time. Tapad's bidding system can correct for underpacing, but sometimes conditions are such that a tactic just will not pace evenly. The following table includes some of the common factors responsible for underpacing.

Sources of underpacing	Description
Supply constraints	These elements limit the bid opportunities available to a specific tactic and to a campaign overall: <ul style="list-style-type: none">• Trying to reach very small or extremely specific audiences.• Excessive or restrictive targeting requirements.• Scarce publisher inventory.
Small budgets	Small budgets contribute to underpacing <i>and</i> erratic delivery. With an inadequate budget, a tactic (and the larger campaign) can sit idle for some of its lifetime without bidding at all. This happens because a limited budget is reduced further as it gets distributed across all the available bid segments. As a result, a tactic cannot bid enough to win during one or more segments. When a bid interval ends without spending anything, the left over budget is redistributed. Eventually, a segment accumulates enough budget to bid, but this spending again reduces the available budget below a meaningful threshold, which pushes a campaign back into dormancy. As a result, the tactic has to waste additional time trying to accumulate enough budget to bid and you get odd cycles of activity and quiescence during the lifetime of a tactic or campaign.

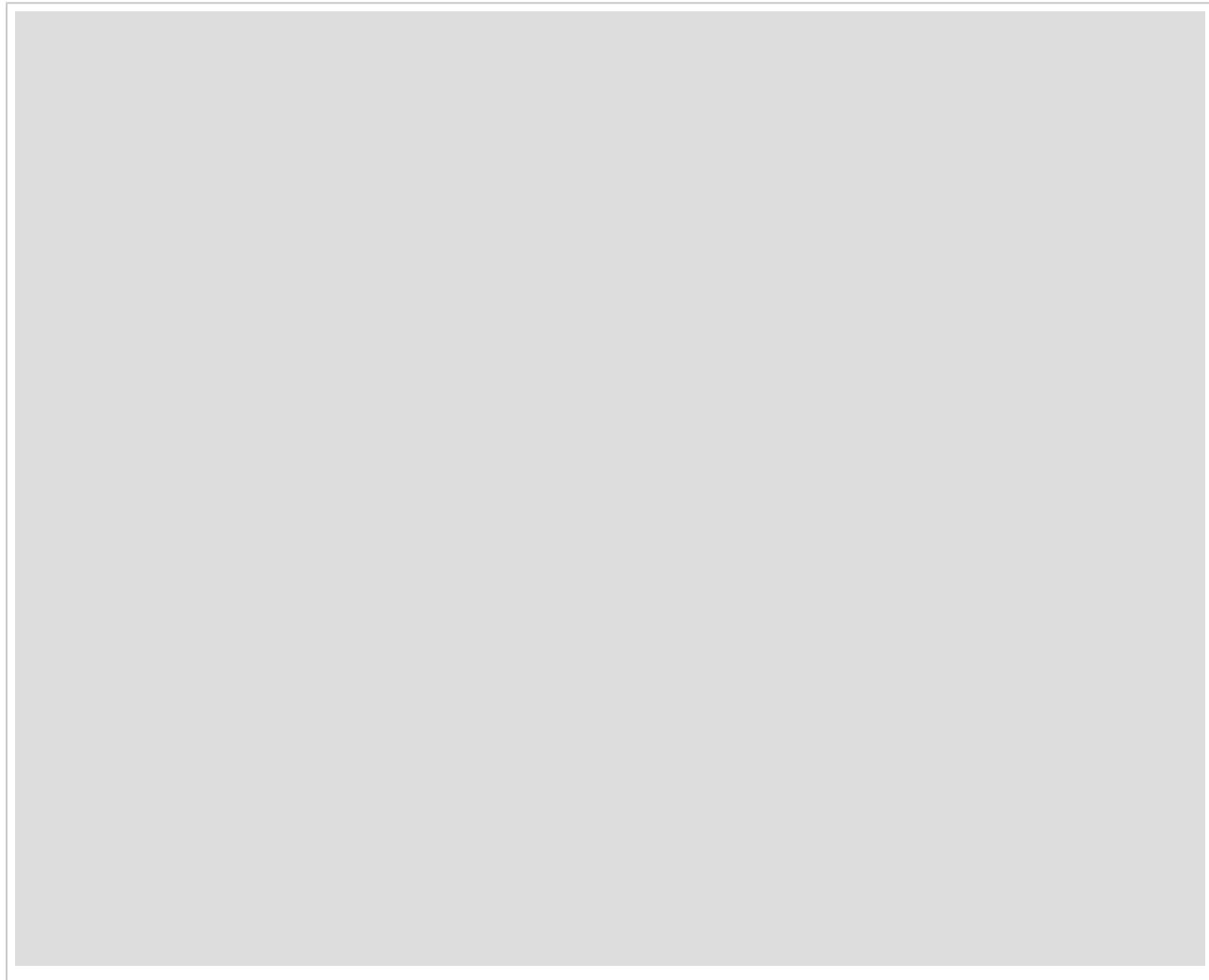
Related concepts:

- [Real Time Bid Optimization with Smooth Budget Delivery in Online Advertising](#)
- [Bid Optimization](#)
- [dCPM Pricing: About](#)
- [Tiger Pricing: About](#)
- [Budget Caps and Flight Dates for Campaigns, Line Items, and Tactics](#)

Marketplace Buyer

Products, features, and related material.

- **Programmatic Real Time Bidding Access** — Tapad's Programmatic Real Time Bidding Access product enables ad networks, demand side platforms and other programmatic buying entities with unique offerings to bid on advertising media while also gaining valuable access to Tapad's Device Graph™ in real time



Programmatic Real Time Bidding Access

Tapad's Programmatic Real Time Bidding Access product enables ad networks, demand side platforms and other programmatic buying entities with unique offerings to bid on advertising media while also gaining valuable access to Tapad's Device Graph™ in real time

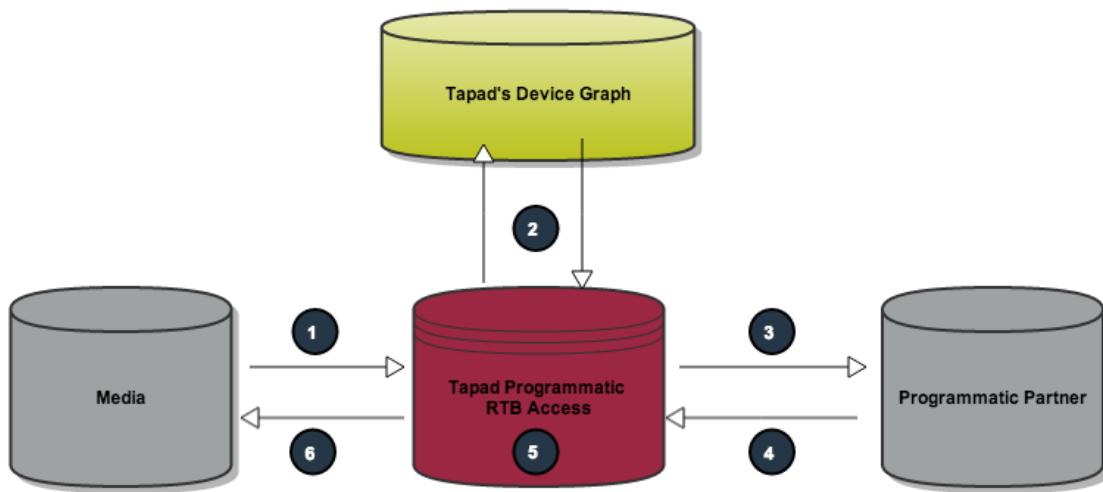
- **Programmatic RTB Access Introduction** — Tapad's Real Time Bidding API enables ad networks, demand side platforms and other programmatic buying entities (collectively known as "Programmatic Partners") with unique offerings to bid on advertising media while also gaining valuable access to Tapad's Device Graph in real time.
- **RTB Access Integration Process** — The integration process with Tapad's Programmatic RTB Access product includes ID synchronization, audience ingestion, creative setup, and integration with the Open RTB 2.1 spec with Tapad's extensions. A Tapad integration expert can help answer questions during setup.
- **Technical Specifications for Programmatic RTB Access** — Tapad Exchange is a way for programmatic buyers to access Tapad's proprietary cross-platform data while also accessing media in real time. The technical specifications are largely based on the OpenRTB 2.1 <http://www.iab.net/media/file/OpenRTB-API-Specification-Version-2-1-FINAL.pdf>, with some clarifications and extensions added.
- **Creative Formats and Creative Approval API** — Programmatic partners are required to abide by Tapad's creative specifications and are required to submit creatives via Tapad's creative approval API.
- **Programmatic RTB Access Version and Revisions** — This page will designate the current version of the spec and any revisions made to prior specs.

Programmatic RTB Access Introduction

Tapad's Real Time Bidding API enables ad networks, demand side platforms and other programmatic buying entities (collectively known as "Programmatic Partners") with unique offerings to bid on advertising media while also gaining valuable access to Tapad's Device Graph in real time.

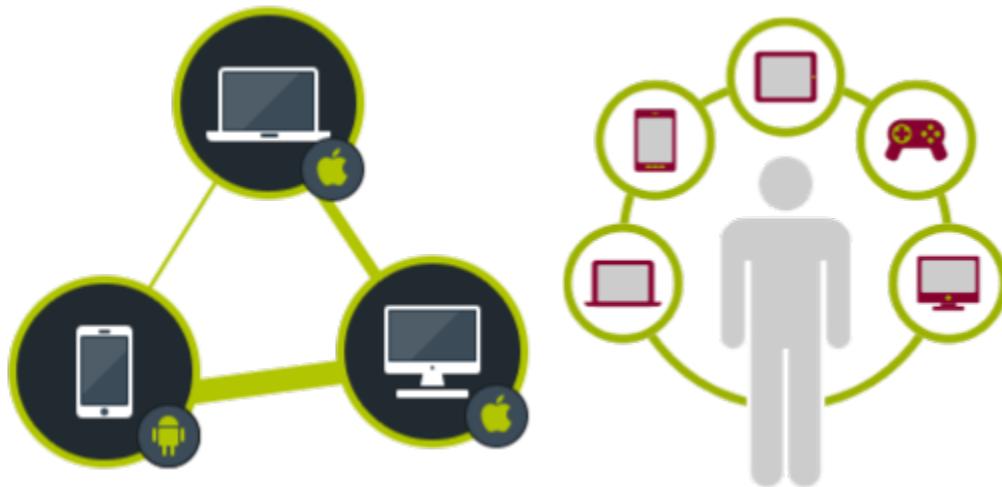
The Tapad RTB API follows the [OpenRTB 2.1](#) specifications with some clarifications and extensions. The process by which impressions and cross-platform data are delivered through RTB works as follows:

1. Bid request created from ad space on publisher which may contain a single device ID (cookie or hardware ID).
2. Tapad looks up that device ID in the [Device Graph](#) and aggregates the audience IDs associated to that device. This allows the Programmatic Partner to identify their users regardless of the device on which the ad is being delivered.
3. Bid request is passed to the Programmatic Partner as a JSON payload to their endpoint (URL). The bid request will include a direct cookie ID or hardware ID, if available, and all audience IDs associated with that device.
4. Programmatic Partner decides to bid, and sends back the bid value to Tapad as a JSON response along with the Creative ID.
5. Tapad conducts a 2nd price auction for multiple bids, and the highest bid wins the impression at the 2nd highest price +\$0.01.
6. Tapad serves the Creative for the won impression.



Tapad's Device Graph

Tapad's proprietary Device Graph technology unifies all of the devices linked to a single consumer.



The Device Graph:

- Anonymously associates multiple devices with a single user.
- Bridges computers, smartphones, tablets, connected TVs, and gaming consoles.
- Is populated through partnerships, wifi linking, and geo-clustering.
- Comprises the largest database of devices (+1B).

RTB Access Integration Process

The integration process with Tapad's Programmatic RTB Access product includes ID synchronization, audience ingestion, creative setup, and integration with the Open RTB 2.1 spec with Tapad's extensions. A Tapad integration expert can help answer questions during setup.

Integration process

Process	Description
Getting Started	<p>Programmatic Partners are first introduced to Tapad's integration experts. During this time, Programmatic Partners will:</p> <ul style="list-style-type: none"> • Sign an MSA and MNDA before integration begins. • Meet their Tapad demand manager and solutions consultant. • Set up a call for Q&A on technical documentation. <p>For information see:</p> <ul style="list-style-type: none"> • Cookie syncing and audience ingestion documents that your solutions consultant will send you separately • Technical Specifications for Programmatic RTB Access API • Tapad's Creative Approval API
ID Synchronization	<p>An ID synchronization is important so that Tapad and the Programmatic Partner can jointly identify devices.</p> <p>For maximum coverage, ID sync should be bi-directional where both Tapad and the Partner initiate.</p>
Audience Ingestion	<p>Audience ingestion is important so that Tapad knows which audiences the Partner's device IDs belong to, so that Tapad can extend these audiences to associated devices.</p> <p>Tapad requires two files to import and process your audience data:</p> <ul style="list-style-type: none"> • Audience Data File: includes Partner's audience IDs, device IDs, and device types. • Audience Taxonomy File: includes Partner's audience IDs, advertiser IDs, and audience names.
Creative Approval	<p>Programmatic Partner sends creatives for approval via Tapad's Creative Approval API.</p> <ul style="list-style-type: none"> • Bidders are still expected to respect the banner.battr list of blocked creative attributes found in Table 6.3 of the Open RTB 2.1 spec. <p>For information see: Creative Approval API V1.2</p>
RTB Integration	<p>Programmatic Partner integrates with Tapad's RTB Access using the Technical Specifications for Programmatic RTB Access.</p> <ul style="list-style-type: none"> • This document should be considered an addendum to the IAB Open RTB 2.1 spec. • Programmatic Partner sends Tapad their bid endpoint(s) for East and West Coast traffic.
Setup and Testing	<p>The testing process will include:</p> <ul style="list-style-type: none"> • A functional test to ensure proper JSON protocol • A latency test • A discrepancy comparison test for impressions <p>Once testing is complete, Tapad will send traffic through the Programmatic RTB Access Endpoint. Partner may specify initial QPS for ramp-up period.</p>
Post Setup and Deployment	Stay in touch! Partner will have a designated Tapad demand manager who will work with Partner on new features available, performance, reporting and Q&A.

Protocols and Connectivity

Requirement	Description
HTTP Protocol	Bid requests are passed to the bidders as JSON payloads via an HTTP 1.1 POST invocations to the Programmatic Partner's end-point(s) (URLs). Connections from Tapad's servers will remain persistent if the Partner's end-point supports HTTP Persistent connections (AKA keep-alive).

Load-balancing	If the Programmatic Partner provides multiple end-points, traffic will be distributed using a least-outstanding requests load-balancing algorithm. This means that servers with higher response times generally will receive fewer requests than servers with low response times. Still, it is highly recommended that the Partner handles load-balancing internally as this gives better control and insight into any performance issues that may occur. Tapad currently has data centers in the United States for East Coast and West Coast. Tapad strongly recommends that the Partner provide separate endpoints for each data center.
Timeouts	Timeouts and response time percentiles are tracked by Tapad servers. All bid responses must be received within the designated threshold as specified in the tmax value of the bid request. This will range from 70ms to 200ms. Responses exceeding the tmax value will be ignored.

Technical Specifications for Programmatic RTB Access

Tapad Exchange is a way for programmatic buyers to access Tapad's proprietary cross-platform data while also accessing media in real time. The technical specifications are largely based on the [OpenRTB 2.1](#) framework, with some clarifications and extensions added.

- **Bid Request Object** — The top-level bid request object contains a globally unique bid request or auction ID. This "id" attribute is required as is at least one impression object.
 - **Impression Object** — The "imp" object describes the ad position or impression up for auction. At least one impression is required for a valid bid request.
 - **Banner Object** — The "banner" object is included in the impression object for creative unit specifications. All attributes are made available for bidding decision, but Tapad will enforce the creative constraints.
 - **Site Object** — Root object for site information when bid request is for a website (standard or mobile optimized). Some extensions are specified in the site object for Tapad audited inventory, COPPA regulation and mobile optimized websites.
 - **App Object** — Root object for app information, when bid request is for a mobile application. Some extensions are specified in the site object for Tapad audited inventory and COPPA regulation.
 - **Device Object** — Root object for device / browser information. The "device" object provides information for the device including its hardware platform and location. The device can refer to a mobile handset, tablet, or desktop computer. The device object includes the geo object and an extension object.
 - **Geo Object** — Root object for geographical information, including city and zip codes. Some extensions are specified in the geo object for Tapad DMA and timezone information.
- **User Object** — Root object for user information. The "user" object contains information about the user of the device and includes the data and segment objects. The user object also includes an extension for Tapad's cross-platform user mapping information.
 - **Data Object** — An array of objects with data information about the user. This data may be sourced from Tapad proprietary data along with first and third-party data providers. A bid request can include information from multiple data providers. The data object includes the segment object.
 - **Segment Object** — An array of segment objects. The "segment" object provides specific IDs from the provider identified in the parent data object.
 - **Related Buyer User ID Object** — Array of objects with data about the buyer's related users. This data is sourced from Tapad's proprietary Device Graph.
- **Bid Request Format Example** — This is an example JSON format for a bid request sample. Please note: formatting may be incorrect due to PDF so please ask for .txt file from Tapad for proper formatting.
- **Bid Response Object** — The top-level bid response object. No-Bids on all impressions should be indicated as a HTTP 204 response. For no-bids on specific impressions, the bidder should omit these from the bid response.
 - **Seatbid Object** — Root object of the seat's bid. A bid response can contain multiple "seatbid" objects, each on behalf of a different bidder seat, however, Tapad can only support one seatbid for each bid request. The "bid" object must include the impression ID to which it pertains as well as the bid price.
 - **Bid Object** — Array of bid objects; each bid object relates to an imp object in the bid request. Note: Tapad only supports one bid for each bid request. For each bid, the "nurl" attribute contains the win notice URL. If the bidder wins the impression, the exchange calls this notice URL a) to inform the bidder of the win and b) to convey certain information using substitution macros (see Section 4.6 <http://www.iab.net/media/file/OpenRTB-API-Specification-Version-2-1-FINAL.pdf> Substitution Macros).
 - **RTB Macros** — The macros object will allow you to dynamically replace macros in the markup of the creative that you previously submitted to the Creative Approval API.
 - **Substitution Macros** — Substitution macros are used to dynamically convey information to the winning bidder. Below are the substitution macros that Tapad supports.
- **Bid Response Format Example** — This is an example JSON format for a bid response sample. Please note: formatting may be incorrect due to PDF so please ask for .txt file from Tapad for proper formatting. Please be sure to provide valid JSON.

Bid Request Object

The top-level bid request object contains a globally unique bid request or auction ID. This "id" attribute is required as is at least one impression object.

Field	Scope	Type	Example value	Description
id	required	string	"2cfb3f10-f6cc-11e1-a108-123139314d8c"	Unique ID of the bid request, provided by the exchange
imp	required	array of objects	/imp	See Impression Object
site	recommended for websites	object	/site	See Site Object
app	recommended for apps	object	/app	See App Object
device	recommended	object	/device	See Device Object
user	recommended	object	/user	See User Object
at	optional	integer	2	Auction Type. If "1", then first price auction. If "2", then second price auction. Tapad currently only conducts second price auctions.
tmax	optional	integer	70	Maximum amount of time in milliseconds to submit a bid (e.g., 70 means the bidder has 70ms to submit a bid to Tapad before the auction is complete). This will be dynamically populated depending on the requirements from each publisher. Range will vary from 70ms to 200ms. Responses exceeding the tmax will automatically forfeit.
wseat	optional	array of string	["1", "2"]	Array of buyer seat ids that are allowed to bid on the auction.
cur	optional	array of string	["USD"]	Array of allowed currencies for bids on this bid request using ISO-4217 alphabetic codes. Tapad currently only supports "USD".
bcat	optional	array of string	["IAB1", "IAB1-2"]	Array of blocked advertiser IAB categories on where each category is coded as "IAB1", "IAB1-2". The former means "IAB1-*". This is just for informational purposes as the exchange will enforce the blocking as per the creative approval process.
badv	optional	array of string	["advertiser.com"]	Array of blocked advertiser domain names, any www prefix will be omitted. This is just for informational purposes as the exchange will enforce the blocking as per the creative approval process.

Impression Object

The "imp" object describes the ad position or impression up for auction. At least one impression is required for a valid bid request.

Each "imp" object has a required ID so that bids can reference them individually. Multiple impression auctions may be specified in a single bid request, however, this will almost always be a single object instance.

Field	Scope	Type	Example value	Description
id	required	string	"1"	ID of the impression object, unique to this bid request. In essence this is the 1-based index, but represented as a string as per the OpenRTB 2.1 spec .
banner	required	object	/banner	See Banner Object
instl	optional	integer	0	1 if the ad is interstitial or full screen; else 0 (i.e., no).
tagid	optional	string	"agltb3B1Yi1pbmNyDQsSBFNpdGUY7fD0FAw"	Identifier for specific ad placement or ad tag that was used to initiate the auction. This can be useful for debugging of any issues, or for optimization by the buyer.
bidfloor	optional	float	0	Bid floor for this impression (in CPM).

Banner Object

The "banner" object is included in the impression object for creative unit specifications. All attributes are made available for bidding decision, but Tapad will enforce the creative constraints.

Field	Scope	Type	Example value	Description
w	recommended	integer	320	Width of the impression in pixels.
h	recommended	integer	50	Height of the impression in pixels.
pos	optional	integer	0	Ad Position. Use Table 6.5 . Will be unknown for most inventory.
btype	optional	array of integers	[4, 5]	Blocked creative types. See Table 6.2 Banner Ad Types . If blank, assume all types are allowed.
battr	optional	array of integers	[1, 2]	Blocked creative attributes. See Table 6.3 Creative Attributes . If blank assume all types are allowed.
mimes	optional	array of string	["image/png", "text/html"]	Array of allowed creative MIME types. Note that the Creative Approval API will automatically transform simple tags such as a standard anchor-image-tag into their pure banner representation for application contexts that only support static images.

Site Object

Root object for site information when bid request is for a website (standard or mobile optimized). Some extensions are specified in the site object for Tapad audited inventory, COPPA regulation and mobile optimized websites.

Field	Scope	Type	Example value	Description
id	recommended	string	"10027"	A unique id for the website (no overlap with applications). In the case of network sourced blind inventory, this may be unique only for the network - not individual sites.
name	optional	string	"Huffington Post"	A "human readable" / "display friendly" name for the site.
domain	optional	string	"huffingtonpost.com"	The primary domain name of the publisher. Any <i>www</i> prefix will be omitted.
cat	optional	array of string	["IAB1, "IAB1-2"]	Array of IAB content categories for the overall site. See Table 6.1 Content Categories . If inventory is specified as audited ("1"), this will be Tapad specified IAB categories. If inventory is not audited ("0"), this IAB categories declared by the publisher.
sectioncat	optional	array of string	["IAB1, "IAB1-2"]	Array of IAB content categories for the current subsection of the site. See Table 6.1 Content Categories .
page	optional	string	"huffingtonpost.com/news/1234"	URL of the page where the impression will be shown.
privacypolicy	optional	integer	1	Specifies whether the site has a privacy policy. "1" means there is a policy. "0" means there is not.
ext	optional	object	/ext	Extension of the site object.

Site Object: Extension ("ext")

Field	Scope	Type	Example value	Description
audited	optional	integer	1	Specifies whether the site has been audited by Tapad. "1" means that it is audited. "0" means it is unaudited.
maturity	optional	integer	1	Maturity rating of the site, (1 = Gold, 2 = Silver, 3 = Bronze, 4 = None-classified / Potentially brand unsafe).
supplier	optional	string	"Mopub"	Supplier from where impression is sourced.
coppa	optional	integer	0	If value is 1, this indicates that the impression falls under the Children's Online Privacy Protection Act , protecting OBA targeting for children under the age of thirteen. If "0" or field is not present, then impression does not fall under COPPA restrictions. When the value is 1, values in the user object may be missing or blank.
mopt	optional	integer	0	Flag to denote whether the impression comes from a mobile optimized website. 0 = not optimized for mobile 1 = optimized for mobile

App Object

Root object for app information, when bid request if for a mobile application. Some extensions are specified in the site object for Tapad audited inventory and COPPA regulation.

Field	Scope	Type	Example value	Description
id	recommended	string	"10048"	A unique id for the application (no overlap with sites).
name	optional	string	"Angry Birds"	Application name (may be masked at publisher's request)
domain	optional	string	"rovio.com"	Domain of the application
cat	optional	array of string	["IAB1, "IAB1-2"]	Array of IAB content categories for the overall app. See Table 6.1 Content Categories . If inventory is specified as audited ("1"), this will be Tapad specified IAB categories. If inventory is not audited ("0"), this IAB categories declared by the publisher.
sectioncat	optional	array of string	["IAB1, "IAB1-2"]	Array of IAB content categories for the current subsection of the app. See Table 6.1 Content Categories .
pagecat	optional	array of string	["IAB1, "IAB1-2"]	Array of IAB content categories for the current page/view of the app. See Table 6.1 Content Categories .
bundle	recommended	string	"com.foo.mygame"	Application bundle or package name. This is intended to be a unique ID across multiple exchanges.
privacypolicy	optional	integer	1	Specifies whether the site has a privacy policy. "1" means there is a policy. "0" means there is not.
storeurl	optional	string	" https://itunes.apple.com/us/app/angry-birds/id343200656?mt=8 "	For QAG 1.5 compliance, an app store URL for an installed app should be passed in the bid request
ext	optional	object	/ext	Extension of the app object.

App Object: Extension ("ext")

Field	Scope	Type	Example value	Description
audited	optional	integer	1	Specifies whether the site has been audited by Tapad. "1" means that it is audited. "0" means it is unaudited
maturity	optional	integer	1	Maturity rating of the site, (1 = Gold, 2 = Silver, 3 = Bronze, 4 = None-classified / Potentially brand unsafe)
supplier	optional	string	"Mopub"	Supplier from where impression is sourced

coppa	optional	integer	0	If value is 1, this indicates that the impression falls under the Children's Online Privacy Protection Act, protecting OBA targeting for children under the age of thirteen. If "0" or field is not present, then impression does not fall under COPPA restrictions. When the value is 1, values in the user object may be missing or blank.
-------	----------	---------	---	--

Device Object

Root object for device / browser information. The "device" object provides information for the device including its hardware platform and location. The device can refer to a mobile handset, tablet, or desktop computer. The device object includes the geo object and an extension object.

Field	Scope	Type	Example value	Description
dnt	recommended	integer	0	If "0", then do not track is set to false, if "1", then do no track is set to true in browser.
ua	recommended	string	"Mozilla/5.0 (iPhone; U; CPU like Mac OS X; en) AppleWebKit/420+ (KHTML, like Gecko) Version/3.0 Mobile/1A543a Safari/419.3"	Browser user agent string.
ip	recommended	string	"193.212.1.10"	The public IP address of the device. IPv4
geo	recommended	object	/geo	See Geo Object
didsha1	optional	string	"7ea394a47346813493c8bdf87ebe4e631b620789"	SHA1 hashed device ID; IMEI when available, else MEID or ESN. OpenRTB's preferred method for device ID hashing is SHA1.
didmd5	optional	string	"a6e4f2678956e3ce6a8ac7c520bfc2fd"	MD5 hashed device ID; IMEI when available, else MEID or ESN. Should be interpreted as case insensitive.
dpidsha1	optional	string	"f53349f40a080db85b2f23f002b8285f66199670"	SHA1 hashed platform-specific ID (e.g., Android ID or IDFA for iOS). OpenRTB's preferred method for device ID hash is SHA1.
dpidmd5	optional	string	"5bb5449901b122a8a1fc319a10ab8054"	MD5 hashed platform-specific ID (e.g., Android ID or IDFA for iOS). Should be interpreted as case insensitive.
language	optional	string	"en"	Browser language; use alpha-2/ISO 639-1 codes.
make	optional	string	"Apple"	Device make
model	optional	string	"iPhone"	Device model
os	optional	string	"iOS"	Device operating system
osv	optional	string	"3.1.2"	Device operating system version
devicetype	optional	integer	1	Return the device type being used. See Table 6.16 Device Type .
ext	optional	object	/ext	Extension of the device object.

Device Object: Extension ("ext")

Field	Scope	Type	Example value	Description
ext.idfa	optional	string	"3d89ce79-db1c-45a3-ac51-fdc2da5d0998"	Raw IDFA

Geo Object

Root object for geographical information, including city and zip codes. Some extensions are specified in the geo object for Tapad DMA and timezone information.

Field	Scope	Type	Example value	Description
lat	optional	float	40.746791	Latitude from -90 to 90. South is negative. This should only be passed if known to be accurate (For example, not the centroid of a postal code).
lon	optional	float	-73.993321	Longitude from -180 to 180. West is negative. This should only be passed if known to be accurate.
country	optional	string	"USA"	Country using ISO-3166-1 Alpha-3.
region	optional	string	"US-NY"	Region using ISO 3166-2
regionfips104	optional	string	"US36"	Region of a country using fips 10-4 notation (alternative to ISO 3166-2)
metro	optional	string	"501"	Pass the metro code (see Metro Codes). Metro codes are similar to but not exactly the same as Nielsen DMAs.
city	optional	string	"MNH"	City using United Nations Code for Trade and Transport Locations (http://www.unece.org/cefact/locode/service/location.htm)
zip	optional	string	"10001"	Zip/postal code
type	recommended	integer	[1, 2]	Indicate the source of the geo data (GPS, IP address, user provided). See Table 6.15 Location Type for a list of potential values. Type should be provided when lat/lon is provided.
ext	optional	object	/ext	Extension of the geo object.

Geo Object: Extension ("ext")

Field	Scope	Type	Example value	Description
dma	optional	integer	501	Nielsen DMA code
timezone	optional	integer	-400	User UTC timezone offset. "-430" means - 4h30min

User Object

Root object for user information. The "user" object contains information about the user of the device and includes the data and segment objects. The user object also includes an extension for Tapad's cross-platform user mapping information.

Field	Scope	Type	Example value	Description
id	recommended	string	"5609881348217144658"	Tapad's Direct Cookie ID for this user
buyeruid	recommended	string	"5610143615252830233"	Programmatic Partner's Direct Cookie ID for this user as mapped by Tapad
data	optional	array of objects	/data	See Data Object
ext	optional	object	/ext	Extension of the user object

User Object: Extension ("ext")

Field	Scope	Type	Example value	Description
relatedbuyeruid	optional	array of objects	/relatedbuyeruid	See Related Buyer User ID Object .

Data Object

An array of objects with data information about the user. This data may be sourced from Tapad proprietary data along with first and third-party data providers. A bid request can include information from multiple data providers. The data object includes the segment object.

Field	Scope	Type	Example value	Description
id	optional	string	"BK"	Exchange specific ID for the data provider
name	optional	string	"BLUEKAT"	Data provider name
segment	optional	array of objects	/segment	See Segment Object

Segment Object

An array of segment objects. The "segment" object provides specific IDs from the provider identified in the parent data object.

Field	Scope	Type	Example value	Description
id	optional	string	"BK-123"	ID of a data provider's segment applicable to the user.
name	optional	string	"Bluekai_autointenders"	Name of the data provider's segment applicable to the user. This will usually not be present.

Related Buyer User ID Object

Array of objects with data about the buyer's related users. This data is sourced from Tapad's proprietary Device Graph.

Field	Scope	Type	Example value	Description
id	optional	string	"12345"	Programmatic Partner's ID for this related user as mapped by Tapad
type	optional	string	"Tapad_Cookie"	Type of ID for Programmatic Partner's related user. See below for list of device types.

Type	Description
THIRD_PARTY	Partner Cookie ID
HARDWARE_ANDROIDID	Android ID
HARDWARE_MD5ANDROIDID	MD5 Hashed Android ID
HARDWARE_SHA1ANDROIDID	SHA1 Hashed Android ID
HARDWARE_IDFA	Raw IDFA
HARDWARE_SHA1IDFA	SHA1 Hashed IDFA
HARDWARE_MD5IDFA	MD5 Hashed IDFA
HARDWARE_IDFV	Raw IDFV
HARDWARE_ANDROID_AD_ID	Google Advertising ID

Bid Request Format Example

This is an example JSON format for a bid request sample. Please note: formatting may be incorrect due to PDF so please ask for .txt file from Tapad for proper formatting.

```
{  
    "id": "2cfb3f10-f6cc-11e1-a108-123139314d8c",  
    "imp": [{  
        "id": "1",  
        "banner": {  
            "w": 320,  
            "h": 50,  
            "pos": 0,  
            "btype": [4, 5],  
            "battr": [1, 2],  
            "mimes": ["image/png", "image/jpeg", "image/gif"]  
        },  
        "instl": 0,  
        "tagid": "agltb3B1Yi1pbmNyDQssSBFNpdGUY7fD0FAw",  
        "bidfloor": 0  
    }],  
    "app": {  
        "id": "10048",  
        "name": "Angry Birds",  
        "domain": "rovio.com",  
        "cat": ["IAB1", "IAB1-2"],  
        "sectioncat": ["IAB1", "IAB1-2"],  
        "pagecat": ["IAB1", "IAB1-2"],  
        "bundle": "com.foo.mygame",  
        "privacypolicy": 1,  
        "storeurl": "https://itunes.apple.com/us/app/angry-birds/id343200656?mt=8",  
        "ext": {  
            "audited": 1,  
            "maturity": 1,  
            "supplier": "Mopub",  
        },  
        "coppa": 0  
    },  
    "device": {  
        "dnt": 0,  
        "ua": "Mozilla/5.0 (iPhone; U; CPU like Mac OS X; en) AppleWebKit/420+ (KHTML, like Gecko) Version/3.0 Mobile/1A543a Safari/419.3",  
        "ip": "108.14.252.247",  
        "geo": {  
            "lat": 40.746902,  
            "lon": -73.993431,  
            "country": "USA",  
            "region": "US-NY",  
            "zip": "10001",  
            "type": 2,  
            "ext": {  
                "dma": 501,  
                "timezone": -400  
            },  
        },  
        "ext": {  
            "idfa": "3d89ce79-db1c-45a3-ac51-fdc2da5d0998"  
        },  
    },  
}
```

```
"user": {
    "id": "5609881348217144658",
    "buyeruid": "5610143615252830233",
    "data": [
        {
            "id": "BK",
            "name": "Bluekai",
            "segment": [
                {
                    "id": "BK-123"
                }
            ]
        },
        {
            "relatedbuyeruid": [
                {
                    "id": "12345",
                    "type": "Tapad_Cookie"
                }
            ]
        }
    ],
    "ext": {
        "at": 2,
        "tmax": 30,
        "wseat": ["1", "2"],
        "cur": ["USD"]
    }
},
```

```
"bcat": [ "IAB1", "IAB1-2"],  
"badv": [ "advertiser.com"]  
}
```

Bid Response Object

The top-level bid response object. No-Bids on all impressions should be indicated as a HTTP 204 response. For no-bids on specific impressions, the bidder should omit these from the bid response.

Field	Scope	Type	Example value	Description
id	Required	String	"ded5b530-f799-11e1-a108-123139314d8c"	Id of the bid request this bid is for (as passed from the ad exchange).
seatbid	Required	Array of Objects	/seatbid	See Seatbid Object . Note: Tapad only supports one bid for each bid request.
bidid	Optional	String	"d5a4d930-f2d2-11e1-b434-0a0027000000"	A unique id for the bid response (generated by the bidder).
cur	Optional	String	"USD"	The currency of the bid. Defaults to "USD" and alternative currencies are not supported at this time.

Seatbid Object

Root object of the seat's bid. A bid response can contain multiple "seatbid" objects, each on behalf of a different bidder seat, however, Tapad can only support one seatbid for each bid request. The "bid" object must include the impression ID to which it pertains as well as the bid price.

Field	Scope	Type	Example value	Description
bid	Required	Array of Objects	/bid	See Bid Objects .
seat	Required	String	"2"	The id of the seat this bid is for. The seat id is provided by Tapad.

Bid Object

Array of bid objects; each bid object relates to an imp object in the bid request. Note: Tapad only supports one bid for each bid request. For each bid, the "nurl" attribute contains the win notice URL. If the bidder wins the impression, the exchange calls this notice URL a) to inform the bidder of the win and b) to convey certain information using substitution macros (see [Section 4.6 Substitution Macros](#)).

Field	Scope	Type	Example value	Description
id	required	string	"d5a4d930-f2d2-11e1-b434-0a0027000000"	A unique id for the bid
impid	required	string	"1"	A reference to the impression object that this bid is for
price	required	decimal(2)	1.50	The bid in \$CPM
nurl	required	string	" http://www.bidder.com/win?\${AUCTION_PRICE} "	Win notification URL. If the bidder wins the impression, Tapad will call this URL to inform the bidder of the win and to convey certain information using substitution macros (see Substitution Macros or Table 4.6 of the Open RTB 2.1 spec).
adomain	optional	array of string	["http://www.advertiser.com"]	The "adomain" attribute can be used to check advertiser block list compliance.
iURL	optional	string	" http://campaign-tapad.s3.amazonaws.com/1234567-d183-4527-b5d2-2c78ec4ba1c0.png "	Sample image URL (without cache busting) for content checking. The "iurl" attribute can provide a link to an image that is representative of the campaign's content.
cid	optional	string	"9"	Campaign ID or similar that appears within the ad markup
crid	required	string	"278"	The creative identifier as returned from the Creative Approval API

<code>attr</code>	optional	array of integers	<code>[1, 2]</code>	Array of creative attributes. See table 6.3 of Open RTB 2.1 for creative attributes.
<code>macros</code>	Optional	object	<code>/macro</code>	See RTB Macros

RTB Macros

The macros object will allow you to dynamically replace macros in the markup of the creative that you previously submitted to the Creative Approval API.

Macro field descriptions

Field	Scope	Type	Example value	Description
CUSTOM_MACRO	Optional	String	"CUSTOM_VALUE"	Macro name and value to be expanded in markup

Substitution Macros

Substitution macros are used to dynamically convey information to the winning bidder. Below are the substitution macros that Tapad supports.

Macro descriptions

Macro	Description
<code> \${AUCTION_BID_ID}</code>	ID of the bid; from “bidid” attribute.
<code> \${AUCTION_PRICE}</code>	Settlement price using the same currency and units as the bid.
<code> \${AUCTION_CURRENCY}</code>	The currency used in the bid (explicit or implied); for confirmation

Bid Response Format Example

This is an example JSON format for a bid response sample. Please note: formatting may be incorrect due to PDF so please ask for .txt file from Tapad for proper formatting. Please be sure to provide valid JSON.

```
{  
    "id": "d5a3c7c0-f2d2-11e1-8998-0a0027000000",  
    "bidid": "d5a4d930-f2d2-11e1-b434-0a0027000000",  
    "cur": "USD",  
    "seatbid": [{  
        "seat": "2",  
        "bid": [{  
            "id": "d5a4d930-f2d2-11e1-b434-0a0027000000",  
            "impid": "1",  
            "price": 9.98,  
            "nurl":  
                "http://bidder.com/win.png?creative_id=278&ts=1346352313923&ta_win_prc=${AUCTION_PRICE}  
        }],  
        "adomain": ["http://www.advertiser.com"],  
        "iurl":  
            "http://campaign-tapad.s3.amazonaws.com/1234567-d183-4527-b5d2-2c78ec4ba1c0.png",  
            "cid": "9",  
            "crid": "278",  
        "attr": [1, 2],  
        "macros": {  
            "CUSTOM_MACRO": "CUSTOM_VALUE"  
        }  
    }]  
}
```

Creative Formats and Creative Approval API

Programmatic partners are required to abide by Tapad's creative specifications and are required to submit creatives via Tapad's creative approval API.

Overview

Most OpenRTB implementations allow the buyer to pass arbitrary creative markup in the bid response. Although somewhat convenient, this makes technical creative verification and block-list enforcement hard. The former is particularly complicated in the mobile space as the supported creative formats vary greatly. The latter is important as Tapad needs to enforce block-lists with upstream supply sources in order to access premium inventory.

Due to this, all creatives need to be submitted to the Creative Approval API and bids will only reference the Creative ID as provided by this API. Upon submission, Tapad will manually classify the creative (who is the advertiser, which IAB categories does the product belong to, what supply will the creative work on, etc) before it is approved or rejected.

The APIs support bid-time expanded macro expansions for click and impression trackers as well as custom macros.

Tapad's Technical specifications for creative approval API can be found [here](#).

Tapad's Technical specifications for creatives can be found [here](#).

Creative Approval API V1.2

Introduction

The Creative Approval API must be used for submitting creatives to RTB Access. The Approval process ensures that each creative is correctly classified correctly and functions properly across all applicable inventory sources.

Creative submission format

Creatives can be submitted for approval by POSTing a JSON document to <https://api.tapad.com/api/creatives>. The sandbox is available at <https://api-test.stg.tapad.com/>.

Creatives can be manually approved in the sandbox by Posting to <https://api-test.stg.tapad.com/>. This is required for bidder testing.

Authentication

All buy-side initiated API calls require a valid API authentication token passed as the HTTP header `API-Token`. The token is associated with your bidder end-point and bidder seats and is provided to you by Tapad. E.g, if your token is `1:xxyyzz`, then the API call should include the following HTTP header:

```
API-Token: 1:xxyyzz
```

You must receive a Creative Approval Token and a Seat ID before you can begin testing. If you have not received a token or Seat ID, please reach out to your Account Manager.

API Version

All buy-side initiated API calls require the API version to be passed as an HTTP header `api-version`. The API call should include the following HTTP header:

```
api-version: 1.2
```

Creative Submission Field descriptions

Field	Scope	Type	Example value	Description
seatId	Required	Integer	4	Seat ID provided by Tapad. This is set up when the account is set up. It allows access to multiple bidder seats.
width	Required	Integer	320	The width of the creative in pixels. This is the equivalent of 320x250 pixels. This should be set for desktop creatives.
height	Required	Integer	250	The height of the creative in pixels. This is the equivalent of 320x250 pixels. This should be set for desktop and mobile creatives.
mimeType	Required	String	"image/png"	The mime type of the creative. See Creative Type for more information. Note: application/x-shockwave-flash and video/mp4 are only supported for desktop environments.

attributes	Optional	Array of Integers	[1,2]	The attributes of the creative attributes Classification
secure	Optional	Integer	0	0 for non-secure; 1 value is 0 (non-sec
/bannerImage	Required when creative is image based	Root object for simple banner image creatives		
imageUrl	Required for image creatives	String	"http://campaign-tapad.s3.amazonaws.com/a50d2819-d183-4527-b5d2-2c78ec4ba1c0.png"	The URL to the banner image. Macros are allowed.
relocateToCdn	Optional, defaults to false	Boolean	true	If true, the creative will be uploaded to the Tapad CDN from there. If this value is false, the image must already be certified by a CDN.
clickUrl	Required	String	"http://www.tapad.com/cid?\${MACRO1}"	The click-through URL.
expandedClickUrl	Required for clickURLs with macros	String	"http://www.tapad.com/cid?123456"	The click-through URL with the macros expanded. Tapad will validate click-through URLs.
/markup	Required when creative is markup based	Root object for markup based creatives		
content	Required for markup creatives	String	""	The actual markup including any macros.
expandedContent	Required for creatives with macros	String	""	The actual markup with any macros expanded. Tapad will validate click-through URLs.
apiSupport	Optional	Integer	1	The API code that is supported. See API support code for more frameworks.
/vast	Required when serving VAST based creatives	Root object for VAST based creatives		Tapad only currently supports VAST wrappers. We will support VPAID video files.
vastwrapperURL	Required if serving VAST wrapper	String	"https://tapad.com/cn=is&c=23&pl=VAST&cid=1111"	The URL of the VAST wrapper.
mediafileURL	optional	String	"http://tapad-video.s3.amazonaws.com/video-upload/1111"	The URL of the local media file.
duration	optional	Integer	30	Duration, in seconds.
bitrate	optional	Integer	600	Bit rate in Kbps. Bit rates up to +600 kps kept at a constant bitrate for mobile is supported.
/advertiser				
domains	Required	Array of Domain name strings	["advertiser.com", "product.com"]	The domain of the advertiser that the creative is targeting.

categories	Required	Array of IAB Category Strings	["IAB1-2"]	Array of IAB categories advertiser / producer to pick IAB level 2 IAB category code Table 6.3 of the OI
------------	----------	-------------------------------	--------------	--

Creative Submission JSON Examples

Example of submitting a **banner** creative

```
{
    "seatId": 4,
    "width": 320,
    "height": 50,
    "mimeType": "image/png",
    "attributes": [1000],
    "bannerImage": {
        "imageUrl": "http://url-to-image",
        "relocateToCdn": true,
        "clickUrl": "http://click-through-url"
    },
    "advertiser": {
    "domains": ["advertiser.com", "product.com"],
    "categories": ["IAB1-2"]
    }
}
```

Example of submitting a **markup** based creative

```
{
    "seatId": 4,
    "width": 320,
    "height": 50,
    "mimeType": "image/png",
    "attributes": [1000],
    "markup": {
        "content": "<a href...."
    },
    "advertiser": {
    "domains": ["advertiser.com", "product.com"],
    "categories": ["IAB1-2"]
    }
}
```

Example of submitting a **VAST** based creative

```
{
    "seatId": 4,
    "width": 320,
    "height": 50,
    "mimeType": "video/x-flv",
    "attributes": [1000],
    "vast": {
        "vastwrapperURL": "https://tapad.com/cn=is&c=23&pl=VAST&cid=1111"
    },
    "advertiser": {
        "domains": ["advertiser.com", "product.com"],
        "categories": ["IAB1-2"]
    }
}
```

Creative Submit Response

Field descriptions

The following fields are added *in addition* to the fields that were specified during the submission.

Field	Scope	Type	Example value	Description
/				
id	Required	Integer	1000	The creative id
createdAt	Required	Timestamp	1346947249159	The time the creative was submitted / created
/auditStatus		Root object for the creative audit status		
updateAt	Required	Timestamp	1346947249159	The time the status was last updated
code	Required	String	"APPROVED"	Status code for the approval process, (PENDING, APPROVED, REJECTED)
errors	Required, if code == "REJECTED"	Array of String	["Error 1", "Error 2"]	Description of reason why the creative was rejected. For a full list of rejections, see Error Rejections

A successful submission will return HTTP 201 Created and the Location header will point to where creative information and audit status can be polled. In addition, a JSON document with brief creative info will be returned, including the Creative ID identifier that later should be passed with bid responses:

```
HTTP/1.1 201 Created
Location: http://api.tapad.com/creatives/1000
...
{
    "status": "OK",
    "creative": {
        "id": 10000
    }
}
```

In the case of an unsuccessful submission, the service will return HTTP 400 Bad Request and a JSON document describing the errors:

```
HTTP/1.1 400 Bad Request
...
{
    "status": "ERROR",
    "errors": [
        {
            "field": "bannerImage.imageUrl",
            "message": "Invalid url"
        }
    ]
}
```

Checking for approval status

The current approval status can be retrieved by invoking a HTTP GET to the Location header given on submission, e.g <http://api.tapad.com/creatives/1000>. This will yield a JSON document describing the creative and it's approval status.

Example of a **pending** creative

```
{
    id: 1000,
    createdAt: 1346947239159,
    auditStatus: {
        updatedAt: 1346947239159,
        code: "PENDING"
    }
}
```

Example of an **approved** creative

```
{
    "id": 1000,
    "seatId": 4,
    "createdAt": 1346947239159,
    "auditStatus": {
        "updatedAt": 1346947249159,
        "code": "APPROVED"
    },
    "width": 300,
    "height": 250,
    "mimeType": "application/javascript",
    "attributes": [1000],
    "markup": {
        "content": "<a href=...",
        "apiSupport": 1
    },
    "advertiser": {
        "domains": [ "advertiser.com", "advertisedproduct.com" ],
        "categories": [ "IAB1-2" ]
    }
}
```

Example of a **rejected** creative

```
{
    "id": 1000,
    "seatId": 4,
    "createdAt": 1346947239159,
    "auditStatus": {
        "updatedAt": 1346947249159,
        "code": "REJECTED",
        "errors": [ "Creative does not click through properly" ]
    }
}
```

Handling creative rejection

Once a creative has been rejected, it must be resubmitted - it cannot be changed in place.

API support codes

Below is a list of API frameworks. Please note not all API codes may be supported at this time.

API Type	Api code	Supply applicability
VPAID 1.0	1	Web inventory only
VPAID 2.0	2	Web inventory only
MRAID	3	In-app inventory only
ORMMA	4	In-app inventory only

Creative Attribute Classification

We require partners to declare attributes of the creative to enforce block lists from publishers.

Values 1-16 are IAB standard creative attribute values. 1000+ are Tapad values that are required for specific inventory sources. Please ensure proper classification of each creative to avoid potential blocking from publishers.

Value	Description
1	Audio Ad (Auto Play)
2	Audio Ad (User Initiated)
3	Expandable (Automatic)
4	Expandable (User Initiated - Click)
5	Expandable (User Initiated - Rollover)
6	In-Banner Video Ad (Auto Play)
7	In-Banner Video Ad (User Initiated)
8	Pop (e.g., Over, Under, or upon Exit)
9	Provocative or Suggestive Imagery
10	Shaky, Flashing, Flickering, Extreme Animation, Smileys
11	Survey
12	Text Only
13	User Interactive (e.g., Embedded Games)
14	Windows Dialog or Alert Style
15	Has audio on/off button
16	Ad can be skipped (e.g., skip button on preroll video)
1000	Animated <15 seconds
1001	Animated >15 seconds
1002	Starts a download
1003	Mimics web or app content
1004	Public Service Announcement
1005	iFrame
1006	English
1007	Spanish
1008	Portuguese
1009	French
1010	Italian
1011	German
1012	Japanese
1013	Chinese
1014	Korean
1015	Other Language

1016	Gambling
1017	Smiley/Cursors

Creative type classification

The creative media type will be classified by MIME type and any API support required for screen take-over, expansion etc. Currently the following creative media types are supported:

Creative Type

Creative type	Mime type	Supply applicability
Banner image	image/png image/jpeg or image/gif	Works in all inventory, with the possible caveat of some publishers not supporting multiple impression trackers
HTML Markup fragment without Javascript	text/html	Excludes some in-app and email inventory
HTML Markup fragment with Javascript	application/javascript	Excludes some in-app and email inventory
Flash	application/x-shockwave-flash	Display only. Excludes all mobile inventory
Video	video/x-flv, video/mp4	Tapad only supports flv and mp4 video, mp4 video being the preferred type. Flv video is only supported on desktop while mp4 is supported on all platforms

Error Rejections

The creative approval API will respond with an error if there is an issue with the creative. Below are a list of errors. Please fix the issue and then resubmit the creative for it to be approved.

Technical Criteria

Creatives will be rejected for the following technical reasons.

Issue	Causes for rejection
Creative does not display properly	<ul style="list-style-type: none">• The creative does not render; blank screen is shown• The creative only displays intermittently• A portion of the creative is not displayed properly; for example a pixel could be implemented incorrectly and the creative shows "/>" in the bottom corner
Creative does not click through properly	<ul style="list-style-type: none">• The creative does not click through to a landing page• The creative clicks through to a landing page with no content• The creative's landing page loads in the same page instead of opening a new window or tab

Ad Content Criteria

All creative content must abide by the below criteria.

Criteria	Description
Creative branding is insufficient	<ul style="list-style-type: none">• The creative does not click through to the landing page of the specified landing page domain. Brand of the creative and landing page must match.• Creative displays more than one brand• Creative clicks through to more than one brand (multiple landing pages with different brands)• Creative does not contain sufficient brand information<ul style="list-style-type: none">• No clearly defined logo• No brand name
Prohibited content	<ul style="list-style-type: none">• Pornography or adult content• Hate speech• Malware• Piracy• Weapons• Violence• Illegal Drugs• Auto-downloads• Illegal content• Government forms or services• Fake errors and warnings
Language of landing page does not match content	The Language of the landing page must match the language of the content
Creative initiates download	Click on an ad must not initiate a download of any type

Programmatic RTB Access Version and Revisions

This page will designate the current version of the spec and any revisions made to prior specs.

Current Programmatic RTB Version

Version 1.2

Revisions

Date	Changes
July-2014	<ul style="list-style-type: none">Added instl, tagid and bidfloor to the imp objectAdded btype and bttr to the banner objectAdded audited, maturity, supplier and coppa to the site and app objectsAdded didsha1, didmd5, dpidsha1, dpidmd5 and idfa to device objectAdded lat and lon to geo objectAdded user object, including data object and segment object
August-2014	<ul style="list-style-type: none">Added related user ID object
October-2014	<ul style="list-style-type: none">Added related buyer ID type

Data Integrations

Send data to Tapad from other products for cross-device targeting and retargeting.

- **Analytics Plug-in** — The analytics plug in is JavaScript code you place on a page to send Tapad data to other reporting systems like Google Analytics or Adobe Analytics (SiteCatalyst).
- **Data Flow: Send Audiences To Tapad** — Tapad needs an audience data file and an audience taxonomy to match your audience IDs with related IDs in the Device Graph. An audience data file consists of device and audience IDs in a JSON file. A taxonomy file consists of metadata related to the audience data file (also in a JSON format). Refer to this guide for information about file formats, syntax, examples, and delivery options.
- **Data Flow: Receive Connected IDs From Tapad** — ID matching is a process that takes device IDs you've sent to Tapad and matches them to other device IDs. As determined by our Device Graph, these new IDs represent devices that share a connection with your original ID. When the matching process is complete, Tapad returns this data to you as a JSON object or in a flat file. You can use this data to target your customers across all their known devices, deliver personalized content, and use it with your own analytics systems or internal processes.
- **Mapping Primary and Related IDs** — As you go through the records in a data file, it's normal to see primary IDs appear as a related ID and for some related IDs to appear as a primary ID. This happens with a related ID that belongs to devices that have been synchronized previously with Tapad. This behavior is the result of how the Device Graph builds a map of related devices and checks for connections between them. These results are useful because they show you relationships between your devices and other devices stored in the Device Graph.
- **Understanding the ID Mapping File** — The ID mapping file contains device ID data for separate Tapad clients. The information in this file is generated from the unique IDs clients send to Tapad for synchronization with other IDs in the Device Graph. Data in this file can help you show prospective clients the size of the connected audiences they could reach when working with Tapad or help existing clients interpret the data they receive in the weekly device mapping file.
- **The Tapad User Opt-out Process: About** — An overview of how Tapad opt-out processes work and how we send consumer opt-out data to our partners. See the Tapad website for more information about our privacy policy <http://www.tapad.com/privacy-policy/>.
- **Technical Specifications for the Impression Log File** — The Impression Log File provides data about impressions and clicks across bridged devices in the Tapad Device Graph. It includes all impressions and clicks served to your first or third-party audiences. Tapad can deliver the report to you in a compressed, tab-delimited file via FTP. Information in the log file can help you understand how your audience data is used and help you create more accurate client reports.
- **Audience and User ID Synchronization** — Data integration partners can share user IDs, audiences, and related audience metadata (optional) with Tapad. Exchanging user and audience IDs with Tapad helps extend campaign reach because it lets you deliver content to individual users across all of their devices (desktop, laptop, mobile, tablet, etc.). Refer to this material for technical details about our user ID and audience data integration/synchronization processes. Work with your Tapad Implementation Consultant for more information and support.
- **Using Tapad as a DMP** — Describes how you can synchronize page metadata with Tapad and return DMP or ad server IDs to the publisher or other platforms and systems.
- **AudienceManager Integration Guide** — A Tapad - AudienceManager data integration helps you target AudienceManager segment members on multiple devices such as desktop and laptop computers, mobile phones, tablets, or game consoles.
- **BlueKai Integration Guide** —
- **Lotame Data Integration Guide** — A Tapad - Lotame data integration helps you reach audiences across multiple devices such as desktop and laptop computers, mobile phones, tablets, game consoles, and other platforms.
- **Turn Integration Overview** — The Turn Audience Suite is a platform that provides its customers with a system to manage and build audiences with online and offline data. This data integration lets Tapad share synchronized audience segments with Turn and make that data available to their customers.
- **Technical Specifications for ID Bridging** — ID bridging is a process that lets data partners share hashed user IDs with Tapad. This user information helps us match authenticated user names to corresponding Tapad cookie IDs. Tapad accepts bridged ID data in real-time from browser pixel calls, from mobile devices via server-to-server data transfers (batch files), or programmatically through our Tapestry SDK. Refer to this material for requirements and code samples used to set up ID bridging.
- **Audience Amplify** — An overview of how it works and data file formats.

Analytics Plug-in

The analytics plug in is JavaScript code you place on a page to send Tapad data to other reporting systems like Google Analytics or Adobe Analytics (SiteCatalyst).

- [Tapad - Adobe Analytics \(SiteCatalyst\) Data Integration](#) — Bring Tapad data into your Adobe Analytics dashboard. Includes instructions and code samples.
- [Tapad - Google Analytics Data Integration](#) — Bring Tapad data into a custom report in Google Analytics. This material applies to classic analytics and Universal Analytics.
- [Analytics Plug-in SDKs](#) — Set up instructions and code samples for Android and iOS devices. These SDKs send custom Tapad data to Google Analytics from Android or iOS mobile devices.
- [Analytics Plug-in Metrics Defined](#) — Contains information and descriptions about how the plug-in metrics work. Currently, a few of these metrics don't work properly or have misleading labels. Review this material for your own understanding, but do not share this content with external customers.

Tapad - Adobe Analytics (SiteCatalyst) Data Integration

Bring Tapad data into your Adobe Analytics dashboard. Includes instructions and code samples.

- [Tapad - Adobe Analytics Data Integration: About](#) — Purpose, how it works, getting started requirements, and implementation questionnaire.
- [Requirements and Implementation Questionnaire for Adobe Analytics](#) — To use this service you must have an Adobe Analytics account and a Tapad customer ID. Contact your Tapad consultant to request an ID. Also, please complete the implementation questionnaire in this section.
- [Tapad Parameters Available in Adobe Analytics](#) — Defines the available Tapad report metrics you can bring into Adobe Analytics. This integration uses a single Analytics eVar http://microsite.omniture.com/t2/help/en_US/reference/conversion_var_admin.html to handle all your Tapad data.
- [Deactivate a Tapad - Adobe Data Connection](#) — Follow these steps to deactivate and remove the Tapad data connector from your Adobe Analytics account.
- [Integration Code Sample for Adobe Analytics](#) — Sample JavaScript code for a Tapad - Adobe Analytics data integration.
- [Tapad and Adobe Analytics Integration Guides](#) — Reference material in PDF format and video (.mp4) format.

Tapad - Adobe Analytics Data Integration: About

Purpose, how it works, getting started requirements, and implementation questionnaire.

Purpose

Tapestry Web Analytics is a Tapad service that collects data on the devices visitors use when they interact with your website or mobile app. For example, it can determine what type of device is used to first access your site, how many devices belong to a visitor, and which device visitors use most frequently when browsing your site. This integration lets you bring Tapad data into your Adobe Analytics account and view it on the reporting dashboard. Try a Tapad - AdobeAnalytics integration learn more about how customers access your content or as a trial evaluation of our cross-platform targeting and analytics technology.

How it works

Adobe Analytics uses the SiteCatalyst `s_code.js` file to request device data from Tapad and push it to your reporting dashboard. The Adobe Integration Wizard generates the required JavaScript code that you add to your `s_code.js` file. After you update your `s_code.js` file the data integration process works like this:

1. When your page loads, it invokes the custom JavaScript in the `s_code.js` file.
2. The `s_code.js` file makes a call to Tapad for data.
3. Tapad returns device data to Adobe.
4. Device data collected by Tapad is available in your selected Analytics dashboard.

Questions?

Contact us at:

Tapad
60 Madison Ave, 3rd floor
New York, NY 10010
info@tapad.com
(646) 561-6500

Requirements and Implementation Questionnaire for Adobe Analytics

To use this service you must have an Adobe Analytics account and a Tapad customer ID. Contact your Tapad consultant to request an ID. Also, please complete the implementation questionnaire in this section.

Please let us know about the following:

eVar availability

The analytics plug-in requires a SiteCatalyst eVar to push data to your Analytics dashboard. Do you have at least 1 eVar available?

- Yes
- No

Where will requests come from?

List all the domains (including staging environments) that a request to Tapad could come from (e.g., domain1.com, staging.domain1.com, etc.).

Domain list

Analytics version

What version of Adobe Analytics are you using?

Tag management

Do you use a tag management system?

- Yes
- No

If you answered "Yes", which tag management system/service do you use?

Request volume

How many requests-per-second can expect from your web properties? An average value is fine.

Tapad Parameters Available in Adobe Analytics

Defines the available Tapad report metrics you can bring into Adobe Analytics. This integration uses a single Analytics eVar to handle all your Tapad data.

The following table describes the variables and data collected by the Tapestry Web Analytics plug-in. The Adobe Integration Wizard generates code that contains these variables. Put the code in the custom plug-ins section of your Analytics `s_code.js` file. For a code sample, see the [Integration Code Sample for Adobe Analytics](#). For details on each report metric, see [Analytics Plug-in Metrics Defined](#).

Tapad report metric	<code>s_code.js</code> parameter
Visited Platforms	<code>(vp?vp:0)</code>
Platforms Associated	<code>(pa?pa:0)</code>
Platform Types	<code>(pt?pt:"")</code>
First Visited Platform	<code>(fvp?fvp:"")</code>
Most Often Visited	<code>(movp?movp:"")</code>
Most Recent Visited	<code>(mrvp?mrvp:"")</code>

Deactivate a Tapad - Adobe Data Connection

Follow these steps to deactivate and remove the Tapad data connector from your Adobe Analytics account.

To delete a data connection from Adobe Analytics:

1. Go to *Adobe Marketing Cloud > Analytics > Data Connectors*.
2. Select the **Configure** tab.
3. In **General Settings**, select **Deactivate**.

Related concepts

[Tapad and Adobe Analytics Integration Guides](#)

Integration Code Sample for Adobe Analytics

Sample JavaScript code for a Tapad - Adobe Analytics data integration.

Sample code

The Adobe Integration Wizard creates this code for your Analytics account after you [set up the data connector](#). Place this code in the plug-ins section of your `s_code.js` file. You can find this code in your Adobe Analytics account in *Data Connectors > Support > Resources*. Click the **Integrate Plugin** link to download the code. Your code could look similar to this:

Oh, Snap!

Code provided as an example only. *Do not copy this to your `s_code.js` file.*

```

// ---Tapad Integrate JavaScript Plugin---
// This code snippet must be placed at the appropriate plugin section of the s_code.js
file in order for the integration to work. Please refer to the integration Guide for
further details.
var tapadConfig = {
    tEvar: 'eVar6',
    gID: 'TAP:',
    tID: '12345',
    tVar: 'tVar',
    visitCookie: 'tcookie',
    requestURL:
    'http://tapestry.tapad.com/tapestry/1?ta_partner_id=[tID]&rnd=[RAND]&ta_analytics={"is
NewSession":true}&ta_set_local_var=[VAR]'
};
/* Tapad Start */
s.maxDelay=750
s.loadModule("Integrate")
s.Integrate.onLoad=function(s,m) {
    var tapadCheck = s.partnerTapadCheck(tapadConfig);
    if (tapadCheck) {
        s.Integrate.add("Tapad");
        s.Integrate.Tapad.tEvar=tapadConfig.tEvar;
        s.Integrate.Tapad.gID=tapadConfig.gID;
        s.Integrate.Tapad.tID=tapadConfig.tID;
        s.Integrate.Tapad.get(tapadConfig.requestURL);
        s.Integrate.Tapad.setVars=function(s,p) {
            if (p.errors) {
                if (p.errorEvar) s[p.errorEvar] = p.errors;
            }
            else if (p.analytics && !tcookie) {
                var a=p.analytics;
                var
                fvp=a.fvp,
                pa=a.pa,
                pt=a.pt,
                vp=a.vp,
                movp=a.movp,
                mrvp=a.mrvp;

                s[p.tEvar]=tapadConfig.gID+(fvp?fvp:"")+"："+(pa?pa:0)+"："+(pt?pt:"")+"："+(vp?vp:0)+"："+"
+(movp?movp:"")+"："+(mrvp?mrvp:"");
                s.c_w(tapadConfig.visitCookie, true);
            } else if (p.timeoutEvent) {
                s.events = ((!s.events || s.events == '') ? '' : (s.events + ',')) +
p.timeoutEvent;
            }
        }
    }
}
//endif
}
// TapAd Plugin: Restrict TapAd API calls once a visit
s.partnerTapadCheck=function(cfg) {
    var c = cfg.visitCookie;
    if (s.c_r(c)) return false;
    return true;
}

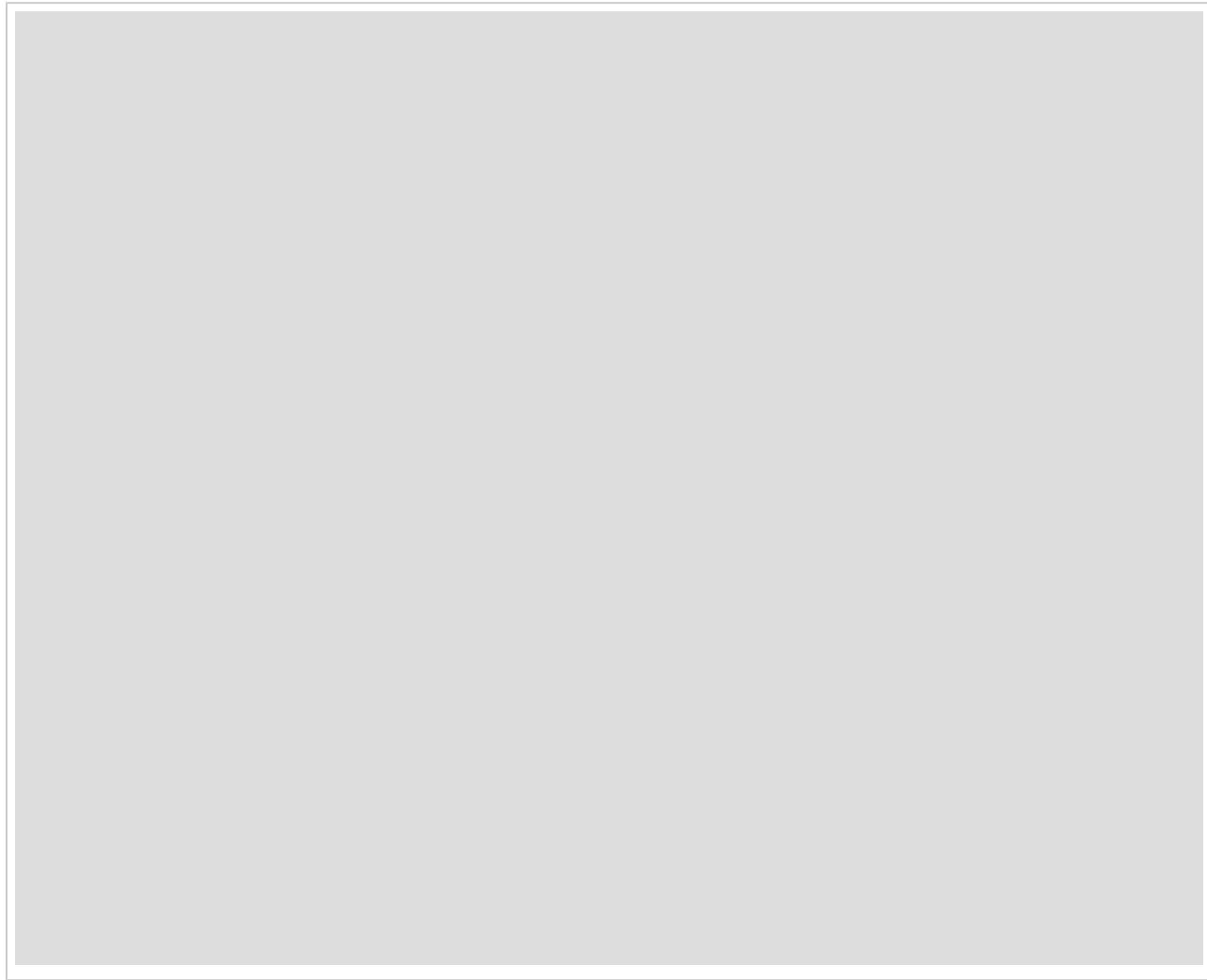
```

Tapad and Adobe Analytics Integration Guides

Reference material in PDF format and video (.mp4) format.

Tapad - Adobe Analytics integration guide

To download, click in the PDF frame.



Adobe Analytics (SiteCatalyst) help and reference

To download, click in the PDF frame.



Video

Setup steps in .mp4 format.

With narration (4 minutes)

[Click to view.](#)

No narration, includes captions (1:42)

[Click to view.](#)

No narration, no captions (2 minutes)

[Click to view.](#)

Tapad - Google Analytics Data Integration

Bring Tapad data into a custom report in Google Analytics. This material applies to classic analytics and Universal Analytics.

- **Tapad - Google Analytics Integration: About** — The Tapestry Web Analytics service collects data on the types of devices visitors use when they interact with your website and the connections between them. Tapestry sends this data to your Google Analytics dashboard. You can use this data to better understand how customers access your content or as a trial evaluation of our cross-device targeting/analytics technology. Tapestry Web Analytics can collect data for typical desktop/laptop browser interactions or from mobile devices with our SDK code
- **Data Collection With Google's Classic Analytics** — This code uses Tapad's JavaScript library (TAJ) to collect data and send it to Google Analytics. Use this code if you're working with the "classic" version of Google Analytics.
- **Data Collection With Google's Universal Analytics** — This code uses Tapad's JavaScript library (TAJ) to collect data and send it to Google Analytics. Use this code if you're working with the "universal" version of Google Analytics. See About Universal Analytics <https://support.google.com/analytics/answer/2790010?hl=en> for more information.
- **Tapad Parameters in Google Analytics: Definitions and Syntax** — A Tapad - Google Analytics integration provides 3 custom variables you can use for data collection.
- **Viewing Report Data in Google Analytics** — To see your custom report in Google Analytics, go to the Standard Reports sidebar and select Audience > Custom > Custom Variables.

Tapad - Google Analytics Integration: About

The Tapestry Web Analytics service collects data on the types of devices visitors use when they interact with your website and the connections between them. Tapestry sends this data to your Google Analytics dashboard. You can use this data to better understand how customers access your content or as a trial evaluation of our cross-device targeting/analytics technology. Tapestry Web Analytics can collect data for typical desktop/laptop browser interactions or from mobile devices with our [SDK code](#) for Android and iOS. This service works with the "classic" version of Google Analytics and their newer, Universal Analytics code.

How it works

This integration uses TAJ (Tapad's JavaScript API), or our SDK code for mobile devices, to collect and send data to your Google Analytics reporting dashboard. To get started, simply place our TAJ code on the pages you want to use for data collection or implement the SDK on your mobile app or site. The data collection process works like this:

1. Your page or app loads and analytics starts collecting data about the devices visiting your site.
2. Analytics code sends data to Tapad.
3. Tapad returns data to Google where it is available in your Google Analytics dashboard.

Getting started

To use this service you must have a Tapad customer ID and use Google Analytics. Contact your Tapad consultant to request an ID. To get started, please complete the on-line [implementation questionnaire](#).

Questions?

Contact us at:

Tapad
60 Madison Ave, 3rd Floor
New York, NY 10010
info@tapad.com
(646) 561-6500

Data Collection With Google's Classic Analytics

This code uses Tapad's JavaScript library (TAJ) to collect data and send it to Google Analytics. Use this code if you're working with the "classic" version of Google Analytics.

Placement

We recommend that you place this analytics code within the `<head>` block on each page you want to use for data collection. However, our code will work from within the `<body>` of your page.

```
<head>
...
<script>
  <!-- Analytics code here -->
</script>
...
</head>
```

Plug-in code

To bring Tapad data into Google Analytics, you need:

- A Tapad partner ID.
- The Tapestry data collection code provided below.
- Your Google Analytics tracking code and tracking ID. See [find your tracking code](#) for instructions about how to find these IDs.

Place this code on the pages you want to use for data collection. For information about our custom code, see [Tapad Parameters in Google Analytics: Definitions and Syntax](#).

```
<script>

<!-- Classic Google Analytics -->
var _gaq = _gaq || [];
_gaq.push(['_setAccount', 'UA-XXXXXXX-X']); //Provide your Google tracking ID
_gaq.push(['_setDomainName', 'your_domain_name.com']); //Provide your domain name
_gaq.push(['_trackPageview']);
(function() {
  var ga = document.createElement('script');
  ga.type = 'text/javascript';
  ga.async = true;
  ga.src = ('https:' == document.location.protocol ? 'https://ssl' : 'http://www') +
'.google-analytics.com/ga.js';
  var s = document.getElementsByTagName('script')[0];
  s.parentNode.insertBefore(ga, s);
})();
<!-- End Classic Google Analytics -->

/*
 * Tapad's TAJ JS code (required).
 */
var _tapestry = _tapestry || [];
_tapestry.push({
  'initialize': '12345' //Initialize with your Tapad partner ID.
});
(function() {
  var taj = document.createElement('script');
  taj.type = 'text/javascript';
  taj.async = true;
```

```

taj.src = ('https:' == document.location.protocol ? 'https://' : 'http://') +
'assets.tapad.com/tapestry/tapestry-0.7.min.js';
var s = document.getElementsByTagName('script')[0];
s.parentNode.insertBefore(taj, s);
})();

/*
 * Tapad code that sends data to Google Analytics.
 */
(function() {
    window['tapestryPushAnalytics'] = function(response) {
        var analytics = response.analytics;
        if (analytics) {
            // You can modify the custom variables and scope here (2 = session-level scope)
            _gaq.push(['_setCustomVar', 2, 'Visited_Platforms', analytics.vp, 2]);
            _gaq.push(['_setCustomVar', 1, 'Platforms_Associated', analytics.pa, 2]);
            _gaq.push(['_setCustomVar', 3, 'Platform_Types', analytics.pt, 2]);
            // In order to send the custom vars, an event fire must occur afterwards
            _gaq.push(['_trackEvent', 'tapestry', 'web', undefined, undefined, true]);
        }
    }
    // If there is no session cookie, we're in a new session.
    var isNewSession = document.cookie.indexOf('TAJ_ga') === -1;
    // On reach run (page load), bump the current session expiration by 30 minutes.
    document.cookie = 'TAJ_ga=' + new Date(new Date().getTime() + 30 * 60 * 1000).toUTCString() + ';';
    _tapestry.push({
        'configure': {
            'register_plugins': ['analytics']
        }
    });
    _tapestry.push({
        'analytics': {
            'isNewSession': isNewSession
        },
        'cache_policy': 'never',
        'callback': 'tapestryPushAnalytics'
    });
});

```

```
    });
})();
</script>
```

Data Collection With Google's Universal Analytics

This code uses Tapad's JavaScript library (TAJ) to collect data and send it to Google Analytics. Use this code if you're working with the "universal" version of Google Analytics. See [About Universal Analytics](#) for more information.

Placement

We recommend that you place this analytics code within the `<head>` block on each page you want to use for data collection. However, our code will work from within the `<body>` of your page.

```
<head>
...
<script>
  <!-- Analytics code here -->
</script>
...
</head>
```

Plug-in code

To bring Tapad data into Google's Universal Analytics, you need:

- A Tapad partner ID.
- The Tapestry data collection code provided below.
- Your Google Analytics tracking code and tracking ID. See [find your tracking code](#) for instructions about how to find these IDs.

Place this code on the pages you want to use for data collection. For information about our custom code, see [Tapad Parameters in Google Analytics: Definitions and Syntax](#).

```
<html>
  <head>
    <script type="text/javascript">

      <!-- Google Analytics -->
      (function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){
        (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o),
        m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m)
      })(window,document,'script','//www.google-analytics.com/analytics.js','ga');

      ga('create', 'UA-xxxxxxxx-x', { //Provide your Google tracking ID
        cookieDomain: 'my_domain.com' //Provide your domain name
      });
      ga('send', 'pageview');
      <!-- End Google Analytics -->

      /*
       * Tapad's TAJ JS code (required).
       */
      var _tapestry = _tapestry || [];
      _tapestry.push({
        'initialize': '12345' //Initialize with your Tapad partner ID.
      });

      (function() {
        var taj = document.createElement('script');
        taj.type = 'text/javascript';
        taj.async = true;
        taj.src = ('https:' == document.location.protocol ? 'https://': 'http://') +
```

```

'assets.tapad.com/tapestry/tapestry-0.7.js';
    var s = document.getElementsByTagName('script')[0];
    s.parentNode.insertBefore(taj, s);
})();

/*
 * Tapad code that sends data to Google Analytics.
 */
(function() {
    window['tapestryPushAnalytics'] = function(response) {
        var analytics = response.analytics;
        if (analytics) {
            ga('send', 'event', 'tapestry', 'web', {
                'nonInteraction': 1,
                'dimension18': analytics.vp,
                'dimension19': analytics.pa,
                'dimension20': analytics.pt
            });
        }
    }

    // If there is no session cookie, we're in a new session.
    var isNewSession = document.cookie.indexOf('TAJ_ga') === -1;

    // On reach run (page load), bump the current session expiration by 30
    minutes.
    document.cookie = 'TAJ_ga=;expires=' + new Date(new Date().getTime() + 30 * 60
    * 1000).toUTCString() + ';';

    _tapestry.push({
        'configure': {
            'register_plugins': ['analytics']
        }
    });

    _tapestry.push({
        'analytics': {
            'isNewSession': isNewSession
        },
        'cache_policy': 'never',
        'callback': 'tapestryPushAnalytics'
    });
}

})();
</script>

```

```
</head>
<body></body>
</html>
```

Tapad Parameters in Google Analytics: Definitions and Syntax

A Tapad - Google Analytics integration provides 3 custom variables you can use for data collection.

Tapad data collection parameters

The following table defines the 3 custom variables that are available for data collection and integration with Google Analytics. For a code samples, see the [classic](#) and [universal](#) code sample documentation. For details on each report metric, see [Analytics Plug-in Metrics Defined](#).

Variable name	Variable	Description
Visited_Platforms	vp	Visited Platforms counts user visits by platform group. Over time, this data can help you understand how many platform types site visitors are using to interact with your online properties.
Platforms_Associated	pa	Platforms Associated counts visits for devices grouped together as a collection of identical platforms. By counting visits by groups of devices, Platforms Associated helps you understand the levels of connectivity among your site visitors. For example, data provided by this metric lets you know if your visitors have 1 device or different types of devices.
Platform_Type	pt	Platform Types counts user visits by groups of named platform types (e.g., iPhone, computer, iPad, etc.). Over time, this data helps provide insight into what types of specific devices and device combinations site visitors are using to interact with your online properties.

Syntax and methods

Google uses 2 variables and a `.push` event to pass custom data into your reporting dashboard. Format each custom Tapad variable within these methods according to the following syntax (*italic* type indicates variable data):

```
_gaq.push(['_setCustomVar', index, 'name', variable, scope]);  
_gaq.push(['_trackEvent', 'tapestry', 'name']);
```

setCustomVar method

The parameters for this method are defined below. All parameters are required unless indicated otherwise. For more information, see [Using Custom Variables](#).

Parameter	Description
index	The index corresponds to the number of custom variables that Google lets you use to bring data into its analytics dashboard. The index numbers range from 1 - 5. However, Tapad provides only 3 custom variables for Google Analytics.
name	The custom variable name (e.g., Visited_Platforms, Platforms_Associated, Platform_Type.).
variable	The actual variable (e.g., pa, vp, pt, etc.).
scope	Defines the type of user activity on your site. Scope values include: <ul style="list-style-type: none">• 1, (visitor-level, default)• 2, (session-level, required for a Tapad - Google Analytics data integration)• 3, (page-level)

trackEvent method

This method provides tracking code that records user interaction with website elements. Match the `trackEvent` variable name to the same name used in `setCustomVar` shown in the example below. For more information, see the [Anatomy of Event Tracking](#).

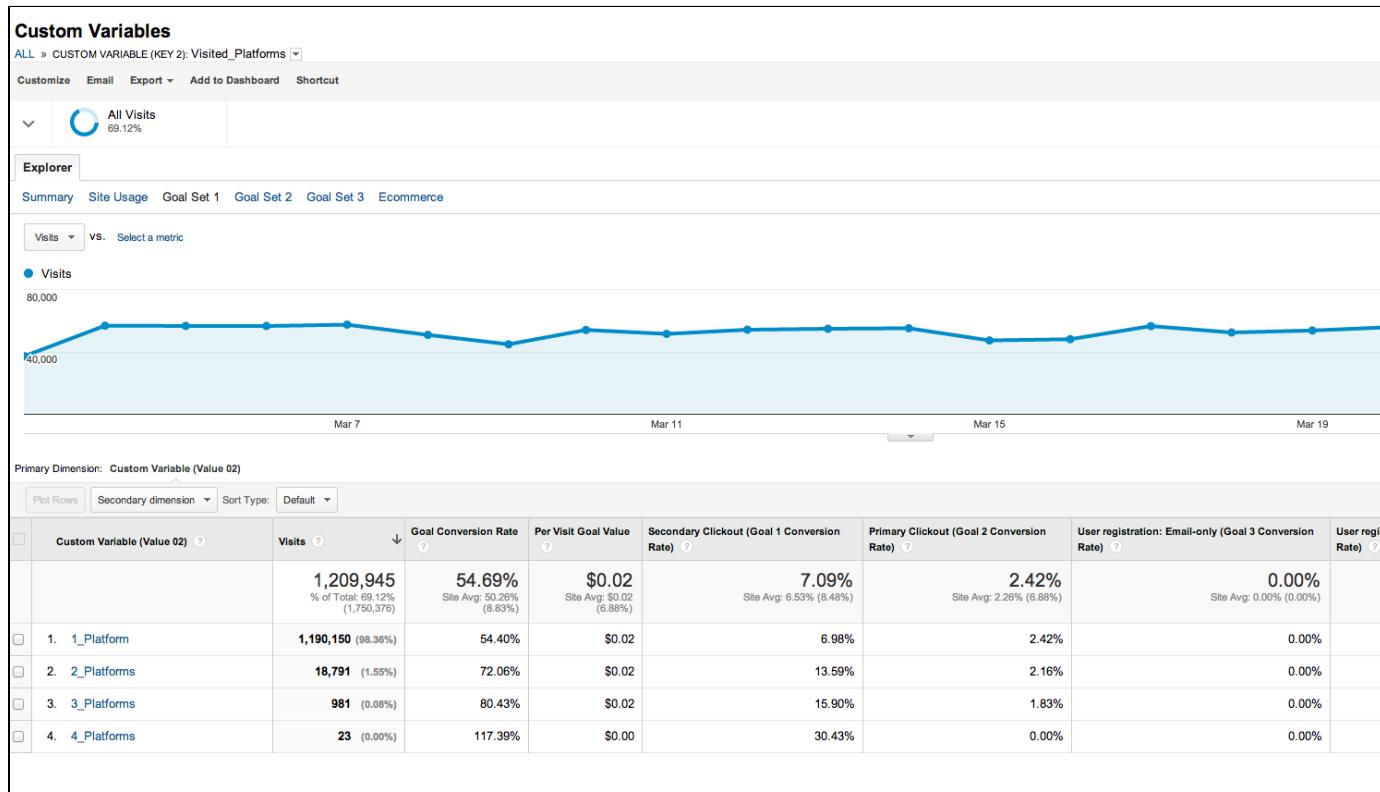
```
_gaq.push(['_setCustomVar', 1, 'Visited_Platforms', vp, 2]);  
_gaq.push(['_trackEvent', 'tapestry', 'Visited_platforms']);
```

Viewing Report Data in Google Analytics

To see your custom report in Google Analytics, go to the Standard Reports sidebar and select *Audience > Custom > Custom Variables*.

See the Google help topic [Customizing Standard Reports](#) for an overview and set up instructions.

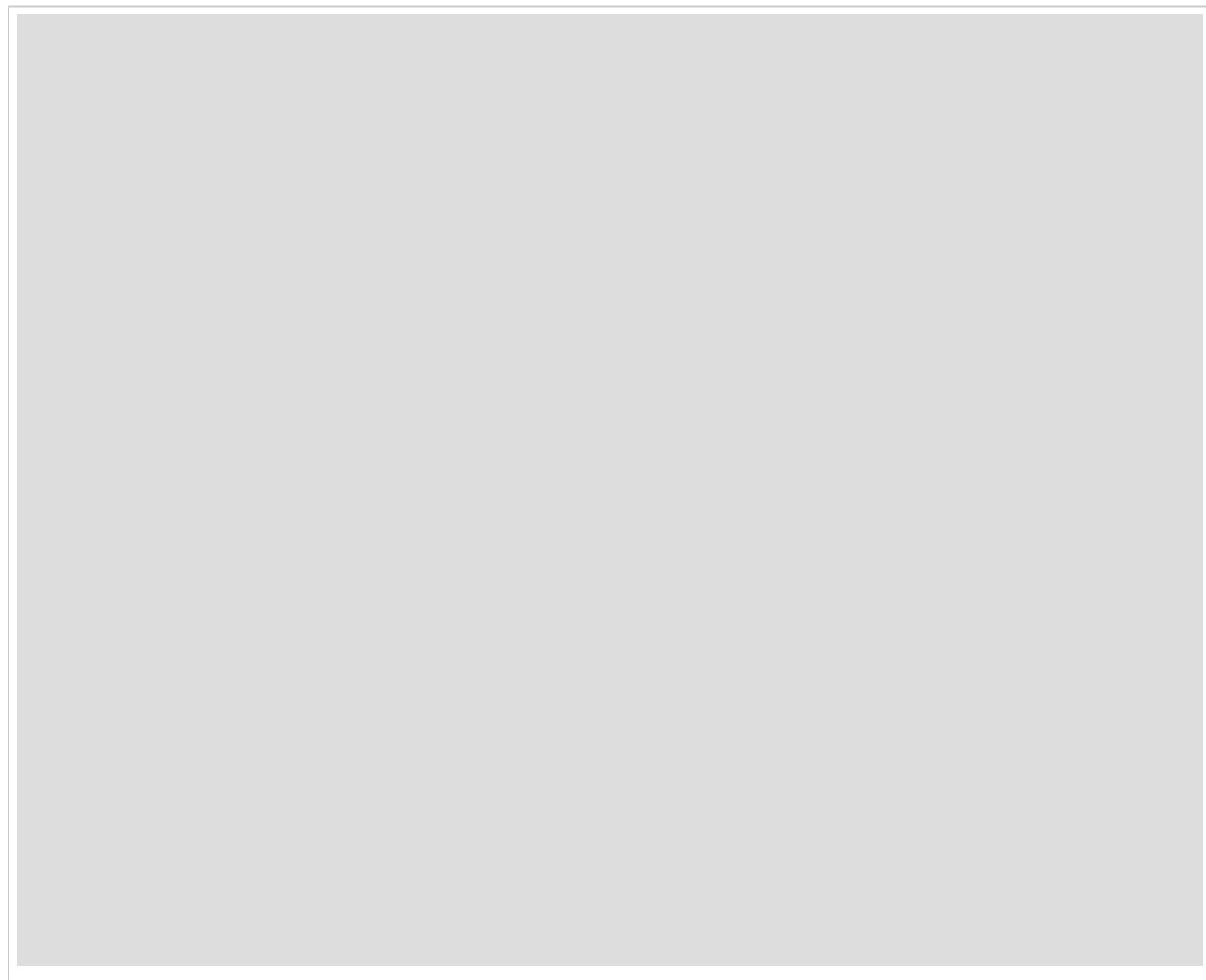
Your analytics data could look similar to this:



Tapad - Google Analytics Integration Guide

Reference material in PDF format. Updated 8/2014.

Download or view the PDF.



Analytics Plug-in SDKs

Set up instructions and code samples for Android and iOS devices. These SDKs send custom Tapad data to Google Analytics from Android or iOS mobile devices.

- [Android Plug-in Guide for the Analytics SDK](#) — Follow these instructions to set up the Analytics SDK plug-in for Android or see our quick setup guide <https://github.com/Tapad/tapestry-android-sdk/blob/master/README.md##%20Quick%20Setup> on GitHub.
- [iOS Plug-in Guide for the Analytics SDK](#) — Includes instructions and code samples that guide you through the iOS set up process.
- [Analytics Plug-in PDF](#) — Plug-in guide content in PDF format.

Android Plug-in Guide for the Analytics SDK

Follow these instructions to set up the Analytics SDK plug-in for Android or see our [quick setup guide](#) on GitHub.

True story

Tapad provides 6 custom variables, but Google Analytics accepts only 3 (see [Custom Analytics Variables for Android](#)). In Google Analytics, each custom variable gets an index number from 1-5. You must have an index number available for each Tapad variable that you want to use. For more information, see:

- [GitHub: Using the Tapestry - Google Analytics Plugin](#)
- [Google: Using Custom Variables](#)

To set up the Analytics plug-in for Android:

1. [Download](#) the latest Tapestry SDK from GitHub.
2. Copy the Tapestry .jar file to the libs folder of your Android project.
3. Open and edit the `AndroidManifest.xml` file:
 - Provide your Tapad-provided partner ID in the `android:name` parameter.
 - Set permissions to access the phone ID. At least 1 is required. See the code sample below.
 - (*Optional*): To cache requests when the phone is offline or a network connection is unavailable, add `<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />` to the XML file.
4. Assign an index number to the `value=` parameter for each Tapad variable you want to use (5 maximum). See the examples in the following code snippet.

Indexed custom variables

```
<meta-data android:name="ga.VISITED_PLATFORMS_DIM_IDX" android:value="1" />
<meta-data android:name="ga.PLATFORMS_ASSOC_DIM_IDX" android:value="2" />
<meta-data android:name="ga.PLATFORM_TYPES_DIM_IDX" android:value="3" />
```

Code sample

Your completed `AndroidManifest.xml` file could look similar to the sample below. Don't copy and paste this code. Use the download link at the top of the page to get the latest SDK code.

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.tapad.tapestry"
    android:versionCode="6"
    android:versionName="1.1" xmlns:tools="http://schemas.android.com/tools">

    <uses-sdk android:minSdkVersion="8" android:targetSdkVersion="18"/>

    <!-- Optional permissions for detecting phone connectivity and caching requests
when offline -->
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />

    <!-- Permissions for accessing the phone IDs (at least one is required) -->
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.READ_PHONE_STATE" />
    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />

    <!-- Required permissions for using Google Analytics SDK -->
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <application android:theme="@android:style/Theme.Black.NoTitleBar.Fullscreen"
tools:ignore="AllowBackup" android:icon="@drawable/ic_launcher">
        <!-- Tapestry partner ID provided by Tapad -->
        <meta-data android:name="tapad.PARTNER_ID" android:value="partner_id_here"/>

        <!-- Google Analytics custom dimension indices for Tapestry data -->
        <meta-data android:name="ga.VISITED_PLATFORMS_DIM_IDX" android:value="1" />
        <meta-data android:name="ga.PLATFORMS_ASSOC_DIM_IDX" android:value="2" />
        <meta-data android:name="ga.PLATFORM_TYPES_DIM_IDX" android:value="3" />

        <activity android:name="com.tapad.sample.GoogleAnalyticsActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>

```

Custom Analytics Variables for Android

Push Tapad data into Google Analytics with these Android SDK variables.

Google Analytics accepts custom data from up to 5 custom variables. However, you can only send data from the following 3 Tapad variables to Google Analytics. Also, each custom variable requires its own index value, which is a number between 1 - 5. You must have a free index value to send Tapad data to Google. For more information about configuring these variables, see the [Android Plug-in Guide for the Analytics SDK](#). For more information about custom variables, see [Using Custom Variables](#) in the Google Analytics documentation.

Note, *italic* type indicates variable data. Click a variable name for more information.

Variable	Android parameter
Visited Platforms	<meta-data android:name="ga.VISITED_PLATFORMS_DIM_IDX" android:value=" <i>value</i> " />
Platforms Associated	<meta-data android:name="ga.PLATFORMS_ASSOC_DIM_IDX" android:value=" <i>value</i> " />
Platform Types	<meta-data android:name="ga.PLATFORM_TYPES_DIM_IDX" android:value=" <i>value</i> " />

iOS Plug-in Guide for the Analytics SDK

Includes instructions and code samples that guide you through the iOS set up process.

- **Configure the Tapestry iOS SDK** — Follow these instructions to download and set up the Analytics plug-in for iOS or see our quick setup guide <https://github.com/Tapad/tapestry-ios-sdk#quick-setup> on GitHub.
- **Edit the AppDelegate.m File** — Follow these instructions to configure the AppDelegate.m file and specify custom variables for data collection.
- **Custom Analytics Variables for iOS** — Push Tapad data into Google Analytics with these iOS SDK variables. Define these in the iOS SDK AppDelegate.m file.

Configure the Tapestry iOS SDK

Follow these instructions to download and set up the Analytics plug-in for iOS or see our [quick setup guide on GitHub](#).

To set up the Tapestry SDK for iOS:

1. [Download](#) the latest Tapestry SDK from GitHub.
2. Decompress the file and drag Tapestry.xcodeproj into your app project.
3. Select your target and add the following under the **Build Phases** tab:
 - Target dependency: Tapestry
 - Link binary: libTapestry.a
 - Link binary: SystemConfiguration.framework
 - Link binary: AdSupport.framework
4. Choose the **Build Settings** tab and add:
 - Header search paths: \${BUILD_PRODUCTS_DIR}/include/ (select recursive when adding)
 - Other linker flags: -ObjC
5. Choose the **Info** tab and add the key TapestryPartnerID to the custom iOS target properties. Set the key's value to your partner ID.

Next steps

Add custom Tapad variables to the `AppDelegate.m` file. See [Edit the AppDelegate.m File](#).

Edit the AppDelegate.m File

Follow these instructions to configure the `AppDelegate.m` file and specify custom variables for data collection.

True story

Tapad provides 6 custom variables, but Google Analytics accepts only 3 (see [Custom Analytics Variables for iOS](#)). In Google Analytics, each custom variable gets an index number from 1-5. You must have an index number available for each Tapad variable that you want to use. For more information, see:

- GitHub: [Using the Tapestry - Google Analytics Plugin](#)
- Google: [Using Custom Variables](#)

Prerequisite

Before you can complete these steps, download and set up the basic SDK code. See [Configure the Tapestry iOS SDK](#) for more information.

To set up the `AppDelegate.m` file:

1. Edit the code to include your Google Analytics tracking ID.

```
***** Set your tracking ID here *****/
static NSString *const kTrackingId = @"YOUR TRACKING ID HERE";
static NSString *const kAllowTracking = @"allowTracking";
```

2. Edit the code to include the Tapad variables that you want to use.

```
// Visited Platforms
[self.tracker send:[[[GAIDictionaryBuilder createAppView] set:[[analytics
objectForKey:@"vp"] description]
forKey:[GAIFields
customDimensionForIndex:1]] build]];
// Platforms Associated
[self.tracker send:[[[GAIDictionaryBuilder createAppView] set:[[analytics
objectForKey:@"pa"] description]
forKey:[GAIFields
customDimensionForIndex:2]] build]];
// Platform Types
[self.tracker send:[[[GAIDictionaryBuilder createAppView] set:[[analytics
objectForKey:@"pt"] description]
forKey:[GAIFields
customDimensionForIndex:3]] build]];
```

Custom Analytics Variables for iOS

Push Tapad data into Google Analytics with these iOS SDK variables. Define these in the iOS SDK `AppDelegate.m` file.

Google Analytics accepts custom data from up to 5 custom variables. However, you can only send data from the following 3 Tapad variables to Google Analytics. Also, each custom variable requires its own index value, which is a number between 1 - 5. You must have a free index value to send Tapad data to Google. For more information about configuring these variables, see the [Edit the AppDelegate.m File](#). For more information about custom variables, see [Using Custom Variables](#) in the Google Analytics documentation.

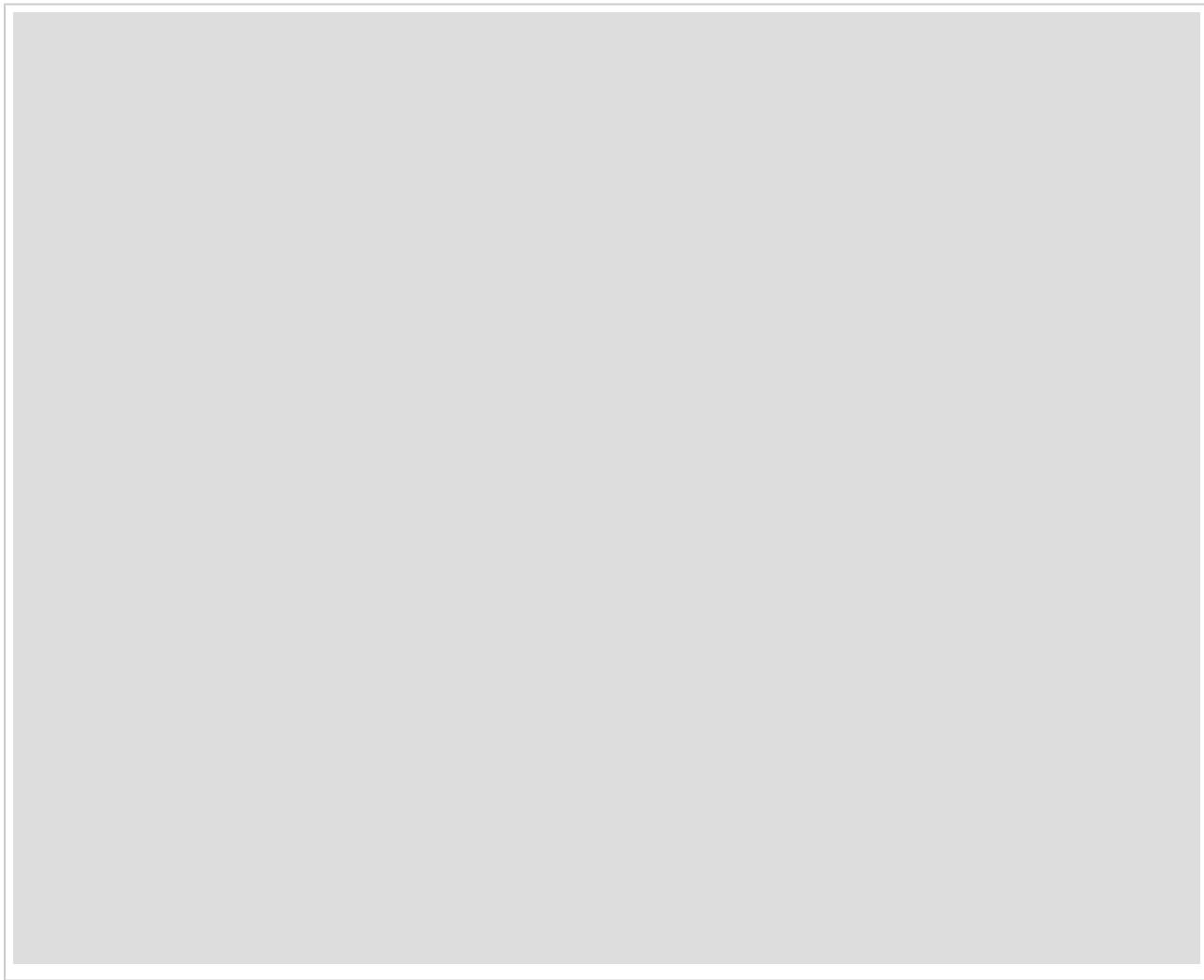
Note, *italic* type indicates variable data. Click a variable name for more information.

Variable	iOS parameter
Visited Platforms	<pre>[self.tracker send:[[[GAIDictionaryBuilder createAppView] set:[[analytics objectForKey:@"vp"] description] forKey:[GAIFields customDimensionForIndex:<i>index value</i>]] build]];</pre>
Platforms Associated	<pre>[self.tracker send:[[[GAIDictionaryBuilder createAppView] set:[[analytics objectForKey:@"pa"] description] forKey:[GAIFields customDimensionForIndex:<i>index value</i>]] build]];</pre>
Platform Types	<pre>[self.tracker send:[[[GAIDictionaryBuilder createAppView] set:[[analytics objectForKey:@"pt"] description] forKey:[GAIFields customDimensionForIndex:<i>index value</i>]] build]];</pre>

Analytics Plug-in PDF

Plug-in guide content in PDF format.

To download, click in the PDF frame.



Analytics Plug-in Metrics Defined

Contains information and descriptions about how the plug-in metrics work. Currently, a few of these metrics don't work properly or have misleading labels. Review this material for your own understanding, but do not share this content with external customers.

- **Visited Platforms** — Visited Platforms counts user visits by platform group. Over time, this data can help you understand how many platform types site visitors are using to interact with your online properties.
- **Platforms Associated** — Platforms Associated counts visits for devices grouped together as a collection of identical platforms. By counting visits by groups of devices, Platforms Associated helps you understand the levels of connectivity among your site visitors. For example, data provided by this metric lets you know if your visitors have 1 device or different types of devices.
- **Platform Types** — Platform Types counts user visits by groups of named platform types (e.g., iPhone, computer, iPad, etc.). Over time, this data helps provide insight into what types of specific devices and device combinations site visitors are using to interact with your online properties.
- **Platforms Associated and Visited Platforms Compared** — An overview of how Platforms Associated and Visited Platforms count site visits and associate visits to a ranked platform group.
- **First Visited Platform** — First Visited Platform counts the devices a visitor uses to access your site for the first time. More importantly, subsequent visits by the same user on another new device do not add a first visit record for that new device. Instead of adding a device, FVP increments the original device count. For example, let's say a first time visitor comes to your site on their iPhone. The FVP metric would count 1 iPhone. Good enough so far, right? Well, here's where it gets tricky. Let's say the same user re
- **Most Often Visited** — Most Often Visited counts the device used to visit your site most frequently. However, it does not provide a true count of the number of most often visited devices. The count increments only once each time a device gets a majority of total visits. This metric is scheduled to be updated in a future release. It is available to Adobe Analytics only.
- **Most Recent Visited** — Most Recent Visited is a frequency count of the number and types of devices that are used to visit your website. It doesn't really tell you which devices visited most recently, just the totals for each. This is basic data that publishers can probably get through their own web analytics. As a result, Most Recent Visited may not be all that useful. This metric is scheduled to be updated in a future release. It is available to Adobe Analytics only.

Visited Platforms

Visited Platforms counts user visits by platform group. Over time, this data can help you understand how many platform types site visitors are using to interact with your online properties.

Understanding platform groups

For Visited Platforms, a platform group consists of sets of devices connected together in the Device Graph. The report ranks these platform groups into categories like 1-platform, 2-platform, 3-platform, etc. The ranking is based on how many different *types* of devices (not individual devices) are connected to the device that visits a site. For example, a one-platform group could contain a single device or multiple devices of the same type (e.g., 1 computer or 10 computers). A multiple platform group contains different types of devices. For example, a 2-platform group might contain one or more computers and iPhones. A 3-platform group might contain one or more computers, iPhones, and tablets.

Counting visits for Visited Platform groups

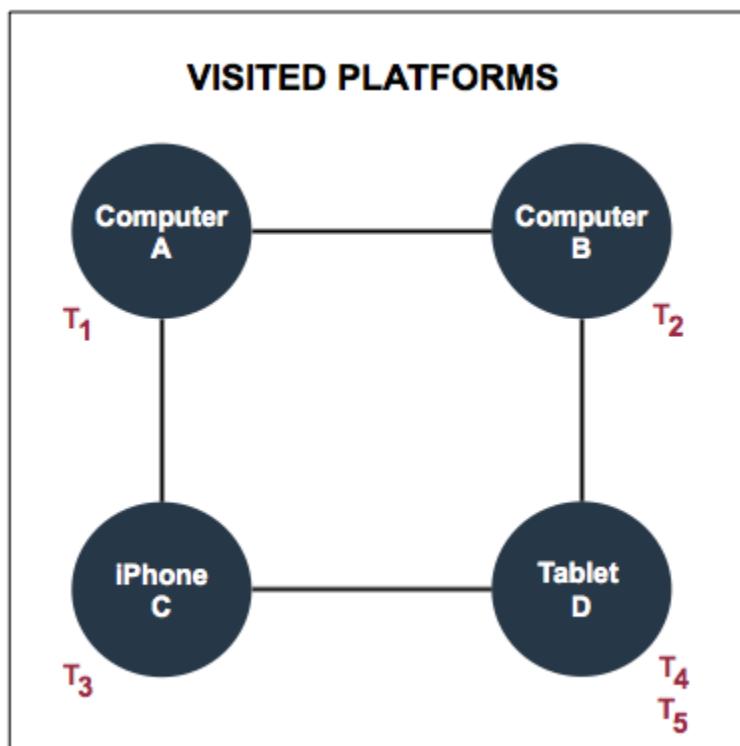
Visited Platforms counts visits as follows:

- **1-platform group:** The report increments the count each time a device from this group visits your site.
- **Multiple platform group:** The first visit from any device is counted as a 1-platform visit. The next visit from a different device would be counted under the 2-platform category. A third visit from still yet another different device would be counted under the 3-platform category and so on.
- **Subsequent visits from identical devices:** Multiple visits from the same device increment the visit count for the current platform level.

How it works

To help you understand how Visited Platforms works, let's take a look at the diagram and table below.

In this example, we have 4 devices connected in the Device Graph. However, because 2 devices (the computers) are identical, they're counted only once for platform classification purposes. This gives us a 3-platform group composed of computers, an iPhone, and a tablet. Each device visits a site as indicated by the T_n time sequence number.



Now, let's take a look at the table. It tracks how the visit count changes as each connected device visits your site.

	Visits per unit time (T_n)
Visited platforms	

	T₁	T₂	T₃	T₄	T₅
1 platform	1	2	2	2	2
2 platform			1	1	1
3 platform				1	2

In this example, computers A and B are the first devices to visit the site (at T₁ and T₂). Because these visits involve the same device type, each visit falls into the 1-platform category. This gives us 2 visits for 1 platform. After the computers, the iPhone is the next device to visit the site at T₃. Because the iPhone is a new device type, the Visited Platform metric records 1 visit in the 2 platform category. After the iPhone, the tablet visits the website at T₄. Because the tablet is another new device type, the Visited Platform metric records 1 visit in the 3 platform category. Finally, the tablet returns again as the last visitor at T₅. This action gives us 2 visits for the 3 platform category. Given these visits, Visited platforms would show you results like this:

Visited platforms	Visits
1 platform	2
2 platforms	1
3 platforms	2

Related concepts

[Platforms Associated and Visited Platforms Compared](#)

Platforms Associated

Platforms Associated counts visits for devices grouped together as a collection of identical platforms. By counting visits by groups of devices, Platforms Associated helps you understand the levels of connectivity among your site visitors. For example, data provided by this metric lets you know if your visitors have 1 device or different types of devices.

Understanding platform groups

For Platforms Associated, a platform group consists of sets of devices connected together in the Device Graph. The report ranks these platform groups into categories like 1-platform, 2-platform, 3-platform, etc. The ranking is based on how many different *types* of devices (not individual devices) are connected to the device that visits a site. For example, a one-platform group could contain a single device or multiple devices of the same type (e.g., 1 computer or 10 computers). A multiple platform group contains different types of devices. For example, a 2-platform group might contain one or more computers and iPhones. A 3-platform group might contain one or more computers, iPhones, and tablets. The platform group for a collection of connected devices only changes when a new type of device gets added to the existing set.

Counting visits for Platform Associated groups

Platforms Associated counts visits as follows:

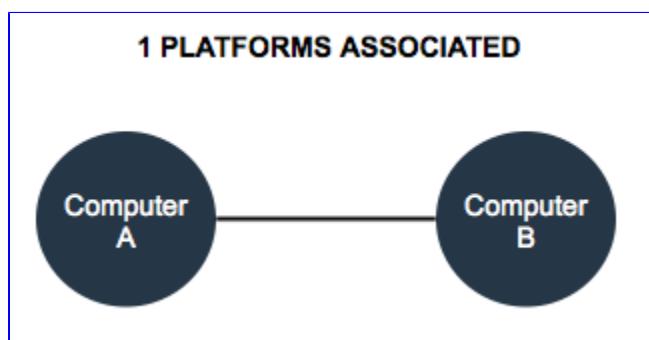
- **First visit:** The first visit from any device in the platform group is counted under that ranked platform. For example, if you have a 2-platform group, the visit count always appears in that category.
- **Subsequent visits:** Multiple visits from any device in the platform group increments the visit count for that platform.

Examples

The following examples walk you through how this metric returns data at the 1, 2, and 3-platform level.

1 associated platform

In this example, computer A and computer B are connected in the Device Graph. Because Platforms Associated groups identical device types together, the devices in this collection are classified as 1 associated platform (computer only).



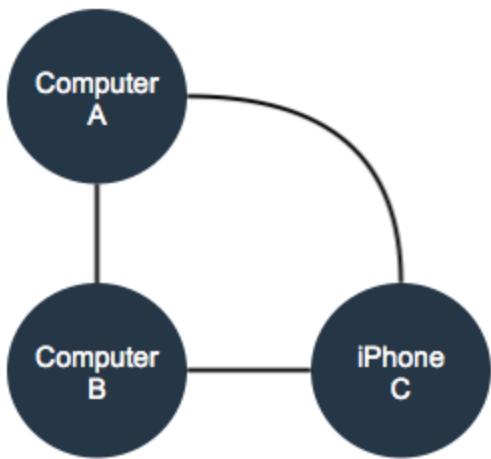
The Platforms Associated variable increments the visit count each time either computer visits your site. For example, if each computer visits your site 5 times, Platforms Associated would count 10 visits (2 devices x 5 visits each). Your Platforms Associated data would look like this:

Platforms associated	Visits
1 Platform	10

2 associated platforms

In this example, computers A and B and an iPhone are connected in the Device Graph. Because Platforms Associated groups identical device types together, the devices in this collection are classified as 2 associated platforms (computer and iPhone).

2 PLATFORMS ASSOCIATED



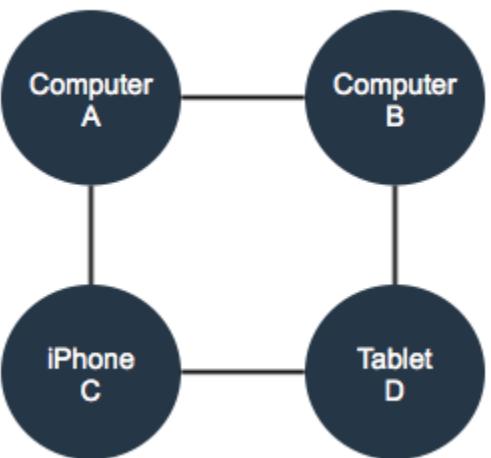
The Platforms Associated variable increments the visit count each time any of these devices visit your site. For example, if each device visits your site 5 times, Platforms Associated would count 15 visits (3 devices x 5 visits each). Your Platform Associated data would look like this:

Platforms associated	Visits
2 Platform	15

3 associated platforms

In this example, computers A and B, an iPhone, and a tablet are connected in the Device Graph. Because Platforms Associated groups of identical device types together, the devices in this collection are classified as 3 associated platforms (computer, iPhone, and tablet).

3 PLATFORMS ASSOCIATED



The Platforms Associated variable increments the visit count each time any of these devices visits your site. For example, if each device visits your site 5 times, Platforms Associated would count 20 visits (4 devices x 5 visits each). Your Platform Associated data would look like this:

Platforms associated	Visits
3 Platform	20

Related concepts

[Platforms Associated and Visited Platforms Compared](#)

Platform Types

Platform Types counts user visits by groups of named platform types (e.g., iPhone, computer, iPad, etc.). Over time, this data helps provide insight into what types of specific devices and device combinations site visitors are using to interact with your online properties.

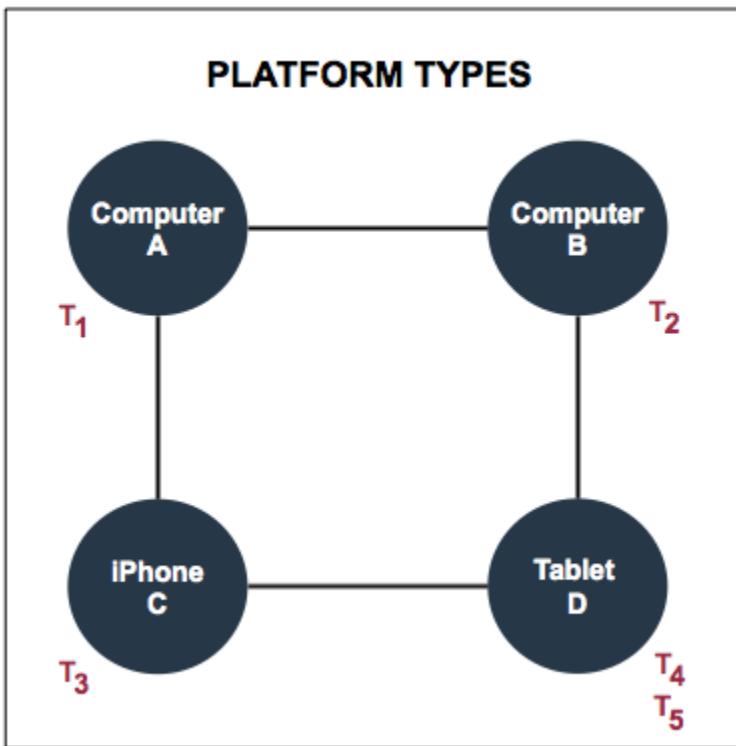
Understanding named platform groups

Like the groups used by [Visited Platforms](#), the named platform groups used by Platform Types are sets of devices connected together in the Device Graph. However, unlike a generic platform group, a named group lists the types of devices used to visit your site. For example, a single named group could contain a single device or multiple devices of the same type (e.g., 1 computer or 10 computers). A named group that lists different devices consists of 2 or more different device types (e.g., a computer and an iPhone). Each new device adds its type to the Platform Types list and starts recording visits for the newly created typed platform group.

Counting visits for typed groups

To help you understand how Platform Types works, let's take a look at the diagram and table below.

In this example, we have 4 devices connected in the Device Graph. However, because 2 devices (the computers) are identical, they're grouped into a single "computer" named group only. This gives us a maximum of 3 different groups of device types, computers, an iPhone, and an iPad. Each device visits the same site as indicated by the T_n time sequence number.



Now, let's take a look at the table. It tracks how the named platform group and device visit count changes as each connected device visits your site.

Platform Types	Visits per unit time (T_n)				
	T ₁	T ₂	T ₃	T ₄	T ₅
Computer	1	2	2	2	2
Computer, iPhone			1	1	1
Computer, iPad, iPhone			1	2	

In this example, computers A and B are the first devices to visit the site (at T₁ and T₂). Because these visits involve the same device type, these visits are classified as "computer" visits. This gives us 2 visits for 1 platform. After the computers, the iPhone is the next device to visit the site at T

3. Because the iPhone is a new device type, the Platform Types records 1 visit for 2 named platforms (computer, iPhone). After the iPhone, the tablet visits the website at T_4 . Because the tablet is another new device type, the Platform Types records 1 visit for the "Computer, iPad, iPhone" named types. Finally, the tablet returns again as the last visitor at T_5 . This action gives us 2 visits for the 3 named devices group. Given these visits, Platform Types would show you results like this:

Platform Types	Visits
Computer	2
Computer, iPhone	1
Computer, iPad, iPhone	2

Platforms Associated and Visited Platforms Compared

An overview of how Platforms Associated and Visited Platforms count site visits and associate visits to a ranked platform group.

Similarities: Platform groups

Both [Platforms Associated](#) and [Visited Platforms](#) organize devices into collections called a *platform group*. A platform group contains sets of devices that are connected together in the Device Graph. These metrics rank platform groups into categories like 1-platform, 2-platform, 3-platform, etc. The platform group ranking is based on how many different *types* of devices (not individual devices) are connected to the device that visits a site. For example, a single platform group could contain just one device or multiple devices of the same type (e.g., 1 computer or 10 computers). A multiple platform group contains 2 or more different types of devices. For example, a 2-platform group might contain one or more computers and iPhones. A 3-platform group might contain one or more computers, iPhones, and tablets. Data for both metrics is grouped by these ranked platform categories.

Differences: Counting devices by platform group and changing group rankings

Platforms Associated and Visited Platforms count visits by group and assign visits to a platform group as described below.

Platforms Associated

Platform group action	Description
Assigning visits	Platforms Associated records and assigns site visits to the largest platform group for that collection of devices. For example, let's say you use an iPhone to visit a website. In the Device Graph, that iPhone is connected to one or more computers and tablets. This set (the iPhone, the computers, and the tablets) constitute a 3-platform group. As a result, the iPhone visit is recorded under the 3-platform category. Also, subsequent visits by any of the devices in this set increment the 3-platform visit count as well.
Changing groups	The platform group for a collection of connected devices only changes when a new type of device gets added to the existing set. For example, let's say you have a 2-platform group. This group contains one or more iPhones and computers. When the Device Graph adds a new type of device to this group, a tablet for example, the Platforms Associated metric considers this a 3-platform group. Any new visits from devices in this set get added to the 3-platform category. Visits recorded when this was a 2-platform group do not change.

Visited Platforms

Platform group action	Description
Assigning visits	Visited Platforms records and assigns site visits to largest platform group known at the time of the visit. For example, let's say you use an iPhone to visit a website. In the Device Graph, that iPhone is connected to one or more computers and tablets. This set (the iPhone, the computers, and the tablets) constitute a 3-platform group. However, Visited Platforms sees the iPhone as a single device type and records a visit in the 1-platform category. In this case, even though we have a 3-platform group, Visited Platforms knows about only one of them during the visit.
Changing groups	The platform group for a collection of connected devices changes when a new type of device visits the site. Given the previous example, let's say one of the computers visits the site. Because it's a new device type, Visited Platforms records a visit in the 2-platform category. Additionally, as other new device types are used to visit the site, those are counted by the next ranking platform. Subsequent visits by a previous platform type increment the count for the highest platform level only. Previously recorded visits do not change platform categories

First Visited Platform

First Visited Platform counts the devices a visitor uses to access your site for the first time. More importantly, subsequent visits by the same user on another new device *do not* add a first visit record for that new device. Instead of adding a device, FVP increments the original device count. For example, let's say a first time visitor comes to your site on their iPhone. The FVP metric would count 1 iPhone. Good enough so far, right? Well, here's where it gets tricky. Let's say the same user returns 9 more times from any other device. In this case, FVP would show 10 iPhone visits even though we have (and can identify) first visits from other devices. This metric is scheduled to be updated in a future release. It is available to Adobe Analytics only.

Most Often Visited

Most Often Visited counts the device used to visit your site most frequently. However, it does not provide a true count of the number of most often visited devices. The count increments only once each time a device gets a majority of total visits. This metric is scheduled to be updated in a future release. It is available to Adobe Analytics only.

Counting most often visited devices

We'll refer to the series of visits below to help you understand how this metric identifies a "most often visited device" and increments the device count. In this case, let's say we have 10 visits to a site from an iPhone and computer as shown in the following table.

Table 1: Visit count by device										
Device type	Visit 1	Visit 2	Visit 3	Visit 4	Visit 5	Visit 6	Visit 7	Visit 8	Visit 9	Visit 10
	Computer	iPhone	iPhone	iPhone	Computer	iPhone	Computer	Computer	Computer	Computer

Based on the sample data in Table 1, the MOV metric would show results like this:

Table 2: MOV results	
Most often visited	Visits
Computer	2
iPhone	1

Certainly, these are unusual results. We had 10 visits total, but Table 2 counts the computer twice and the iPhone once. It's clear there's something wrong with how the metric classifies the most often visited device and counts the number of visits. After all, how is the computer counted twice when we've seen it 5 times and how is the iPhone counted once when we've seen it 4 times? Well, for MOV data keep in mind that:

- Out of the total number of visits, the device seen the majority of the time is considered the device seen most often.
- Even if other devices access your site, the device considered "most often" doesn't change until another device comes to your site a majority of times.
- The MOV count changes by 1 only when a device becomes the majority device. For example, if it takes a computer 20 visits to become the most seen device, it only gets counted 1 time not 20.

Let's apply these rules to the visit count and recreate the results from Table 1.

Table 3: Counting MOV			
Visit	Device	Results	MOV count
1	Computer	The computer constitutes the majority of all the visits. By default, it is the most often visited platform.	Computer: 1 iPhone: 0
2	iPhone	The iPhone visits next and the device count split evenly. Because the device count is even (no majority device), the computer remains the most often visited device.	Computer: 1 iPhone: 0
3	iPhone	The iPhone now accounts for the majority of all visits: <ul style="list-style-type: none">• 2 iPhones• 1 computers The MOV count gets incremented by 1.	Computer: 1 iPhone: 1
4	iPhone	The iPhone still accounts for the majority of all visits: <ul style="list-style-type: none">• 3 iPhones• 1 computer	No change
5	Computer	The iPhone still accounts for the majority of all visits: <ul style="list-style-type: none">• 3 iPhones• 2 computers	No change
6	iPhone	The iPhone still accounts for the majority of all visits: <ul style="list-style-type: none">• 4 iPhones• 2 computers	No change

7	Computer	The iPhone still accounts for the majority of all visits: <ul style="list-style-type: none">• 4 iPhones• 3 computers	No change
8	Computer	The device count is split evenly: <ul style="list-style-type: none">• 4 iPhones• 4 computers Because the device count is even (no majority device), the iPhone remains the most often visited device.	No change
9	Computer	The computer now accounts for the majority of all visits: <ul style="list-style-type: none">• 4 iPhones• 5 computers The MOV count gets incremented by 1.	Computer: 2 iPhone: 1
10	Computer	The computer still accounts for the majority of all visits: <ul style="list-style-type: none">• 4 iPhones• 6 computers	No change

In this scenario, we've seen 10 visits from 2 devices (4 iPhone, 6 computer). During these visits, the computer became the majority device twice and the iPhone achieved a majority count only once. Additionally, because the MOV metric increases the count only by 1 when most often visited device changes, we get the results shown in [Table 2](#).

Fixing the problem

The way MOV counts platforms is inaccurate and misleading. It should count each visit and show you which devices were used to view a site in descending order. Given the original data in [Table 1](#) above, a MOV metric that works properly should return data shown in Table 4. This metric is scheduled to be fixed in an upcoming sprint.

Table 4: Fixed MOV results	
Most often visited	Visits
Computer	6
iPhone	4

Not used by Google Analytics

Currently, Google Analytics does not accept MOV data.

Most Recent Visited

Most Recent Visited is a frequency count of the number and types of devices that are used to visit your website. It doesn't really tell you which devices visited most recently, just the totals for each. This is basic data that publishers can probably get through their own web analytics. As a result, Most Recent Visited may not be all that useful. This metric is scheduled to be updated in a future release. It is available to Adobe Analytics only.

Data Flow: Send Audiences To Tapad

Tapad needs an audience data file and an audience taxonomy to match your audience IDs with related IDs in the Device Graph. An audience data file consists of device and audience IDs in a JSON file. A taxonomy file consists of metadata related to the audience data file (also in a JSON format). Refer to this guide for information about file formats, syntax, examples, and delivery options.

- [Audience Data File](#) — An audience data file contains JSON data with information about audiences and devices. Tapad uses this data to create and manage your audiences. Format your audience data according to these specifications.
- [Audience Taxonomy File](#) — An audience taxonomy file contains human-readable data about the contents of your audience data files. Tapad uses the taxonomy to help our systems ingest, identify, and manage your data. Format taxonomic data as JSON objects according to these specifications.
- [Delivery Methods for Audience Data and Taxonomy Files](#) — You can send audience and taxonomy data to Tapad by batch file or real-time delivery methods. This section includes information about communication protocols, delivery schedules, and file formatting.
- [Audience Data File PDF](#) — Requirements and specifications in PDF format. Updated March, 2015.

Audience Data File

An audience data file contains JSON data with information about audiences and devices. Tapad uses this data to create and manage your audiences. Format your audience data according to these specifications.

- **JSON Format for Audience Data Files** — An audience data file contains strings, integers, arrays, and mapped data in a JSON object <http://www.json.org/>. Format the parameters in your data file according to these specifications. All parameters are required unless indicated otherwise.
- **Name and Update Your Audience Data Files** — Name and update an audience data file according to these specifications.
- **Text Format for Audience Data Files (Not Used)** — A text-formatted audience data file contains alphanumeric device and segment IDs in a simple text file. You must format the parameters in the audience data file according to these specifications. This section contains a sample data file. All parameters are required unless indicated otherwise.

JSON Format for Audience Data Files

An audience data file contains strings, integers, arrays, and mapped data in a [JSON object](#). Format the parameters in your data file according to these specifications. *All parameters are required unless indicated otherwise.*

The following table lists and defines the data elements in a properly formatted data file. In the data file, write each JSON object a separate line. See our GitHub repository for a [sample data file](#).

Parameter	Data type	Description
device_id	String	A device ID (e.g., cookie or hardware ID).
device_id_type	String	<p>Tapad-defined device type.</p> <p>When you send us your own data, set the <code>device_id_type</code> to "THIRDPARTY". You should do this because even though this is your own data (first party), it's third-party data to your friends here at Tapad. Yes, this is weird, we know, we know...</p> <p>See Device ID Types for a complete list.</p>
audience_ids	Array	<p><code>audience_ids</code> contains 2 key-value pairs in an array:</p> <ul style="list-style-type: none"> <code>audience_id</code>: Your audience IDs (data type: string). This ID must match the ID passed in by the <code>audience_id</code> parameter in the separate audience taxonomy file. <code>advertiser_ids</code>: An array of integers listing Tapad IDs for your clients. These IDs link first-party audiences to their respective advertisers. For third-party audiences, this array may not contain any IDs. Please check with your Tapad Implementation Engineer if you're unsure about which IDs to use. <p>See our GitHub repository for a sample data file.</p>
operation	Object	<p>The <code>operation</code> object includes 2 key-value pairs:</p> <ul style="list-style-type: none"> <code>updated_at</code>: (<i>Optional</i>) Uses the <code>yyyy-mm-ddhh:mm:ss</code> extended format time stamp to record when a device visited a website or other online property. <code>action</code>: A named key that accepts 3 numeric values: <ul style="list-style-type: none"> 1: Adds/appends audience IDs to a device (Tapad systems add audience IDs by default). 2: Removes audience IDs from a device. 3: Replaces all audience IDs on a device with new IDs. <p>See the object data type or sample data file for an example.</p>

Data types and format

The following table describes the data types used in the data file and their formatting requirements.

Data type	Description
String	<p><i>Strings</i> consists of alphanumeric characters like abcd, 1234 (sometimes numbers are strings), or abc1234.</p> <p>Indicate string data with either double quotes or single quotes. Whatever you decide, be consistent and don't mix punctuation marks in the same data file.</p>
Integer	An <i>integer</i> (or <i>INT</i>) consists of whole numbers only. When declared as an integer, numbers <i>do not</i> require punctuation marks.
Array	<p>An <i>array</i> is a collection of similar objects or data. When working with arrays:</p> <ul style="list-style-type: none"> Put array data between square brackets []. Enclose array elements with single or double quotes. Separate each array element with a comma. For example, <code>"advertiser_ids": [123, 456]</code>
Object / Map	<p>An <i>object</i> (or <i>map</i>) is an unordered collection of key-value pairs.</p> <p>Indicate a map with curly braces and separate each map element with a comma. For example, your data file could contain an object similar to this:</p> <pre>"operation" : {"updated_at": "2014-02-26T22:50:34+00:00", "action": 2}</pre>

Name and Update Your Audience Data Files

Name and update an audience data file according to these specifications.

Naming conventions

The following syntax defines the structure of a well-formed data file name. Note, *colored italics* indicates a variable placeholder. The other elements are required constants and do not change.

*partnerName*_audience_*yyyymmdd_hhmmss*.json.gz

In the file name, replace:

- *partnerName* with your Tapad-provided partner name.
- *yyyymmdd_hhmmss* with a date and timestamp that records when the file was created.

A properly named data file could look similar to this:

MyCompany_audience_20150217_143245.json.gz

File updates

To update your data file, create and post a new data file to your SFTP endpoint. During an update, devices remain in an audience:

- Until removed by an action value set on the [operation object](#).
- Up to 30 days since the device was added to an audience. To ensure that devices are not removed after 30 days, send us a complete snapshot of your audience data files at least every 2 weeks.

Compress the file with gzip (.gz) compression.

Text Format for Audience Data Files (Not Used)

Oh snap!

Tapad *does not* use this text file format. Documentation retained for legacy purposes only. See



[DOC-261](#) - Update "Data Flow: Send Audiences To Tapad" documentation RESOLVED.

A text-formatted audience data file contains alphanumeric device and segment IDs in a simple text file. You must format the parameters in the audience data file according to these specifications. This section contains a sample data file. *All parameters are required unless indicated otherwise.*

The following table lists and defines the data elements in a properly formatted audience data file. Note, *italics* indicates a variable.

Parameter	Data type	Description
<i>device_id</i>	String	A device ID such as a cookie or hardware ID (e.g., fb725e65-53d2-7d37-589e-7c2e159b65c0)
<i>segment_id</i>	String	A segment ID associated with the device ID (e.g., <i>segment_123</i> , or a simple ID like 1234, 5678, etc.)

Sample data file

Your audience file could look similar to the following example. The first entry in each record contains the device ID. Subsequent records in the same row contain the segment ID.

```
fb725e65-53d2-7d37-589e-7c2e159b65c0,segment_123,segment_456,segment_789
cc2f3cbe14b24390304b53701f351506,segment_123
cb824f65-42c2-7c77-958e-6c3e260b56c1,segment_456,segment_789
```

Audience Taxonomy File

An audience taxonomy file contains human-readable data about the contents of your audience data files. Tapad uses the taxonomy to help our systems ingest, identify, and manage your data. Format taxonomic data as JSON objects according to these specifications.

- [JSON Format for Audience Taxonomy Files](#) — An audience taxonomy file contains strings and integers in JSON objects <http://www.json.org/>. Format the parameters in your taxonomy file according to these specifications. All parameters are required unless indicated otherwise.
- [Name and Update Your Audience Taxonomy File](#) — Name and update an audience taxonomy file according to these specifications.
- [Text Format for Audience Taxonomy Files \(Not Used\)](#) — Format the taxonomy text file according to these specifications.

JSON Format for Audience Taxonomy Files

An audience taxonomy file contains strings and integers in **JSON objects**. Format the parameters in your taxonomy file according to these specifications. *All parameters are required unless indicated otherwise.*

The following table lists and defines the parameters used in a properly formatted audience taxonomy file. In the taxonomy file, write each JSON object on a separate line. See our GitHub repository for a [sample data file](#).

Parameter	Data type	Description
audience_id	String	An alphanumeric audience ID. This ID must match the ID passed in by the <code>audience_id</code> array in the separate audience data file .
advertiser_ids	Array	An array of integers listing Tapad IDs for your clients. These IDs link first-party audiences to their respective advertisers. For third-party audiences, this array may not contain any IDs. Please check with your Tapad Implementation Engineer if you're unsure about which IDs to use. See our GitHub repository for a sample data file .
name	String	An audience name.
description	String	(Optional) A short, informative description about the audience.

Data types and format

The following table describes the data types used in the taxonomy file and their formatting requirements.

Data type	Description
String	<i>Strings</i> consists of alphanumeric characters like abcd, 1234 (sometimes numbers are strings), or abc1234. Indicate string data with either double quotes or single quotes. Whatever you decide, be consistent and don't mix punctuation marks in the same taxonomy file.
Integer	An <i>INT</i> (or <i>integer</i>) consists of whole numbers only. When declared as an integer, numbers <i>do not</i> require any punctuation marks.
Array	An <i>array</i> is a collection of similar objects or data. When working with arrays: <ul style="list-style-type: none">Put array data between square brackets [].Enclose array elements with single or double quotes.Separate each array element with a comma. For example, "advertiser_ids": [123, 456]

Name and Update Your Audience Taxonomy File

Name and update an audience taxonomy file according to these specifications.

Naming conventions

The following syntax defines the structure of a well-formed data file name. Note, *colored italics* indicates a variable placeholder. The other elements are required constants and do not change.

*partnerName*_taxonomy_*yyyymmdd_hhmmss*.json.gz

In the file name, replace:

- *partnerName* with your Tapad-provided partner name.
- *yyyymmdd_hhmmss* with a date and timestamp that records when the file was created.

A properly named taxonomy file could look similar to this:

MyCompany_taxonomy_20140214_142033.json.gz

The following table lists and defines the elements in a well-formed file name.

File updates

To update your taxonomy, post a new file to your SFTP endpoint. Updates add new records and modify existing records only. They do not duplicate or delete records. Compress the file with gzip (.gz) compression.

Text Format for Audience Taxonomy Files (Not Used)

Oh snap!

Tapad *does not* use this text file format. Documentation retained for legacy purposes only. See



[DOC-261 - Update "Data Flow: Send Audiences To Tapad" documentation](#) RESOLVED.

Format the taxonomy text file according to these specifications.

The following table lists and defines the data elements in a properly formatted audience taxonomy file. Note, *italics* indicates a variable.

Parameter	Data type	Description
<i>segment_id</i>	String	A segment ID associated with the device ID (e.g., <code>segment_123</code> , or a simple ID like <code>1234</code> , <code>5678</code> , etc.).
<i>name</i>	String	The segment name.
<i>description</i>	String	A brief, informative description about the segment (e.g., <code>auto intenders</code> , <code>electronics shoppers</code> , etc.).

Sample taxonomy file

Your taxonomy file could look similar to the following example. The first entry in each record contains the segment ID. Subsequent records in the same row contain the segment name, and a description.

```
segment_123,documentary fans,movie segment
segment_456,auto repair,automotive enthusiasts
segment_789,college life,higher ed interests
```

Delivery Methods for Audience Data and Taxonomy Files

You can send audience and taxonomy data to Tapad by batch file or real-time delivery methods. This section includes information about communication protocols, delivery schedules, and file formatting.

Batch file delivery

Batch file delivery method relies on the following:

- **Communication protocols:** SFTP only (no FTP). Tapad prefers to download audience and taxonomy data from your SFTP server. Tapad can provide an SFTP endpoint if you do not have one.
- **Delivery schedule:** Nightly.
- **File format:** Save the audience data and taxonomy files as separate gzip compressed files on your SFTP server. See the [Audience Data File](#) and [Audience Taxonomy File](#) sections for information about file format and name requirements.
- **Compression:** Compact each file with gzip compression before sending.

Real-time file delivery

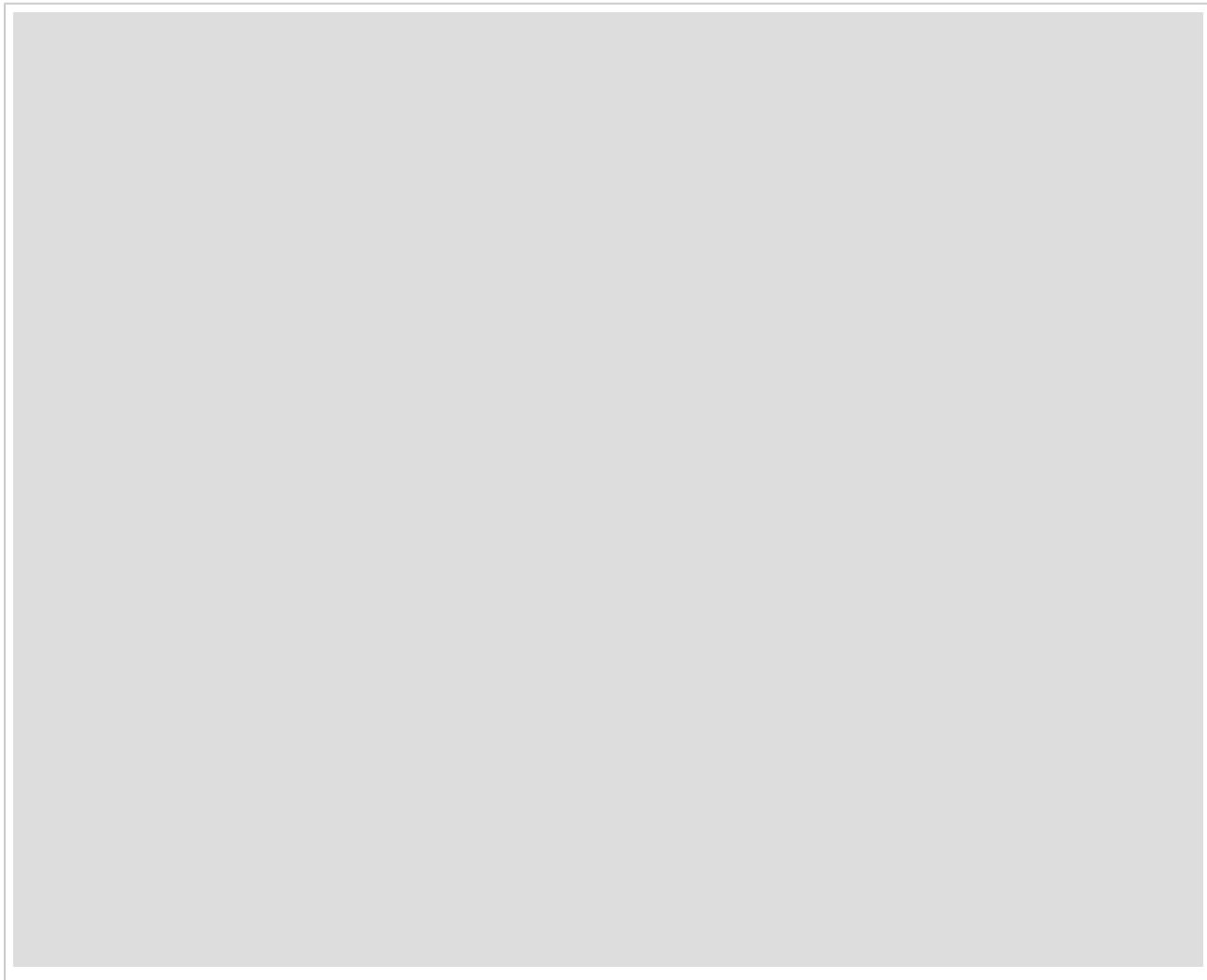
Send data to Tapad in real-time with a `POST` method (for audience data) and `PUT` method (for taxonomy data). Note, *italic* type indicates variable data

Data type	Description
Audience data	<p>Send audience data to Tapad with the following <code>POST</code> method:</p> <pre>POST http://data.tapad.com/dmp/dmp_prefix/device/encoded_device_ID</pre> <p>Variable elements in this method include:</p> <ul style="list-style-type: none">• DMP prefix: A Tapad-provided abbreviation for your company.• URL encoded device ID <div style="border: 1px solid #fca; padding: 5px; margin-top: 10px;"><p>Heads up! You must URL-encode device IDs.</p></div>
Taxonomy data	<p>Send taxonomy data to Tapad with the following <code>PUT</code> method:</p> <pre>PUT http://data.tapad.com/dmp/dmp_prefix/device/audience_ID</pre> <p>Variable elements in this method include:</p> <ul style="list-style-type: none">• DMP prefix: A Tapad-provided abbreviation for your company.• Audience ID: The alphanumeric audience ID.

Audience Data File PDF

Requirements and specifications in PDF format. *Updated March, 2015.*

To download, click in the PDF frame.



Data Flow: Receive Connected IDs From Tapad

ID matching is a process that takes device IDs you've sent to Tapad and matches them to other device IDs. As determined by our Device Graph, these new IDs represent devices that share a connection with your original ID. When the matching process is complete, Tapad returns this data to you as a JSON object or in a flat file. You can use this data to target your customers across all their known devices, deliver personalized content, and use it with your own analytics systems or internal processes. Because partner systems and data requirements vary, Tapad can structure and deliver this information to you using the file format, content, and delivery options described in this material.

- [JSON Data Files: Syntax and Content Options](#) — Syntax, definitions, and examples for synchronized IDs in a JSON object. Content options include basic ID data, IDs with scores, and IDs with platform types.
- [Text Data Files: Syntax and Content Options](#) — Syntax for receiving synchronized IDs in tab-separated (.tsv) text file. Content options include basic ID data, IDs with scores, and IDs with platform types.
- [Data File Delivery Methods and Filter Options](#) — Basic and enhanced delivery options.
- [Data Flow Appendix](#) — Supplemental reference material related to Tapad Data Flow processes.
- [Data Flow PDFs](#) — PDF formatted content that describes options for sending synchronized IDs to Tapad partners. Updated March, 2015.

JSON Data Files: Syntax and Content Options

Syntax, definitions, and examples for synchronized IDs in a JSON object. Content options include basic ID data, IDs with scores, and IDs with platform types.

- [Basic JSON Data](#) — Syntax, definitions, and examples for synchronized ID data in a JSON object.
- [JSON Data With Platform Types](#) — Syntax, definitions, and examples for synchronized ID data in a JSON object, including optional platform types. The platform type is a human-readable term that describes the device itself and can include results like iPad, Android, Kindle, iPhone, etc.
- [JSON Data With Scores](#) — Syntax, definitions, and examples for synchronized ID data in JSON data, including optional scores that quantify the relationships between connected devices. A higher score suggests a closer connection between devices than a lower score. These scores help make accurate, cross-device targeting possible.

Basic JSON Data

Syntax, definitions, and examples for synchronized ID data in a JSON object.

File syntax and descriptions

A basic JSON data object contains device IDs for your users that have been matched to other, corresponding devices in the Device Graph. The file will contain data for your account only. The basic JSON object uses the syntax shown below. Note, *italic* type indicates variable data.

Syntax:

```
"primary_id" : { "id_type" : "hardware_or_cookie" , "id" : "device_id" }
"related_ids" : [{ "id_type" : "hardware_or_cookie", "id" : "device_id"}, {"id_type" : "hardware_or_cookie",
 "id" : "device_id" }]
```

These parameters are defined as follows:

Parameter	Description
primary_id	IDs sent to Tapad for synchronization with other devices in the Device Graph. Primary IDs and related IDs represent your synchronized data.
related_ids	An array of device IDs connected to the primary device ID in the Device Graph. Primary IDs and related IDs represent your synchronized data.
id_type	A variable for a hardware or cookie type, for example: <ul style="list-style-type: none">• "id_type": "HARDWARE_IDFA"• "id_type": "BDC" (In this case, ID types for cookies are identified by a 2-3 character abbreviation of the partner they belong to) See the device ID type list for more information.
id	The actual device ID. For example "id": "c1573b64c661d3d76ee4869e5f1712de" <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">Note, hardware IDs are always lower case.</div>

Sample content

The contents of your JSON file could look similar to the example below. The data has been truncated for brevity. Your JSON file will contain a complete, unencoded ID string.

```
{
  "primary_id": {"id_type": "bdc", "id": "12345"},
  "related_ids": [{"id_type": "HARDWARE_MD5ANDROIDIDPHONEID", "id": "c1573b64c661d3d76ee4869e5f1712de"}, {"id_type": "HARDWARE_IDFA", "id": "fca73806-3426"}, {"id_type": "HARDWARE_SHA1IDFA", "id": "68753a44-4d6f"}, {"id_type": "BDC", "id": "c1573b64"}]
```

JSON Data With Platform Types

Syntax, definitions, and examples for synchronized ID data in a JSON object, including optional platform types. The platform type is a human-readable term that describes the device itself and can include results like iPad, Android, Kindle, iPhone, etc.

File syntax and descriptions

A JSON data file with optional platform types contains device IDs that have been matched to other, corresponding devices in the Device Graph. This file contains data for your account only. A JSON data file with platforms uses the syntax shown below. Note, *italic* type indicates variable data.

Syntax

```
"primary_id" : {"id_type" : "hardware_or_cookie", "id" : "device_ID", "platform" : "platform_type" },
"related_ids" : [{"id_type" : "hardware_or_cookie", "id" : "device_ID", "platform" : "platform_type"}]
```

These parameters are defined as follows:

Parameter	Description
primary_id	IDs sent to Tapad for synchronization with other devices in the Device Graph. Primary IDs and related IDs represent your synchronized data.
related_ids	An array of device IDs connected to the primary device ID in the Device Graph. Primary IDs and related IDs represent your synchronized data.
id_type	A variable for a hardware or cookie type, for example: <ul style="list-style-type: none">• "id_type": "HARDWARE_IDFA"• "id_type": "BDC" (In this case, ID types for cookies are identified by a 2-3 character abbreviation of the partner they belong to) See the device type ID list for more information.
id	The actual device ID. For example "id": "c1573b64c661d3d76ee4869e5f1712de" <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">Note, hardware IDs are always lower case.</div>
platform	A key-value pair that contains a human-readable term that describes the device (see Platform Types). Returns UNKNOWN when we cannot determine the platform type or if that information is missing.

Sample content

The contents of your JSON file could look similar to the example below. The data has been truncated for brevity. Your JSON file will contain a complete, unencoded ID string.

```
{
  "primary_id": {"id_type": "bdc", "id": "12345", "platform": "Computer"},
  "related_ids": [
    {"id_type": "HARDWARE_MD5ANDROIDIDPHONEID", "id": "c1573b64c661d", "platform": "Android"},
    {"id_type": "HARDWARE_IDFA", "id": "fca73806-3426", "platform": "iPhone"}
  ]
}
```

JSON Data With Scores

Syntax, definitions, and examples for synchronized ID data in JSON data, including optional scores that quantify the relationships between connected devices. A higher score suggests a closer connection between devices than a lower score. These scores help make accurate, cross-device targeting possible.

File syntax and descriptions

A JSON data file with optional scores contains device IDs that have been matched to other, corresponding devices in the Device Graph. This file contains data for your account only. JSON data files use the syntax shown below. Note, *italic* type indicates variable data.

Syntax

```
"primary_id" : {"id_type" : "hardware_or_cookie", "id" : "device_ID",},  
"related_ids" : [{ "id_type" : "hardware_or_cookie", "id" : "device_ID", "strength" : "score_value"}]
```

These parameters are defined as follows:

Parameter	Description
primary_id	IDs sent to Tapad for synchronization with other devices in the Device Graph. Primary IDs and related IDs represent your synchronized data.
related_ids	An array of device IDs connected to the primary device ID in the Device Graph. Primary IDs and related IDs represent your synchronized data.
id_type	A variable for a hardware or cookie type, for example: <ul style="list-style-type: none">• "id_type": "HARDWARE_IDFA"• "id_type": "BDC" (In this case, ID types for cookies are identified by a 2-3 character abbreviation of the partner they belong to) See the device type ID list for more information.
id	The actual device ID. For example, "id": "c1573b64c661d3d76ee4869e5f1712de" <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">Note, hardware IDs are always lower case.</div>
strength	A key in the named score data pair that defines the relationship between a primary ID and a related ID. The key's score value is an integer with a range of 1-10. Scores are returned on devices in the related_ids array.

Sample content

The contents of your JSON file could look similar to the example below. The data has been truncated for brevity. Your JSON file will contain a complete, unencoded ID string.

```
{  
  "primary_id": {"id_type": "bdc", "id": "12345"},  
  "related_ids": [  
    {"id_type": "HARDWARE_MD5ANDROIDIDPHONEID", "id": "c1573b64c661d", "strength": "3"},  
    { "id_type": "HARDWARE_IDFA", "id": "fca73806-3426", "strength": "2"}  
    //This line shows the platform type and score.  
    { "id_type": "HARDWARE_IDFA", "id": "fca73806-3426", "platform": "ipad",  
    "strength": "2"}]  
}
```

Text Data Files: Syntax and Content Options

Syntax for receiving synchronized IDs in tab-separated (.tsv) text file. Content options include basic ID data, IDs with scores, and IDs with platform types.

- [Basic Text Data File](#) — Syntax, definitions, and examples for synchronized ID data in a tab-separated (.tsv) text file.
- [Text Data File With Platform Types](#) — Syntax, definitions, and examples for synchronized ID data in a tab-separated (.tsv) text file, including optional platform types. The platform type is a human-readable term that describes the device itself and can include results like iPad, Android, Kindle, iPhone, etc.
- [Text Data File With Scores](#) — Syntax, definitions, and examples for synchronized ID data in a text file, including optional scores that quantify the relationships between connected devices. A higher score suggests a closer connection between devices than a lower score. These scores help make accurate, cross-device targeting possible.
- [Content Separators and Delimiters for Text Data Files](#) — Describes the available data separators, delimiters, and how to choose the right combination for your data file.

Basic Text Data File

Syntax, definitions, and examples for synchronized ID data in a tab-separated (.tsv) text file.

Syntax and contents

Tapad can send you synchronized ID data in a tab-delimited flat file which uses this syntax (*italic* type indicates variable data):

```
PRIMARY_ID = id <tab> RELATED_IDS = id <tab> RELATED_IDS = id
```

This basic text file contains:

- **Primary IDs:** Device IDs you've sent to Tapad.
- **Related IDs:** Hardware or cookie IDs related to the primary ID.

Sample data

This data has been truncated for brevity, but contents of your text file could look similar to the example below. In each text file:

- New records start with a cookie or hardware identifier on each line.
- IDs can consist of encoded and unencoded strings.
- Long entries can span multiple lines.

```
BDC=%2B%2B%2BZ      BDC=PBhABtkpOx      BDC=6i990McenE      HARDWARE_IDFA=5e85e971-2m4
HARDWARE_OPENUDID=a96de15e1a
HARDWARE_IDFA=1abc4e57-7c5c      HARDWARE_SHA1IDFA=c8b25fd214
HARDWARE_MD5IDFA=d55bbffcf8      HARDWARE_SHA1ANDROIDID=a8efb32074
b0640413
BDC=zzzPnRRe99      HARDWARE_OPENUDID=3941f35910
BDC=6i99YiYFMY      BDC=%2BeNuzS6i      HARDWARE_MD5IDFA=1jq-r8hm9r5
```

Definitions

Parameters in the text file are defined as follows:

Parameter	Description
<i>PRIMARY_ID</i>	The first entry in each row contains the primary IDs you send to Tapad. Subsequent entries in the same row contain synchronized data. A single entry can span multiple rows. The partner ID type can be either a hardware or cookie identifier as shown below: <ul style="list-style-type: none">• HARDWARE_IDFA=1234: Indicates data is synchronized with partner's device type and ID.• BDC=1234: Indicates data is synchronized with a partner cookie. A cookie gets a 2-3 character abbreviation based on the company name. For example, if your is named the Big Data Company, your cookie ID would be BDC. See Device ID Types for more information. <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">Note, hardware IDs are always lower case.</div>
<i>RELATED_ID</i>	Tapad device or cookie ID synchronized to the primary ID. Uses the same format as the primary ID above. See Device ID Types for more information.

Text Data File With Platform Types

Syntax, definitions, and examples for synchronized ID data in a tab-separated (.tsv) text file, including optional platform types. The platform type is a human-readable term that describes the device itself and can include results like iPad, Android, Kindle, iPhone, etc.

Syntax and contents

Tapad can send you synchronized ID data in a tab-delimited format which uses this syntax (*italic* type indicates variable data):

```
PRIMARY_ID = id = platform_type <tab> RELATED_ID = id = platform_type <tab> RELATED_ID = id = platform_type
```

A text file with platform types contains:

- **Primary IDs:** Device IDs you've sent to Tapad.
- **Related IDs:** Hardware or cookie IDs related to the primary ID.
- **Platform types:** The type of device associated with the returned IDs.

Sample data

This data has been truncated for brevity, but contents of your text file could look similar to the example below. In each text file:

- New records start with a partner cookie or hardware identifier on each line.
- IDs can consist of encoded and unencoded strings.
- Each record contains a platform type.
- Long entries can span multiple lines.

```
BDC=%2B%2B%2BZ=computer    BDC=PbhABtkpOx=computer    BDC=6i990McenE=computer
HARDWARE_IDFA=5e85e971-2m4=iPad    HARDWARE_OPENUDID=a96de15e1a=iPod
HARDWARE_IDFA=1abc4e57-7c5c=iPad    HARDWARE_SHA1IDFA=c8b25fd214=Android
HARDWARE_MD5IDFA=d55bbffcf8=iPhone    HARDWARE_SHA1ANDROIDID=a8efb32074
b0640413=Android tablet
BDC=zzzPnRRe99=computer    HARDWARE_OPENUDID=3941f35910=iPad
BDC=6i99YiYFMY=computer    BDC=%2BeNuzS6i=computer
HARDWARE_MD5IDFA=ljq-r8hm9R5=iPhone
```

True story

Sometimes, you may want text file data that contains platform types *and* strength scores. In this case, the extra parameter is simply appended to the related ID data as shown below:

```
BDC=6i99YiYFMY    BDC=%2BeNuzS6i=cookie=1    HARDWARE_MD5IDFA=ljq-r8hm9R5=Android_phone=3
```

Definitions

Parameters in the text file are defined as follows:

Parameter	Description
PRIMARY_ID	The first entry in each row contains the primary IDs you send to Tapad. Subsequent entries in the same row contain synchronized data. A single entry can span multiple rows. This ID parameter can be either a hardware or cookie identifier as shown below: <ul style="list-style-type: none">• HARDWARE_IDFA=1234: Indicates data is synchronized with partner device type and ID.• BDC=1234: Indicates data is synchronized with a partner cookie. A cookie gets a 2-3 character abbreviation based on the company name. For example, if your company name is Big Data Company, your cookie ID would be BDC. See Device ID Types for more information. <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">Note, hardware IDs are always lower case.</div>

<i>RELATED_ID</i>	Tapad device or cookie ID synchronized to the primary ID. Uses the same format as the partner parameter above. See Device ID Types for more information.
<i>platform</i>	A human-readable term that describes the device. Returns UNKNOWN when we cannot determine the platform type or if that information is missing. See Platform Types List for more information.

Text Data File With Scores

Syntax, definitions, and examples for synchronized ID data in a text file, including optional scores that quantify the relationships between connected devices. A higher score suggests a closer connection between devices than a lower score. These scores help make accurate, cross-device targeting possible.

Syntax and contents

Tapad can send you synchronized ID data in a tab-delimited format which uses this syntax (*italic* type indicates variable data):

```
PRIMARY_ID = id    <tab> RELATED_ID = id = strength <tab> RELATED_ID = id = score
```

Parameters in the text file are defined as follows:

Parameter	Description
<i>PRIMARY_ID</i>	The first entry in each row contains the primary IDs you send to Tapad. Subsequent entries in the same row contain synchronized data. This ID parameter can be either a hardware or cookie identifier as shown below: <ul style="list-style-type: none">• HARDWARE_IDFA=1234: Indicates data is synchronized with partner device type and ID.• BDC=1234: Indicates data is synchronized with a partner cookie. A cookie gets a 2-3 character abbreviation based on the company name. For example, if your company name is Big Data Company, your cookie ID would be BDC. See Device ID Types for more information. <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">Note, hardware IDs are always lower case.</div>
<i>RELATED_ID</i>	Tapad data synchronized to the primary ID. Uses the same format as the primary ID above. See Device ID Types for more information.
<i>score</i>	A numeric value between 1-10 that defines the relationship between a primary ID and related ID. Note, only the related IDs get this score. For example, in this data file entry: <code>BDC=1234 HARDWARE_IDFA=1234=4 HARDWARE_OPENUDID=5678=2</code> The related devices are connected to the partner cookie BDC=1234 with a score of 4 and 2, respectively.

Sample data

This data has been truncated for brevity, but contents of your text file could look similar to the example below. In each text file:

- New records start with a partner cookie or hardware identifier on each line.
- IDs can consist of encoded and unencoded strings.
- Each related ID record contains a score.
- Long entries can span multiple lines.

```
BDC=%2B%2B%2BZ      BDC=PBhABtkpOx=5      BDC=6i990McenE=5      HARDWARE_IDFA=5e85e971-2m4=3
HARDWARE_OPENUDID=a96de15e1a=2
HARDWARE_IDFA=1abc4e57-7c5c      HARDWARE_SHA1IDFA=c8b25fd214=1
HARDWARE_MD5IDFA=d55bbffcf8=4      HARDWARE_SHA1ANDROIDID=a8efb32074
b0640413=3
BDC=zzzPnRRe99      HARDWARE_OPENUDID=3941f35910=5
BDC=6i99YiYFMY      BDC=%2BeNuzS6i=cookie=1
HARDWARE_MD5IDFA=1jq-r8hm9R5=Android_phone=3
```

Sometimes you may want data in a text file that contains scores *and* platform types. In this case, the extra parameter is simply appended to the related ID data as shown below:

BDC=6i99YiYFMY BDC=%2BeNuzS6i=cookie=1
HARDWARE_MD5IDFA=1jq-r8hm9R5=Android_phone=3

Content Separators and Delimiters for Text Data Files

Describes the available data separators, delimiters, and how to choose the right combination for your data file.

Key-value separators and delimiters

These options are available to help your systems parse and ingest device ID data.

Heads up!

Data and device ID examples have been truncated for brevity. Your data file will contain the full ID string and can include multiple types and IDs for each record.

Key-value separators

The separator is a special character that joins a key to its related value. For example, in the string `color:blue` the key (`color`) is joined to its value (`blue`) by a colon. The colon, and elements on either side of it, form a complete key-value pair. The following table lists the special characters you can choose to define key-value pairs in your data file.

Separator	Example
:	HARDWARE_MID5IDFA:12345
=	HARDWARE_MID5IDFA=12345
	HARDWARE_MID5IDFA 12345

Key-value delimiters

The delimiter is a special character that separates sets of key-value pairs. For example, say you have a set of key-value pairs such as "color=blue product=new price=100." Without a delimiter, it's difficult to determine where one key-value pair stops and another begins. A delimiter, such as a comma for instance, defines each key-value pair like this: `color=blue,product=new,price=100`. The following table lists the special characters you can choose to separate key-value pair sets.

Delimiter	Example
,	HARDWARE_MID5ANDROIDID:1234,HARDWARE_IDFA:5678,TAPAD_COOKIE:9876
	HARDWARE_MID5ANDROIDID:1234 HARDWARE_IDFA:5678 HTAPAD_COOKIE:9876
;	HARDWARE_MID5ANDROIDID:1234;HARDWARE_IDFA:5678;TAPAD_COOKIE:9876
space (a single space between key-value pairs)	HARDWARE_MID5ANDROIDID:1234 HARDWARE_IDFA:5678 TAPAD_COOKIE:9876
tab (a tabbed space between key-value pairs)	HARDWARE_MID5ANDROIDID:1234 HARDWARE_IDFA:5678 TAPAD_COOKIE:9876

Choose different separators and delimiters

The separator must be different from the delimiter in your data file. For example, if you decide to identify a key-value pair with a pipe, you cannot use that character to separate sets of key-value pairs in the activation file. See the table below for examples.

Correct	Incorrect
---------	-----------

HARDWARE_MD5ANDROIDID:1234,HARDWARE_IDFA:5678
,TAPAD_COOKIE:9876

HARDWARE_MD5ANDROIDID=1234;HARDWARE_IDFA=5678;
TAPAD_COOKIE=9876

HARDWARE_MD5ANDROIDID:1234:HARDWARE_IDFA:5678:TAPAD_COOKIE:

HARDWARE_MD5ANDROIDID|1234|HARDWARE_IDFA|5678|TAPAD_COOKIE|

Data File Delivery Methods and Filter Options

Basic and enhanced delivery options.

- **Standard Delivery Methods** — Information about available transfer methods, naming conventions, and delivery schedule.
- **Enhanced Delivery Methods** — Data Activation File transfer methods, file type, and frequency.
- **Filter Options** — Filters let you choose how you would like Tapad to return synchronized ID data. Filter options include return all data (default), filter for "seen devices" (IDs you already know about), filter by device IDs, filter by platforms, and filter by Confidence Scores. Also, you can combine filters to limit the amount of data returned or to create specific data sets.

Standard Delivery Methods

Information about available transfer methods, naming conventions, and delivery schedule.

Standard delivery methods include:

- **Protocol:** SFTP
- **Delivery method:** Push to partner (partner provides login credentials to Tapad)
- **Update type:** Full update only
- **Delivery schedule:** Weekly only

Tapad uses the .tmp file extension and a trigger file to indicate data transfer status.

- **Transfer in progress:** Partnername_ids_full_yyyyymmdd_HHMMSS.gz.tmp
- **File complete:** Partnername_ids_full_yyyyymmdd_HHMMSS.gz.trigger

The trigger file contains just 2 lines: The file name and file size in bytes. The contents of your completed trigger file would look similar to the following example:

```
FILE=json_partnername_ids_full_20150311_160732.gz
SIZE=8270525492
```

Enhanced Delivery Methods

Data Activation File transfer methods, file type, and frequency.

Tapad uses SFTP to transfer the data file. You can get your data by either of the following methods:

- **Push:** Tapad sends the file to your designated FTP dropbox.
- **Pull:** You download the file from Tapad.

Tapad uses the .tmp file extension and a trigger file to indicate data transfer status.

- **Transfer in progress:** Partnername_ids_full_yyyyymmdd_HHMMSS.gz.tmp
- **File complete:** Partnername_ids_full_yyyyymmdd_HHMMSS.gz.trigger

The trigger file contains just 2 lines: The file name and file size in bytes. The contents of your completed trigger file would look similar to the following example:

```
FILE=json_partnername_ids_full_20150311_160732.gz
SIZE=8270525492
```

Delivery frequency:

- Daily
- Weekly
- Monthly

Filter Options

Filters let you choose how you would like Tapad to return synchronized ID data. Filter options include return all data (default), filter for "seen devices" (IDs you already know about), filter by device IDs, filter by platforms, and filter by Confidence Scores. Also, you can combine filters to limit the amount of data returned or to create specific data sets.

Filter by all devices

By default, Tapad returns all synchronized data for every device mapped to the IDs you send us. This includes data for devices and IDs that your systems do not know about. These "unknown" devices get included because they're connected to your known IDs in the Device Graph. The following diagram illustrates how the "all devices" filter works.

Filter by seen devices

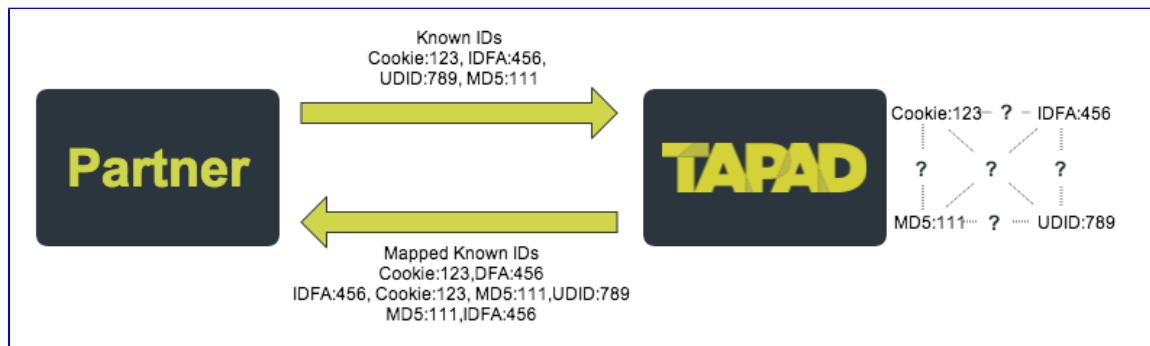
Seen devices include IDs:

- Sent *to* Tapad *from* a data partner.
- Sent *from* Tapad *to* a data partner.
- Associated with a particular audience.
- Passed in on API or other data synchronization calls.

Sometimes you may not want to receive data for every device mapped to an ID you send us, particularly when the device is unknown to you. Tapad can exclude unknown devices from the data file and send you results for your known devices only. For example, this filter may be useful when you want to know:

- How Tapad has mapped your IDs to known IDs from other data partners you work with.
- How Your devices are connected to each other in the Tapad Device Graph.

The following diagram illustrates how the "seen devices" filter works.



Filter by Device ID Type

Filter by device ID type to return data for the specified device type only. For example, if you filter by IDFA, Tapad returns data for IDFA devices only. For a complete ID list, see [Device ID Types](#).

Filter by Platform

Filtering by platform returns data for the specified platform type only. For example, if you filter by iPad, Tapad returns data iPad devices only. Available platforms include:

- Android
- Android tablet
- Blackberry
- Computer
- Feature phone (Includes uncommon or specialized mobile devices, software, or communication protocols.)
- iPad
- iPhone
- iPod
- Kindle
- Kindle Fire
- Palm
- Playstation
- Symbian
- Television
- Wii

- Windows Phone
- Windows Tablet
- Xbox

Filter by Score

The optional scores quantify the relationships between connected devices in the Device Graph. Higher scores suggest that devices are more likely to be related to each other than devices with lower scores. Scores range from 1-10. Filter by score when you want to return data for devices with relationships = > the selected score.

Data Flow Appendix

Supplemental reference material related to Tapad Data Flow processes.

Data Flow PDFs

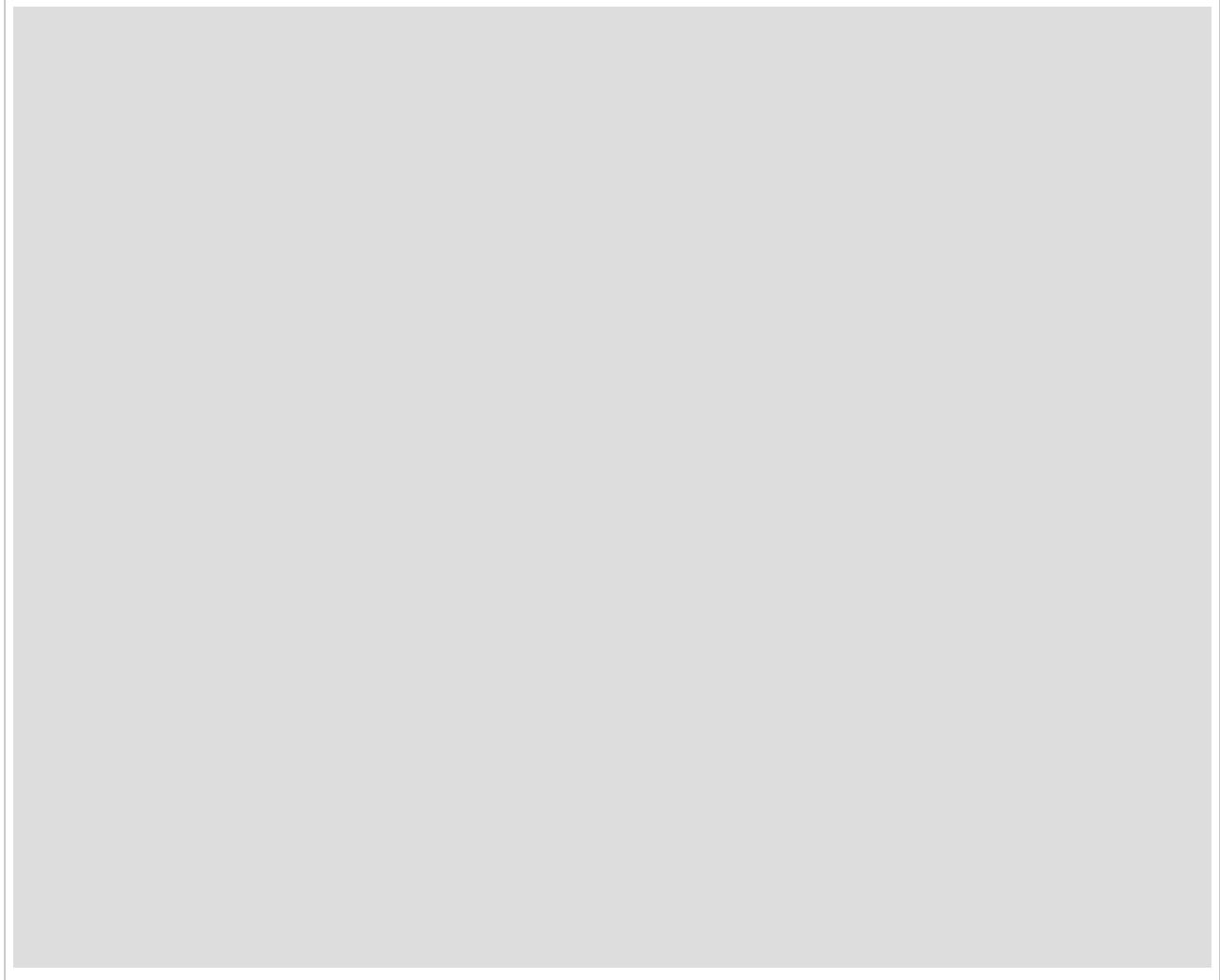
PDF formatted content that describes options for sending synchronized IDs to Tapad partners. *Updated March, 2015.*

- Basic content and delivery options
- Basic content, delivery, and data filter options
- JSON and text file data with platforms
- JSON and text file data with scores
- The Full Monty
- Filter options

To download, click in the PDF frame.

Basic content and delivery options

Includes basic JSON and text data, content separators/delimiters, and standard delivery methods.



Basic content, delivery, and data filter options

Includes basic JSON and text data, content separators/delimiters, standard delivery methods, and ID filtering options.



JSON and text file data with platforms

Includes JSON and text data with platform details, content separators/delimiters, and standard delivery methods.



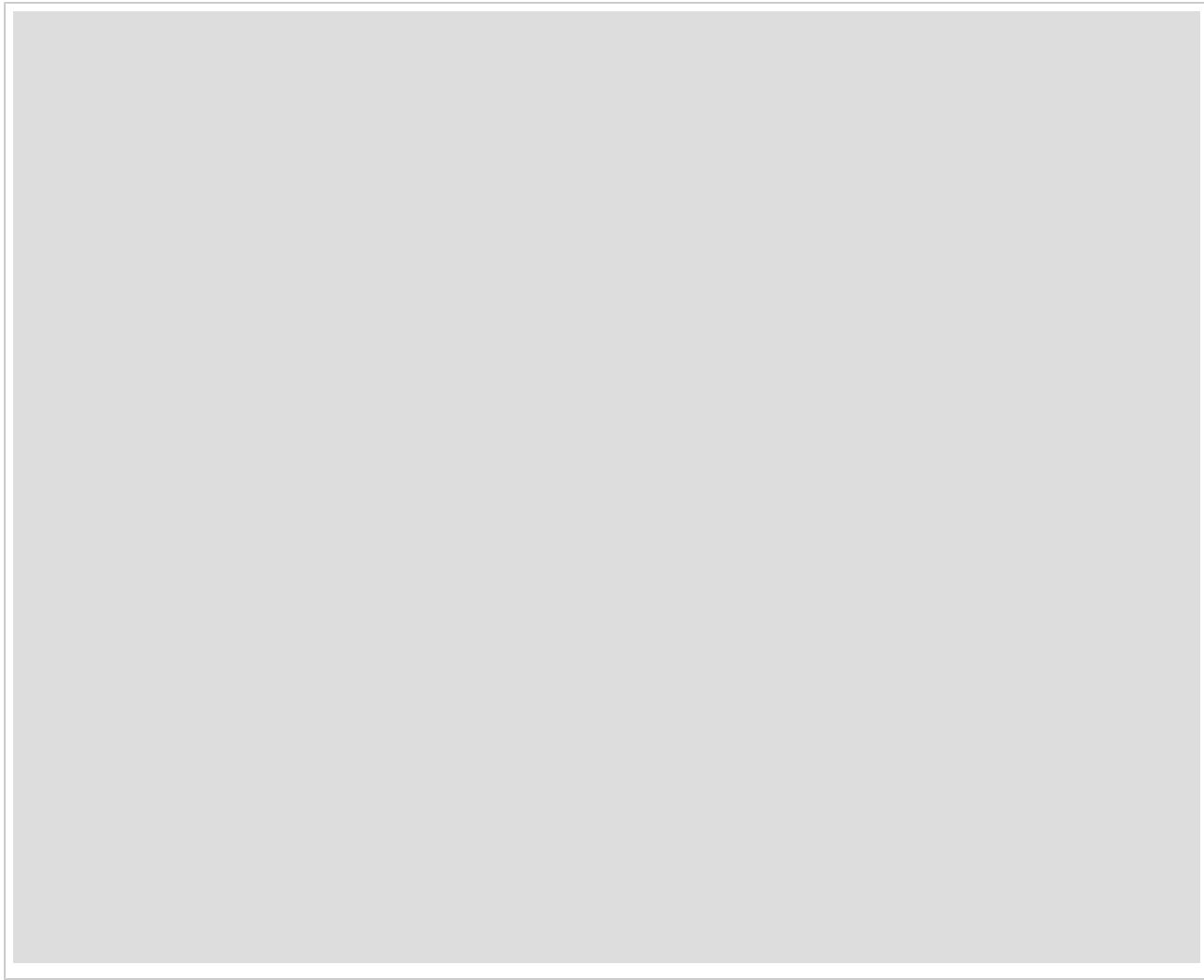
JSON and text file data with scores

Includes JSON and text data with scores, content separators/delimiters, and standard delivery methods.



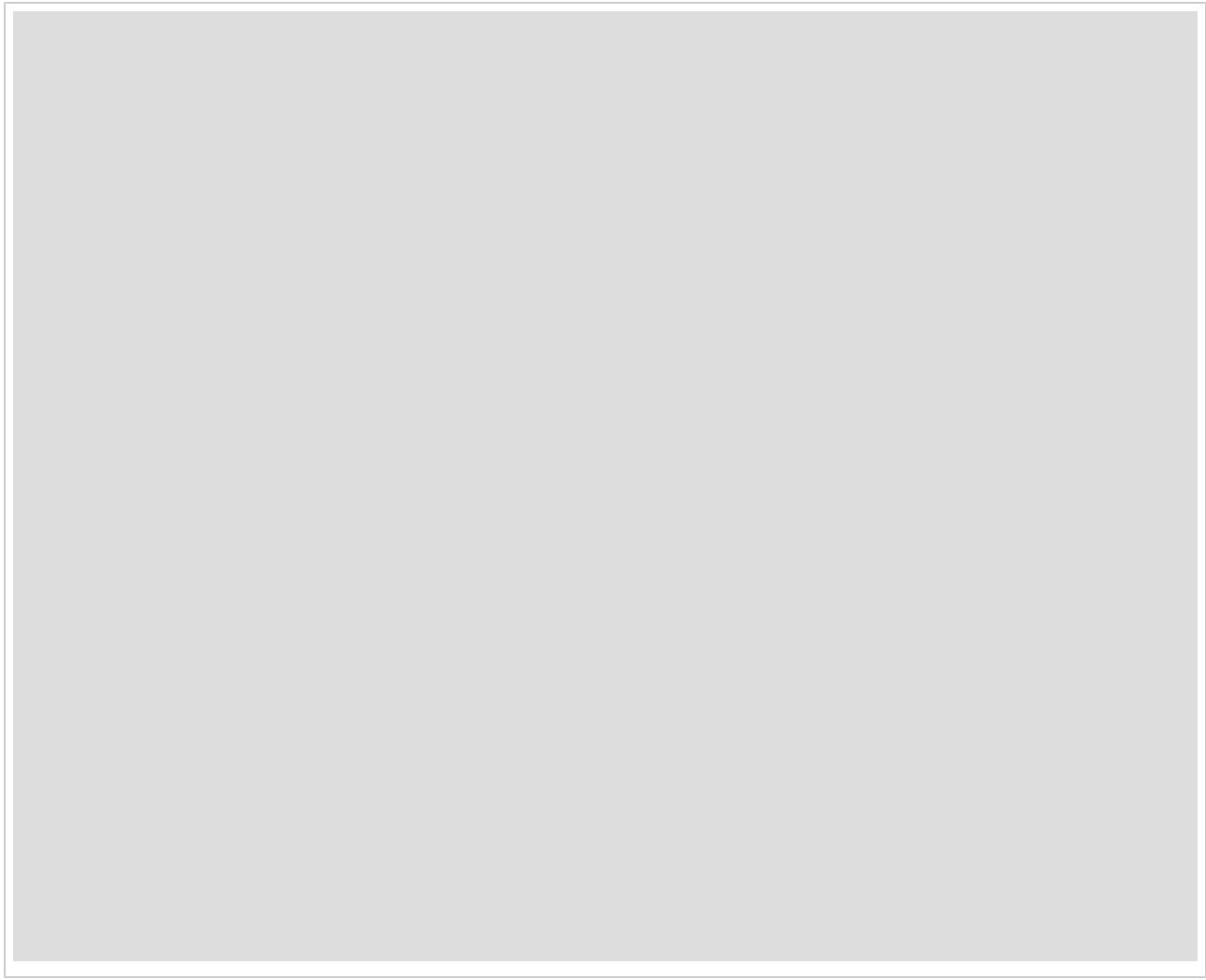
The Full Monty

All options for JSON and text data files.



Filter options

Limit synchronized ID data with these filters.



Mapping Primary and Related IDs

As you go through the records in a data file, it's normal to see primary IDs appear as a related ID and for some related IDs to appear as a primary ID. This happens with a related ID that belong to devices that have been synchronized previously with Tapad. This behavior is the result of how the Device Graph builds a map of related devices and checks for connections between them. These results are useful because they show you relationships between your devices and other devices stored in the Device Graph.

Data file contents: A review

Part of the Tapad data flow process returns mapped device IDs to our partners. Mapped data is organized into sets that contain a primary ID and related IDs. A primary ID is the device ID you send to Tapad and related IDs are devices that share a connection with the primary ID.

How it all works

To help you understand this process lets use a simple ID file, build a graph with this data, and step through the matching process used to generate an ID mapping file. This example uses simplified device IDs for each record. A real data file is far more complex. This synchronization and matching process works like this:

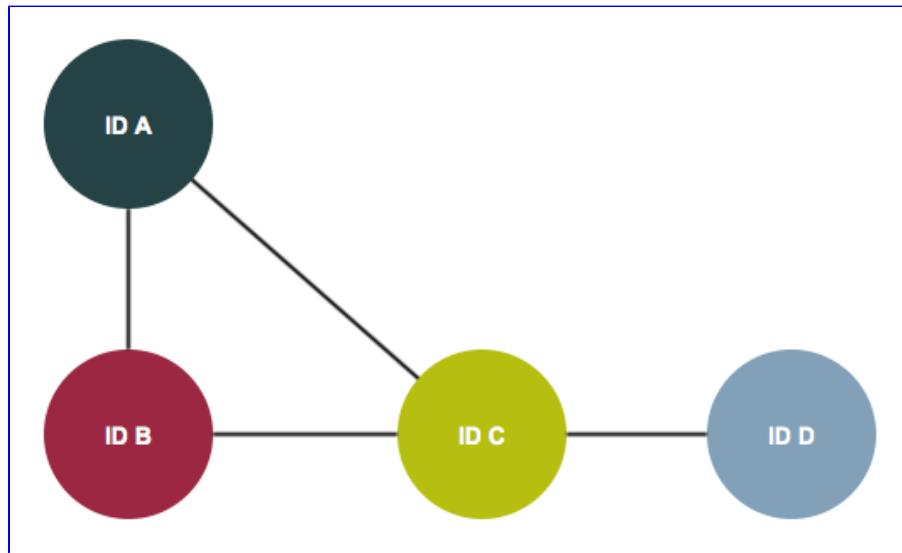
Send data

In this example, let's say you send the following device IDs to Tapad.

Simple ID file
ID A, ID B, ID C, ID D

Connect devices

After receiving this data, the Device Graph determines these devices are connected as shown below.



Synchronize IDs and map connections

Tapad creates an ID mapping file once we know these devices are connected in the Device Graph. As shown below, the mapping process checks each device ID and its immediate neighbor only. This means:

- Each synchronized ID gets evaluated as a primary ID.
- Primary IDs become related IDs and vice versa as the matching process steps through each ID in the graph.
- Related IDs are devices connected at depth 1 only (see [Understanding Depth](#) for more information).
- Depending on graph structure, and because we check devices at depth 1 only, the [transitive property for equality](#) does not always apply

to each device combination. For example, in our sample graph (above) and data file results (below), A = C, C = D, but D > A because A is at depth 2.

ID Synchronization (Depth 1)		
Data file	Primary ID	Related IDs
primary_id: A related_ids: B, C	ID A	ID B ID C
primary_id: B related_ids: A, C	ID B	ID A ID C
primary_id: C related_ids: A, B, D	ID C	ID A ID B ID D
primary_id: D related_ids: C	ID D	ID C

The end result is a data mapping file that lists all your connected devices in the device graph. Given the available data, a sample data file would look like this:

```
primary_id: A  
related_ids: B, C
```

```
primary_id: B  
related_ids: A, C
```

```
primary_id: C  
related_ids: A, B, D
```

```
primary_id: D  
related_ids: C
```

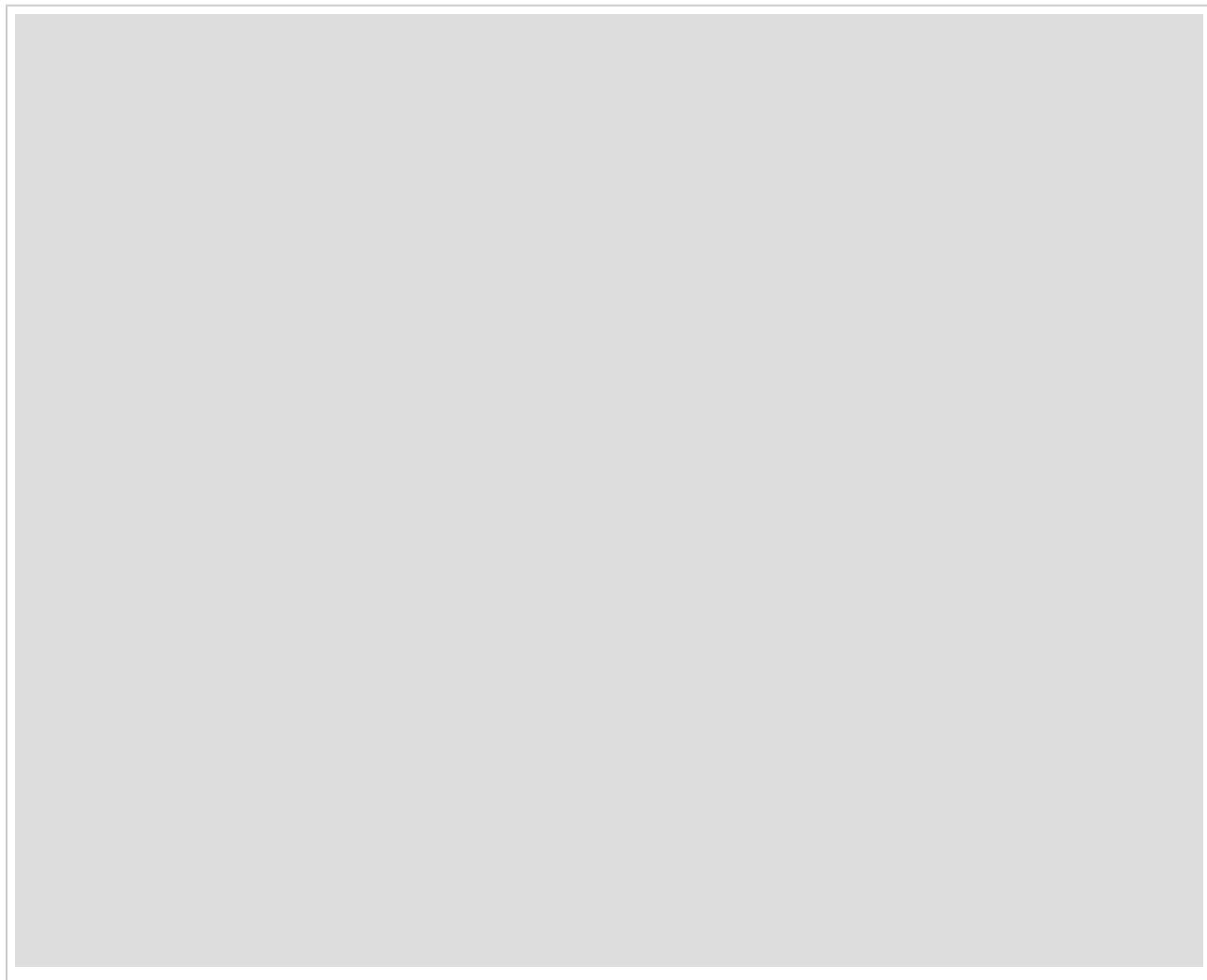
Related concepts

[Merge Logic](#)

Mapping Primary and Related IDs PDF

PDF version of this material.

Download or view the PDF.



Understanding the ID Mapping File

The ID mapping file contains device ID data for separate Tapad clients. The information in this file is generated from the unique IDs clients send to Tapad for synchronization with other IDs in the Device Graph. Data in this file can help you show prospective clients the size of the connected audiences they could reach when working with Tapad or help existing clients interpret the data they receive in the weekly device mapping file.

- [ID Mapping File Metrics Defined](#) — Lists and defines the metrics included in the ID mapping file for v.1.0
- [Sample ID Mapping File](#) — An ID mapping file contains data that looks similar to the example in this section. Note, this report has been truncated for brevity.

ID Mapping File Metrics Defined

Lists and defines the metrics included in the ID mapping file for v.1.0

ID mapping file includes metrics described in the following table. For a report example, see the [Sample ID Mapping File](#).

Metric	Example	Description
Number of hardware IDs lowercased	68177	The number of IDs that had at least 1 upper case character. The reporting process transforms upper case characters to lower case and then uses the ID.
Number of malformed hardware IDs	652099	The number of hardware IDs that cannot be identified by our regular expressions for device IDs. These IDs are removed from the report.
Primary IDs removed	<pre>Primary ids removed: { "hardwareIds":0, "cookieIds":151915, "supplierIds":0, "total":151915 }</pre>	Sometimes a data file contains multiple rows with the same primary ID. In this case, we remove the duplicated primary IDs and merge the data into one row. This metric counts how many duplicated primary IDs are removed, by type, during the merging process.
Related IDs removed	<pre>Related ids removed: { "hardwareIds":6853, "cookieIds":94151, "supplierIds":0, "total":101004 }</pre>	Sometimes a data file contains multiple rows with the same related ID. In this case, we remove the duplicated related IDs and merge the data into one row. This metric counts how many duplicated IDs are removed, by type, during the merging process.
Duplicate ID values	<pre>Duplicate id values: { "HARDWARE_ANDROID_ID":6608, "HARDWARE_IDFA":6608 }</pre>	Sometimes a single record in a data file has the same numeric ID but a different device type. This metric counts the number of duplicated ID values.
Clustering percentiles	<pre>"clusteringPercentiles":{ "100.0":113002, "99.0":19, "99.9":48, ... }</pre>	<p>The number of IDs in the largest cluster for that percentile. For example, say you have results like this:</p> <pre>"99.0":19</pre> <p>For results at the 99th percentile, the largest cluster contains 19 IDs. Results at the 100th percentile return the size of the largest cluster (in IDs) for that client.</p> <p>Data is filtered at the 99.99 percentile by default.</p>

Sample ID Mapping File

An ID mapping file contains data that looks similar to the example in this section. Note, this report has been truncated for brevity.

For a list of report metrics and their definitions, see [ID Mapping File Metrics Defined](#).

Sample data

```
Number of hardware ids lowercased: 68177,  
  
Number of malformed hardware ids: 5652099,  
  
Clustering percentiles: {  
    "99.0":20,  
    "99.9":45,  
    "99.99":90,  
    "99.999":149,  
    "99.9999":220,  
    "100.0":22027141  
},  
  
Clustering number of ids filtered: 22204520,  
  
Primary ids removed: {  
    "hardwareIds":0,  
    "cookieIds":151915,  
    "supplierIds":0,  
    "total":151915  
},  
  
Related ids removed: {  
    "hardwareIds":1006853,  
    "cookieIds":1794151,  
    "supplierIds":0,  
    "total":2801004  
},  
  
Duplicate id values: {  
    "HARDWARE_ANDROID_AD_ID":66088,  
    "HARDWARE_IDFA":66088  
}
```

The Tapad User Opt-out Process: About

An overview of how Tapad opt-out processes work and how we send consumer opt-out data to our partners. See the Tapad website for more information about our [privacy policy](#).

- [Tapad Opt-out Overview](#) — Tapad's opt-out policies and procedures support industry-standard consumer privacy initiatives and we expect our partners to adhere to these same high standards. As a Tapad partner, you need to be aware that:
- [Batch File Opt-out Process: About](#) — The batch file option uses FTP to deliver opt-out data to your dropbox on a nightly schedule. Batch file opt-out sends data in a tab-separated flat file that contains a timestamp and a device ID for the user who elected to opt-out from data collection. Visit our website for detailed information about our privacy policy <http://www.tapad.com/privacy-policy/>.
- [Real-time Opt-out Process: About](#) — This method uses API methods to transfer opt-out information via direct server-to-server data calls. As a Tapad partner, you will need to communicate with the opt-out API to receive this data. Real-time transfers send data in a URL string to the partner's domain. This opt-out response URL contains the device ID. Visit our website for detailed information about our privacy policy <http://www.tapad.com/privacy-policy/>.
- [Opt-out ID types Defined](#) — Lists and defines the IDs used by Tapad's batch file opt-out process. This ID is returned by the device_id_type parameter in the opt-out file.
- [Opt-out PDFs](#) — Opt-out content in PDF format.

Tapad Opt-out Overview

Tapad's opt-out policies and procedures support industry-standard consumer privacy initiatives and we expect our partners to adhere to these same high standards. As a Tapad partner, you need to be aware that:

- Tapad process opt-outs in real-time.
- The [ID mapping file](#) is sent on a weekly basis.
- As a result, you may be working with slightly outdated information and still targeting users.

To meet our privacy standards, Tapad requires partners process opt-out data in real-time and in a daily batch file. This ensures that consumer opt-out requests are respected as soon as possible by Tapad and our partners. For more information about Tapad's privacy policy, visit our [privacy page](#).

Oh snap!

Before you can activate the synchronized IDs that Tapad sends to you through a data flow, you *must* be able to:

- Receive opt-out data in real-time and in batch files.
- Share that opt-out information with other partners that you are passing the cross-device data to.

Privacy opt-out for browsers

Consumers who want to opt-out of data collection for browsers that use cookies can visit our [privacy page](#) and click **Opt-out**. When a user opts-out, Tapad systems:

1. Replace the Tapad ID in the Tapad cookie with the value `OPTED_OUT`.
2. Send the opted-out ID to our partners via FTP or server-to-server data transfers.

Privacy opt-out for mobile apps

Consumers who want to opt-out of data collection for mobile applications can visit our [privacy page](#) and download the opt-out apps for [Android](#) or [iOS](#) systems. When a user opts-out, Tapad systems:

1. Add an opt-out flag for all device IDs belonging to that device.
2. Write opted-out device IDs to a database for archival and backup.
3. Run a nightly batch job on the opt-out file to consolidate and verify data.
4. Send the opted-out IDs to partners we share data with.

Receive opt-out data from Tapad

You can receive user opt-out data from Tapad via FTP or by real-time data transfers. For more information, see:

- [Batch File Opt-out Process: About](#)
- [Real-time Opt-out Process: About](#)

Batch File Opt-out Process: About

The batch file option uses FTP to deliver opt-out data to your dropbox on a nightly schedule. Batch file opt-out sends data in a tab-separated flat file that contains a timestamp and a device ID for the user who elected to opt-out from data collection. Visit our website for detailed information about our [privacy policy](#).

Prerequisites

This method requires login credentials to your FTP server.

Requirements

You must ingest this file daily.

Batch file contents

A sample opt-out batch file could look similar to this:

timestamp	device_id	device_id_type
2014-01-09 22:26:57	08BA5FEE	3
2014-01-09 22:26:30	123ff34ab453e	3
2014-01-09 22:10:12	D9BB4EE2-8D10	23
2014-01-09 23:18:02	4a083ff2	18
2014-01-09 21:25:25	5af62300	26

Parameters are defined as follows:

Parameter	Description
timestamp	A GMT timestamp that records when the user opted-out of data collection. Uses the yyyy-mm-dd hh:mm:ss format.
device_id	The user's device ID. IDs can include: <ul style="list-style-type: none">• Hardware ID• Tapad ID• Partner Device ID
device_id_type	An ID that identifies the device type. See Opt-out ID types Defined for more information.

Real-time Opt-out Process: About

This method uses API methods to transfer opt-out information via direct server-to-server data calls. As a Tapad partner, you will need to communicate with the opt-out API to receive this data. Real-time transfers send data in a URL string to the partner's domain. This opt-out response URL contains the device ID. Visit our website for detailed information about our [privacy policy](#).

Real-time opt-out call

Each call sends one ID and corresponding ID type. If there are multiple IDs for a given device (i.e. cookie ids, SHA1, MD5, and RAW IDFA) we will send you all versions in separate calls. A sample real-time call uses a POST method and the following syntax to send opt-out data to your domain.

```
POST http://tapad_partner_domain.com/?device_id=${device_id}&device_id_type=${device_id_type}
```

These parameters are defined as follows:

Parameter	Description
<code> \${device_id_type}</code>	A macro that expands to contain a numerical ID corresponding to a particular device ID type. See Opt-out ID types Defined for more information.
<code> \${device_id}</code>	A macro that expands to contain the ID for the user. IDs can include: <ul style="list-style-type: none">• Hardware ID• Your partner cookie ID• Tapad ID

Opt-out ID types Defined

Lists and defines the IDs used by Tapad's batch file opt-out process. This ID is returned by the device_id_type parameter in the opt-out file.

Batch file opt-out IDs can include the following:

ID	Description
1	Tapad cookie ID
3	Your cookie ID
18	Android ID
19	MD5 ANDROID ID
20	SHA1 Android ID
23	Raw IDFA
24	SHA1 IDFA
25	MD5 IDFA
26	Raw IDFV
29	Google Advertising ID

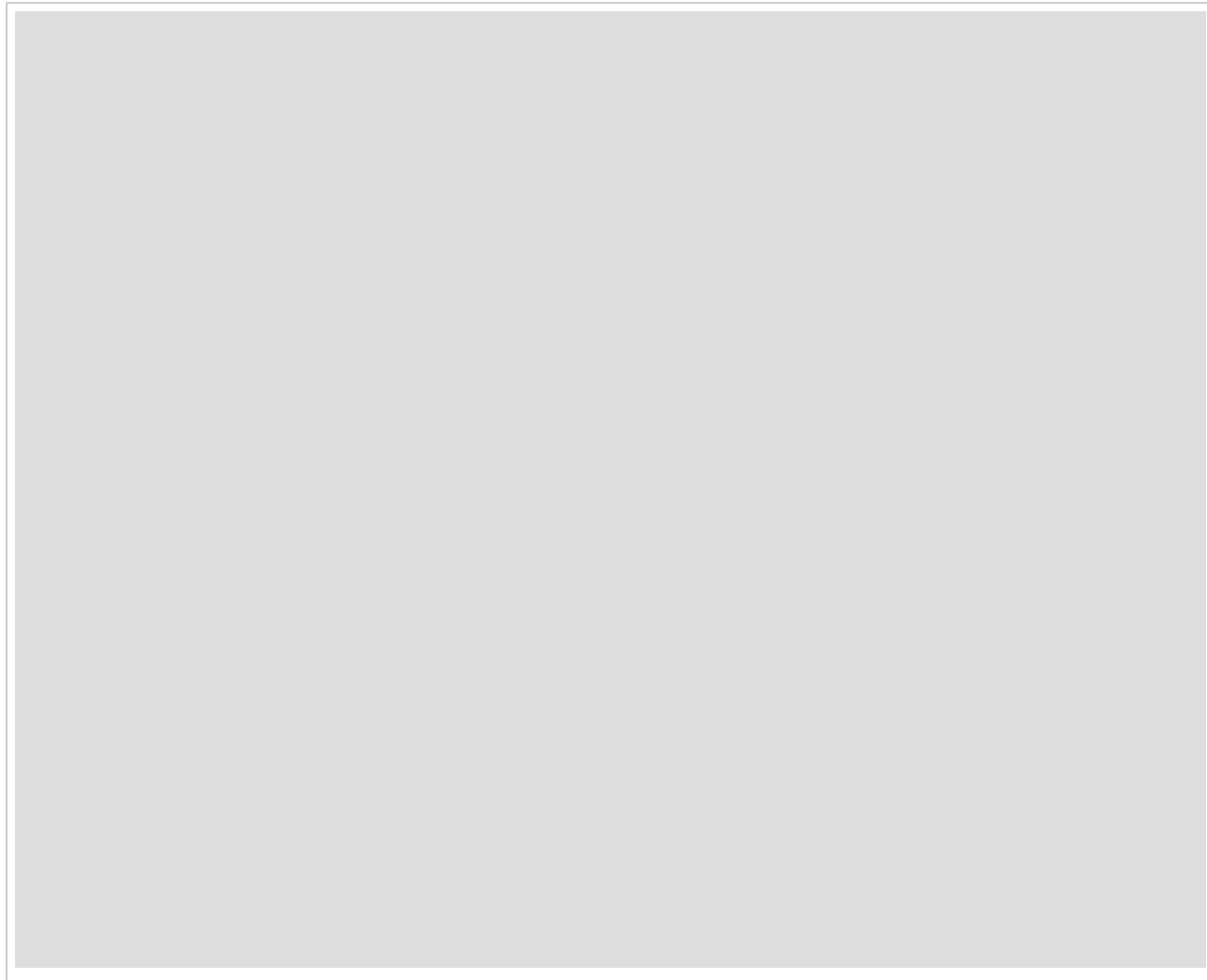
Opt-out PDFs

Opt-out content in PDF format.

Heads up: PDF link issues!!

This PDF content contains links to the [data flow documentation](#). You need to tell clients that clicking on some links will take them to Confluence for that information. If they need the data flow docs, use the previous link to go to that section and download the PDF.

To download, click in the PDF frame.



Technical Specifications for the Impression Log File

The Impression Log File provides data about impressions and clicks across bridged devices in the Tapad Device Graph. It includes all impressions and clicks served to your first or third-party audiences. Tapad can deliver the report to you in a compressed, tab-delimited file via FTP. Information in the log file can help you understand how your audience data is used and help you create more accurate client reports.

Sample file

The report contains tab-separated data saved as a `.tsv` file. Your Impression Log File could look similar to the following sample report. Note, IDs have been truncated for brevity.

timestamp	device_id	related_tapad_id	creative_id	tactic_id	action_id
2014-01-09 22:26:57	08BA5FEE	123ff34ab453e	75541	16224	impression
2014-01-09 22:26:30	08BA5FEE	ba2510de-4e8e	75541	17244	impression
2014-01-09 22:10:12	08BA5FEE	2013092717015	75543	16472	click
2014-01-09 23:18:02	4a083ff2	D9BB4EE2-8D10	37200	18535	impression
2014-01-09 21:25:25	5af62300	08BAEE5F-FE0A	37199	18539	click

Metrics

The Impression Log File contains impressions and click data for the following metrics.

Report metric	Description
timestamp	Records the time of an impression, click, or tap. Uses <code>yyyy-mm-dd hh:mm:ss</code> date/time format.
device_id	An alphanumeric identifier for a device. Can contain upper and lower case characters and/or hyphens. This is the ID of a device that recorded an action for an ad. An action can include activity like serving an impression or recording a user click. As a Tapad partner, devices listed in the <code>device_id</code> column could be known or unknown to you.
related_tapad_id	This is Tapad's alphanumeric ID or hardware ID (IDFA, Android, etc.) for a device. It's used when we do not have a <code>related_client_id</code> for a device in the report. These IDs can contain upper and lower case characters and/or hyphens.
creative_id	Creative ID
tactic_id	Tactic ID
action_id	Flags the record as an impression, click, or tap (taps are impressions + clicks).

Filters

Report filters let you control the type and amount of information returned in a report. Filtering is useful if you manage multiple campaigns and/or advertisers. Filter options for the Impression Log let you receive data for a specific advertiser or campaign only. Rather than receive *all* your data in a single file, request only that data you need for each advertiser or campaign.

Naming conventions, format, and delivery options

File names and format

A properly named Impression Log File uses this syntax:

`partnerName_fileContent_campaignId_yyyymmdd_HHMMSS.gz`

The following table describes the elements in the file name.

File element	Description

Naming conventions	You have some options regarding the file name. Generally, the file name contains the following elements separated by an underscore: <ul style="list-style-type: none"> • Your Tapad partner name. Note, <i>do not</i> separate multiple words in your partner name with an underscore. The underscore is a reserved character that separates other blocks of information in the file name. • File content identified as impressions, clicks, or taps (taps are impressions + clicks). • Tapad's Campaign ID. • A file creation date/timestamp.
File type and content format	The Impression Log File is a flat file that contains tab-separated data. It uses the <code>.tsv</code> file extension. It is delivered to you as a compressed <code>.gz</code> file.

Delivery options

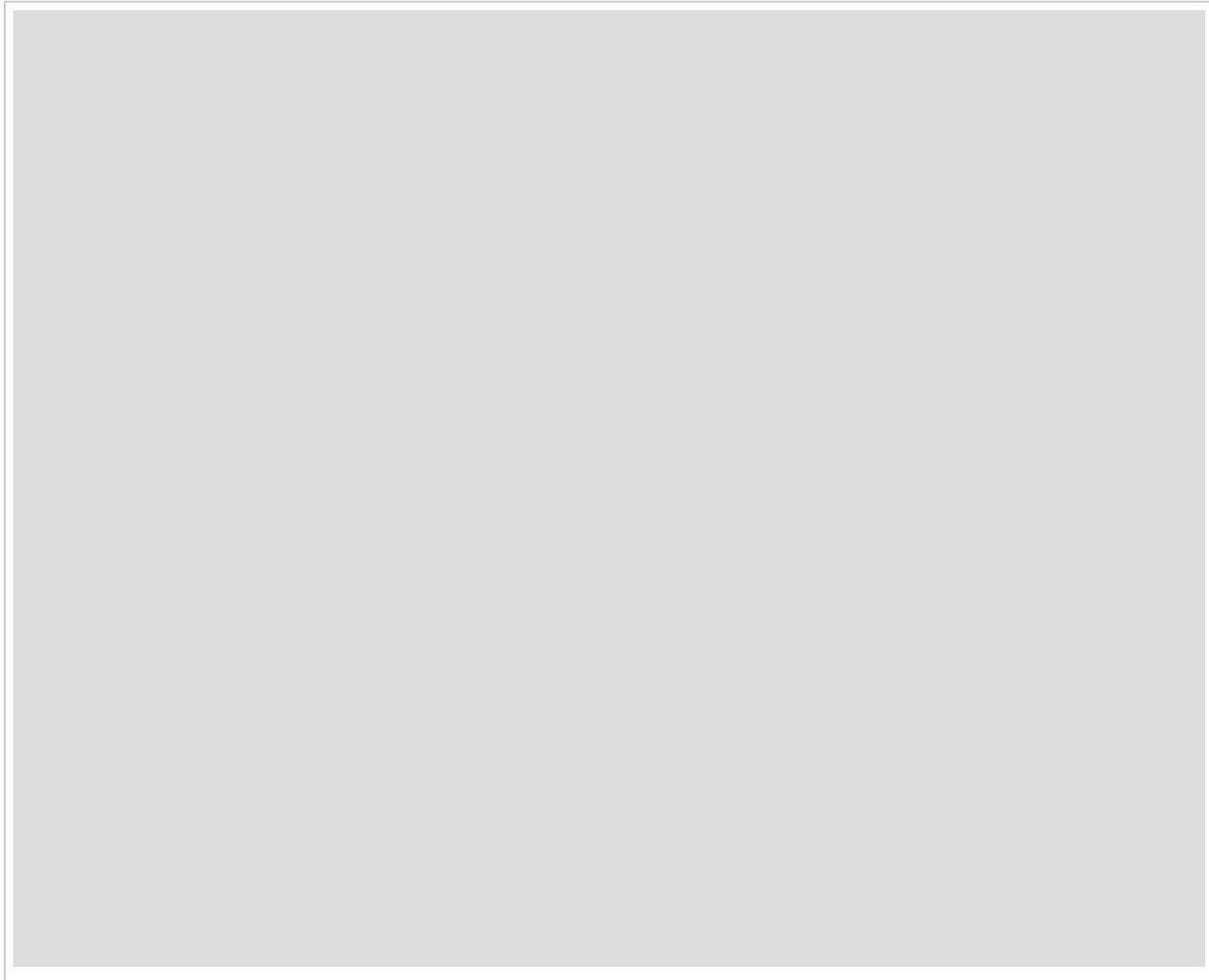
Delivery methods and options include the following.

- **File transfer method:** FTP or SFTP push to your dropbox.
- **Delivery frequency:** 24-hour delivery intervals.

Impression Log File PDF

Options and specifications in PDF format.

To download, click  in the PDF frame.



Audience and User ID Synchronization

Data integration partners can share user IDs, audiences, and related audience metadata (optional) with Tapad. Exchanging user and audience IDs with Tapad helps extend campaign reach because it lets you deliver content to individual users across all of their devices (desktop, laptop, mobile, tablet, etc.). Refer to this material for technical details about our user ID and audience data integration/synchronization processes. Work with your Tapad Implementation Consultant for more information and to get started.

- [Partner Initiated ID Synchronization Code: Syntax, Parameters, and Sample Request](#) — ID synchronization code and variables defined. Put this code in the <head> section of your client's website when you set up user ID synchronization with Tapad.
- [Tapad Initiated ID Synchronization Code: Syntax, Parameters, and Sample Request](#) — Tapad will send a user it sees to the URL specified.
- [User ID Synchronization Methods](#) — An overview of how a Tapad or partner-initiated user ID synchronization works.
- [Audience Metadata File](#) — Although it is not required, you can send us data about your audience segments. This information helps us understand your audience data.
- [Getting Started Questionnaire](#) — Review and answer these questions to help us set up your Tapad data integration.

Partner Initiated ID Synchronization Code: Syntax, Parameters, and Sample Request

ID synchronization code and variables defined. Put this code in the <head> section of your client's website when you set up user ID synchronization with Tapad.

Syntax

Our ID synchronization code can send data to secure (HTTPS) and un-secure (HTTP) websites alike. You place this code on the page as a standard 1x1 image pixel. This syntax example uses the HTTP version.

```
http://pixel.tapad.com/idsync/ex/receive?partner_id=<tapad_partner_id>&partner_device_id=<your_id_macro>&partner_url=<hostname><:port><redirect_uri_prefix>${TA_DEVICE_ID}<redirect_uri_postfix>
```

Use this syntax when you place the Tapad call on your site but do not return your IDs to us.

```
http://pixel.tapad.com/idsync/ex/push?partner_url=<hostname><:port><redirect_uri_prefix>${TA_DEVICE_ID}<redirect_uri_postfix>
```

Parameters

All variables are required unless indicated otherwise.

Parameter	Description
tapad_partner_id	Your Tapad partner ID.
partner_device_id	A macro that inserts your user ID into the synchronization call.

If you want to receive the Tapad ID to store the ID map, then provide the params as indicated in the table.

Parameter	Description
partner_url	Your domain URL.
:port	(Optional) Port number used by your domain URL to receive the request. For example: <code>http://pixel.tapad.com/.../receive?http://data_partner_website:80...</code>
redirect_uri_prefix	(Optional) A variable that lets you add additional parameters to the URL string.
\${TA_DEVICE_ID}	A Tapad macro that inserts the Tapad ID user ID into the synchronization call.
redirect_uri_postfix	(Optional) A variable that lets you customize or add additional parameters to the URL string.

Partner initiated without sending IDs

Sample request

The following examples show sample ID synchronization calls with and without an optional redirect.

With redirect

```
http://pixel.tapad.com/idsync/ex/receive?partner_id=1111&partner_device_id=2222&partner_url=http://endpoint.partnername.com/received?id=3333&partner=tapad
```

Without redirect

http://pixel.tapad.com/idsync/ex/receive?partner_id=1111&partner_device_id=2222

Partner initiated without returning IDs

[http://pixel.tapad.com/idsync/ex/push?partner_url=http://endpoint.partnername.com/received?id=3333&partner= tapad](http://pixel.tapad.com/idsync/ex/push?partner_url=http://endpoint.partnername.com/received?id=3333&partner=tapad)

Tapad Initiated ID Synchronization Code: Syntax, Parameters, and Sample Request

Tapad will send a user it sees to the URL specified.

Syntax

Our ID synchronization code can send data to secure (HTTPS) and un-secure (HTTP) endpoints. This syntax example uses the HTTP version.

```
http://<hostname><:port><redirect_uri_prefix>${TA_DEVICE_ID}<redirect_uri_postfix>
```

Parameters

All variables are required unless indicated otherwise.

Parameter	Description
hostname	Your domain URL.
:port	(Optional) Port number used by your domain URL to receive the request. For example: <code>http://data_partner_website:80...</code>
redirect_uri_prefix	(Optional) A variable that lets you add additional parameters to the URL string.
\${TA_DEVICE_ID}	A Tapad macro that inserts the Tapad ID.
redirect_uri_postfix	(Optional) A variable that lets you customize or add additional parameters to the URL string.

Sample request

A sample ID synchronization call could look like this:

```
http://endpoint.partnername.com:80/receiveid?id=1111&partner=TAPAD
```

Note that Tapad may request that you also redirect your partner ID back to Tapad to store the mapping as well.

User ID Synchronization Methods

An overview of how a Tapad or partner-initiated user ID synchronization works.

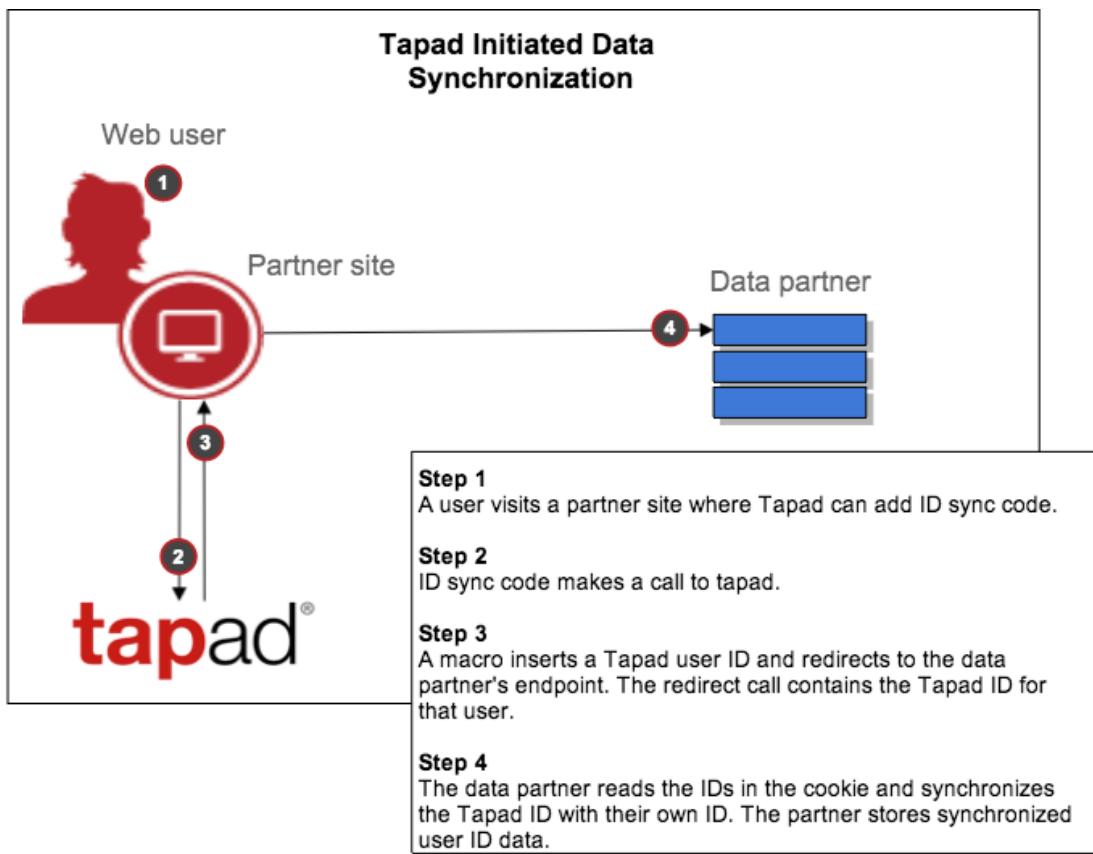
Audience synchronization begins by matching user IDs between Tapad and a data integration partner. During the user ID synchronization process either Tapad or our data partner can match, map, store, and redirect user IDs. However, for maximum coverage, both sides (Tapad and the data partner) should set up this code to synchronize user IDs. The following sections describe these processes.

Tapad initiated user ID synchronization

During a Tapad initiated user ID sync, we collect and sends user IDs to you. In this case, you can:

- Map Tapad user IDs to your own user IDs.
- Synchronize mapped IDs to your own audience data.
- Return synchronized IDs and audience data to Tapad through a real-time or server-to-server data transfer. This lets Tapad target your users across all their known devices via the Device Graph.

The following diagram illustrates the Tapad to data partner user ID mapping process. For ID code syntax, parameters, and sample calls see [Tapad Initiated ID Synchronization Code](#).

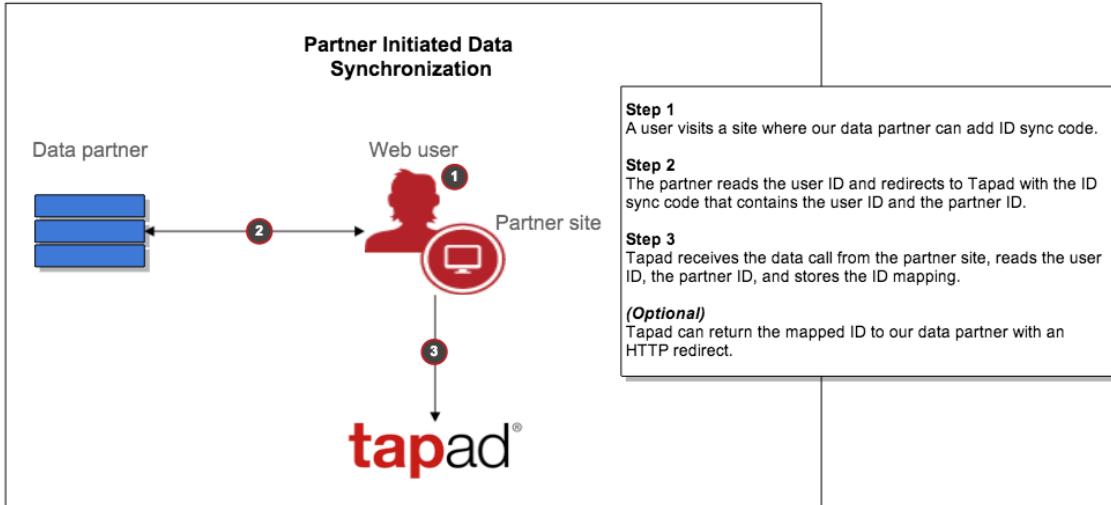


Partner initiated data synchronization

During a partner initiated user ID sync, you collect and send user IDs to Tapad. In this case, Tapad:

- Maps your user IDs to our own.
- Synchronizes mapped IDs to your audience data passed in to Tapad through a real-time or server-to-server data transfer.
- Targets your users across all their known devices via the Device Graph.

The following diagram illustrates the data partner to Tapad user ID mapping process. For ID code syntax, parameters, and sample calls see [Partner Initiated ID Synchronization Code](#).



Audience Metadata File

Although it is not required, you can send us data about your audience segments. This information helps us understand your audience data.

File format

Format your metadata as a simple CSV or XLS file. Also, the metadata file structure may differ depending on how you work with audience data. Some metadata files are organized by category-ID combinations while others group information into key-value pairs. See the following examples.

Example 1: Categories and IDs

If you organize data around categories and IDs, your metadata file should include a category name, subcategory names (if used), and the audience ID. Your metadata file could look similar to this:

Main category	Subcategory	Audience Id
Demography	Age 18-25	123
Automobiles	Domestic-used	456
Education	University-undergraduate	789

Example 2: Key-value pairs

If you organize data with key-value pairs your metadata file should include the key, value, and audience name. Your metadata file could look similar to this:

Key	Value	Name
Gender	m	Gender-male
Gender	f	Gender-female
Age group	18-25	Age 18-25
Signed up	y	Has signed up
Signed up	n	Has not signed up

Case, space, and character sensitivity

Metadata files contain alphanumeric characters that are not case, space, or character sensitive.

Updates and revisions

Follow these guidelines when you need to update your metadata file with Tapad:

- **Update limits:** Restrict updates to once-per-month.
- **Update contents:** Send updated content in a flat file or spreadsheet.
- **Maintain consistency:** Try to avoid significant structural changes in your taxonomy. These can be difficult to integrate into an active data transfer process.

Send the Taxonomy to Tapad

Send the metadata file and updates to your Tapad account manager.

Getting Started Questionnaire

Review and answer these questions to help us set up your Tapad data integration.

Basic data file information

Before we get started with an integration, please provide information about your data file including:

- **Methods:** Is this a real-time or batch data transfer?
- **Frequency:** How often do you expect to send data (daily, weekly, etc.)?
- **Size:** How large is an average batch file in MB? If real-time, how many queries per second do you expect to send?
- **Format and contents:** How is data organized (syntax, parameter naming conventions, etc.)? Does a data file contain headers, metadata, or other parameters we should be aware of?
- **Device ID:** We need to know what gets passed in with the device ID (e.g., user agent, IP addresses, etc.).
- **Completion/verification:** What do you use to indicate that the file transfer process is complete? For example, some integration partners send an empty file with the name like `filename.done` to indicate the data transfer process is complete. Also, do you use a checksum value?

Technical specifications for data transfers

If available, send us your technical data integration/transfer documentation and specifications. This information helps our engineering teams configure our systems to exchange data.

Using Tapad as a DMP

Describes how you can synchronize page metadata with Tapad and return DMP or ad server IDs to the publisher or other platforms and systems.

- [Tapad as a DMP: Overview and Requirements](#) — Tapad's JavaScript API lets you use our service as a data management platform (DMP). This can be useful if your current DMP does not provide support for mobile devices or other technologies and platforms.
- [DMP Code Placement and Configuration](#) — Follow these instructions to set up our JavaScript data collection code.
- [Tapad DMP Code Sample](#) — Put this JavaScript code on the pages you want to use for meta tag data collection. This code sends meta tag data to Tapad and returns audience IDs in a JSON object to your page. You can make function calls against the returned data or send it to an ad server for content delivery.

Tapad as a DMP: Overview and Requirements

Tapad's JavaScript API lets you use our service as a data management platform (DMP). This can be useful if your current DMP does not provide support for mobile devices or other technologies and platforms.

How it works

TAJ, Tapad's JavaScript API, provides methods and function calls that let you send page metadata to Tapad. This lets you work with Tapad as a lightweight DMP. This process works as follows:

1. **Page load:** TAJ API code reads the metadata attributes on your page (e.g., `<meta name="category" scheme="SCHEME1" content="wild planet" />`) and sends that information to Tapad.
2. **Synchronization processes:** After receiving the data, Tapad:
 - Matches the variable in the meta tag `content` attribute to an audience ID in a mapping file that you've sent us previously (e.g., "wild planet" = `DMP_ID_1234`).
 - Associates the matching audience IDs to the requesting device.
3. **Callback:** After synchronization, Tapad returns audience IDs to your page in a JSON object. At this point, you can make your own function calls against that ID or (optionally) pass it on to your DMP or ad server.

Requirements

To use Tapad as a DMP you need:

- A Tapad partner ID.
- To send us a file that maps all your meta tag data to the audience IDs used by your DMP or ad server service. We cannot associate meta tag data to an audience ID without this mapping file.
- To place the TAJ API code on the pages you want to collect data from.

Next steps

After sending us the ID mapping file, you're ready to implement our JavaScript API code. See [DMP Code Placement and Configuration](#) for more information.

DMP Code Placement and Configuration

Follow these instructions to set up our JavaScript data collection code.

Placement

Put the API code in the <head><script> block on each page you want to use for data collection.

```
<head>
...
<script>
<!-- Tapad code here --&gt;
&lt;/script&gt;
...
&lt;/head&gt;</pre>
```

Configure authentication and data collection parameters

Edit the JavaScript to include variables used for authentication and meta tag data collection. *Italics* indicate variable code.

Authentication

Provide you Tapad partner ID in the JavaScript .push function as shown below:

```
_tapestry.push ({'initialize': 'your_partner_id'})
```

Identify meta tags for data collection

A standard meta tag contains attributes that define the tag's name, scheme, and content like this:

```
<meta name="name variable" scheme="scheme variable" content="content variable" />
```

In the [JavaScript code](#), the `GetAudience()` function handles these meta tag attributes and variables. To implement this function properly, edit the following code elements:

Code element	Takes the variable used by the
<code>.getElementsByName(" ")</code>	<code>name=</code> attribute As an example, let's say your meta tag attribute looks like this: <code>name="category"</code> . You would format this method as <code>.getElementsByName("category")</code> .
<code>.getAttribute("scheme") ==== ""</code>	<code>scheme=</code> attribute As an example, let's say your meta tag attribute looks like this: <code>scheme="SAMPLE1"</code> . You would format this method as <code>.getAttribute("scheme") === "SAMPLE1"</code> .
<code>.getAttribute("content")</code>	No change required. This method automatically captures the value specified in the <code>content=</code> attribute. However, if you change the name of this attribute, you must modify this method to include that new name. For example, say you want to use an attribute called "category" instead of "content." That's fine, just change the method parameter to match the new name like this: <code>.getAttribute("category")</code> .

Tapad DMP Code Sample

Put this JavaScript code on the pages you want to use for meta tag data collection. This code sends meta tag data to Tapad and returns audience IDs in a JSON object to your page. You can make function calls against the returned data or send it to an ad server for content delivery.

Heads up!

See the [configuration instructions](#) *before* using this code in a production environment.

Tapad DMP code

```
/** Initialize Tapestry */
var _tapestry = _tapestry || [];
_tapestry.push({'initialize': 'xxxxx'}); // Provide your Tapad partner ID
(function() {
    var taj = document.createElement('script');
    taj.src = ('https:' == document.location.protocol ?
        'https://' : 'http://') + 'assets.tapad.com/tapestry/tapestry-0.7.min.js';
    taj.setAttribute('async', 'true');
    document.documentElement.firstChild.appendChild(taj);
})();
/** Returns the page's audience. Returns null if not found. */
function GetAudience() {
    var tags = document.getElementsByName("xxxxxxxx"), // Provide the variable for the
meta tag 'name' attribute
        i,
        audienceTag;
    for (i = 0; i < tags.length; i++) {
        if (tags[i].nodeName === "META" && tags[i].getAttribute("scheme") === "xxxxx") {
// Provide the variable for the meta tag 'scheme' attribute
            audienceTag = tags[i].getAttribute("content"); // Gets the variable from the
meta tag 'content' attribute
            if (audienceTag) {
                return audienceTag;
            }
        }
    }
    return null;
};
/** 
 * Sends the page's audience to Tapestry and invokes a callback on the response:
 *
 * function(response) {
 *   console.log(response.audiences);
 * }
 *
 */
function SendAudiences(callback) {
    _tapestry.push({"add_audiences": [GetAudience()], "callback": callback });
}
```

AudienceManager Integration Guide

A Tapad - AudienceManager data integration helps you target AudienceManager segment members on multiple devices such as desktop and laptop computers, mobile phones, tablets, or game consoles.

Overview

The unique data sharing capabilities in AudienceManager and Tapad's cross-device targeting technology are the core components of this integration. To get started, have your AudienceManager account representative set up Tapad as a real-time and FTP destination. Once Tapad gets added to your account as destination, you can select and send AudienceManager segments to Tapad. When Tapad receives these segments, our system synchronizes AudienceManager IDs with corresponding Tapad IDs. After matching segment IDs, Tapad can target your Adobe audience members across all their known devices and browsers. A principal benefit of a Tapad - AudienceManager integration is that it helps you accurately expand audience reach to multiple devices and mobile platforms.

Prerequisites: You must be a Tapad customer to complete this integration.

- [Tapad - Adobe AudienceManager Integration Details](#) — Data synchronization methods and examples.
- [AudienceManager Configuration Process](#)

Tapad - Adobe AudienceManager Integration Details

ID synchronization

Both Tapad and Adobe initiate the ID synchronization process and store mapped (synchronized) ID data.

Tapad synchronization: Sample call

When Tapad makes the ID synchronization call, it looks like the example below. In this call, \${TA_DEVICE_ID} is a macro that contains Tapad device ID.

```
http://dpm.demdex.net/ibs:dpid=540&dpuid=${TA_DEVICE_ID}
```

AudienceManager synchronization: Sample call

When Adobe makes the ID synchronization call, it looks like the example below. This call includes the Adobe device ID and the corresponding Tapad device ID (\${TA_DEVICE_ID}), which is returned to AudienceManager.

```
http://pixel.tapad.com/idsync/ex/receive?partner_id=ADB&partner_url=http://px.demdex.net/ibs:dpid=540&dpuid=${TA_DEVICE_ID}&partner_device_id=USERS-ADOBED-DEVICE-ID
```

Methodologies for audience synchronization

With the Tapad/AudienceManager integration, Tapad synchronizes IDs with either of the following methods:

- Real-time, server-to-server data transfers
- HTTP POST that contains data in a JSON object

Real-time, server-to-server data transfers

With this method, the first_party_audience_advertiser_id contains the ID that lets Tapad identify which joint customer that user belongs to. A real-time, server-to-server call looks like the code provided in the example below. For more information about Adobe data integrations, see [Server to Server Data Integration](#).

```
http://data.tapad.com/dmp/adobe-import?first_party_audience_advertiser_id=SOME-PARTNER-ID
```

HTTP POST data transfers

Data comes from Adobe in a JSON object as shown below. In this example, the DataPartner_UUID parameter contains the Tapad cookie ID that gets sent to Adobe during an ID synchronization. For more information see the Adobe documentation, [Outbound Realtime Data Transfers](#).

```
{
    "ProcessTime": "Fri Jul 15 04:52:21 GMT 2012",
    "Client_ID": "74323",
    "AAM_Destination_Id": "423",
    "User_count": "2",
    "Users": [
        {
            "AAM_UUID": "19393572368547369350319949416899715727",
            "DataPartner_UUID": "4250948725049857",
            "Segments": [
                {
                    "Segment_ID": "14356",
                    "Status": "1",
                    "DateTime": "Fri Jul 14 01:23:21 GMT 2012"
                },
                {
                    "Segment_ID": "12176",
                    "Status": "0",
                    "DateTime": "Fri Jul 14 02:59:01 GMT 2012"
                }
            ]
        },
        {
            "AAM_UUID": "0578240750487542456854736923319946899715232",
            "DataPartner_UUID": "848457757347734",
            "Segments": [
                {
                    "Segment_ID": "10329",
                    "Status": "1",
                    "DateTime": "Fri Jul 15 01:23:21 GMT 2012"
                },
                {
                    "Segment_ID": "23954",
                    "Status": "1",
                    "DateTime": "Fri Jul 15 11:02:21 GMT 2012"
                }
            ]
        }
    ]
}
```

Batch files

Typically, batch files are a simple comma-separated lists that contain client, device, and audience or segment IDs. Tapad requires a full data file the first time we synchronize/integrate data. Subsequent updates can include another full batch file, containing old and new data, or an incremental file that contains new data only. Tapad prefers to receive real-time *and* daily batch files. Also, only the daily batch files contain audience descriptions, incremental batch file updates do not. In this integration, Tapad downloads the batch file from an FTP server set up by Adobe.

Audience Description File

A full batch file is required for the first audience synchronization. Additional synchronizations can use a full or incremental data. A sample description file may look similar to this:

```
Client ID, Segment ID, Segment_Name, Path
111,123,Segment A,"/All segments/funnel"
111,456,Segment B,"/All segments/funnel"
111,789,Segment C,"/All segments/funnel"
```

Incremental batch file

In a Tapad/Adobe integration, an incremental batch file contains new data only. Also, when Adobe appends a :0 or :-1 to a segment ID, that means an Adobe user has been removed from the segment. Data without those flags marks new segments that should be added to the database. In the following example, `device_a` gets added and `device_b` gets removed. For details, see the Adobe documentation, [Inbound Data File Contents](#).

```
cookie ID, Audience ID
```

```
device_a,1000  
device_b,1001:0
```

AudienceManager Configuration Process

Provide this information to your AudienceManager account manager. This section contains instructions on how to use AudienceManager administrative tools to send data to Tapad.

- [Send AudienceManager segments to Tapad](#) — Follow these instructions to send AudienceManager segment data to Tapad.
- [Set up Tapad as an FTP destination in AudienceManager](#) — Follow these instructions to add Tapad as an AudienceManager FTP destination.
- [Set up Tapad as a real-time HTTP destination in AudienceManager](#) — Follow these instructions to add Tapad as an AudienceManager HTTP destination.

Send AudienceManager segments to Tapad

Follow these instructions to send AudienceManager segment data to Tapad.

Prerequisites

Before you can complete these steps, Tapad must be set up as an FTP and real-time destination in AudienceManager. Work with your AudienceManager consulting contact to get started.

In the AudienceManager user interface (UI), the Destinations section lets you send segment data to Tapad.

To send AudienceManager segment data to Tapad:

1. Go to the Destinations section in the AudienceManager UI.
2. Search the destinations list for your Tapad HTTP destination and click the destination.
3. In the Segment Mappings section, use the search/browse feature to find for the segment you want to work with.
4. In the Edit Mapping pop, put the segment ID in the **Mapping** field and click **Add Mapping**.
5. Repeat these steps to set up your FTP destination.

See [Destinations Mappings](#) in the AudienceManager documentation for additional information.

Set up Tapad as an FTP destination in AudienceManager

Follow these instructions to add Tapad as an AudienceManager FTP destination.

These steps require access to the AudienceManager Admin UI. Work with your AudienceManager consulting contact to get started.

Find and edit the client profile

1. From the AudienceManager Admin UI, click **Partner Profiles**, search for the customer's account, and click **Edit**.
2. Scroll to the bottom of the partner's profile and click **Manage Server-to-server Destinations**.
3. Choose **Tapad** from the **Enable new server-to-server destination** dropdown. This opens the Destination information pop.

Configure Destination information settings

The Destination information pop contains 3 sections. Set up each section as follows:

Section	Settings
Server to server Configuration	<ul style="list-style-type: none">• Select Tapad (FTP) from the Destination dropdown.• <i>(Optional)</i> In the Description field, provide a short description.• Select Tapad from the Template dropdown.• Select the Sync all first party checkbox.• In the <partner name> ID in Tapad field, provide the Tapad customer ID for your client.• Leave the Label and Master Data Provider fields blank.
Transfer interval	Accept the default settings.
Configure server	<ul style="list-style-type: none">• Select SFTP from the Protocol dropdown.• In the:<ul style="list-style-type: none">• Host field, set the FTP address to <code>ftp21.demdex.com</code>.• Port field, specify port 22.• Username and Password fields, provide the Tapad user name and password.• Leave the Remote Path and File Suffix fields blank.• When finished, review your settings to make sure they're accurate and click Save.

Sample FTP destination

A properly configured Tapad FTP destination should look similar to the following:

Destination information

Server to Server Configuration

Destination	Tapad (FTP)
Description	<input type="text"/>
Destination Type	Server-to-server
Template	Tapad <input checked="" type="checkbox"/>
Sync all first party	<input checked="" type="checkbox"/>
<Partner name> ID in Tapad	<input type="text"/> 1234
Label	<input type="text"/>
Method	FTP
Master Data Provider	<input type="text"/>

Transfer Interval

Full Sync Type:
Customer Life Time Platform

Full Sync Schedule Never

Incremental Sync Schedule 4 hours

Configure Server

Protocol	SFTP <input checked="" type="checkbox"/>
Host	<input type="text"/> ftp21.demdex.com
Port	<input type="text"/> 22
Username	<input type="text"/> Tapad
Password	<input type="text"/> *****
Remote Path	<input type="text"/>
File Suffix	<input type="text"/>

Set up Tapad as a real-time HTTP destination in AudienceManager

Follow these instructions to add Tapad as an AudienceManager HTTP destination.

These steps require access to the AudienceManager Admin UI. Work with your AudienceManager consulting contact to get started.

Find and edit the client profile

1. From the AudienceManager Admin UI, click **Partner Profiles**, search for the customer's account, and click **Edit**.
2. Scroll to the bottom of the partner's profile and click **Manage Server-to-server Destinations**.
3. Choose **Tapad** from the **Enable new server-to-server destination** dropdown. This opens the Destination information pop.

Configure Destination information settings

The Destination information pop contains 3 sections. Set up each section as follows:

Section	Settings
Server to server Configuration	<ul style="list-style-type: none">• Select Tapad from the Destination dropdown.• (<i>Optional</i>) In the Description field, provide a short description.• Select Server-to-server from the the Destination Type dropdown.• Select S2S Publisher Seg from the Template dropdown.• In the <partner name> ID in Tapad field, provide the Tapad customer ID for your client.• Select HTTP from the Method dropdown.• Leave the Label and Master Data Provider fields blank.
Transfer interval	Accept the default settings.
Configure server	<ul style="list-style-type: none">• In the:<ul style="list-style-type: none">• Host field, set the HTTP address to <code>http://data.tapad.com</code>.• Port field, specify port 80.• URL Prefix field, set the prefix to <code>/dmp/adobe-import</code>.• Leave the Authentication URL field blank.• Select the Realtime checkbox.• When finished, review your settings to make sure they're accurate and click Save.

Sample real-time destination

A properly configured Tapad real-time destination should look similar to the following:

Destination information

Server to Server Configuration

Destination	Tapad <input checked="" type="checkbox"/>
Description	<input type="text"/>
Destination Type	Server-to-server <input checked="" type="checkbox"/>
Template	S2S Publisher Seg <input checked="" type="checkbox"/>
<Partner name> ID in Tapad	1234
Label	<input type="text"/>
Method	HTTP <input checked="" type="checkbox"/>
Master Data Provider	<input type="text"/>

Transfer Interval

Full Sync Type:
Customer Life Time Platform

Full Sync Schedule Never

Incremental Sync Schedule 4 hours

Configure Server

Host	ftp21.demdex.com
Port	22
URL Prefix	/dmp/adobe-import
Authentication URL	<input type="text"/>
Realtime	<input checked="" type="checkbox"/>

Save

BlueKai Integration Guide

- [Overview and Getting Started](#) — A Tapad - BlueKai data integration helps you reach BlueKai audiences on multiple devices such as desktop and laptop computers, mobile phones, tablets, game consoles, and other platforms.
- [Debugging](#)
- [BlueKai Integration Details](#)

Overview and Getting Started

A Tapad - BlueKai data integration helps you reach BlueKai audiences on multiple devices such as desktop and laptop computers, mobile phones, tablets, game consoles, and other platforms.

Overview

The unique audience sharing capabilities available in BlueKai and Tapad's cross-device targeting technology are the core components of this integration. This process starts after you create and share your selected BlueKai audiences with Tapad. When Tapad receives your audience information, our system synchronizes BlueKai IDs with corresponding Tapad IDs. After matching IDs, Tapad can target your BlueKai audiences across all their known devices and browsers. A principal benefit of a Tapad - BlueKai integration is that it helps you accurately expand audience reach to multiple devices and mobile platforms.

Getting started

Prerequisites: You must be a Tapad customer to complete this integration.

As a BlueKai customer it is easy to set up and send audience data to Tapad. Basically, you create an audience in BlueKai and share that audience with Tapad. See the BlueKai documentation for details, specifically:

- [Creating and Managing Audiences](#)
- [Sharing Audiences](#)

Debugging

And this: <https://jira.tapad.com/confluence/x/BIAuAQ>

BlueKai Third Party Segments

- Tracking tag is the segment id prefixed with "BK-" (e.g. BK-12345)
- If tracking tag does not exist in the UI/db
 - check that it is a member of com.tapad.dmpBluekaiAudiences#TolImport
 - check tapellite server logs for errors
- If name and description are not populated
 - check to see if the data is actually available by getting the Json data from <http://services.bluekai.com/Services/WS/Taxonomy?fullPath=1&description=1&bkuid=00c8c4045919c2c024fc6bda8bf98df3fc64e20e0f1facee0862e60fbcbca04&bksig=5ilX7nWgiLMvF9sXnGnRM3Fh6oW7lr2i5LOsp7UFJ6U%3D>
 - check the tapellite server logs for any errors in the scheduled jobs for "Bluekai unknown audience updater". It should be running every hour on the hour.

BlueKai First Party Campaigns

- Tracking tag is the campaign id prefixed with "BK-C-" (e.g. BK-C-567)
- If the campaign does not exist in the UI/db
 - check that it is configured and active by getting the Json data from <http://services.bluekai.com/Services/WS/Campaign?bkuid=00c8c4045919c2c024fc6bda8bf98df3fc64e20e0f1facee0862e60fbcbca04&bksig=x1lqV7ko8QFqJvEWOrXxe5BefiuGJAXqoayymL%2B%2FjE%3D>
 - check the above data to make sure the given id is actually the campaign id
 - check the tapellite server logs for errors
- If name and description are not populated
 - check that it is set up correctly by getting the Json data from <http://services.bluekai.com/Services/WS/Campaign?bkuid=00c8c4045919c2c024fc6bda8bf98df3fc64e20e0f1facee0862e60fbcbca04&bksig=x1lqV7ko8QFqJvEWOrXxe5BefiuGJAXqoayymL%2B%2FjE%3D>
 - check the tapellite server logs for any errors in the scheduled jobs for "Bluekai unknown audience updater". It should be running every hour on the hour.

BlueKai Integration Details

BlueKai

Type	<ul style="list-style-type: none">DMPTapestry Data ActivationDelivery File Exporter Product Requirements
Business Owner	Ryan Hotaling
Technical Owner	Shelly Weingarten
Status	NEW
Business Objective	
Development Work	
Documentation	BlueKai Integration Guide

Notes from 7/17 meeting:

- 1: Do an ID swapping- both sides initiate
2. We will send over a list of ID's we currently collect.
3. We can export BK ID mapping back to them
4. How to send it to them- real time, or when DG gets updated,
5. One problem- Need to get all the hashes so we figure out how to best aggregate them or create a universal ID

Open Issues	Need to get documentation for the delivery exporter file
Next Steps	<ul style="list-style-type: none">Tech call scheduled for 7/17Sent them an agreement and they need to replyWaiting for feedback on formatting

Overview

- Company description: BlueKai is a DMP (Data Management Platform) which provides a system to manage data on behalf of their customers. This provides a single platform for customers to manage their data and build audiences.
- Partnership description: Tapad works with BlueKai in order to receive audience data to allow joint customers to leverage the Device Graph. By doing a sync with BlueKai IDs, Tapad can make the Device Graph available to joint customers.
- How implemented:
 - ID Sync
 - Initiated by: Tapad
 - [BlueKai Endpoint](#)
 - ID Map Storage: BlueKai only
 - Audience Sync
 - Methodologies available
 - Real-time server-to-server:
 - HTTP Post of JSON data. If audience doesn't have descriptions, Tapad will use HTTP API to pull descriptions from BK.
 - [Tapad Endpoint](#)
<http://data.tapad.com/dm/p/bluekai-import>
 - [Sample real-time call](#)

```
{
  "DeliveryTime": "Wed Mar
13 19:00:21 CDT 2013",
  "DestinationId": 13583,
  "PixelCount": 1,
  "Pixels": [
    {
      "BkUuid": "ZOn5/f/i99OS
NBwf",
      "CampaignId": 33516,
      "CategoryId": "1,32790,3
2838,33729,38838",
      "PartnerUuid": "50dfa8eb
-07f7-4dfb-bd83-f4337cc1
6f76",
      "PixelId": 36725,
      "PixelUrl": "http://tags.blu
ekai.com/site/13645?r=1",
      "Rank": 6,
      "Timestamp": "Wed Mar
13 18:58:51 CDT 2013",
      "UtcSeconds": 13632191
31
    }
  ]
}
```

- Daily batch file: N/A

Lotame Data Integration Guide

A Tapad - Lotame data integration helps you reach audiences across multiple devices such as desktop and laptop computers, mobile phones, tablets, game consoles, and other platforms.

Overview

User ID sharing and synchronization are the key components in a Tapad - Lotame data integration. This process starts when you share your Lotame user IDs with Tapad. When we receive Lotame IDs, our system synchronizes them with corresponding Tapad IDs. After matching IDs, you can use our APIs and SDKs to receive synchronized data and take action on it from the page or in your Lotame Crowd Control account. A principal benefit of a Tapad - Lotame integration is that it helps you accurately expand audience reach across multiple devices and platforms.

Audience synchronization methods

Tapad can send/receive and synchronize IDs in real-time with server-to-server data transfers or on a set schedule with a daily batch file.

Set Up ID Synchronization and Audience Passing

To set up Tapad - Lotame ID sharing and synchronization:

1. Contact your Lotame account manager and give them approval to place ID synchronization calls on your pages.
2. Enable Tapad ID synchronization in Lotame. You can do this in the Lotame user interface or by contacting your account manager.
3. Select an audience synchronization method. We recommend using both server-to-server *and* batch file synchronization. Inform Lotame of the method, and Lotame will begin passing all your audiences to Tapad.

Lotame Integration Details

Lotame Overview

Lotame is a data management platform (DMP) that provides its customers with a system to manage their on-line data and build audiences. A Tapad-Lotame data integration lets Lotame share and synchronize data with Tapad and make it available to Lotame customers.

Lotame IDs

In the mobile web environment with iOS browsers Lotame uses a first party cookie for tracking iOS devices.

Lotame also uses 2 other ID types:

- A basic ID for every audience.
- Targeting codes. Instead of the audience ID, many Lotame customers often use the targeting code with their ad server.

ID synchronization

Lotame makes the ID synchronization call and Tapad stores mapped (synchronized) data.

```
http://pixel.tapad.com/idsync/ex/receive?partner_id=LOTAME&partner_device_id=LOTAME_UID_MACRO
```

Methodologies for audience synchronization

ID synchronization lets Tapad match our IDs to Lotame IDs. Lotame can send IDs to Tapad via real-time, server-to-server data transfers (preferred) or by daily batch file updates.

Real-time, server-to-server synchronization

Tapad prefers to receive user and audience IDs from Lotame by server-to-server data transfers. With this method, Lotame makes a call to <http://data.tapad.com/dmp/lotame-import> that includes user and audience IDs in a JSON object as shown below.

```
{
  "deliveryTime" : 1312312323,
  "users": [
    {
      "userID": "60773733b1f258e9ac24d666dee1463a",
      "audiences" : [ 111111, 22222, 333333 ]
    },
    {
      "userID": "9837a7f3b1f2re8e0kc24d642idqe1473a",
      "audiences": [ ]
    }
  ]
}
```

Batch file synchronization

With batch file synchronization, Lotame sends several files to Tapad by SFTP (to <ftp://data.tapad.com>). The table below describes each file, by type.

File type	Description
-----------	-------------

Audience ID and name	These data files consist of: <ul style="list-style-type: none"> • A parent folder that uses a YYYYMMDD timestamp naming convention. For example, the name 20130817 indicates the folder contains audience ID and name data for August 17, 2013. • Subfolders for each Lotame client. The subfolders contain audience IDs and names in an array as shown below.
Lock file	This is an empty file with a .done extension. It is used to indicate that Lotame has finished uploading files for the day and that all data transfers are complete.
Metadata	The file metadata.json.md5 contains hashed audience ID and name data. It is used as a check to ensure that ID and name data is not corrupted during transfer.
Audience membership	A simple text file that contains user and audience IDs. This file is compared with the audience ID/name file to synchronize Lotame user and audience IDs with corresponding Tapad data.

File syntax for Lotame audience IDs and names

As shown in the example below, Lotame batch file subfolders contain audience IDs and names in an array.

```
{
  "audience": [
    {"id": 1,
     "name": "Sports
Lovers"},
    {"id": 2,
     "name": "Exposed to
Campaign X"},
    {"id": 3, "name": "Political
Junkies"}
  ]
}
```

Turn Integration Overview

The Turn Audience Suite is a platform that provides its customers with a system to manage and build audiences with online and offline data. This data integration lets Tapad share synchronized audience segments with Turn and make that data available to their customers.

Methodologies for audience synchronization

Tapad uses pixel fires to make real time ID synchronization call to Turn. A standard Tapad pixel call uses this syntax:

```

```

In return, Turn sends data to their Tapad endpoint at <http://data.tapad.com/dmp/turn-import>.

File syntax for Turn audience data

Turn sends data to Tapad in a JSON object as shown below.

```
{"users": [
  {
    "user_id": "7321",
    "segments": [
      {"segmentId_advertiserId": "A123_735", "data_cost": 5.0E-4,
       "avg_cost": 5.0E-4}
    ],
    "user_id": "9846",
    "segments": [
      {"segmentId_advertiserId": "A123_735",
       "data_cost": 250E-4, "avg_cost": 5.0E-4}
    ]
  }
]
```

Technical Specifications for ID Bridging

DRAFT Reference is JIRA DOC-165 and [this Google doc](#).

ID bridging is a process that lets data partners share hashed user IDs with Tapad. This user information helps us match authenticated user names to corresponding Tapad cookie IDs. Tapad accepts bridged ID data in real-time from browser pixel calls, from mobile devices via server-to-server data transfers (batch files), or programmatically through our Tapestry SDK. Refer to this material for requirements and code samples used to set up ID bridging.

- **Web ID Bridging** — Web ID bridging uses a JavaScript pixel call to send authenticated user names to Tapad. This is a real-time data transfer method and is ideal when you can implement our pixel on your site. Use this method to send data from desktop/laptop browsers and mobile websites.
- **Batch File ID Bridging** — Tapad partners can collect authenticated user IDs and send that data to Tapad in a batch file. Batch file bridging does not require website pixel calls or SDK code. Instead, you can collect data from your own servers, format it according to these specifications, and make it available via FTP methods.
- **Mobile SDK ID Bridging** — The Tapestry SDKs let you send authenticated user IDs to Tapad from mobile devices and apps. This method is in development and will be available soon.

Web ID Bridging

Web ID bridging uses a JavaScript pixel call to send authenticated user names to Tapad. This is a real-time data transfer method and is ideal when you can implement our pixel on your site. Use this method to send data from desktop/laptop browsers and mobile websites.

Sample code and variable definitions

Implement this code on your site so it collects an authenticated user name after login. Note, you do not need to include the < > symbols when you implement this code. Those simply indicate variable data.

```
function getUrl(userName) {  
    return  
  
    'http://tapestry.tapad.com/tapestry/1?ta_partner_id=<partner_id>&ta_partner_user_id=%7B%22<partner_prefix>%22:%22'+userName+'%22%7D';  
}  
var img = new Image();  
img.src = getUrl("<user_name>");
```

The following table describes the variables you need to configure to capture authenticated user names for ID bridging.

Variable	Data type	Description and example
partner_id	Int	Your Tapad-provided partner ID. <code>http://tapestry.tapad.com/.../?ta_partner_id=1234</code>
partner_prefix	String	A variable you can use to identify your partner name to Tapad. Try to limit the string to 3-4 characters. For example, if your company name is Big Data Company, you could shorten that to BDC and instantiate the variable like this: <code>http://tapestry.tapad.com/.../?ta_partner_id=1234...%7B%22BDC...</code>
user_name	String	The hashed user name you're sending to Tapad. <i>Do not</i> pass any PII in this field. <code>http://tapestry.tapad.com/.../...158f5cfb0220575375cca0a6cd386b0e</code>

Sample calls

The JavaScript pixel sends URL encoded data to Tapad. Calls from your site could look similar to the following example.

Encoded URL example

```
http://tapestry.tapad.com/tapestry/1?ta_partner_id=4357&ta_partner_user_id=%7B%22BDC%22%3A%2222158f5cfb0220575375cca0a6cd386b0e%22%7D
```

An unencoded URL call would look similar to the following example.

Unencoded URL example

```
http://tapestry.tapad.com/tapestry/1?ta_partner_id=4357&ta_partner_user_id={"BDC": "22158f5cfb0220575375cca0a6cd386b0e"}
```

Batch File ID Bridging

DRAFT

Tapad partners can collect authenticated user IDs and send that data to Tapad in a batch file. Batch file bridging does not require website pixel calls or SDK code. Instead, you can collect data from your own servers, format it according to these specifications, and make it available via FTP methods.

FTP info needed:

- Do we pull from a partner's dropbox or do they send to us?
- If they send us data, we need to include:
 - An FTP endpoint
 - File type extension (.csv, .txt, .tsv)
 - File name requirements
 - Any compression options
 - Sample FTP call maybe
 - Sample file

Batch file requirements

A properly formatted batch file contains the following required and optional headers.

Required fields

Header	Data type	Description and example
user_id	String	The hashed user name you're sending to Tapad. <i>Do not</i> pass any PII in this field. 158f5cfb0220575375cca0a6cd386b0e
hardware_id_type	String	A hardware type identifier (e.g., IDFA, Android ID, etc.).
hardware_id_value	String	An actual hardware ID (e.g., 98aff49e-1957-0566-c130-33a1ef0670ed).
platform	String	Device type name (e.g., iPhone, iPad, Android tablet, etc.).

Optional fields

Header	Data type	Description and example
user_agent	String	A basic user agent, or <i>user-agent string</i> , contains information such as browser type and version, operating system information, and other data. Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_3) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/35.0.1916.114 Safari/537.36
ip_address	String	The device IP address (e.g., 38.67.254.194).
timestamp	Timestamp	Date and time of a user's interaction with your site. Uses <i>yyyy-mm-dd hh:mm:ss</i> format (e.g., 2014-05-15 14:07:44).

Mobile SDK ID Bridging

DRAFT

The Tapestry SDKs let you send authenticated user IDs to Tapad from mobile devices and apps. This method is in development and will be available soon.

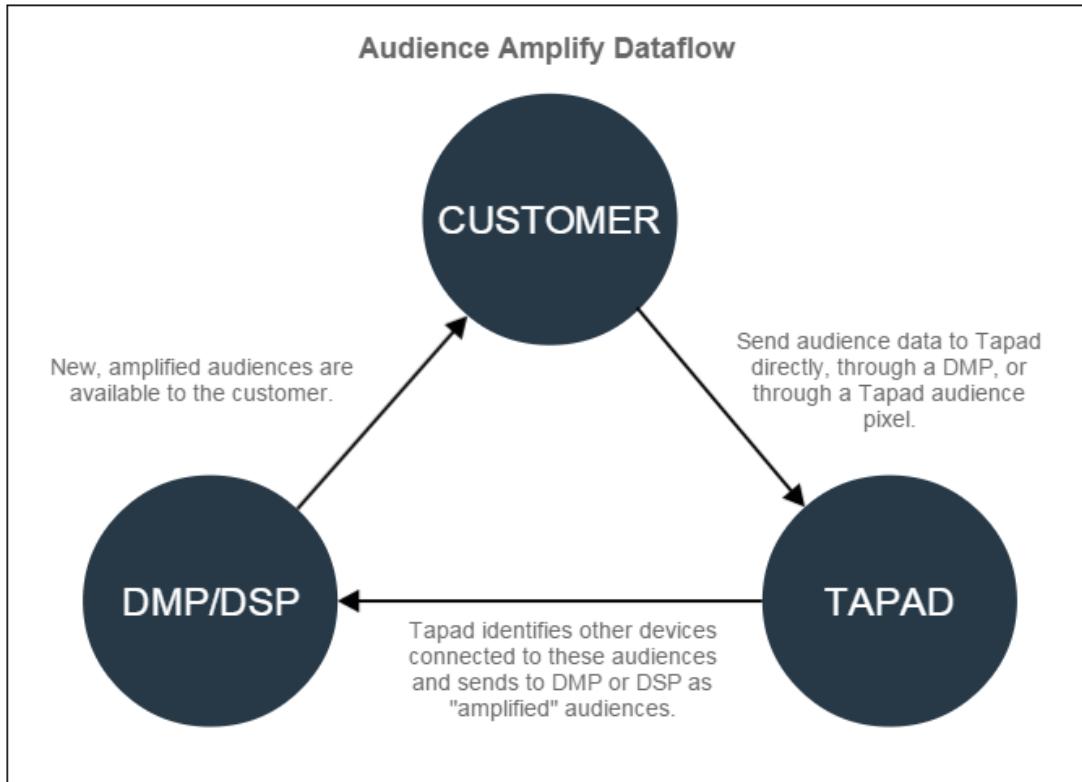
Audience Amplify

An overview of how it works and data file formats.

- [Audience Amplify: About](#) — Audience Amplify is a Tapad data integration that helps increase audience size. This basic integration starts when Tapad receives audience data from a customer directly, from their DMP, or from a Tapad audience pixel. Tapad then looks up the devices from these audiences in the Device Graph and finds those users' other, connected devices. The device IDs for these users' other devices are what comprise the "amplified" audiences created by Tapad. Finally, Tapad sends these amplified audiences back
- [Data File Formats for Audience Amplify](#) — An Audience Amplify data file contains device IDs, audience IDs, and device types. There are 2 types of data files, each with their own, unique syntax. A standard data file is for clients who do not use the AppNexus DSP. An AppNexus data file is for clients who already exchange data with that DSP.
- [Delivery Methods](#) — Information about available transfer methods, naming conventions, and delivery schedule.
- [Audience Amplify Appendix](#) — Supplemental reference material related to Audience Amplify.
- [Audience Amplify PDF](#) — Audience Amplify documentation in PDF format.

Audience Amplify: About

Audience Amplify is a Tapad data integration that helps increase audience size. This basic integration starts when Tapad receives audience data from a customer directly, from their DMP, or from a Tapad audience pixel. Tapad then looks up the devices from these audiences in the Device Graph and finds those users' other, connected devices. The device IDs for these users' other devices are what comprise the "amplified" audiences created by Tapad. Finally, Tapad sends these amplified audiences back to the customer or directly to their DMP or DSP.



Available DMPs

Tapad can receive audience data from most DMPs to create amplified audiences. Check with your Tapad Solutions Engineer or Implementation Engineer to make sure that we can receive audiences from your preferred DMP.

Data File Formats for Audience Amplify

An Audience Amplify data file contains device IDs, audience IDs, and device types. There are 2 types of data files, each with their own, unique syntax. A standard data file is for clients who *do not* use the AppNexus DSP. An AppNexus data file is for clients who already exchange data with that DSP.

- **Standard Format for Audience Amplify Data Files** — When a client does not use the AppNexus DSP, Tapad formats an Audience Amplify data file according to these specifications. If your clients exchange data with the AppNexus DSP, see the AppNexus DSP Format for Audience Amplify Data Files.
- **AppNexus DSP Format for Audience Amplify Data Files** — When clients use the AppNexus DSP, Tapad formats an Audience Amplify data file according to these specifications. When clients do not exchange data with the AppNexus DSP, see the Standard Format for Audience Amplify Data Files.

Standard Format for Audience Amplify Data Files

When a client *does not* use the AppNexus DSP, Tapad formats an Audience Amplify data file according to these specifications. If your clients exchange data with the AppNexus DSP, see the [AppNexus DSP Format for Audience Amplify Data Files](#).

Syntax

A standard Audience Amplify data file is formatted according to the syntax shown below.

- **Full syntax:** device_id|device_id_type|platform|add_to_audience1,add_to_audience2,...|remove_from_audience1,remove_from_audience2,...
- **Add an audience:** device_id|device_id_type|platform|add_to_audience1,add_to_audience2,...|
- **Remove an audience:** device_id|device_id_type|platform||remove_from_audience1,remove_from_audience2,...

In the file, a pipe character (the vertical bar "I") defines and separates each data element. Audiences can be added or removed based their location in the last 2 positions of the data string. If multiple audiences are being added or removed from a device ID, these audiences are separated by a comma (","). This position-sensitive syntax provides a generic file structure applicable to most use cases. In a standard Audience Amplify file:

- Each parameter is replaced with actual data in a real file.
- Each ID is written to its own, separate line.

Parameter definitions

Parameter	Description
device_id	An alphanumeric string that identifies a device.
device_id_type	See Device ID Types .
platform	See the Platform Types List .
add_to_audience	A comma-separated string of audiences the device is being added to.
remove_from_audience	A comma-separated string of audiences the device is being removed from.

Example

Given this syntax, the contents of a standard Audience Amplify data file look similar to the following:

```
00b853ba-65d7-49ea-ae9f-c1f2be7fc4dd|THIRDPARTY|COMPUTER|BDC-1|
029b320c78679764007199ad70592575|HARDWARE_MD5ANDROIDID|ANDROIDTABLET||BDC-1,BDC-2
01cf460b-1102-4061-ba5a-cce185b27a52|HARDWARE_IDFA|IPHONE|BDC-1,BDC-2|BDC-3,BDC-4
```

In the sample data:

- Line 1 shows a computer being added to audience BDC-1. THIRDPARTY is the identifier for the partner's cookie ID.
- Line 2 shows an Android tablet (identified with an MD5-hashed Android ID) being removed from audience BDC-1 and BDC-2. The double pipe || appears because the device is not being added to any audiences.
- Line 3 shows an iPhone (identified with an IDFA) being added to audiences BDC-1 and BDC-2 and removed from audiences BDC-3 and BDC-4.

AppNexus DSP Format for Audience Amplify Data Files

When clients use the AppNexus DSP, Tapad formats an Audience Amplify data file according to these specifications. When clients *do not* exchange data with the AppNexus DSP, see the [Standard Format for Audience Amplify Data Files](#).

Syntax

An Audience Amplify data file is formatted as shown below. Spaces have been added to the syntax examples for clarity. Normally, there are no spaces in the data file. In each data string, the position of the ID before or after a tab delimiter determines if it's added to or removed from a device.

- **Add audiences:** Device ID , Audience ID ; Audience ID ; Audience ID <tab> ^ Mobile Device Type ID
- **Remove audiences:** Device ID , <tab> Audience ID ; Audience ID ^ Mobile Device Type ID

Parameter definitions

Parameter	Description
Device ID	An alphanumeric string that identifies a device.
Audience ID	A numeric audience ID that's added to or removed from a device.
Mobile Device Type ID	Defined by the carat operator ^ this value identifies the device itself as one of the following: <ul style="list-style-type: none">• ^3: IDFA• ^4: SHA1UDID• ^5: MD5UDID• ^6: SHA1MAC• ^7: OPENUDID• ^8: AAID See also, Device IDs .

Example

Given this syntax, the contents of a sample Audience Amplify data file look similar to the following:

```
000a8b6ac66ced547df5a90ee9e064cccebec445,1866771;1866772 ^3
000266ac8859e430ec626555240d2fac, 1866770;1863385^3
000776eae39ffa9a75d40e5117f705270228b6c8,1866771;1866770;1866772;1863385 ^4
000DD56CABDEEF983F659D1A8A540019,1866770;1866771;1866772 ^5
000ac177-0654-4c9d-b18b-a5d83c32a7b3,1866621;1866770;1866771;1863385;1866772 ^3
000cd2a8-c915-42b3-897d-77c636d8b420,1863385;1866770 ^3
```

The data file can get pretty complicated looking. For simplicity sake, just look at lines 1 and 2. In this data file example:

- Line 1 shows audiences being *added to* a device because they appear *before* the tab.
- Line 2 shows audiences being *removed from* a device because they appear *after* the tab.
- Both device IDs are IDFA IDs as indicated by the ^3 parameter.

Delivery Methods

Information about available transfer methods, naming conventions, and delivery schedule.

Standard delivery methods include:

- **Protocol:** SFTP
- **Delivery method:** Push to partner (partner provides login credentials to Tapad)
- **Update type:** Incremental
- **Delivery schedule:** Weekly

Tapad uses the .tmp file extension to indicate that a data transfer is in progress.

- **Transfer in progress:** <Customername>_audiences_yyyyMMdd_HHmmss.gz.tmp
- **File complete:** <Customername>_audiences_yyyyMMdd_HHmmss.gz

Note: "<Customername>" will be replaced with your company's name. For example, an actual file for Acme Corp might look like:

- **Transfer in progress:** Acmecorp_audiences_20150101_153844.gz.tmp
- **File complete:** Acmecorp_audiences_20150101_153844.gz

Audience Amplify Appendix

Supplemental reference material related to Audience Amplify.

- **Device ID Types** — Tapad's Device Graph Access (DGA), and other systems, identifies devices and cookies with the IDs contained in this list.
- **Platform Types List** — Platform types are the commercial or commonly used names for various devices (e.g., Android tablet, iPhone, Kindle, etc.). The enumerated platform types are different than Device ID types, which are Tapad-specific descriptors for device hardware or cookies.
- **Feature Phones** — Feature phones are uncommon or specialized mobile devices, software, or communication protocols.

Device ID Types

Tapad's Device Graph Access (DGA), and other systems, identifies devices and cookies with the IDs contained in this list.

Device IDs are defined as follows.

ID	Device or cookie type	id_type	Description
3	Partner Cookie ID	2-4 character abbreviation	In the data file, Tapad identifies synchronized cookies with a 2-4 character abbreviated company name. For example, if your company name is Big Data Company, the cookie ID would be <code>BDC</code> .
18	Android ID	<code>HARDWARE_ANDROIDID</code>	An ID generated by the Android OS. If your engineering team needs a way to identify, parse, or validate these types of IDs, apply this regular expression to the data file: <code>"[0-9a-f]{16}"</code>
19	MD5 Android ID	<code>HARDWARE_MD5ANDROIDID</code>	An Android ID with an MD5 hash. This ID is generated by the Android OS. If your engineering team needs a way to identify, parse, or validate these types of IDs, apply this regular expression to the data file: <code>"[0-9a-f]{32}"</code>
20	SHA1 Android ID	<code>HARDWARE_SHA1ANDROIDID</code>	An Android ID with a SHA1 hash. This ID is generated by the Android OS. If your engineering team needs a way to identify, parse, or validate these types of IDs, apply this regular expression to the data file: <code>"[0-9a-f]{40}"</code>
23	Raw IDFA	<code>HARDWARE_IDFA</code>	An iOS IDFA ID. This is an identifier that is consistent/persistent across all apps. If your engineering team needs a way to identify, parse, or validate these types of IDs, apply this regular expression to the data file: <code>"[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}"</code>
24	SHA1 IDFA	<code>HARDWARE_SHA1IDFA</code>	An iOS IDFA ID with a SHA1 hash. If your engineering team needs a way to identify, parse, or validate these types of IDs, apply this regular expression to the data file: <code>"[0-9a-f]{40}"</code>
25	MD5 IDFA	<code>HARDWARE_MD5IDFA</code>	An iOS IDFA ID with an MD5 hash. If your engineering team needs a way to identify, parse, or validate these types of IDs, apply this regular expression to the data file: <code>"[0-9a-f]{32}"</code>
26	Raw IDFV	<code>HARDWARE_IDFV</code>	An iOS ID type. This ID is different for each app. It lets you track users, but does let you correlate activities on one app against activities on another app. Note, if you don't send in an IDFV, we will not return these IDs.
29	Google AdID	<code>HARDWARE_ANDROID_AD_ID</code>	A Google-specific ID used instead of a cookie ID. If your engineering team needs a way to identify, parse or validate these types of IDs, apply this regular expression to the data file: <code>"[0-9a-f]{8}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{4}-[0-9a-f]{12}"</code>

For the Tapad list of commercial or commonly used names for various devices (e.g., Android tablet, iPhone, Kindle, etc.), see the [Platform Types List](#).

Platform Types List

Platform types are the commercial or commonly used names for various devices (e.g., Android tablet, iPhone, Kindle, etc.). The enumerated platform types are different than [Device ID types](#), which are Tapad-specific descriptors for device hardware or cookies.

In the formatted output of many Tapad processes and data files, named platform types are often displayed in all caps and as a single word.

Platform type	Formatted as
Android	ANDROID
Android tablet	ANDROIDTABLET
Blackberry	BLACKBERRY
Computer	COMPUTER
Feature phone	FEATUREPHONE
iPad	IPAD
iPhone	IPHONE
iPod	IPOD
Kindle	KINDLE
Kindle Fire	KINDLEFIRE
Palm	PALM
Playstation	PLAYSTATION
Symbian	SYMBIAN
Television	TELEVISION
Wii	WII
Windows phone	WINDOWSPHONE
Windows tablet	WINDOWSTABLET
Xbox	XBOX

Feature Phones

Feature phones are uncommon or specialized mobile devices, software, or communication protocols.

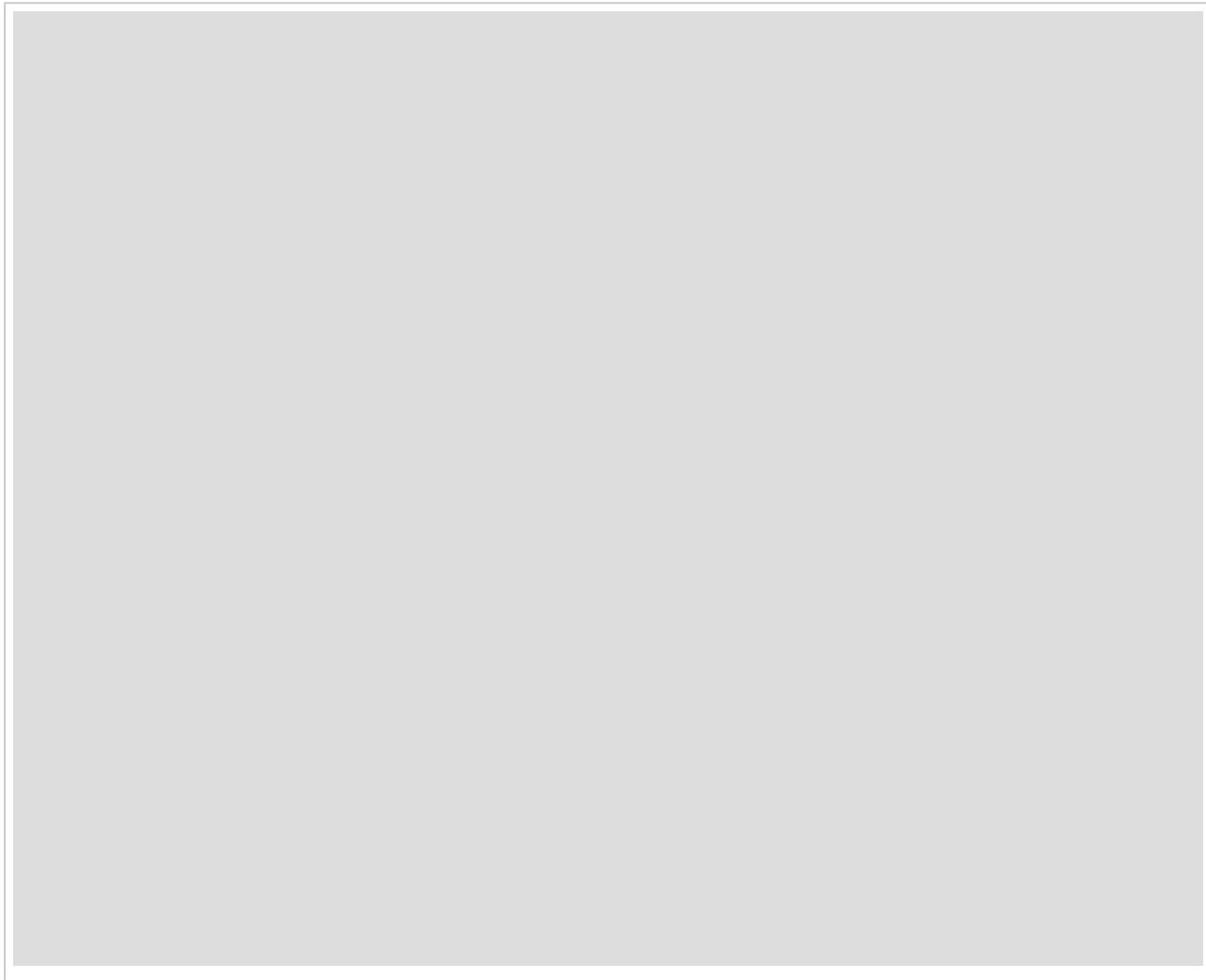
As defined by Tapad, feature phones include:

- [MIDP](#): Mobile Information Device Profile
- [MOBI](#): MobiPocket Reader
- [NetFront](#): Browser technology for game consoles and handheld devices.
- Nokia
- [Openwave](#): Proprietary communication protocols.
- Samsung
- [SEMC](#): Sony Ericsson Mobile Communications
- Sony

Audience Amplify PDF

Audience Amplify documentation in PDF format.

To download, click in the PDF frame.



Cross-plan Analytics

Overview, reports, and set up instructions.

- [Cross-plan Analytics: About](#) — Monitor campaign performance on different devices with Cross-plan Analytics.
- [Cross-plan Analytics: Available Reports](#) — Descriptions and sample reports.
- [Set Up Cross-plan Analytics](#) — Instructions, code samples and links required to configure cross-plan analytics.

Cross-plan Analytics: About

Monitor campaign performance on different devices with Cross-plan Analytics.

Purpose of Cross-plan Analytics

Cross-plan Analytics produces a report that gives advertisers a unified view of campaign performance. The report contains de-duplicated campaign data for each of your media partners across various platforms (display, laptop, mobile phone, tablet, etc.). Typically, these reports contain impressions, clicks, conversion, and app installation data. Tapad generates this report and sends it to you. Typically, marketers use Cross-plan Analytics to:

- Understand how media partners contribute to campaign performance.
- Set up cross-plan, cross-platform brand studies.
- Generate reports to show performance by media partner and device type.
- Measure actions on creatives or in-app activities such as installations, clicks, purchases, sharing, etc.
- Reduce the technical effort needed to set up data collection from separate sources and different devices.

How it works

To generate this report, Tapad collects and process data from:

- Conversion, impression, and click-tracking pixels.
- In-app activity captured by our Universal Software Development Kit (SDK).
- Partners who cannot implement our SDK code and return data by server-to-server methodologies.

Getting started

Your Tapad account manager can help you set up analytics reporting. This process involves 2 main steps. To use Cross-plan Analytics add our:

- Conversion, impression, and click tracking pixels to your display and mobile creatives.
- Universal SDK code to your mobile app. You can also set up server-to-server data transfers if you cannot implement the Universal SDK.

Note:

Tapad requires install tracking with any other event tracking.

Related tasks:

- Set up conversion tracking
- Mobile ads: Set up impression and click tracking
- Mobile video: Set up impression, quartile, and click tracking
- Set up the Universal SDK

Cross-plan Analytics: Available Reports

Descriptions and sample reports.

Exposure group reports

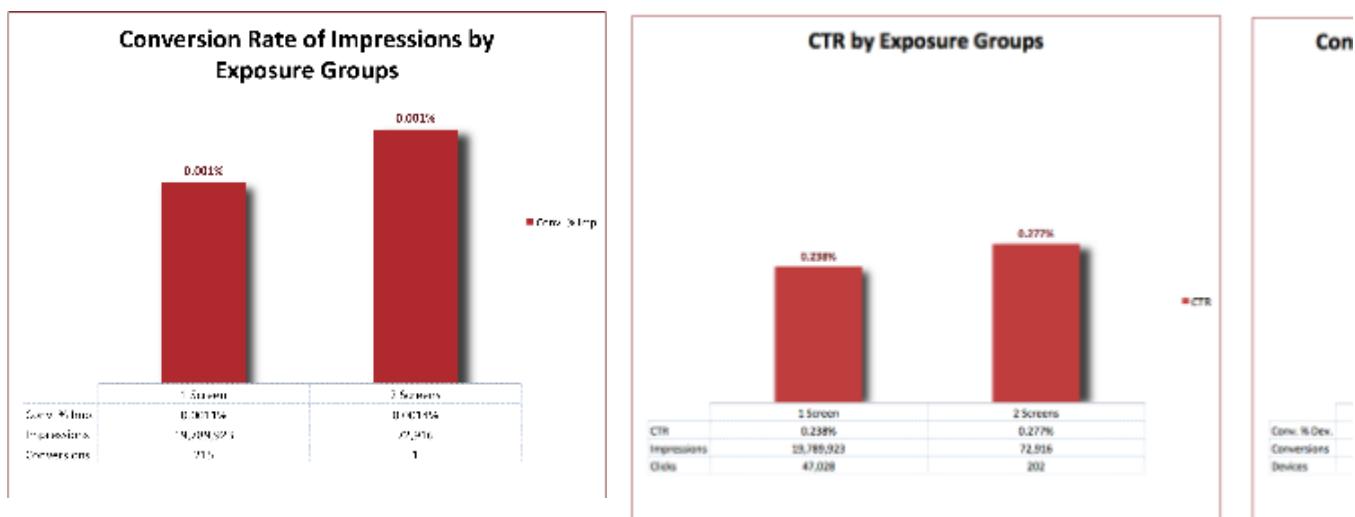
Exposure reports group users according to the number of devices they've used to see an ad. For example, in the conversion rate report below, the data measures the difference in conversion rates between users who have seen an ad on one device compared to users who have seen it on 2 devices. With cross-plan analytics, these reports can include:

- Conversion rates by impressions.
- Conversion rates by device.
- Click through rates.

This report is not available for CPA campaigns.

Click the sample report to expand it.

Exposure group report samples



Platform summary reports

The platform summary reports shows the platform that initially delivered an ad that a user eventually converted on. The conversion may or may not be attributable to the actual device itself, but it was the initial source of the ad. For example, say Tapad delivers 40,000 impressions to a Kindle. Ads delivered to the Kindle show a 1.3% conversion rate. The conversion could have happened on the Kindle or later, on another device. See detailed platform reports below for reports that show conversions by delivery and conversion. Basically, the platform summary report shows conversions by initial delivery device.

Click the sample report to expand it.

Platform summary reports

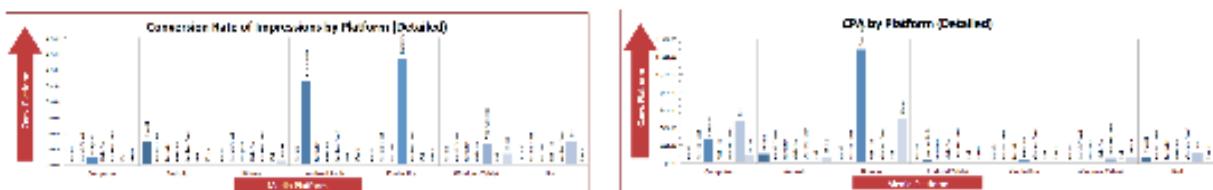


Platform detail reports

The platform detail reports show you the platform that first delivered an ad and the subsequent platform that recorded a conversion.

Click the sample report to expand it.

Platform detail reports

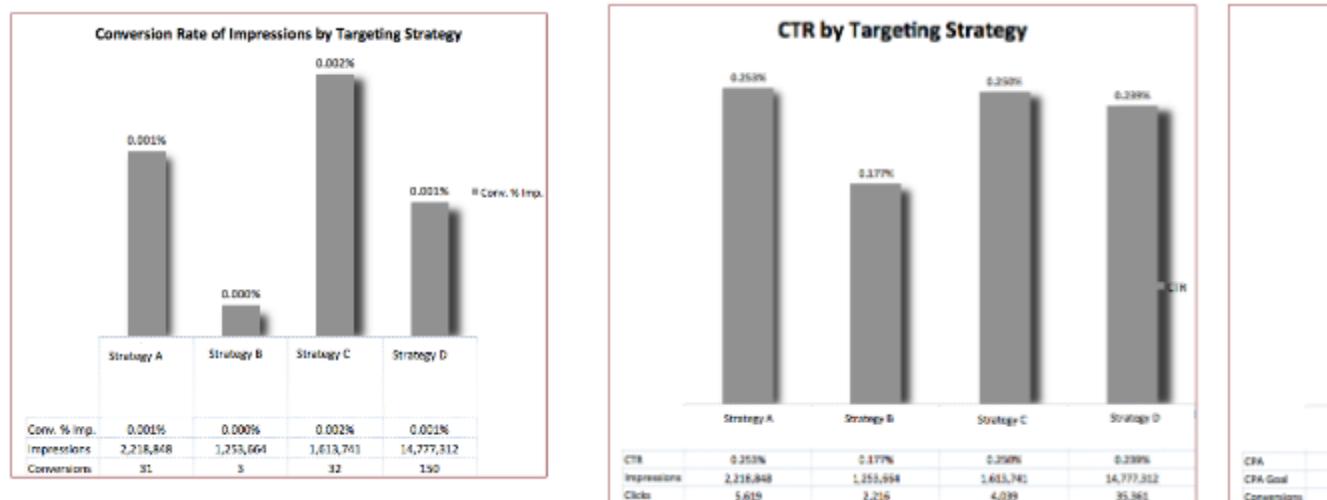


Strategy reports

The strategy reports measure and compare the effectiveness of a client's ad delivery plans. For example, a client wants to target ads based on different combinations of criteria such as age, gender, location, income, etc. This report shows how well each of those strategic combinations work.

Click the sample report to expand it.

Strategy reports



Set Up Cross-plan Analytics

Instructions, code samples and links required to configure cross-plan analytics.

- [Set up the Universal SDK](#) — SDK code and set up instructions for mobile devices.
- [Set up conversion tracking](#) — Code sample for a Tapad conversion tracking pixel.
- [Mobile ads: Set up impression and click tracking](#) — Follow these steps to set up impression and click tracking from display and mobile creatives.
- [Mobile video: Set up impression, quartile, and click tracking](#) — Follow these steps to set up to set up impression, quartile, and click tracking for display and mobile video creatives

Set up the Universal SDK

SDK code and set up instructions for mobile devices.

A requirement for Cross-plan Analytics includes setting up the Universal SDK code in your mobile app. For more information about the Universal SDK see [Universal SDK: About](#).

SDK code

Tapad provides a Universal SDK for iOS and Android devices. Download the source code from Github for your required device here:

- [iOS SDK](#)
- [Android SDK](#)

SDK set up instructions

To configure the SDKs for each device, see:

- Set up the iOS SDK
- Set up the Android SDK

Set up conversion tracking

Code sample for a Tapad conversion tracking pixel.

A requirement for Cross-plan Analytics includes placing a conversion-tracking pixel on your site. Add the pixel to the conversion page of your site or use it to record data on other pages or user actions. Be sure to use a unique ID for the different conversions you want to capture. Your pixel code could look similar to the following example:

Sample conversion tracking pixel

```

```

Important conversion-tracking variables include:

- `ta_campaign_id`: Tapad campaign ID (provided by Tapad).
- `ta_action_id`: ID of the conversion or other in-browser action you want to record.

Testing and support

The Tapad team can help make sure your code is set up properly and verify that we're receiving data. Contact us at support@tapad.com.

Mobile ads: Set up impression and click tracking

Follow these steps to set up impression and click tracking from display and mobile creatives.

A requirement for Cross-plan Analytics includes placing impression and click-tracking pixels on your site. To set up impression and click tracking:

1. Obtain the impression and click tracking code from Tapad.
2. Distribute these pixels to your partners from your ad server or by other automated or manual processes.
3. Ensure your partners add their cache buster to the code.

Code samples

Impression and click tracking code for your banner ads could look similar to the following examples:

Sample impression tracking code

```
http://analytics.tapad.com/tap/pxl.png?ta_action_id=<action_id>&ta_campaign_id=<campaign_id>&tid=1&crid=1&ssp=EXTERNAL&ctx=A&ts=${CACHE_BUSTER}
```

Sample click tracking code

```
http://analytics.tapad.com/click?ta_campaign_id=<campaign_id>&tid=1&crid&ssp=EXTERNAL&ctx=A&redirect=<insert_redirect_url_here>
```

Important impression and click-tracking variables include:

- **ta_campaign_id:** Tapad campaign ID (provided by Tapad).
- **ta_action_id:** ID of the conversion or other in-browser action you want to record.
- **redirect:** A URL that redirects the user to a landing page after a click action.

Testing and support

The Tapad team can help make sure your code is set up properly and verify that we're receiving data. Contact us at support@tapad.com.

Mobile video: Set up impression, quartile, and click tracking

Follow these steps to set up to set up impression, quartile, and click tracking for display and mobile video creatives

A requirement for Cross-plan Analytics includes configuring impression, quartile, and click-tracking pixels for your video creatives. To set up video pixels:

1. Obtain the impression and click tracking code from Tapad.
2. Distribute these pixels to your partners from your ad server or by other automated or manual processes.
3. Ensure your partners add their cache buster to the code.

Code samples

Impression, quartile, and click tracking code for your display and mobile video ads could look similar to the following examples:

Sample impression tracking code

```
http://analytics.tapad.com/tap/pxl.png?ta_action_id=<action_id>&ta_campaign_id=<campaign_id>&tid=1&crid=1&ssp=EXTERNAL&ctx=A&ts=${CACHE_BUSTER}
```

Sample quartile tracking code

```
http://analytics.tapad.com/tap/pxl.png?ta_action_id=<action_id>&ta_campaign_id=<campaign_id>&tid=1&crid=1&ssp=EXTERNAL&ctx=A&ts=${CACHE_BUSTER}
```

Sample click tracking code

```
http://analytics.tapad.com/click?ta_campaign_id=<campaign_id>&tid=1&crid=1&ssp=EXTERNAL&ctx=A&redirect=<insert_redirect_url_here>
```

Important impression, quartile, and click-tracking variables include:

- **ta_campaign_id:** Tapad campaign ID (provided by Tapad).
- **ta_action_id:** ID of the conversion or other in-browser action you want to record.
- **redirect:** A URL that redirects the user to a landing page after a click action.

Testing and support

The Tapad team can help make sure your code is set up properly and verify that we're receiving data. Contact us at support@tapad.com.

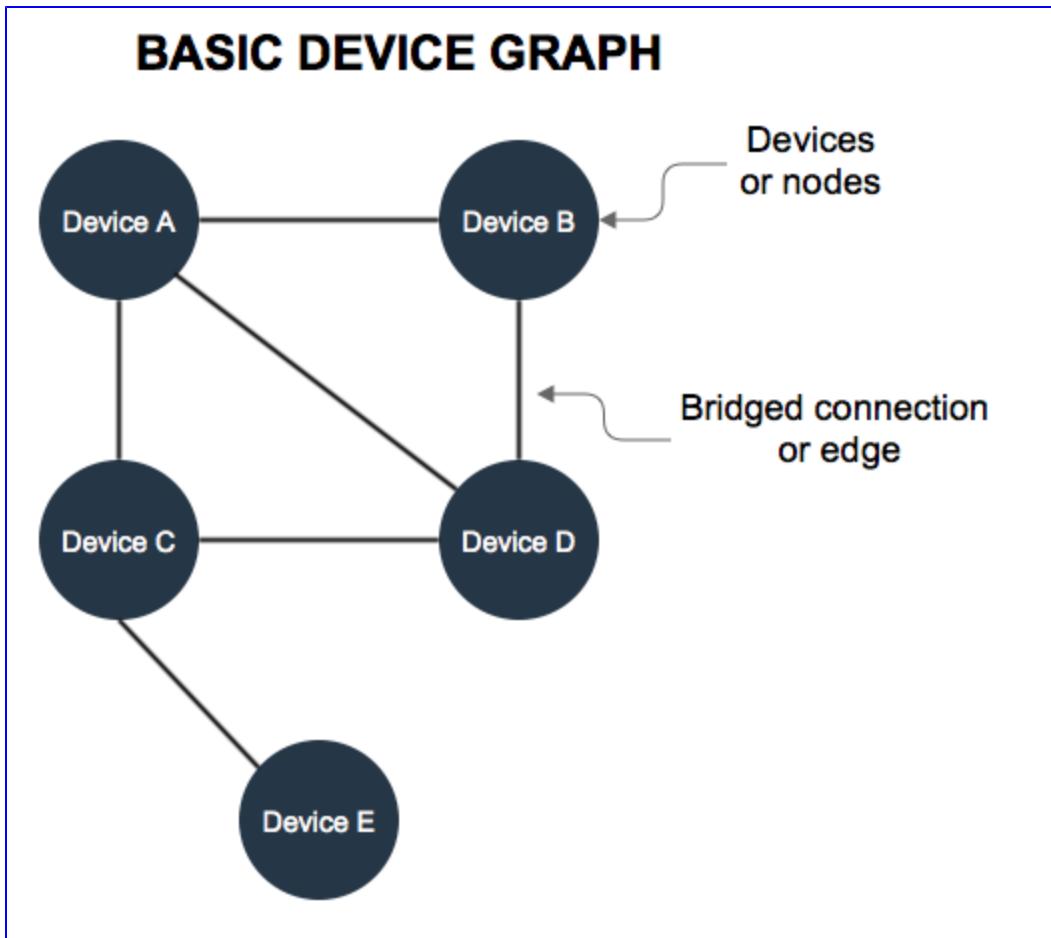
Device Graph

Overview and functionality.

- **Device Graph Basics** — The Device Graph (or graph) is Tapad technology that lets our systems target and deliver content to the same user across multiple devices. The Device Graph consists of internet-enabled devices seen on a network and the connections (or edges) shared between each device. In the Device Graph, a "connection" is not a physical link between devices. In this case, the "connection" or "edge" refers to a virtual relationship between devices. This relationship is based on how recently and frequently 2 (or
- **Household Scores** — Household scores measure the strength of the relationship between nodes (devices) in the Device Graph. Higher scores suggest a closer connection between devices in a household than lower scores. High scores improve accuracy, but reduce reach. Lower scores are less accurate, but increase reach. You can select a household score when synchronizing IDs or targeting devices and audiences. Keep in mind, however, that these scored connecations are abstract representations of different data points. They
- **Individual Scores** — Individual scores measure the strength of the relationship between devices (or nodes) within a household group in the Device Graph. They also help you sync IDs and reach devices at the user level. Higher individual scores between devices suggests they're more likely to belong to an individual user than devices that share lower scores. Higher scores improve accuracy, but reduce reach. Lower scores are less accurate, but increase reach. However, keep in mind that individual scores are based on st
- **Understanding Depth** — In the Device Graph, depth (sometimes called reach) indicates how far each device in the graph is from the root device. The root device is the first device discovered by the Device Graph during the bridging process.
- **Merge Logic** — Tapad receives cookie and hardware IDs (or signals) that get used to build a map of devices in the Device Graph. During ID synchronization, these signals can indicate that different devices are really the same device or that a device should have a different ID than the one that's already been assigned to it. This ID evaluation process relies on a set of rules known as merge logic. Merge logic tells the Device Graph when to merge devices, when to keep them separate, or when to replace/add IDs on
- **Device ID Superclusters** — Device ID superclusters are unusually large groups of device IDs connected to a single device ID in the Device Graph. Unfortunately, there's no minimum number of IDs that define or identify a supercluster. For example, a typical household might have a dozen device IDs associated with it. However, you might have a supercluster if an ID in that household group is connected to 20, 50, 100, or even thousands of other IDs. Fortunately, our technology can identify these clusters, prune them from the D
- **Device Identifiers** — ID types, adding IDs to the Device Graph, and detecting browser support for third-party cookies.
- **IDs Used for Attribution Between Devices and Apps** — IDs have several of functions that include user identification (which helps deliver targeted content), frequency capping, and unique user reporting. IDs also perform the crucial role of tracking attribution, which links conversions on one browser or device back to related impressions or clicks on another browser or device. Tapad can track attributions on computers, mobile devices, and apps across different operating systems.
- **Device Graph Methodology**
- **Device Graph Data Structures**

Device Graph Basics

The *Device Graph* (or *graph*) is Tapad technology that lets our systems target and deliver content to the same user across multiple devices. The Device Graph consists of internet-enabled devices seen on a network and the connections (or edges) shared between each device. In the Device Graph, a "connection" is not a physical link between devices. In this case, the "connection" or "edge" refers to a virtual relationship between devices. This relationship is based on how recently and frequently 2 (or more) devices are seen together on a network.



Building the Device Graph

Tapad builds the device graph with a process known as *bridging*. Bridging lets Tapad identify and make connections between devices on the same network. The result is a map of all the devices on a particular network. In addition to bridging, Tapad creates and assigns various numeric scores to the connections between each device. These scores are designed to quantify relationship between devices in the graph (see [Household Scores](#) and [Individual Scores](#) for more information).

Composition

A typical Device Graph contains:

- Desktop and laptop computers
- Tablets
- Mobile phones
- Game consoles (XBox, PlayStation, etc.)
- Connected TVs
- Other networked devices

Data sources

Tapad builds the Device Graph with data collected from:

- The HTTP header of ad impression.
- API and SDK calls.
- Information provided by contractual relationships with publishers or other site owners.

Heads up!

- Note that Tapad does not have business or contractual relationships with data networks or ISPs.
- Tapad does not (and cannot) track everything a user does online. Returned data is limited to what comes back on an impression call only.

See the following table for details.

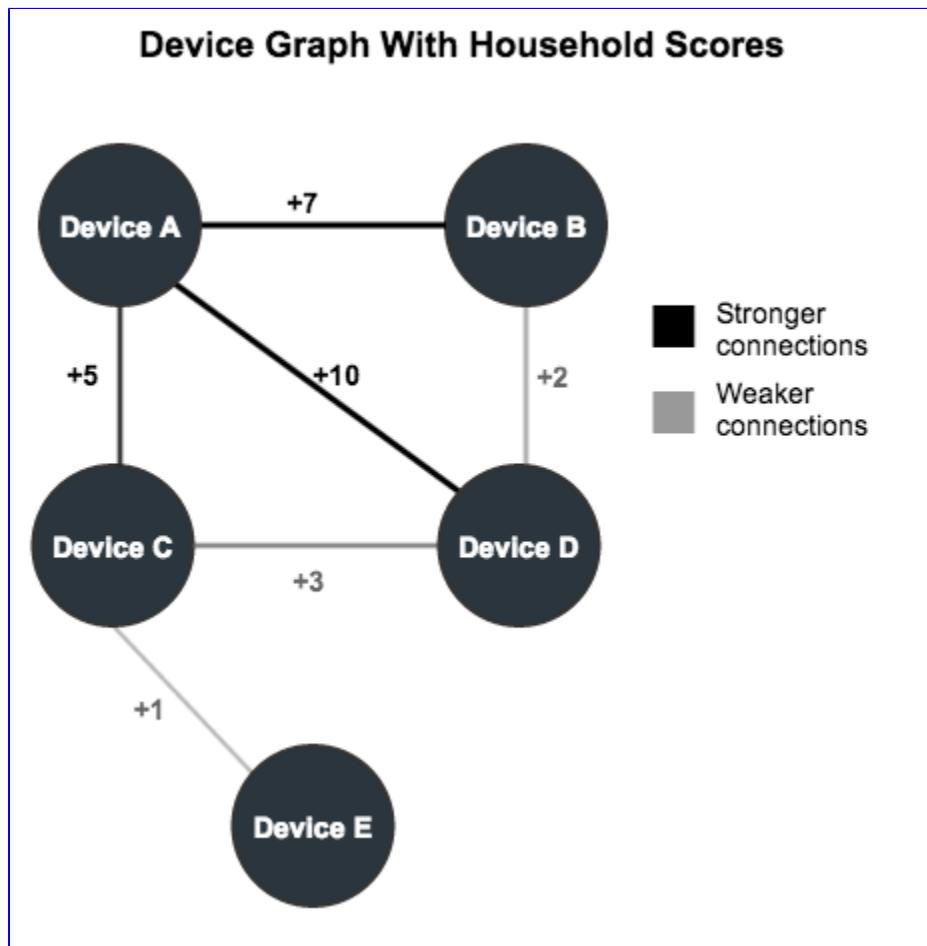
Data source	Description
HTTP Calls	<p>HTTP calls provide the largest data sets used to build the Device Graph. These calls come directly from:</p> <ul style="list-style-type: none">• Webpage code• Server-to-server integrations with other data providers or sources <p>This data includes, but is not limited to:</p> <ul style="list-style-type: none">• Standard values passed in an HTTP call• Any other data points passed by Tapad clients and other partners. <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">Tapad does not use or accept Personally Identifiable Information (PII) to bridge devices.</div>
Authenticated sources	Tapad purchases data from sites that require user authentication. User authentication provides us with data we can use to help build the Device Graph.
APIs or SDKs	Includes data returned by our APIs (Tapestry or TAJ) or by our Android and iOS SDKs.
Location-based sources	Includes data from airport, stadium, and hotel networks.

Household Scores

Household scores measure the strength of the relationship between nodes (devices) in the Device Graph. Higher scores suggest a closer connection between devices in a household than lower scores. High scores improve accuracy, but reduce reach. Lower scores are less accurate, but increase reach. You can select a household score when synchronizing IDs or targeting devices and audiences. Keep in mind, however, that these scored connections are abstract representations of different data points. They are not physical links between nodes.

Purpose of household scores

As described in the [Device Graph overview](#), a basic device graph is a map of connected devices on a network. By itself, a simple map is not very useful because the nodes and connections (or [edges](#)) between them look identical. As a result, we need a way to identify and define the relationships between each device. To do this, Tapad applies a metric called a *household score* to the edges between each device. These scores range from 1 (weakest) to 10 (strongest). Devices with higher scores are more strongly related than devices with lower scores. As an example, take a look at the sample device graph below. This diagram shows devices A through E and their corresponding household scores. Within this household, device A shares strong connections with devices B, C, and D. These scores quantify the relationships between devices in a household and helps make accurate, cross-device targeting possible.



Building household scores

Tapad uses proprietary technology and algorithms to identify household devices and calculate their scores. Some key metrics that contribute to score calculations include:

- **Frequency:** Tapad evaluates how frequently devices appeared together on a network. Our technology increments the household score between devices that connect to the same network frequently.
- **Recency:** Tapad evaluates how recently a set of devices appeared together on a network. Our technology increments the household score between devices if they appear together within a set time interval.
- **Authenticated data:** This is device information that comes from Tapad's relationships with sites that require user authentication. These devices usually have strong household scores because the authentication process ties a device to a specific user.

Household scores increase in value only. However, to avoid perpetually increasing scores and maintain data freshness, edges can (and do) expire. Expired edges, and their scores, are removed from the Device Graph.

What do the scores really mean?

Scores span a range of values from 1 to 10. Choosing one score vs another affects how many devices you can target or how many devices are available for synchronization. But what does it mean to select a score of 3 compared to 5, or 7, or any other score. Well, as shown in the table below, a score of 1 represents 100% of the available edges in the Device Graph. Each score greater than 1 reduces the available edges by 10%. This means a score of 2 reaches 90% of all edges. A score of 3 reduces this to 80% and so on until you reach score 10, which represents 10% of all the edges. Finally, scores work on an inclusive *equal to or greater than* basis. For example, a score of 7 includes all those edges with the same value and any other edge with a higher score.

Score > =	% of available edges
1	100%
2	90%
3	80%
4	70%
5	60%
6	50%
7	40%
8	30%
9	20%
10	10%

Reach vs accuracy

Reach represents how many devices are available for targeting or synchronization. Accuracy represents how precise you are at delivering content to those devices. Reach and accuracy are inversely proportional. As you increase the score:

- Reach *decreases* because score targeting includes devices that are *equal to or greater than* the selected score only.
- Accuracy *increases* because you're more likely to reach that smaller device audience.

Let's use the sample device graph above to demonstrate this behavior. Say you want to synchronize device A with all the other devices in the graph and set the household score to 7. In this case, you'll sync with devices B, and D only because those devices share an edge with device A that is => 7. A score of 7 excludes any edges with lower values.

Related concepts:

- Individual Scores
- How Confidence Scores Affect Report Results

Individual Scores

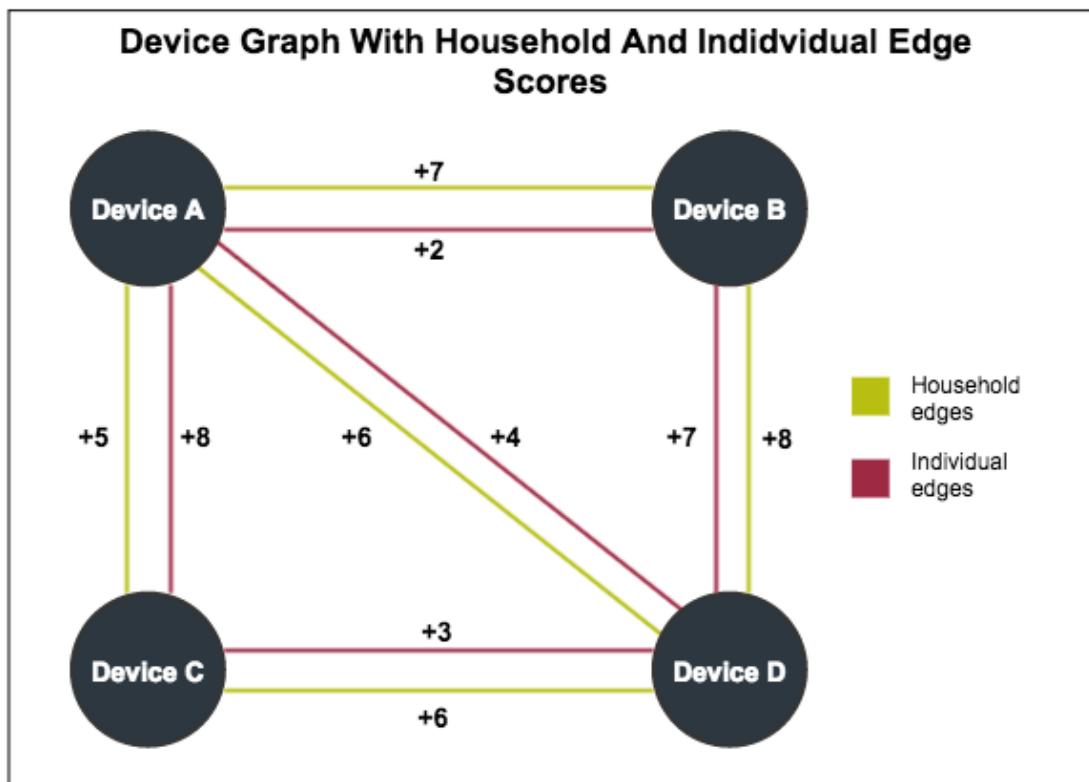
Individual scores measure the strength of the relationship between devices (or nodes) within a household group in the Device Graph. They also help you sync IDs and reach devices at the user level. Higher individual scores between devices suggests they're more likely to belong to an individual user than devices that share lower scores. Higher scores improve accuracy, but reduce reach. Lower scores are less accurate, but increase reach. However, keep in mind that individual scores are based on statistical analysis and represent abstract data points. They are not physical links between devices.

Purpose of individual scores

As described in the [Device Graph overview](#), a basic device graph is a map of connected devices on a network. By itself, a simple map is not very useful because the nodes and connections (or [edges](#)) between devices look identical. As a result, we need a way to identify and define the relationships between each device in the household and attempt to differentiate devices by their individual owners. To do this, Tapad identifies devices in a household and applies a metric called a [household score](#) to the edges between each device. After building a graph with household scores, we can then develop and apply another metric called an [individual score](#) to those devices. Individual scores help determine which devices in a graph belong to a particular user. The scores range from 1 (weakest) to 10 (strongest). Devices with higher scores are more likely to belong to the same user than devices with lower scores. Individual scores quantify the relationships between devices and users in a household and help make accurate, cross-device targeting possible.

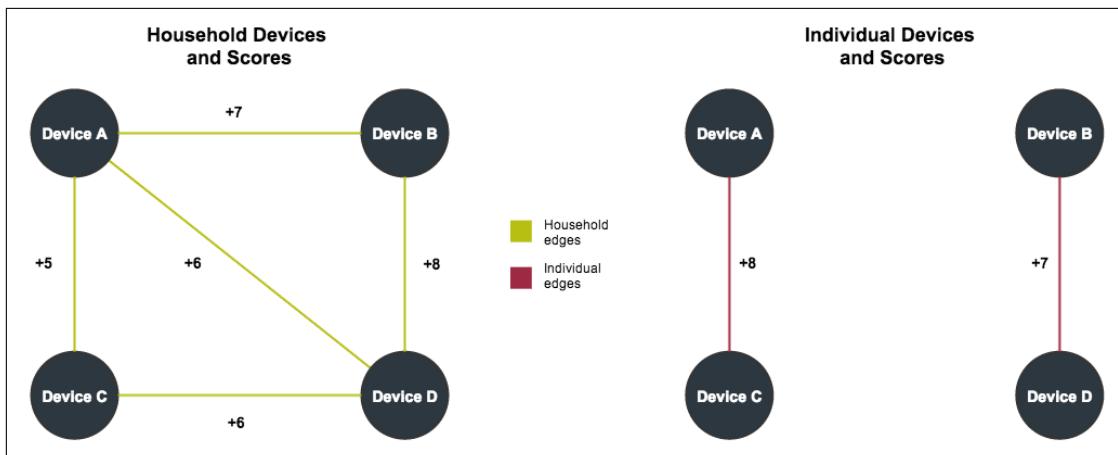
Individual and household scores compared

To help you understand how these scores work, take a look at the following illustration. This shows you how Tapad scores household and individual edges in simple device graph. The graph contains devices A through D and their corresponding household and individual scores.



Let's separate the devices by edge type and drop any connections < 5. This shows us that:

- High household scores between A, B, C, and D indicate these devices probably belong to the same household.
- High individual scores between A - C and B - D indicate these device pairs probably belong to different users. The other individual edge/score combinations for these devices are too low to make statistically confident assertions about device ownership.
- Individual scores suggest there are 2 distinct device users/owners in this household.



Building individual scores

Individual scores are based on comparing and scoring the browsing behaviors of different devices. The following table describes these processes. For detailed examples of each step, see the [numeric example](#) and [Pearson Correlation](#) documentation created by the Data Science team.

Process	Description
Collect data	The collection process gathers activity data about a device for a given context at a particular time. A context consists of browser or app-based activity such as visiting a particular website or using an app and can be extended to include other types of contexts. The end result is a list of sites or apps visited or used by a device.
Correlate data	The correlation process builds an activity matrix based on collected data. The matrix compares devices to site visits or app usage. For example, visiting a Yahoo fantasy sports site on an iPhone might also show a correlation with visits to sites like ESPN.
Score edges	Within a household, Tapad scores the edges between devices that overlap on similar site or app visits. Based on the results from a Pearson correlation , these edges get assigned a raw score that ranges between 0 - 1.
Test edges	Edge testing compares the activity between devices connected by a "real" edge with devices connected by a "false" edge. A real edge is an actual connection between 2 devices. A false edge is an artificial connection between 2 other separate but identical devices. For example, say iPhone A and iPad A share a real edge in the Device Graph. Testing against a false edge would link iPhone A to iPad B and iPad A to iPhone B as shown below.
	<p>The diagram shows four devices arranged in a square: iPhone A (top-left), iPhone B (bottom-left), iPad A (top-right), and iPad B (bottom-right). A solid horizontal line connects iPhone A to iPad A, labeled as a 'True edge'. Two dashed diagonal lines connect iPhone A to iPad B and iPad A to iPhone B, labeled as 'False edges'. A blue box encloses the top row of devices (iPhone A, iPad A).</p>
	<p>The false edge gets a score, which is compared to the score for the real edge. If the real edge score is greater than the false edge score, the real edge wins. Repeating this test for 1000s of edges lets us create a win rate for the real edge. After testing, we apply a function to the win rate to generate the individual score for the real edge between 2 connected devices.</p>
Bin scores	<p>Binning creates the 1 - 10 score range. These scores work like household scores. For more information see:</p> <ul style="list-style-type: none"> • What Do the Scores Really Mean • Reach vs Accuracy

Lower scoring links have value too

Devices with low scores are not automatically less desirable than devices with higher scores. For example, low scores suggest a looser social connection between devices and may indicate the presence of a friend on a known network. These lower scoring links can be valuable because that device owner may share similar interests and demographic characteristics as nodes with stronger household or individual scores. Also, low scores can mean that Tapad has not seen a particular device before, which lets us add that device to our system and expand the Device Graph. As the Device Graph grows, it discovers new devices with low scores that increase over time. Eventually, low scoring devices develop their own strong scores and connections that make desirable targets.

Score expiration

Like household scores, individual scores increase or remain the same, they do not decrease. However, scores and nodes have a finite time-to-live (TTL) value and can expire. For example, if our systems don't see a device within a specific number of days, it will remove the node and its edges from the Device Graph. Applying a TTL to the Device Graph graph helps ensure data remains current and accurate.

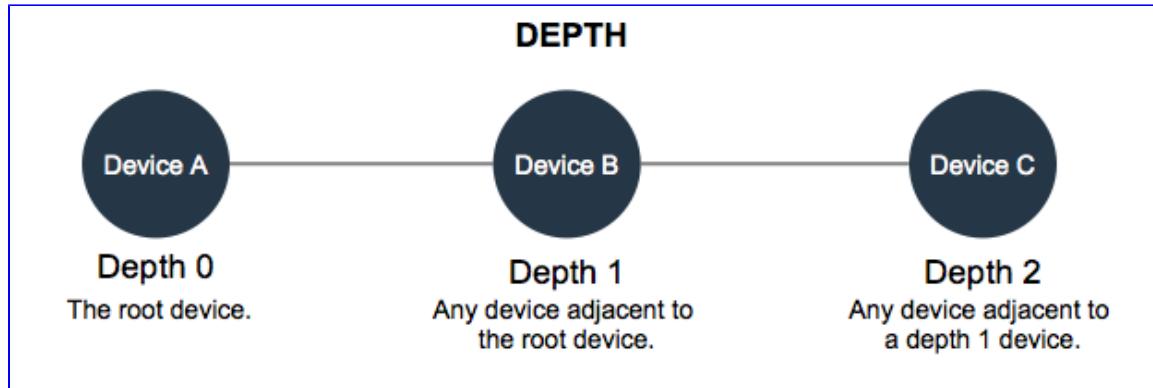
Related concepts:

- [Household Scores](#)
- [How Confidence Scores Affect Report Results](#)
- [Numerical Example - Pearson Correlation with Context Propagation & SVD](#)

Understanding Depth

In the Device Graph, *depth* (sometimes called *reach*) indicates how far each device in the graph is from the root device. The root device is the first device discovered by the Device Graph during the bridging process.

Depth has a range from 0 to 2 as shown below.



Targeting and depth

Targeting to a specific depth means you can reach those devices that have the specified depth or lower. For example, if you target devices at depth 1, you'll reach devices scored as equal to or less than 1.

Merge Logic

Tapad receives cookie and hardware IDs (or signals) that get used to build a map of devices in the Device Graph. During ID synchronization, these signals can indicate that different devices are really the same device or that a device should have a different ID than the one that's already been assigned to it. This ID evaluation process relies on a set of rules known as *merge logic*. Merge logic tells the Device Graph when to merge devices, when to keep them separate, or when to replace/add IDs on a device. The use cases in this section outline the conditions that result in merging and synchronization and those that do not.

- **Merging Conditions** — The merge process combines 2 separate devices into 1 if the assigned Tapad hardware or cookie IDs do not conflict. Also, in some cases, we may simply append IDs to a device instead of merging. These examples describe merging conditions in a 2-device set.
- **Non-merging Conditions** — In most cases, 2 separate devices cannot merge if they have already been identified with a Tapad hardware or cookie ID. These examples describe non-merging conditions in a 2-device set.
- **Replacing and Appending IDs** — During an ID sync, Tapad and cookie IDs get replaced while hardware IDs get appended to existing IDs for a device.

Related concepts

[Mapping Primary and Related IDs](#)

Merging Conditions

DRAFT

The merge process combines 2 separate devices into 1 if the assigned Tapad hardware or cookie IDs do not conflict. Also, in some cases, we may simply append IDs to a device instead of merging. These examples describe merging conditions in a 2-device set.

A note about variables

In the following examples, Tapad and partner cookie and hardware IDs are identified as shown in the table below. When making merge decisions, Tapad hardware and cookie IDs are authoritative when compared to partner-assigned IDs. This is because we know how our ID assignment logic works, but do not have insight into the ID assignment logic used by our data partners. The subscript variable $_n$ is a placeholder.

Symbol	Represents
H_n	Tapad hardware IDs
T_n	Tapad cookie IDs
P_n	Partner cookie IDs

Case 1

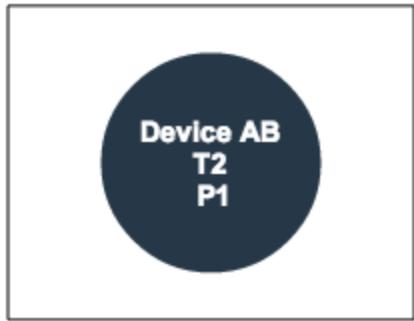
In this example we have 2 devices, A and B. They have been identified with the following Tapad and partner cookie IDs.



Next, a partner sends a signal to synchronize these devices by matching their P1 cookie with Tapad's T2 cookie like this:



As there's no conflict between Tapad IDs, we can synchronize these devices and merge them together. After merging, records for both devices are combined to create AB as shown below. This common use case is an example of most of the ID syncs that happen everyday. Also, devices A and B are not purged so original data is not lost.



Related concepts

- Non-merging Conditions
- Replacing and Appending IDs

Non-merging Conditions

DRAFT

In most cases, 2 separate devices cannot merge if they have already been identified with a Tapad hardware or cookie ID. These examples describe non-merging conditions in a 2-device set.

A note about variables

In the following examples, Tapad and partner cookie and hardware IDs are identified as shown in the table below. When making merge decisions, Tapad hardware and cookie IDs are authoritative when compared to partner-assigned IDs. This is because we know how our ID assignment logic works, but do not have insight into the ID assignment logic used by our data partners. The subscript variable $_n$ is a placeholder.

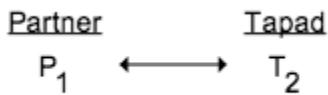
Symbol	Represents
H_n	Tapad hardware IDs
T_n	Tapad cookie IDs
P_n	Partner cookie IDs

Case 1

In this example we have 2 devices, A and B. They have been identified with the following Tapad and partner cookie IDs.



Next, a partner sends a signal to synchronize these devices by matching their P1 cookie with Tapad's T2 cookie like this:



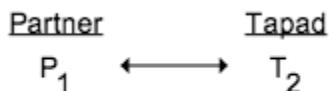
In this case, we can't synchronize or merge these devices because of a collision between the Tapad cookie IDs T_1 and T_2 . These IDs tell us that devices A and B have already been identified as separate devices. As a result, $P1 \neq T2$. The cookie IDs associated with each device, and the devices themselves, remain unchanged.

Case 2

In this example we have 2 devices, A and B. They have been identified with the following Tapad and partner cookie IDs.



Next, a partner sends a signal to synchronize these devices by matching their P1 cookie with Tapad's T2 cookie like this:



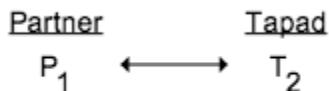
As in Case 1, we can't synchronize or merge these devices because of a collision between the Tapad cookie IDs T₁ and T₂. These IDs tell us that devices A and B have already been identified as separate devices. As a result, P₁ ≠ T₂. The cookie IDs associated with each device, and the devices themselves, remain unchanged.

Case 3

In this example we have 2 devices, A and B. They have been identified with the following Tapad and partner hardware and cookie IDs.



Next, a partner sends a signal to synchronize these devices by matching their P1 cookie with Tapad's T2 cookie like this:



Again, as in Cases 1 and 2, we can't synchronize or merge these devices because of a collision between the Tapad hardware IDs H₁ and H₂ and the cookie IDs T₁ and T₂. Even more so than cookie IDs, the hardware IDs indicate that devices A and B are completely separate devices. As a result, P₁ ≠ T₂. The cookie IDs associated with each device, and the devices themselves, remain unchanged.

Related concepts

- Merging Conditions
- Replacing and Appending IDs

Replacing and Appending IDs

DRAFT

During an ID sync, Tapad and cookie IDs get replaced while hardware IDs get appended to existing IDs for a device.

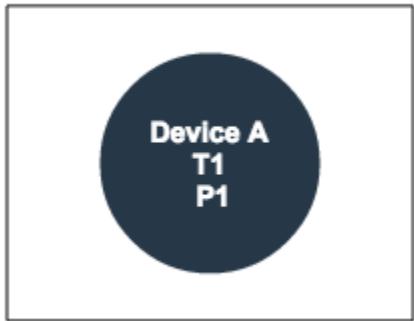
A note about variables

In the following examples, Tapad and partner cookie and hardware IDs are identified as shown in the table below. When making merge decisions, Tapad hardware and cookie IDs are authoritative when compared to partner-assigned IDs. This is because we know how our ID assignment logic works, but do not have insight into the ID assignment logic used by our data partners. The subscript variable $_n$ is a an ID placeholder.

Symbol	Represents
H_n	Tapad hardware IDs
T_n	Tapad cookie IDs
P_n	Partner cookie IDs

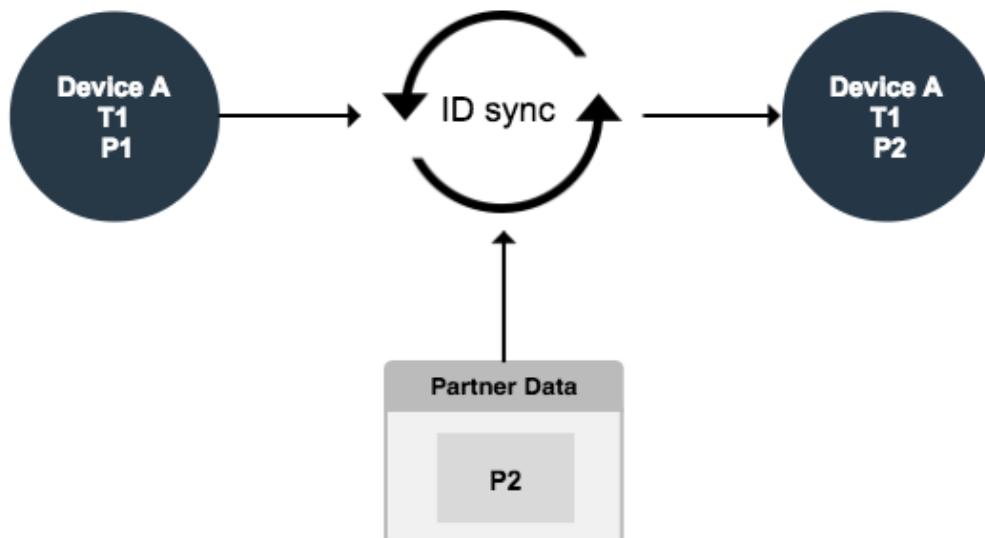
Case 1: Partner cookie ID replacement

In this example, Device A has been identified with Tapad cookie ID T_1 and partner cookie ID P_1 .



During an ID sync or another data transfer/update process, Tapad receives data that includes a new partner cookie ID (P_2) for Device A. As shown in the illustration below, Tapad replaces the old partner ID (P_1) with the new ID (P_2) on Device A. The Tapad cookie ID T_1 remains unchanged because our IDs are considered authoritative and take precedence over partner cookie IDs.

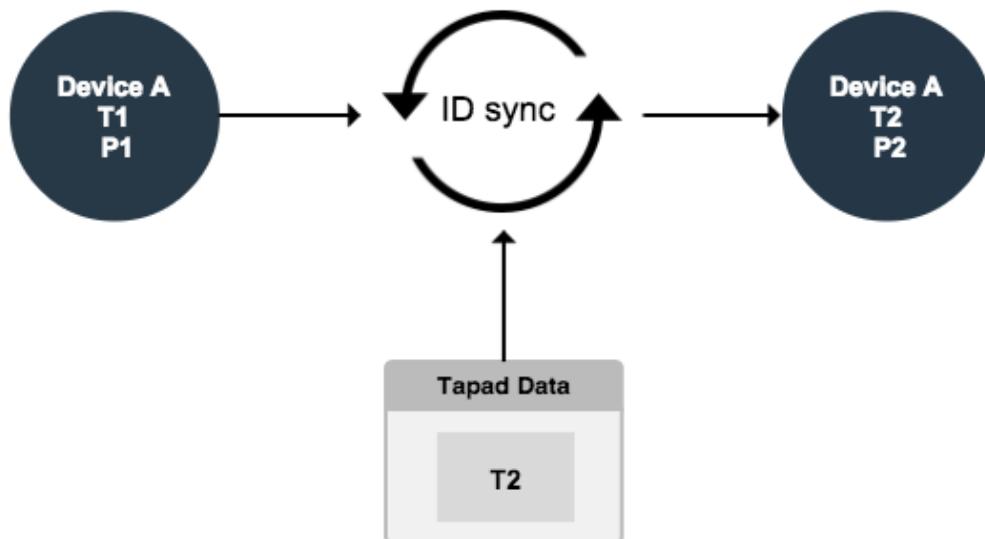
Partner Cookie ID Replacement



Case 2: Tapad cookie ID replacement

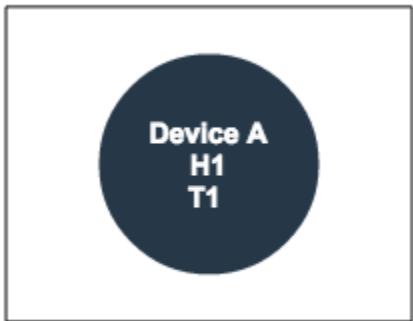
Similar to Case 1 above, Tapad replaces its own cookie ID with a newer cookie ID.

Tapad Cookie ID Replacement

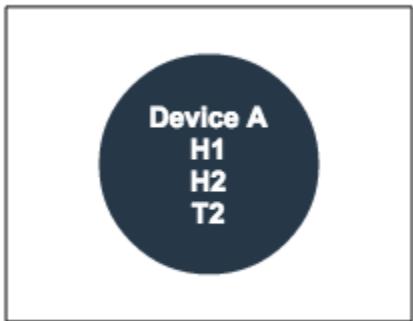


Appending IDs

When we identify a device with another hardware ID, that hardware ID is appended to the device. This process does not remove older hardware IDs. In this example, Device A has been identified with Tapad hardware ID H_1 and cookie ID T_1 .



Later, if we identify this device with another Tapad hardware ID (e.g., H_2), that new ID is appended to the device. Because hardware IDs are not replaced, Device A ends up with 3 IDs as shown below:



Related concepts

- Merging Conditions
- Non-merging Conditions

Device ID Superclusters

Device ID superclusters are unusually large groups of device IDs connected to a single device ID in the Device Graph. Unfortunately, there's no minimum number of IDs that define or identify a supercluster. For example, a typical household might have a dozen device IDs associated with it. However, you might have a supercluster if an ID in that household group is connected to 20, 50, 100, or even thousands of other IDs. Fortunately, our technology can identify these clusters, prune them from the Device Graph, and exclude them from the device ID files sent to clients.

- **Sources of ID Superclusters** — At Tapad, percentile filtering and unknown processes are responsible for most ID superclusters.
- **Filtering ID Superclusters** — To remove superclusters from reports, Tapad sorts device IDs into a series of equal buckets that group clusters by size percentiles. By default, we filter (exclude) clusters larger than those in the 99.99th percentile. You can see this filtering in the clusteringPercentiles and clusteringIdsFiltered results in the ID Mapping Report <https://jira.tapad.com/confluence/display/DOC/ID+Mapping+File+Metrics+Defined>.
- **A Field Guide to ID Supercluster Structures** — Typically, data files display primary and related IDs in rows of text or as key-value pairs in JSON objects. Given the nature of this alphanumeric data, it's often difficult to visualize the structure of an ID supercluster. As a result, we've created a few diagrams that may help you understand the structure of common supercluster variants. This guide includes a basic cluster, a daisy chain supercluster, and a spoke-and-hub supercluster.

Sources of ID Superclusters

At Tapad, percentile filtering and unknown processes are responsible for most ID superclusters.

Percentile filtering

During percentile filtering Tapad merges long strings of IDs together. This process deliberately creates ID superclusters. We do this to find the largest ID cluster for each client, which helps create a baseline for percentile filtering. During filtering, clusters are grouped into buckets from largest to smallest. The largest cluster constitutes the 100th percentile level. Smaller clusters are grouped into lower percentiles. This lets us apply filters to exclude superclusters larger than a selected percentile. Percentile filtering is essential to creating the [device ID mapping file](#) because the results in that file are filtered at the 99.99th percentile. At a high level, percentile filtering works like this:

- Find and merge related IDs into clusters.
- Identify the clusters and group them into percentiles according to the number of IDs in each cluster.
- Apply a filter to exclude clusters larger than the selected percentile. As an example, take a look at the [Sample ID Mapping File](#). The `clusteringPercentiles` results show you the number of IDs in the largest cluster for each percentile.
- Un-merge the remaining clusters.
- Send those IDs to clients and use that data to create the device ID mapping file.

Unknown processes

Unfortunately, sometimes it's not possible to determine the cause or origin of every device ID supercluster. In these cases, an unusually large cluster could be legitimate, but this is unlikely. These very large cluster may be created by automated systems or process unknown to Tapad and outside our control. Fortunately, Tapad has technology that helps reduce or eliminate the spurious effects of these large data formations.

Filtering ID Superclusters

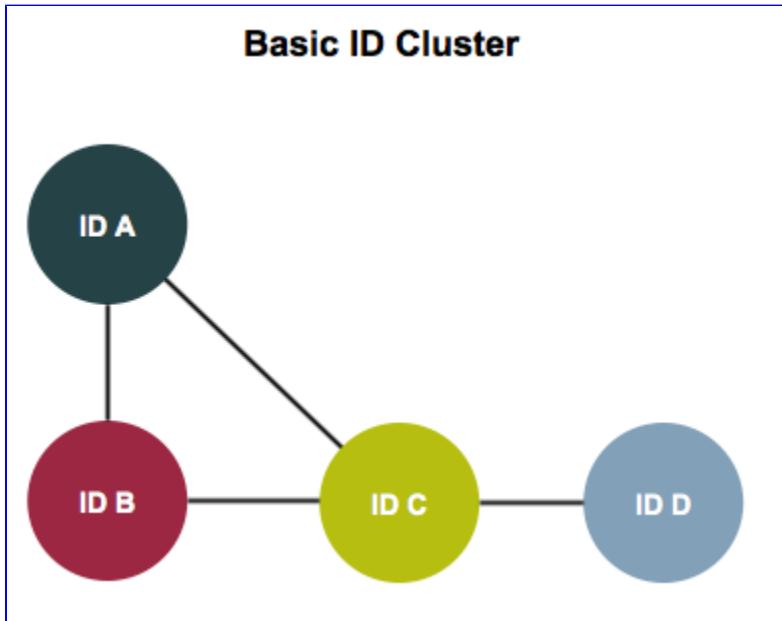
To remove superclusters from reports, Tapad sorts device IDs into a series of equal buckets that group clusters by size percentiles. By default, we filter (exclude) clusters larger than those in the 99.99th percentile. You can see this filtering in the `clusteringPercentiles` and `clusteringIdsFiltered` results in the [ID Mapping Report](#).

A Field Guide to ID Supercluster Structures

Typically, data files display primary and related IDs in rows of text or as key-value pairs in JSON objects. Given the nature of this alphanumeric data, it's often difficult to visualize the structure of an ID supercluster. As a result, we've created a few diagrams that may help you understand the structure of common supercluster variants. This guide includes a basic cluster, a daisy chain supercluster, and a spoke-and-hub supercluster.

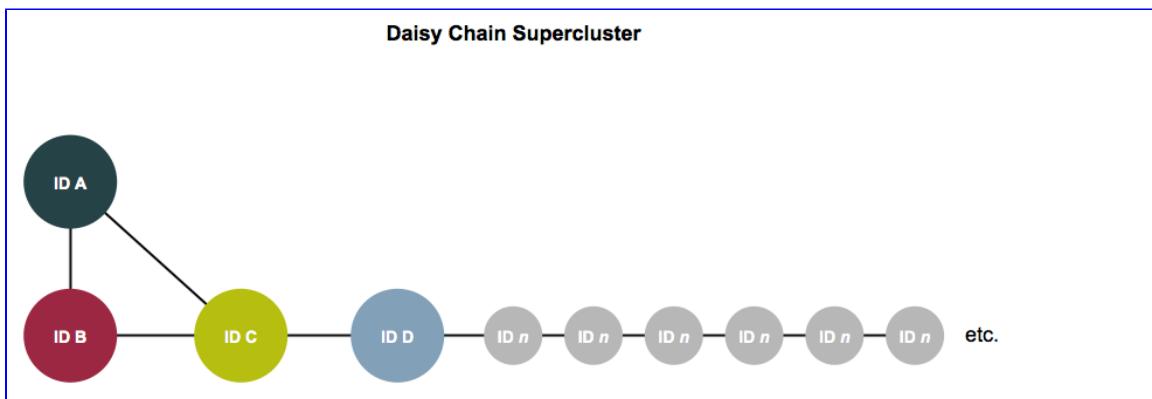
Basic ID cluster

The most common type of cluster consists of several connected device IDs. The structure of this basic cluster could look similar to the following diagram.



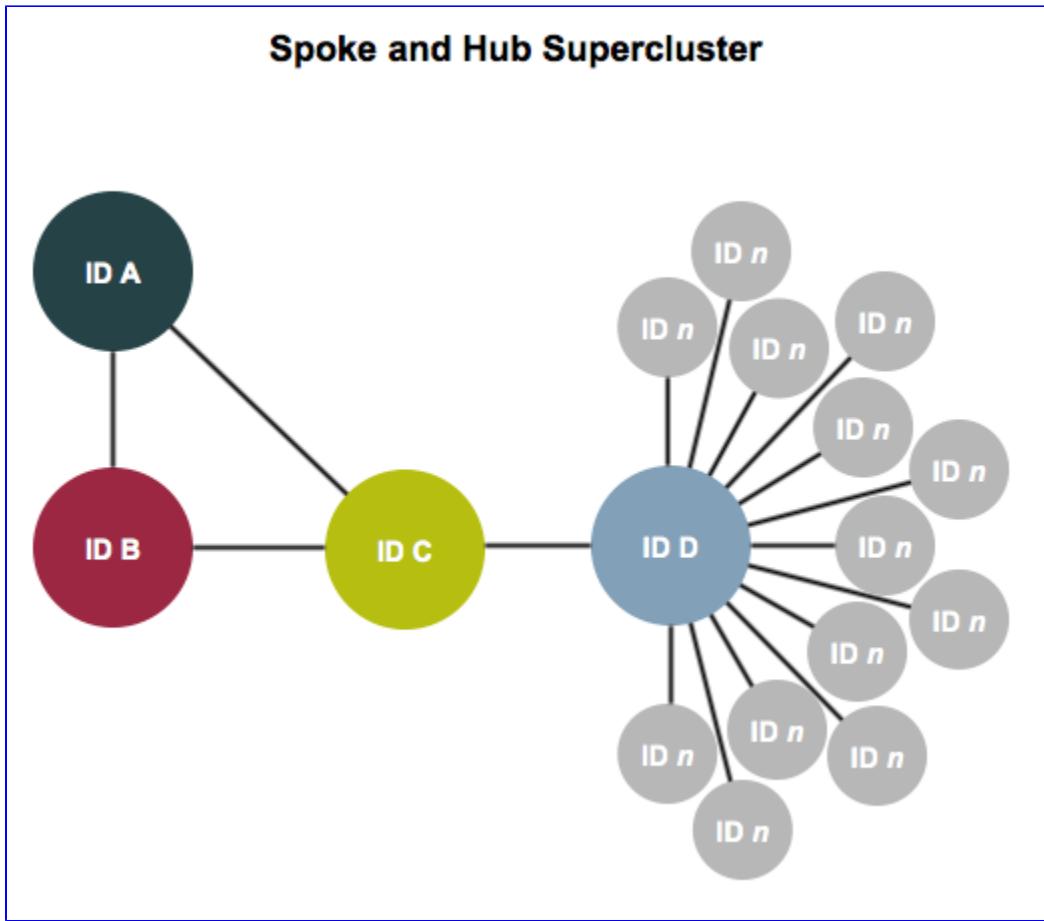
Daisy chain supercluster

A daisy chain consists of a long series of devices connected serially and is another, common type of device ID cluster. This supercluster type is not always suspicious and could represent a normal series of connections among separate devices. However, as a Tapad client, you should avoid merging these IDs together. These connections are not always fully transitive (i.e., an ID in one position is not always = to an ID a few hops away). See [Mapping Primary and Related IDs](#) for more information about the transitive/intransitive nature of device connections. The structure of this supercluster could look similar to the following diagram.



Spoke-and-hub supercluster

Spoke-and-hub superclusters are spectacular but less common types of superclusters. In this case, a single ID is the main node for dozens, hundreds, or even thousands of related IDs. Furthermore, it's often difficult to determine the source or cause of a spoke-and-hub supercluster. Internally, and in the data files sent to clients, Tapad can identify and exclude these clusters from the Device Graph. The structure of this supercluster could look similar to the following diagram.



Device Identifiers

ID types, adding IDs to the Device Graph, and detecting browser support for third-party cookies.

Types of identifiers

The following table describes the different IDs used by Tapad:

ID types	Description
First-party cookies	Tapad can store a unique ID in a first-party cookie and pass that value into the Device Graph. This ID is domain specific, but lets publishers bridge devices across their owned and operated inventory, even in contexts where third-party cookies are disabled.
Third-party cookies	For browser-based delivery. This is a standard, web-based cookie for browsers that support third-party cookies by default, or cookies that are more than 24 hours old.
Hardware IDs	For in-app delivery. Examples include: <ul style="list-style-type: none">• IDFA• Android ID in hashed or plain-text versions

Adding IDs to the Device Graph

Tapad stores IDs in the Device Graph as they become available. New IDs are added to the graph on:

- Bid requests (where contractually allowed)
- Impressions
- Cookie syncs
- Tapestry API calls

Detecting browser support for third-party cookies

It's not possible to determine if cookie support is enabled or disabled at the browser level. As a result, Tapad always tries to set a third-party cookie, even on browsers that don't accept them by default (e.g., Safari). Cookies that persist for at least 24-hours suggest that a particular browser supports third-party cookies or that a user has enabled this feature. A cookie that is 24-hours old is considered a trustworthy identifier for the device and can be used for targeting or attribution.

Most browsers these days are derivatives of Safari. This condition makes it hard to detect the Safari browser alone. Here are some examples:

Browser	Platform	User-agent
Chrome	Android	Mozilla/5.0 (Linux; Android 4.0.4; Galaxy Nexus Build/IMM76B) AppleWebKit/535.19 (KHTML, like Gecko) Chrome /18.0.1025.133 Safari /535.19
	Mac	Mozilla/5.0 (Macintosh ; Intel Mac OS X 10_8_3) AppleWebKit/537.36 (KHTML, like Gecko) Chrome /27.0.1453.116 Safari /537.36
	iPhone	Mozilla/5.0 (iPhone ; U; CPU iPhone OS 5_1_1 like Mac OS X; en-gb) AppleWebKit/534.46.0 (KHTML, like Gecko) CriOS /19.0.1084.60 Mobile/9B206 Safari /7534.48.3
	iPad	Mozilla/5.0 (iPad ; U; CPU iPhone OS 5_1_1 like Mac OS X; en-gb) AppleWebKit/534.46.0 (KHTML, like Gecko) CriOS /19.0.1084.60 Mobile/9B206 Safari /7534.48.3
Safari	Windows	Mozilla/5.0 (Windows ; U; Windows NT 6.1; sv-SE) AppleWebKit/533.19.4 (KHTML, like Gecko) Version/5.0.3 Safari /533.19.4
	Mac	Mozilla/5.0 (Macintosh ; Intel Mac OS X 10_8_3) AppleWebKit/536.29.13 (KHTML, like Gecko) Version/6.0.4 Safari /536.29.13
	iPhone	Mozilla/5.0 (iPhone ; CPU iPhone OS 6_0 like Mac OS X) AppleWebKit/536.26 (KHTML, like Gecko) Version/6.0 Mobile/10A5376e Safari /8536.25
	iPad	Mozilla/5.0 (iPad ; CPU OS 6_0 like Mac OS X) AppleWebKit/536.26 (KHTML, like Gecko) Version/6.0 Mobile/10A5376e Safari /8536.25

Opera	Mac	Opera/9.80 (Macintosh ; Intel Mac OS X 10.8.3) Presto/2.12.388 Version/12.15
	Android	Mozilla/5.0 (Linux; Android 4.1.2; SCH-I535 Build/JZO54K) AppleWebKit/537.31 (KHTML, like Gecko) Chrome /26.0.1410.58 Mobile Safari /537.31 OPR/14.0.1074.58201
	iPhone	Opera/9.80 (iPhone ; Opera Mini/7.0.5/30.3214; U; en) Presto/2.8.119 Version/11.10
Android stock browser	Android	Mozilla/5.0 (Linux; U; Android 4.0.3; ko-kr; LG-L160L Build/IML74K) AppleWebKit/534.30 (KHTML, like Gecko) Version/4.0 Mobile Safari /534.30
Puffin	• Android • iOS • Windows	See Puffin browser user agent format .
SkyFire	Android	Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10_5_7; en-us) AppleWebKit/530.17 (KHTML, like Gecko) Version/4.0 Safari /530.17 Skyfire/2.0

See also:

- [Chrome for Android User-Agent](#)
- [Safari User-Agent String](#)

Pseudo code for determining if third-party cookies are disabled

```
3rdPartyCookiesDisabled = (user agent contains "Safari" and not ("Mobile Safari" |
"CriOS" | "Chrome"))
```

IDs Used for Attribution Between Devices and Apps

IDs have several of functions that include user identification (which helps deliver targeted content), frequency capping, and unique user reporting. IDs also perform the crucial role of tracking attribution, which links conversions on one browser or device back to related impressions or clicks on another browser or device. Tapad can track attributions on computers, mobile devices, and apps across different operating systems.

Attribution IDs types

The following table lists the identifiers used on browsers and mobile apps that help tie impressions and clicks to conversions.

Device, app, or browser	Attribution ID
<ul style="list-style-type: none">Across different browsers (e.g., between Firefox and Chrome)Across apps and browsers	Device Graph
Within same browser (e.g., within Firefox) as long as 3rd party cookies are enabled. See Safari attribution for details about Safari attribution.	Cookie IDs
Within Android apps	Device IDs Identifiers include (plain text or MD5 or SHA1 hashed): <ul style="list-style-type: none">Android IDsMAC addressesPhone IDs
Within iOS apps	Advertiser IDs <ul style="list-style-type: none">IDFA (Identifier for Advertisers)Open UDID UDID (Universal Device ID) and MAC address is deprecated and replaced by these other ID types.
Safari browsers on iOS devices when 3rd party cookies are not enabled See Safari attribution below.	Derived IDs

Attribution with Safari on iOS devices

Browser-based attribution on mobile iOS devices is challenging because Safari does not accept third-party cookies by default. However, attribution tracking is possible. The following table lists the methods used to track attributions with the Safari browser on iOS mobile devices.

Attribution method	Description
Derived ID	Tapad uses a "derived ID" to perform attribution for devices on wireless home networks with Safari browsers that do not accept third-party cookies. The derived ID consists of the device IP address and browser user agent .
Cookies	When a user enables cookies in Safari, Tapad can set a third-party cookie and use the ID to track and attribute conversions. Cookies work with Safari and iOS mobile devices on a home wireless networks.
On a carrier network	Tapad cannot track attributions for iOS devices on a carrier network when Safari blocks third-party cookies. This is a common scenario because most users don't change their Safari preferences to allow third-party cookies. However, Tapad can track and attribute conversions for Safari browsers on a carrier network if a user enables third-party cookies <i>and</i> the cookie is at least 24-hours old.

Related concepts

[Understanding the Post-view and Post-click Attribution Window](#)

Device Graph Methodology

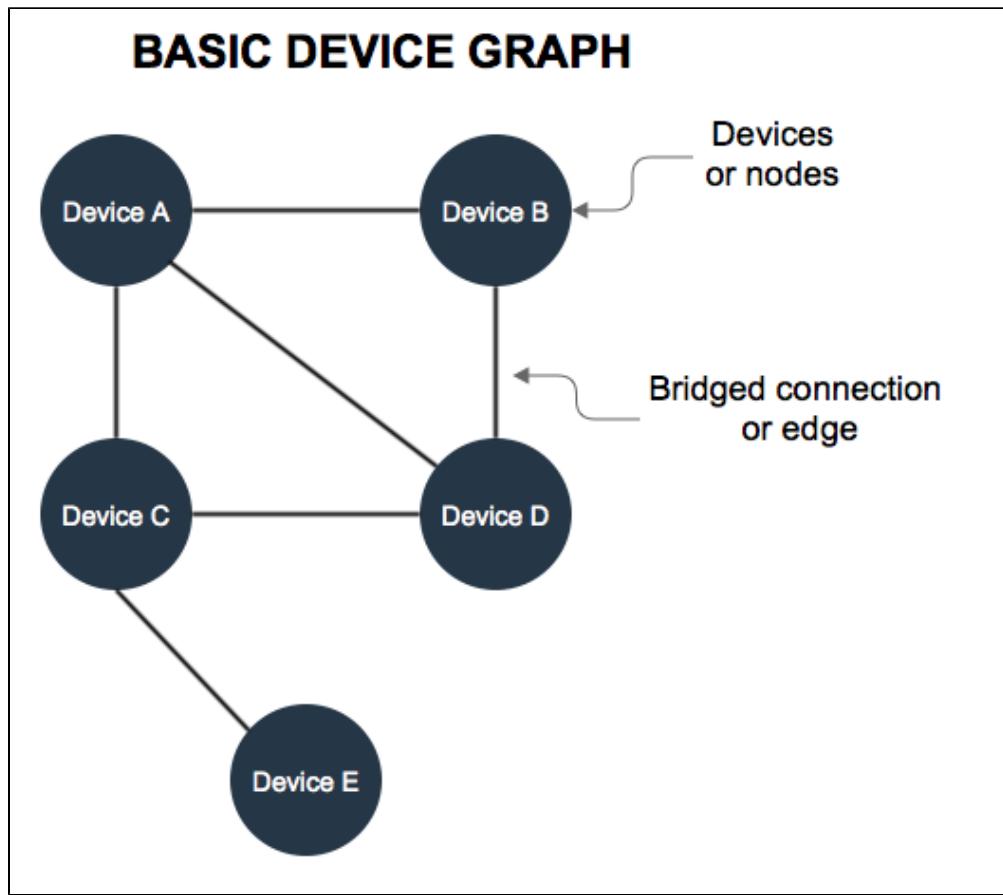
Purpose of the Device Graph

The Device Graph™ identifies and maps the relationships among and between multiple devices, usually belonging to the same user and/or household. It uses various rules, data sets, algorithms, and other metrics to measure these relationships in order to determine if the discovered devices belong to a single household or individual.

Graph Components: Nodes and Edges

The Device Graph consists of *nodes* (devices) and *edges* as described below:

Graph Component	Description
Node	Ideally, a node in the Device Graph represents an individual, physical device. Often, a physical device is represented by multiple nodes within the graph because it can have distinct unique IDs (hardware identifiers vs cookie identifiers). As a result, counting the number of nodes in the device graph may not provide an accurate view of the number of physical devices.
Edge	Bridges are represented in the Device Graph as a connection, or edge, between two nodes. Edges contain multiple properties, including the type of bridges represented by the edge (individual vs. household) and the confidence score associated with each type of bridge on that edge. These confidence scores (sometimes referred to as <i>strength scores</i>) constitute a continuous data set where larger numeric values suggest stronger relationships between bridged devices.



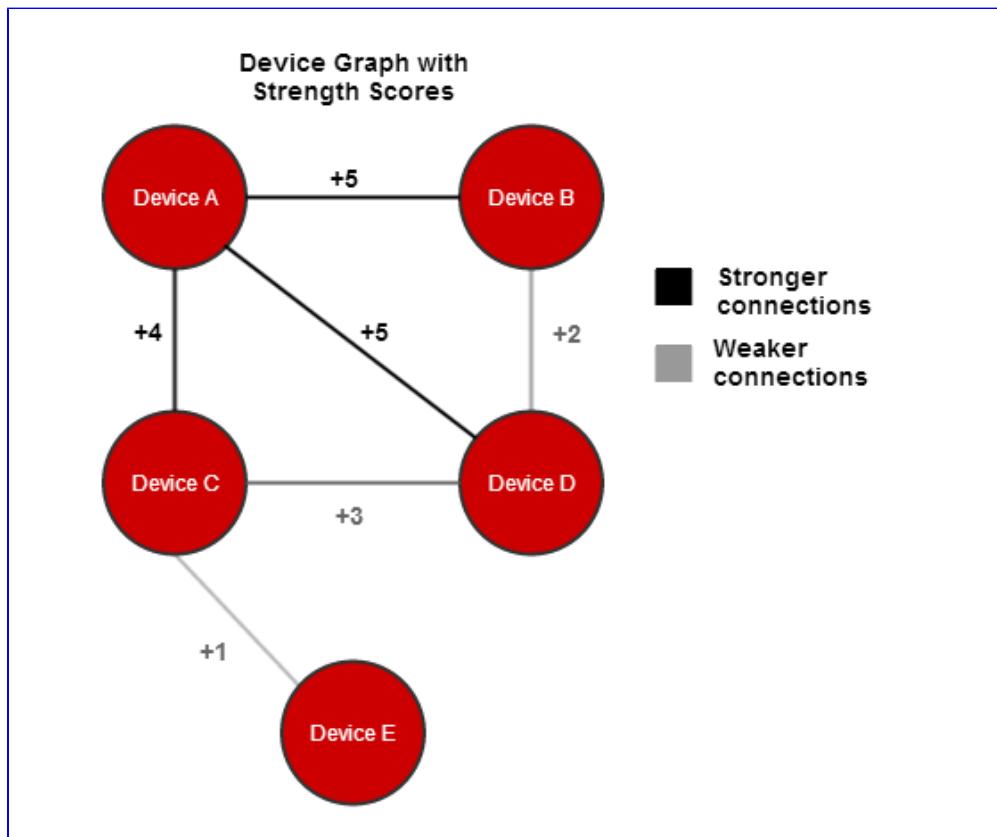
Finally, the device graph is a *disconnected* or *undirected graph*. Until nodes are joined by edges and scored, the graph consists of hundreds of millions connected components that have no relation to each other.

Graph Components: Strength Scores

A *strength score* measures the relationship between connected nodes in the Device Graph. Tapad uses these scores to help identify devices that

are more likely to belong to a single household, user, or group of related users than devices with lower scores. These values quantify the relationship between devices in the graph and help make accurate, cross-device targeting possible.

To illustrate this concept, take a look at the device graph below. This diagram shows devices A through E and their corresponding strength scores. The scores show that device A has strong connections to devices B, C, and D. In this case, Tapad assumes that these devices are more likely to be related than devices with lower scores. Targeting to device IDs with strong scores gives us a greater chance of reaching a particular user across all their devices than by delivering content to any (or all) nodes in the graph or just to IP addresses alone.



Building the Graph

Signals Used for Bridging

Signal Type	Description
Strong	<p>Strong signals are data points that represent a high level of confidence that multiple devices should be bridged or merged at the individual or household level. Tapad treats these strong signals as an immediate trigger to create a bridge or a merge devices or as data points to train and strengthen our models. Examples of strong signals include:</p> <ul style="list-style-type: none"> Getting obfuscated user login information on multiple platforms (e.g., user logins into a service on both their computer and phone). Direct links between mobile application and mobile web contexts that allow us to merge multiple nodes within the graph (e.g., unique ID sent through Android referral URL). Multiple hardware device identifiers coming from the same device.

Weak	<p>Weak signals are data points that allow us to bridge devices after we collected a number of related weak signals to change the device graph. These weak signals can be input for both our rules based and algorithmic models that affect the device graph. Weak signals include:</p> <ul style="list-style-type: none"> • HTTP headers, such as IP address and user agent that specifies device and browser information. • Anonymous data extrapolated from the user's IP address. For example, we may be able to determine the type of network the user is on (e.g., home, business, carrier) as well as their general location, which also lets us infer a demographic profile. • 3rd party audience data, such as demographic (when contractually allowed). <p>Weak data sources include:</p> <ul style="list-style-type: none"> • Pixel fires • Impressions and clicks • SDK or S2S integrations for mobile applications • Bid requests where contractually allowed
-------------	---

Rules Based vs. Algorithmic Bridging

We use a combination of rules-based heuristics and pure machine learning algorithms to build the device graph. Both methodologies rely on the entire data set of strong and weak bridging signals. A subset of our most confident rules-based bridges act as the training set for the machine learning algorithms. This creates manual and automated feedback mechanisms that help the Device Graph become more accurate over time. Bridge accuracy is the primary goal of Device Graph evolution. We do not look at downstream metrics such as media effectiveness as a direct measurement of graph quality. This is an important distinction because we see the use cases for the device graph extending past pure media opportunities, and, even within pure media, the breadth of use cases and quality measurements make it difficult to optimize for everyone.

Household Bridging

Household bridging builds a relationship between devices within a given household. When bridging household devices, Tapad makes no effort to identify separate individuals within the household. We primarily use rules based algorithms for associating devices within a household. Note, however, some platform types can be associated only with a household and not an individual. For example, a connected TV or gaming console will never receive an individual bridging score and can only have an edge with a household score.

We construct household bridges in a variety of ways. First, we'll take any strong signals of devices existing within a household to create high quality bridges. Next, we'll try to identify home networks types (see [network type](#) section) via a combination of existing data sets and machine learning. After this, we use a variety of rules based methods (based on device activity on these networks) to associate devices to that household. Finally all of that data is interpreted by a classification algorithm that predicts the likelihood of any pair of devices belong to the same household.

Individual Bridging

Tapad's individual bridging algorithms build on top of our household bridging capabilities by trying to identify devices that belong to an individual within a given household, but in a way that does not rely on privacy-intrusive methodologies. Instead, Tapad looks at non-PII audience data, rather than individual user data, to build a device profile. Based on the household similarity information and location of the devices, we train a similarity function that uses this audience data only. The similarity function is inspired by an [algorithm used to rank image query results](#).

Compared to other algorithms for similarity learning, Tapad's method relies on minimal assumptions and works well with noisy training data. Although it begins with the assumption that the similarity between devices on the same network should be greater than between devices outside the household, very quickly it starts to identify devices that belong to the same person. After the first iteration, we keep edges between devices in the same household that fit to our model with high confidence and reject the others that seem unrelated. For example, our algorithm assumes device A is more closely related to device B when they belong to the same household. Devices A and B are less likely to have a close relationship with device C when that device does not belong to the same household. In practice, this logic helps Tapad determine if device A is more likely to be used by (or belong to) the same person who uses device B.

Updating the Device Graph

The Device Graph changes constantly in real-time and from batch update processes. These updates are based on the strong and weak signals flowing into our systems. To keep the Device Graph data current, we remove devices and related edges from the graph after 14 days of inactivity if those devices and edges are not refreshed. If a removed device reappears in the graph after 14 days, we do not reuse previous data. In that case, device edges and strength scores get rebuilt from new activity data.

Bridging vs Merging

Most devices have multiple contexts represented by different unique identifiers. For example, a laptop may have multiple browsers that use distinct cookie spaces and mobile platforms have a hardware identifier that is not available in the context of that device's web browser. In the Device Graph, this means a single device can be represented by multiple nodes. These nodes are often bridged together, but sometimes we lack enough weak or strong signals to represent the device as a single node. As the Device Graph gathers data, we can determine if separate nodes are actually a single device. When this happens, the graph merges multiple nodes into a single node.

Blacklists

The Device Graph always makes a best effort to accurately depict the relationship between individuals, households, and their associated devices. However, as with any system that receives imperfect or limited information, occasionally the Device Graph makes some false inferences. Blacklisting helps prevent these inferences from creating large scale problems within the graph. For example, if a given node is bridged to too many other nodes, that node could be blacklisted. This prevents it from being bridged or merged with other nodes. Additionally, the blacklisted node is ignored from a targeting perspective. Eventually, blacklisted nodes get dropped from the Device Graph. Tapad data scientists analyze blacklisted devices on an ongoing basis. This helps improve the data sets and logic used by our systems.

Data Sources

Tapad uses IDs from the following data sources:

Type	Description
First-party Cookies	A unique ID stored in a first-party cookie that gets passed into the Device Graph. This ID is domain specific but allows publishers to bridge devices across their owned and operated inventory even in contexts where third-party cookies are disabled.
Third-party Cookies	Used for browser-based delivery. This is a standard web-based cookie for browsers that support third-party cookies by default <i>or</i> cookies that are more than 24-hours old.
Hardware IDs	Used for mobile in-app contexts. For example, IDs include IDFA or Android IDs in hashed or plain-text versions.

Tapad Is Not a Fingerprinting Technology

The intent of fingerprinting companies is to come up with a universal and persistent identifier to identify a device without cookies. This is not Tapad's intent. Our technology has been designed to avoid fingerprinting. Tapad's practices are substantially different from the associations of device fingerprinting. In particular, Tapad:

- **Is blind to personal information:** Tapad's proprietary technology is blind to all Personally Identifiable Information (PII), including names, email addresses, and phone numbers. We do not see personal data.
- **Is not invasive:** Tapad views only a handful of attributes that are visible to any entity serving content to the browser. Tapad does not leverage any advanced inspection techniques such as intrusive client-side detection through an SDK or JavaScript code that surreptitiously monitors the device or its traffic.
- **Is consistently transparent:** Tapad delivers proactive ad choice and opt-out notices whenever possible, including when we manage the media for a campaign. Tapad is committed to making its technology as visible as possible to the end-user.
- **Gives consumers control over persistence:** Tapad relies on device identifiers that give consumers maximum control and transparency. Additionally, we provide and full Device Graph opt-out functionality
- **Does not use alternative tracking or storage mechanisms:** Tapad does not use "zombie cookies" or history sniffing because these methods are harder for individuals to manage. Additionally, we not use local storage and Flash cookies for any other purpose other than to maintain a persistent opt-out.

Instead, Tapad infers device ownership through the Device Graph, which is built with the techniques and data sources described in this paper. Finally, Tapad is fully committed to consumer privacy. For details see the [opt-out section](#).

Network Type Identification

The network type and signals from those networks are important data sets for the Device Graph. Tapad identifies these networks from purchased data sets that identify ranges of IP addresses as home, business, education, airport, or hotel network types. We then try to improve this information by applying rules-based algorithms to get the most accurate data set possible for each network type. For example if a data set classifies an IP block as a household but shows hundreds of devices active on a daily basis, we may reclassify that block as a range of business address. This data analysis process is not used to identify an individual household or individual. It is one data point that helps build and strengthen the device graph.

Platform Identification

Every node in the device graph is associated with a platform type, which is used for analytics and targeting. Tapad identifies platforms through the user agent. However, the user agent is not used as a persistent platform identifier. Instead, it helps associate a node to a platform type. Tapad platform types include:

- Android
- AndroidTablet
- Blackberry
- Computer
- FeaturePhone
- iPad
- iPhone
- iPod
- Kindle
- KindleFire
- Palm
- PlayStation
- Symbian
- Television
- Wii
- WindowsPhone
- WindowsTablet
- XBox

Schema

The following JSON object represents a node and its related edges in the Device Graph.

```
{  
    "type" : "record",  
    "name" : "Device",  
    "namespace" : "com.tapad.avro",  
    "fields" : [ {  
        "name" : "is_opted_out",  
        "type" : [ "boolean", "null" ]  
    }, {  
        "name" : "is_blacklisted",  
        "type" : [ "boolean", "null" ]  
    }, {  
        "name" : "platform",  
        "type" : [ {  
            "type" : "enum",  
            "name" : "Platform",  
            "symbols" : [ "ANDROID", "IPHONE", "IPAD", "BLACKBERRY", "PALM", "WINDOWSPHONE",  
"FEATUREPHONE", "SYMBIAN", "COMPUTER", "ANDROIDTABLET", "PLAYSTATION", "WII",  
"TELEVISION", "IPOD", "XBOX", "KINDLE", "KINDLEFIRE", "WINDOWSTABLET", "OTHER" ]  
        }, "null" ]  
    }, {  
        "name" : "home_dma",  
        "type" : [ "int", "null" ]  
    }, {  
        "name" : "impression_history",  
        "type" : [ {  
            "type" : "record",  
            "name" : "ImpressionHistory",  
            "fields" : [ {  
                "name" : "campaign_impressions",  
                "type" : {  
                    "type" : "array",  
                    "itemType" : "string"  
                }  
            }  
        }  
    }  
}
```

[Expand source](#)

```

    "type" : "array",
    "items" : {
        "type" : "record",
        "name" : "Impression",
        "fields" : [ {
            "name" : "id",
            "type" : "string"
        }, {
            "name" : "count",
            "type" : "int"
        }, {
            "name" : "window_left",
            "type" : "int"
        }, {
            "name" : "window_size",
            "type" : "int"
        } ]
    }
},
"default" : [ ]
}, {
    "name" : "tactic_impressions",
    "type" : {
        "type" : "array",
        "items" : "Impression"
    },
    "default" : [ ]
}, {
    "name" : "tactic_impression_details",
    "type" : {
        "type" : "array",
        "items" : {
            "type" : "record",
            "name" : "ImpressionDetail",
            "fields" : [ {
                "name" : "id",
                "type" : "string"
            }, {
                "name" : "timestamp",
                "type" : "long"
            }, {
                "name" : "creative_id",
                "type" : [ "string", "null" ]
            }, {
                "name" : "placement_id",
                "type" : [ "string", "null" ]
            } ]
        }
    },
    "default" : [ ]
}, {
    "name" : "tactic_click_details",
    "type" : {
        "type" : "array",
        "items" : "ImpressionDetail"
    },
    "default" : [ ]
}
}, "null" ]

```

```

}, {
  "name" : "audiences",
  "type" : {
    "type" : "array",
    "items" : {
      "type" : "record",
      "name" : "Audience",
      "fields" : [ {
        "name" : "id",
        "type" : "string"
      }, {
        "name" : "fired",
        "type" : "boolean"
      }, {
        "name" : "distance",
        "type" : [ "int", "null" ]
      }, {
        "name" : "platform",
        "type" : [ "Platform", "null" ]
      }, {
        "name" : "attributes",
        "type" : {
          "type" : "map",
          "values" : "string"
        }
      }
    }
  },
  "default" : [ ]
}, {
  "name" : "ids",
  "type" : {
    "type" : "array",
    "items" : {
      "type" : "record",
      "name" : "Identifier",
      "fields" : [ {
        "name" : "type",
        "type" : "string"
      }, {
        "name" : "value",
        "type" : "string"
      }
    ]
  }
},
  "default" : [ ]
}, {
  "name" : "edges",
  "type" : {
    "type" : "array",
    "items" : {
      "type" : "record",
      "name" : "Edge",
      "fields" : [ {
        "name" : "id",
        "type" : "string"
      }, {
        "name" : "strength",
        "type" : "int"
      }
    ]
  }
}

```

```
}, {
  "name" : "timestamp",
  "type" : "long"
}, {
  "name" : "platform",
  "type" : [ "Platform", "null" ]
}, {
  "name" : "partner_ids",
  "type" : [ {
    "type" : "array",
    "items" : "string"
  }, "null" ],
  "default" : [ ]
} ]
},
},
```

```
        "default" : [ ]  
    } ]  
}
```

Opt-Out

Tapad is committed to protecting consumer privacy, providing choice, and opt-out mechanisms. Our privacy protections include:

- **Cross-platform notification:** Tapad displays the Digital Advertising Alliance AdChoices logo to users on computers, mobile devices, and tablets.
- **Opt-out technology:** Tapad has invested resources in mobile opt-out technologies and strategic relationships with top privacy compliance vendors such as Evidon and TRUSTe. In mobile apps, Tapad has partnered with both Evidon and TRUSTe to offer first-to-market mobile app opt-out solutions through hardware device identifiers.
- **Blind to personal information:** Tapad's proprietary Device Graph technology is blind to all Personally Identifiable Information (PII), including names, email addresses & phone numbers. We see no personal data.

See the Tapad [privacy page](#) for a full disclosure about our policies and opt-out capabilities.

Device Graph Data Structures

Definitions and Data Structures

Device =

1. **storage_key** = canonical ID
2. **Deviceld** (array) =
 - ▼ [Expand for Deviceld parameters](#)
 - a. **value** = ID of device
 - b. **Sighting** (map) =
 - i. **key** = ID Type

Note

All ID types prefixed with "HARDWARE" are hardware identifiers and can come either from an SDK/S2S or come from a bid request where the supplier has specified the exact ID type. When the ID comes from bid requests and is not put into a supplier-specific ID type, the supplier who sent the ID is stored in the "provider" field rather than in "ID type".

All ID types prefixed with "SUPPLIER" are received from a supplier in bid request. It comes from a field where the supplier has not specified the exact ID type (e.g., can be any one of these 3 types) so these fields have no specified type connotation and are not necessarily hardware identifiers.

ID Type	Description
TAPAD_COOKIE	Tapad's 3rd party cookie ID
TAPAD_FINGERPRINT	Tapad's device signature (used for analytics only)
THIRDPARTY	Another 3rd party provider's cookie ID. This ID type is used for all pixel and ID sync endpoints that are <u>unrelated to a specific supplier</u> . All supplier-specific endpoints have their own ID types (see below).
TAPAD_FIRST_PARTY_COOKIE	1st party cookie ID generated by Tapad's Javascript
HARDWARE_OPENUDID	OpenUDID (iOS only)
HARDWARE_MD5RAWMAC	Raw MAC address MD5 hashed (Android and iOS)
HARDWARE_SHA1RAWMAC	Raw MAC address SHA1 hashed (Android and iOS)
HARDWARE_MD5UDID	UDID MD5 hashed (iOS only)
HARDWARE_SHA1UDID	UDID SHA1 hashed (iOS only)
HARDWARE_MD5MAC	User-friendly MAC address MD5 hased (Android and iOS)
HARDWARE_SHA1MAC	User-friendly MAC address SHA1 hased (Android and iOS)
HARDWARE_ANDROIDID	Android ID clear text (Android only)
HARDWARE_MD5ANDROIDID	Android ID MD5 hashed (Android only)
HARDWARE_SHA1ANDROIDID	Android ID SHA1 hashed (Android only)
HARDWARE_MD5ANDROIDPHONEID	Android Phone ID MD5 hashed (Android only)
HARDWARE_SHA1ANDROIDPHONEID	Android Phone ID SHA1 hashed (Android only)
HARDWARE_IDFA	IDFA clear text (iOS only) - ID for advertisers
HARDWARE_SHA1IDFA	IDFA SHA1 hashed (iOS only) - ID for advertisers
HARDWARE_MD5IDFA	IDFA MD5 hashed (iOS only) - ID for advertisers
HARDWARE_IDFV	IDFV clear text (iOS only) - ID for vendors

HARDWARE_SECUREUDID	Secure UDID clear text (iOS only) - Yet another implementation of an open source udid prior to the advent of IDFA. Only sent to us from Pubmatic.
HARDWARE_SHA1WINDOWSUID	Windows DUID SHA1 hashed (Windows phones only)
SUPPLIER ADMARVEL	Type: query param Key: "UNIQUE_ID"
SUPPLIER ADMELD	Type: query param Key: "admeld_user_id"
SUPPLIER APPNEXUS	Type: json Path: "bid_request" / "bid_info" / "user_id_64"
SUPPLIER_BRIGHTROLL	Type: protobuf Field: proto.getUser.getId
SUPPLIER_BRIGHTROLLDIDMD5	Type: protobuf Field: proto.getDevice.getDidmd5
SUPPLIER_BRIGHTROLLDIDSHA1	Type: protobuf Field: proto.getDevice.getDidsha1
SUPPLIER_BRIGHTROLLDPIDMD5	Type: protobuf Field: proto.getDevice.getDpidmd5
SUPPLIER_BRIGHTROLLDPIDSHA1	Type: protobuf Field: proto.getDevice.getDpidsha1
SUPPLIER_BURSTLY	Type: query param Key: "user"
SUPPLIER GOOGLEADX	Type: protobuf Field: proto.getGoogleUserId
SUPPLIER LIVERAIL	Type: query param Key: "uid"
SUPPLIER MOBCLIX	Type: query param Key: "deviceld"
SUPPLIER_NEXAGEDID	Type: json Path: "device" / "did"
SUPPLIER_NEXAGEDPID	Type: json Path: "device" / "dpid"
SUPPLIER_NEXAGEDPIDMD5	Type: json Path: "device" / "dpidmd5"
SUPPLIER_NEXAGEUID	Type: json Path: "user" / "uid"
SUPPLIER_OPENRTB10DID	Type: json Path: "device" / "did"
SUPPLIER_OPENRTB10DPID	Type: json Path: "device" / "dpid"
SUPPLIER_OPENRTB10UID	Type: json Path: "user" / "uid"
SUPPLIER_OPENRTB20DID	Type: json Path: "device" / "did"
SUPPLIER_OPENRTB20DPID	Type: json Path: "device" / "dpid"
SUPPLIER_OPENRTB20DIDMD5	Type: json Path: "device" / "didmd5"
SUPPLIER_OPENRTB20DIDSHA1	Type: json Path: "device" / "didsha1"
SUPPLIER_OPENRTB20DPIDMD5	Type: json Path: "device" / "dpidmd5"
SUPPLIER_OPENRTB20DPIDSHA1	Type: json Path: "device" / "dpidsha1"
SUPPLIER_OPENRTB20ID	Type: json Path: "user" / "id"
SUPPLIER_PUBDIRECT	Type: query param Key: "did"
SUPPLIER_PUBMATICDID	Type: json Path: "mobile" / "device" / "did"
SUPPLIER_PUBMATICDPID	Type: json Path: "mobile" / "device" / "dpid" when "mobile" / "device" / "dpidtype" = 0 or not (1-8 inclusive)
SUPPLIER_PUBMATICUID	Type: query param Key: "uid"
SUPPLIER_SIMPLESTATICRTD	Type: query param Key: "uid" or "device_id"
SUPPLIER_SMAATO	Type: json Path: "user" / "id"
SUPPLIER_SPOTX	Type: query param Key: "spotx_user_id"

SUPPLIER_TRUSTE	Type: json Path: "user" / "ext" / "nex_data_truste" / "tpid" (this is from a nexage bid request)
SUPPLIER_NEXAGEDPIDSHA1	Type: json Path: "device" / "dpidsha1"
SUPPLIER_NEXAGEDIDMD5	Type json Path: "device" / "didmd5"
SUPPLIER_NEXAGEDIDSHA1	Type json Path: "device" / "didsha1"
SUPPLIER_RMX	Type xml Element: <bid:bidReq><bid:exchangeId></bid:exchangeId></bid:bidReq>

ii. Value

1. **ContextType** = In which environment the ID was seen in
 - [Full list of options](#)
 - APP
 - BROWSER
2. **provider** = Which partner sent us this ID (e.g., this ID came from BlueKai). This is an integer that is the primary id of the constraint_values table in mysql.
 - [Full list of options](#)

Provider Type	Provider ID	Description
Supplier	44	Admarvel
Supplier	45	Admeld
Supplier	47	Nexage RTB
Supplier	49	Pubmatic
Supplier	50	Where
Supplier	59	MoPub RTB
Supplier	60	mOcean
Supplier	61	LiveIntent
Supplier	63	Mobclix
Supplier	69	Swappit
Supplier	81	AppNexus
Supplier	118	Smaato
Supplier	148	Amobee
Supplier	170	Nexage RTD
Supplier	172	MoPub RTD
Supplier	206	SpotXchange
Supplier	208	BrightRoll
Supplier	210	NYTimes
Supplier	212	PANDORA
Supplier	214	Rubicon Project
Supplier	216	LiveRail
Supplier	237	Millennial
Supplier	241	InMobi
Supplier	243	Adelphic

Supplier	244	Tapjoy
Supplier	245	Todacell
Supplier	246	Jumptap
Supplier	248	Millennial All Caps
Supplier	353	RMX
DMP Partner	232	Admeld id sync
DMP Partner	233	Adobe id sync
DMP Partner	234	Lotame id sync
DMP Partner	247	LiveRamp
DMP Partner	349	BlueKai
DMP Partner	350	DataLogix
DMP Partner	351	Turn Dmp
DMP Partner	354	Rich Relevance
Tapestry Partner	311	Tapad
Tapestry Partner	312	Cobalt
Tapestry Partner	313	Ally Bank
Tapestry Partner	314	Neiman Marcus
Tapestry Partner	315	NY Times
Tapestry Partner	316	Nielsen
Tapestry Partner	317	FC2
Tapestry Partner	318	Tagman
Tapestry Partner	319	Edmunds
Tapestry Partner	320	Turn
Tapestry Partner	321	Lotame
Tapestry Partner	322	Turn Sample Adv ID 1
Tapestry Partner	323	Turn Sample Adv ID 2
Tapestry Partner	324	Turn Sample Adv ID 3
Tapestry Partner	325	Accuen Sample Adv ID1
Tapestry Partner	326	Accuen Sample Adv ID2
Tapestry Partner	327	Accuen Sample Adv ID3
Tapestry Partner	328	PlaceIQ
Tapestry Partner	329	Seamless
Tapestry Partner	330	AutoTrader
Tapestry Partner	331	LiveRamp
Tapestry Partner	332	Demo
Tapestry Partner	333	Demand Media
Tapestry Partner	334	Viacom

Tapestry Partner	335	Conde Nast
Tapestry Partner	336	Adometry
Tapestry Partner	337	BlogHer
Tapestry Partner	355	BlueKai
Tapestry Partner	356	Kodak
Analytics Partner	235	Yahoo Owned and Operated
Analytics Partner	236	Cobalt
Analytics Partner	238	HTC cross plan analytics
Analytics Partner	239	AOL
Analytics Partner	240	External
Analytics Partner	242	Somo Audience
Analytics Partner	352	Tapad Self

- 3. **firstSeen** = Timestamp of when this device ID from this provider was first seen
- 4. **lastSeen** = Timestamp of when this device ID from this provider was last seen (we don't store all timestamps, just the first and last seen)

3. **DeviceEdge** (array) =

 ▼ [Expand for DeviceEdge parameters](#)

- a. **storage_id** = canonical id of the connected device
- b. **strength** = strength score associated with edge
- c. **timestamp** = timestamp of last time the strength was updated
- d. **Platform** = platform of the connected device

 ▼ [Full list of options](#)

- ANDROID
- ANDROID_TABLET
- BLACKBERRY
- IPAD
- IPHONE
- IPOD
- WINDOWS_PHONE
- WINDOWS_TABLET
- PALM
- SYMBIAN
- FEATURE_PHONE
- PLAYSTATION
- XBOX
- WII
- KINDLE
- KINDLE_FIRE
- COMPUTER
- TELEVISION

- e. **partner_ids** (array) = IDs of partners that provided deterministic information to establish/reinforce this edge (this can come from the partner end point hosted on the bidders or through tapestry when tapestry partners provide an user id in addition to the device id)

4. **DeviceAudience** (array) =

 ▼ [Expand for DeviceAudience parameters](#)

- a. **id** = Audience ID (same as audience tracking tag)
- b. **is_direct_fire** (TRUE or FALSE) = did the audience originate on this device or was it propagated onto this device?
- c. **distance** = number of edges (hops) away from the closest device this audience was directly fired on (0 for is_direct_fire = TRUE)
- d. **Platform** = of this device (redundant?)

 ▼ [Full list of options](#)

- ANDROID
- ANDROID_TABLET
- BLACKBERRY
- IPAD
- IPHONE

- IPOD
- WINDOWS_PHONE
- WINDOWS_TABLET
- PALM
- SYMBIAN
- FEATURE_PHONE
- PLAYSTATION
- XBOX
- WII
- KINDLE
- KINDLE_FIRE
- COMPUTER
- TELEVISION

e. **attributes** = I don't know what this was used for. I don't think anyone still uses this. It should be empty for everything.

5. Platform of this device -

[▼ Full list of options](#)

- ANDROID
- ANDROID_TABLET
- BLACKBERRY
- IPAD
- IPHONE
- IPOD
- WINDOWS_PHONE
- WINDOWS_TABLET
- PALM
- SYMBIAN
- FEATURE_PHONE
- PLAYSTATION
- XBOX
- WII
- KINDLE
- KINDLE_FIRE
- COMPUTER
- TELEVISION

6. **is_opted_out** (TRUE or FALSE)

7. **is_blacklisted** (TRUE or FALSE) = devices can be blacklisted if they have too many edges associated with them (device graph rules)

8. **home_dma** = DMA that this device was last bridged in (with a min strength of 2)

9. **expires_at** = expiry timestamp of device

10. **ImpressionHistory** =

[▼ Expand for ImpressionHistory parameters](#)

- campaign_impressions** made up of **impression** (array) = the count of impressions for each campaign that has a campaign level frequency cap

[▼ Expand for impression parameters](#)

 - i. **id** = campaign ID
 - ii. **count** = how many time impressions happened in the window
 - iii. **window_left** = beginning of the frequency cap window as far as the count is concerned (i.e. when the first count occurred)
 - iv. **window_size** = length of the frequency cap window
- tactic_impressions** made up of **impression** (array) = the count of impressions for each tactic that has a tactic level frequency cap

[▼ Expand for impression parameters](#)

 - i. **id** = tactic ID
 - ii. **count** = how many time impressions happened in the window
 - iii. **window_left** = beginning of the frequency cap window as far as the count is concerned (i.e. when the first count occurred)
 - iv. **window_size** = length of the frequency cap window
- tactic_impression_details** made up of **impressionDetail** (array) = last impression for each tactic that had an impression for this device

[▼ Expand for impressionDetail parameters](#)

 - i. **id** = tactic ID
 - ii. **timestamp** = Timestamp of impression delivered
 - iii. **creative_id** = Creative ID associated with delivered impression
 - iv. **placement_id** = Placement ID associated with delivered impression
- tactic_click_details** made up of **impressionDetail** (array) = last click for each tactic that had an click for this device

[▼ Expand for impressionDetail parameters](#)

 - i. **id** = tactic ID

- ii. **timestamp** = Timestamp of click
- iii. **creative_id** = Creative ID associated with click
- iv. **placement_id** = Placement ID associated with click

Device Graph Supporting Avro

```
{
record DeviceIdSighting
{
  EventContextType context_type;
  string provider;
  long firstSeen;
  union { null, long } lastSeen = null;
}

record DeviceId
{
  string value;
  union { null, map<array<DeviceIdSighting>> } sightings_by_id_type = null;
}

record DeviceEdge
{
  string storage_id;
  int strength;
  long timestamp;
  union { null, Platform } platform;
  union { null, array<string> } partner_ids;
}

record DeviceAudience
{
  string id;
  boolean is_direct_fire;
  union { null, int } distance = null;
  union { null, Platform } platform = null;
  union { null, map<string> } attributes = null;
}

record Device
{
  string storage_key;
  array<DeviceId> ids = [];
  array<DeviceEdge> edges = [];
  array<DeviceAudience> audiences = [];
  union { null, Platform } platform = null;
  union { null, boolean } is_opted_out = null;
  union { null, boolean } is_blacklisted = null;
  union { null, int } home_dma = null;
  union { null, long } expires_at = null;
  union { null, ImpressionHistory } impression_history = null;
}

protocol PlatformInterface
{
  enum Platform
```

```

{
    ANDROID,
    ANDROID_TABLET,
    BLACKBERRY,
    IPAD,
    IPHONE,
    IPOD,
    WINDOWS_PHONE,
    WINDOWS_TABLET,
    PALM,
    SYMBIAN,
    FEATURE_PHONE,
    PLAYSTATION,
    XBOX,
    WII,
    KINDLE,
    KINDLE_FIRE,
    COMPUTER,
    TELEVISION
}
}

protocol ImpressionHistoryInterface
{
    record Impression
    {
        string id;
        int count;
        int window_left;
        int window_size;
    }

    record ImpressionDetail
    {
        string id;
        long timestamp;
        union { null, string } creative_id = null;
        union { null, string } placement_id = null;
    }

    record ImpressionHistory
    {
        array<Impression> campaign_impressions = [];
        array<Impression> tactic_impressions = [];
        array<ImpressionDetail> tactic_impression_details = [];
    }
}

```

```
        array<ImpressionDetail> tactic_click_details = [ ];  
    }  
}
```

SDK and Server-to-Server Data Collection

Overview and set up guides for collecting data with our software development kit (SDK) and server-to-server data transfers.

- [SDK Data Collection](#) — Overview and set up instructions for our standard and universal software development kits (SDK).
- [Server-to-server data collection](#) — Send in-app information to Tapad for cross platform analytics and user targeting. Useful if technical or other constraints prevent you from collecting data with an SDK.

SDK Data Collection

Overview and set up instructions for our standard and universal software development kits (SDK).

See the [server-to-server documentation](#) if policy or engineering restrictions prevent you from using an SDK to collect in-app data.

- **Standard SDK: About** — The standard SDK is a code module that lets publishers and media partners collect data from mobile apps installed on iOS or Android platforms and send it to Tapad for reporting, analysis, and cross-device targeting.
- **Universal SDK: About** — The Universal SDK helps publishers simplify code management and data collection from mobile apps installed on iOS or Android devices. Refer to this material for an overview of how the Universal SDK works, a comparison with the standard SDK, and setup requirements.
- **Set up the Android SDK** — Instructions that help you set up this SDK to capture data from mobile applications installed on Android devices.
- **Set up the iOS SDK** — Instructions that help you set up this SDK to capture data from mobile applications installed on iOS devices.
- **Set Up Impression and Click Tracking** — Follow these instructions to place standard 1x1 pixels on creates or your mobile site to return impression and click tracking data to Tapad. This is required for SDK data collection and cross plan analytics.
- **Endpoint Configuration File: Requirements** — An endpoint configuration file tells Tapad where to send your data and is required when you set up the Universal SDK. It contains a destination URL and syntax used to organize data in the URL string.

Standard SDK: About

DRAFT DRAFT

The standard SDK is a code module that lets publishers and media partners collect data from mobile apps installed on iOS or Android platforms and send it to Tapad for reporting, analysis, and cross-device targeting.

The Tapad SDK is designed for media partners that publish ads, optimize campaigns, and manage data collection on behalf of advertisers and other marketing clients. The SDK tracks standard in-app user actions like installations, clicks, purchases, or other customer-specific data points.

How it works

The standard SDK is identical to the Universal SDK, except it does not require an [endpoint configuration file](#). For details on how the Tapad SDKs work and how to set them up, see [Universal SDK: About](#).

Universal SDK: About

The Universal SDK helps publishers simplify code management and data collection from mobile apps installed on iOS or Android devices. Refer to this material for an overview of how the Universal SDK works, a comparison with the standard SDK, and setup requirements.

Purpose

The Universal SDK contains the same code as our standard SDK. It is designed to replace the other SDKs provided by your media partners. When used as a universal SDK, our code sends data to Tapad, where it gets processed and sent to your data partners.

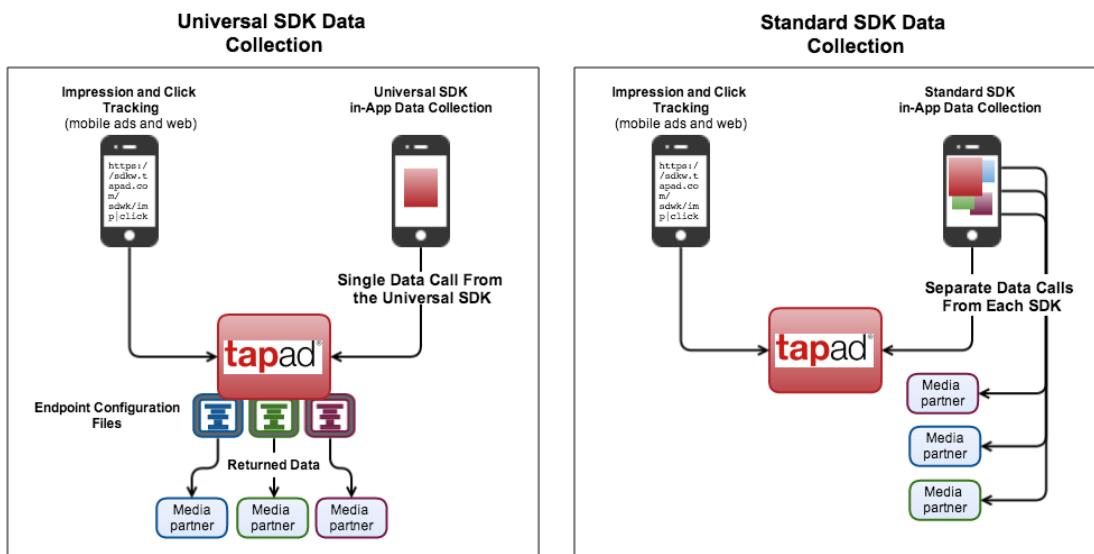
Advantages

The Universal SDK improves in-app data collection because it eliminates the need for multiple, partner-specific SDKs with a single code module. Replacing all your other SDKs with the Tapad Universal SDK provides several benefits that help:

- Preserve the user experience:** A single SDK reduces the burden on your app (and the user's mobile device) by eliminating the need for separate SDK data calls to each media partner. Tapad receives all the SDK data and sends it on to your media partners through a server-to-server data transfer.
- Reduce engineering and code management efforts:** One SDK is easier to deploy, test, and maintain than multiple SDKs from different developers.
- Shorten review and deployment times:** With the Universal SDK, Tapad manages the server-side relationships and data transfers to all your media partners. As a result, you don't have to have our code reviewed by the App Store or Android Marketplace each time you add a new partner and need to update the SDK.

How it works

With the Universal SDK, Tapad sends media tracking and in-app data to your media partners via server-to-server data transfers. This means the Universal SDK makes a single call to Tapad and doesn't burden your app with multiple calls to other destinations. The standard SDK works like any other SDK and makes its own data calls to Tapad. The following illustration describes this process:



Requirements and implementation summary

The following table outlines the process steps required to get started with the standard or Universal SDK.

Process	Implementation step	Description
Media tracking	Add Tapad impression and click trackers to your mobile ads and site.	Impression and click tracking code uses standard 1x1 pixels to record impressions and clicks and send that data to Tapad. See Set Up Impression and Click Tracking . Required when you want to use the standard or Universal SDK.

In-app tracking	Add the SDK code to your mobile app.	The Tapad SDKs collect data on installs, in-app user activity, and other actions you can specify in the SDK code. To implement our SDK code, see: <ul style="list-style-type: none"> • Set up the Android SDK • Set up the iOS SDK Required when you want to use the standard or Universal SDK.
Data transfers	Configure data transfers with media partners.	Tapad works with your media partners to set up server-to-server data transfers. An <i>endpoint configuration file</i> controls the details of these data transfers. Basically, it is a script that establishes the communication protocols between servers controlled by Tapad and your media parter. See Endpoint Configuration File: Requirements . Required for Universal SDK use only.

Set up the Android SDK

Instructions that help you set up this SDK to capture data from mobile applications installed on Android devices.

Purpose of the Android SDK

The Tapad software development kit (SDK) lets our systems and business partners collect data about user activity in mobile applications. The SDK collects data about typical activities such as installations, clicks, purchases, and conversion events. Additionally, you can customize the SDK to collect data about other types of in-app behavior specific to your campaign or business needs.

Android set up instructions and code samples

The following instructions will show you how to set up the SDK to collect in-app data from a mobile device.

Step 1: Download the Tapad Android SDK

Download the Android SDK from Github. Copy the SDK to your Android code.

Step 2: Configure identifiers

From within the Android SDK, open and edit the `AndroidManifest.xml` file and specify the identifiers you want to collect. The following table provides code for the available ID types.

ID type	Sample code
Phone ID	<code><uses-permission android:name="android.permission.READ_PHONE_STATE" /></code>
WiFi MAC address	<code><uses-permission android:name="android.permission.ACCESS_WIFI_STATE" /></code>
Android ID	<code><uses-permission android:name="android.permission.INTERNET" /></code>

Step 3: Specify Tapad credentials

Tapad credentials authenticate data calls to our system. From within the `<application...>` section of the `AndroidManifest.xml` file, replace `credentials_from_Tapad` with your Tapad authentication. Your code could look similar to this:

```
<meta-data android:name="tapad.APP_ID" android:value="credentials_from_Tapad"/>
```

Step 4: Enable identifiers

From within the `<application...>` section of the `AndroidManifest.xml` file, enable ID collection. Remember, specify the value for each ID type you want to collect. For example, to collect all 3 available ID types (phone, WiFi, Android), your code should look similar to this:

```
<meta-data android:name="tapad.ID_SOURCES" android:value="PhoneId, WiFiMac, INTERNET" />
```

Step 5: Add install and event code tracking

The follow table provides code for installation and in-app event data collection.

Tracking type	Add code
Installation	To your start up activity, <code>onCreate()</code> : as shown: <code>com.tapad.tracking.Tracking.init(this);</code>

Events	After the event happens as shown: com.tapad.tracking.Tracking.get().onEvent("some_event"); Replace some_event with the name of the event you want to track.
--------	---

Step 6: (Optional) Enable Google Play referral tracking

To enable referral tracking from Google Play (optional, but recommended), add the following code to the <application...> section of the AndroidManifest.xml file:

```
<application...>
  ...
    <receiver android:name="com.tapad.tracking.InstallReferrerReceiver"
    android:exported="true">
      <intent-filter>
        <action android:name="com.android.vending.INSTALL_REFERRER" />
      </intent-filter>
    </receiver>
</application>
```

Set up the iOS SDK

Instructions that help you set up this SDK to capture data from mobile applications installed on iOS devices.

Purpose of the iOS SDK

The Tapad software development kit (SDK) lets our systems and business partners collect data about user activity in mobile applications. The SDK collects data about typical activities such as installations, clicks, purchases, and conversion events. Additionally, you can customize the SDK to collect data about other types of in-app behavior specific to your campaign or business needs.

iOS set up instructions and code samples

The following instructions will show you how to set up the SDK to collect in-app data from a mobile device.

Step 1: Download the Tapad iOS SDK

Download the iOS SDK from Github. Copy the SDK to your iOS code.

Step 2: Configure identifiers

From within the iOS SDK, open the file `MyEnabledTapadIdentifiers.h` and uncomment the device identifiers that you want to collect. The following lists all the identifiers in this file. At a minimum you should enable the indicated device IDs.

Minimum recommended device IDs	Other available device IDs
<ul style="list-style-type: none">• <code>#define TAPAD_IDENTIFIER_ENABLE_OPENUDID</code>• <code>#define TAPAD_IDENTIFIER_ENABLE_SHA1_HASHED_RAW_MAC</code>• <code>#define TAPAD_IDENTIFIER_ENABLE_SHA1_HASHED_MAC</code>• <code>#define TAPAD_IDENTIFIER_ENABLE_ADVERTISING_IDENTIFIER</code>	<ul style="list-style-type: none">• <code>#define TAPAD_IDENTIFIER_ENABLE_MD5_HASHED_RAW_MAC</code>• <code>#define TAPAD_IDENTIFIER_ENABLE_MD5_HASHED_MAC</code>

Step 3: Specify Tapad credentials

Tapad credentials authenticate data calls to our system. Set your credentials within the `(void) applicationDidFinishLaunching: (UIApplication *)`; function as shown in the following example:

```
[TapadEvent registerAppWithId:@"credentials_from_tapad"];
```

Step 4: Enable identifiers

From within the `(void) applicationDidFinishLaunching: (UIApplication *)`; function, uncomment and add YES to collect IDs selected in Step 2 above. Your code could look similar to this:

```
//Capture open UDID
[TapadIdentifiers sendOpenUDID:YES];

//Capture SHA1 Hashed RAW MAC
[TapadIdentifiers sendSHA1HashedRawMAC:YES];

//Capture SHA1 Hashed MAC
[TapadIdentifiers sendSHA1HashedMAC:YES];

//Capture Advertising ID
[TapadIdentifiers sendAdvertisingIdentifier:YES];
```

Step 5: Add install and event code tracking

To collect installation and in-app event data modify `(void)applicationDidFinishLaunching: (UIApplication *)` of the main application.

Tracking type	Add Code
Installations	<code>[TapadEvent applicationDidFinishLaunching:application];</code>
Events	<code>[TapadEvent send:some-event];</code> In this case, <code>some-event</code> is the name of the in-app event you want to track.

Set Up Impression and Click Tracking

DRAFT

Follow these instructions to place standard 1x1 pixels on creates or your mobile site to return impression and click tracking data to Tapad. This is required for SDK data collection and cross plan analytics.

Impression tracking code

A standard impression tracking call could look similar to the example below. Note, bracketed content like <some_parameter> should be replaced with your own or Tapad provided variables.

```
https://sdkw.tapad.com/sdkw/click?did=<device_ID>&ctx=<site_ID>&tp=<site_type>&ua<user_agent>&ip=<client_IP_address>&cid=<tapad_variable>&tid=<tapad_variable>&crid=<tapad_variable>&ssp=<tapad_variable>&ts=<cachebuster>
```

Click tracking code

A standard click tracking or redirect call could look similar to the example below. Note, bracketed content like <some_parameter> should be replaced with your own or Tapad provided variables.

```
https://sdkw.tapad.com/sdkw/click?did=<device_ID>&ctx=<site_ID>&tp=<site_type>&ua<user_agent>&ip=<client_IP_address>&cid=<tapad_variable>&tid=<tapad_variable>&crid=<tapad_variable>&ssp=<tapad_variable>&ts=<cachebuster>
```

Impression and click tracking code parameters defined

Parameter	Description
did	Device identifier (e.g., OpenUDID, ODIN1, SHA1, Android ID, etc.).
ctx	Your site ID.
tp	Identifies your property as a "site" or "app." Specify: <ul style="list-style-type: none">• tp="S" to flag your property as a site.• tp="A" to flag your property as an app.
ua	User agent (browser type).
ip	IP address used by the application.
cid	Variables provided by Tapad.
tid	
crid	
ssp	
cachebuster	Standard cache busting code such as a random number or date/time stamp.
redirect	Encoded URL used as the click redirect.

Endpoint Configuration File: Requirements

An endpoint configuration file tells Tapad where to send your data and is required when you set up the Universal SDK. It contains a destination URL and syntax used to organize data in the URL string.

Transferred data contains *all* in-app events, not just those from your specific media. For example, if you receive 100 installation notifications, only 50 may be attributable to you.

Sample configuration file

The syntax, URL, and parameters in your configuration file can vary according to how you want to receive data from Tapad. A sample configuration file could look similar to the following:

```
http://your_domain_url.com/some_directory?variable_x=<data_point_1>&variable_y=<data_point_2>
```

Server-to-server data collection

Send in-app information to Tapad for cross platform analytics and user targeting. Useful if technical or other constraints prevent you from collecting data with an SDK.

- [Server-to-Server Data Transfers: Technical Specifications](#) — Technical, engineering, or development teams should use this material to help set up server-to-server (S2S) data transfers with Tapad.
- [Device ID Collection with typed_device_id](#) — In a server-to-server (S2S) data transfer, the typed_device_id parameter sends iOS and Android IDs to Tapad.
- [Android and iOS Device IDs](#) — Lists the IDs and platforms accepted by the <device_id> parameter.

Server-to-Server Data Transfers: Technical Specifications

Technical, engineering, or development teams should use this material to help set up server-to-server (S2S) data transfers with Tapad.

Purpose

Sometimes, policy or engineering restrictions prevent you from collecting in-app data with our Universal Software Development Kit (SDK). As an alternative, you can set up and send in-app event information to Tapad with our S2S data transfer process. S2S data transfers are useful when:

- You can't use the Universal SDK.
- Immediate data transfers are not required.
- You want to reduce data discrepancies and extra, in-app data calls.

Set up S2S data transfers

To set up S2S data transfers with Tapad, you must:

- Get a unique URL from Tapad.
- Add code to call the URL from your app or server when a user triggers an event.

URL syntax

In an S2S data transfer, a properly formatted URL should look like the example below. Tapad supports HTTP and HTTPS data transfers. Replace the bracketed code <variable> with the actual parameter.

Sample URL
<code>https://analytics.tapad.com/app/event?app_id=<app_id>&action_id=<action_id>&device_id=<device_id>&typed_device_id=<typed_device_id>&platform=<device_id>&ip_address=<ip_address>&user_agent=<user_agent></code>

URL parameters

The URL for your inbound ID synchronization call should contain variables described in the table below. All parameters are required unless indicated otherwise.

Parameter	Description
app_id	A unique ID assigned by Tapad.
action_id	A Tapad defined event ID (e.g., install, login, purchase, etc.).
device_id	A device ID created by the advertiser. Optional when passing in typed_device_id.
typed_device_id	Multiple device IDs created by the advertiser. Optional when passing in device_id.
ip_address	User's IP address recorded during the event.
user_agent	(Optional) User agent recorded during the event. This is required if the User-Agent header of the server request does not reflect the user agent of the client.
platform	(Optional) Same as user_agent, kept for backward compatibility.

Event code

An event call is code you create and put in an app or invoke from a server-side call. To enable event tracking, instantiate this code after a specific user event. Remember, an "event" can be any user action within an app. You can make this call asynchronously, but it's best to make an event call within a few minutes of the qualifying activity.

Device ID Collection with typed_device_id

In a server-to-server (S2S) data transfer, the `typed_device_id` parameter sends iOS and Android IDs to Tapad.

typed_device_id: Syntax and example

S2S data transfers capture device IDs and values related to the ID. These IDs are useful because they help you improve conversion tracking accuracy. Instantiate the `typed_device_id` variable to return iOS or Android IDs and corresponding values. When you use this variable, the `<device_id>` parameter accepts a numeric [Android or iOS device ID](#). The `typed_device_id` variable accepts data formatted as key-value pairs:

Syntax

```
typed_device_id=<device_id>:<value>,<device_id>:<value>
```

A properly formed variable could look similar to the following code:

Example

```
typed_device_id=3:value_A,6:value_B,7:value_C
```

In this case, the `<device_id>` parameter passes in these device identifiers:

- **ID 3:** UDID with MD5 hash
- **ID 6:** WIFI MAC address (human readable) with SHA1 hash
- **ID 7:** iOS 6 + IDFA

For a complete list of Android and iOS IDs, see [Android and iOS Device IDs](#).

Parameter order and missing values

When you instantiate `typed_device_id=<device_id>:<value>`

- Device ID order does not matter. For example, returning `3:value_A, 6:value_B` is the same as `6:value_B, 3:value_A`.
- Exclude unavailable identifiers or indicate a missing identifier with 0. For example, `3:value_A, 6:0` is the same as `3:value_A`.

ID recommendations for iOS devices

UDID is deprecated. As an alternative, we recommend that you always collect the following iOS device identifiers:

- ID 2, WIFI MAC address in byte form with SHA1 hash
- ID 6, WIFI MAC address (human readable) with SHA1 hash
- ID 7, iOS 6 + IDFA

Android and iOS Device IDs

Lists the IDs and platforms accepted by the <device_id> parameter.

For more information about how to use the <device_id> parameter, see [Device ID Collection with typed_device_id](#).

iOS IDs and platforms

These Tapad IDs correspond to iOS identifiers.

Because UDID is deprecated, we recommend that you collect the following iOS device identifiers:

- ID 2, WIFI MAC address in byte form with SHA1 hash
- ID 6, WIFI MAC address (human readable) with SHA1 hash
- ID 7, iOS 6 + IDFA

ID	Platform
0	OpenUDID
1	WIFI MAC address in byte form with MD5 hash
2	WIFI MAC address in byte form with SHA1 hash
3	UDID with MD5 hash
4	UDID with SHA1 hash
5	WIFI MAC address (human readable) with MD5 hash
6	WIFI MAC address (human readable) with SHA1 hash
7	iOS 6 + IDFA
10	IDFV

Android IDs and platforms

These Tapad IDs correspond to Android identifiers.

IDs	Platform
101	Android ID with SHA1
102	Android ID with MD5
103	Android phone ID with SHA1
104	Android phone ID with MD5
105	WIFI MAC address with SHA1 hash
106	WIFI MAC address with MD5 hash
107	Google Advertising ID

Reference

General information, terms, useful facts, tables, and other data.

- [Device IDs](#) — A list of cookie and hardware IDs.
- [Glossary](#) — Commonly used terms and concepts defined.
- [IAB Real-time Bidding Specifications](#) — Open RTB Specifications v2.1 in PDF format.
- [JIRA - Confluence Linking](#) — Instructions for setting up links between JIRA and Confluence. Includes examples from both systems.
- [Key-Value Pairs: About](#) — A key-value pair (or key-value data), consists of individual data elements organized into pairs. The key is a constant that does not change. It defines a category of data such as name, color, or model. The value is a variable that can change in each key-value set. For example, sample key-value pair could contain data such as name:my_creative, color:blue, or model:new.
- [Mobile Device Bid Request](#) — Includes sample bid request, definitions, and the IAB specifications for mobile, real-time bidding.
- [Publisher ID: How to find it](#) — Each publisher in our network has a unique numeric ID. Enter this ID in the blacklist or whitelist targeting field to exclude or include a specific publisher in your targeting.
- [Request a Deal ID](#) — A Deal ID is a universal identifier that links private exchange inventory to specific buyers through programmatic real time bidding. This ID is carried into every bid request a private exchange transmits to DSPs and indicates when a private deal impression is available. For instance, a Deal ID can grant access to differentiated or unique placements such as first-look inventory or more premium inventory that is not available on the open exchange. When creating a Deal ID, buyers negotiate directly.
- [Tapad Rate Cards](#) — Prices and descriptions for the various ad placement options available from Tapad.
- [Understanding User Agents](#) — A user agent is an HTTP header sent by your browser to a web server. A basic user agent, or user-agent string, contains tokens that provide details about the application you're using, browser type, operating system, and version information for those variables. User agents can include language information (optional) and additional details about browser enhancements, add-ons, or the application making an HTTP request.
- [Whitelisting Steps and Site/App Evaluation Guide](#) — A whitelist is a list of websites and apps that have been reviewed and approved for ad serving. Sites and apps must be approved because Tapad will not respond to bid requests or serve ads from unreviewed or blacklisted publisher inventory. The whitelisting process involves researching the publisher's website or app, evaluating those objects according to our whitelisting criteria, and applying those results in Backoffice to the Display Context settings for those sites and apps.

Device IDs

A list of cookie and hardware IDs.

Cookie IDs

Tapad and the Tapestry APIs and SDKs identify cookies as shown in the following table.

Cookie type	Description
TAPAD_COOKIE	The Tapad cookie ID.
TAPAD_FIRST_PARTY_COOKIE	The ID of a first-party cookie set by Tapad.
Third-party cookie abbreviation	A third-party cookie ID. As a Tapad partner, this can be your first-party cookie. However, it gets identified as a third-party cookie in our system. In customer reports, Tapad uses a 2-3 character abbreviation to identify cookies that belong to a specific partner. For example, a cookie and ID from the BigDataCompany would look like this: BDC : 1234.

Hardware IDs

Tapad and the Tapestry APIs and SDKs identify devices as shown in the following table.

Device ID	Description
HARDWARE_ANDROIDID	An Android ID generated by the Android OS. Our valid Android IDs are 16 character which is 64 bit hex string. We don't accept any dashes with Android IDs.
HARDWARE_IDFA	An iOS IDFA ID. This is an identifier that is consistent/persistent across all apps.
HARDWARE_IDFV	An iOS ID type. This is an identifier that is different for each app. It lets you track users, but does let you correlate activities on one app against activities on another app.
HARDWARE_MD5ANDROIDID	An MD5 hashed Android ID. The Android ID is generated by the Android OS.
HARDWARE_MD5ANDROIDPHONEID	An MD5 hashed Android ID associated with the device's radio chip.
HARDWARE_MD5IDFA	An iOS IDFA ID with an MD5 hash.
HARDWARE_MD5MAC	An MD5 hash applied to the hexadecimal values of a MAC address associated with the device's ethernet NIC or WiFi chip.
HARDWARE_MD5RAWMAC	An MD5 hash applied to the binary values of a MAC address associated with the device's ethernet NIC or WiFi chip.
HARDWARE_MD5UDID	An iOS UDID with an MD5 hash.
HARDWARE_OPENUDID	An open UDID ID.
HARDWARE_SECUREDID	An iOS ID type.
HARDWARE_SHA1ANDROIDID	A SHA1 hashed Android ID. The Android ID is generated by the Android OS.
HARDWARE_SHA1ANDROIDPHONEID	A SHA1 hashed Android ID associated with the device's radio chip.
HARDWARE_SHA1IDFA	An iOS IDFA ID with a SHA1 hash.
HARDWARE_SHA1MAC	A SHA1 hash applied to the hexadecimal values of a MAC address associated with the device's ethernet NIC or WiFi chip.
HARDWARE_SHA1RAWMAC	A SHA1 hash applied to the binary values of a MAC address associated with the device's ethernet NIC or WiFi chip.
HARDWARE_SHA1UDID	A UDID with a SHA1 hash.
HARDWARE_SHA1WINDOWSUID	A Windows ID with a SHA1 hash.

Deprecated device IDs

These are obsolete, but still recognized by Tapad for legacy purposes.

Device ID	Device type
anid	Android ID
cookie	Cookie ID Sometimes cookies are identified by a 2-3 character abbreviation of the partner name the cookie belongs to.
fpc	First-party cookie
idfa	IDFA
md5anid	MD5 Android ID
md5idfa	IDFA MD5
md5mac	MD5 MAC
md5phid	MD5 Android phone ID
md5rmac	MD5 RAW MAC
md5udid	MD5 UDID
oudid	Open UDID
sh1idfa	IDFA SHA1
shalanid	SHA1 Android ID
sh1mac	SHA1 MAC
sh1phid	SHA1 Android phone ID
sh1rmac	SHA1 RAW MAC
sh1udid	SHA1 UDID
tpc	Tapad cookie

Glossary

Commonly used terms and concepts defined.

10 Terms

4 Terms

16 Terms

16 Terms

2 Terms

5 Terms

4 Terms

4 Terms

8 Terms

J

1 Term

4 Terms

4 Terms

3 Terms

4 Terms

10 Terms

9 Terms

7 Terms

6 Terms

4 Terms

1 Term

...

1 Term

X

Y

...

1 Term

+ Add a new letter

A

A	B	C	D	E	F	G	H	I	J	K	Add term
L	M	N	O	P	Q	R	S	T	U	V	
W	X	Y	Z								

Ad Analytics

Tools and metrics used to measure the effectiveness of advertising campaigns and activities.

Ad Exchange

A unified platform where advertisers, publishers and networks communicate to exchange data, set prices, and serve ads.

Ad Metrics

Measurements that provide data about ad performance.

Ad Servers

Systems that deliver ads to websites, smartphones, tablets, and other digital devices.

Ad Choices

Refers to the Evidon ad choices program (and logo), which helps consumers understand how they're being tracked on the Internet and gives them the ability to opt-out of online tracking. Companies that participate in this program display the triangular AdChoices logo on their online ads. Clicking the icon shows users information about an ad, why they received it, and how to opt-out of future tracking. See also:

- [The Tapad User Opt-out Process: About](#)
- [Customize the Evidon AdChoices Logo and Placement](#)

Algorithm

A process or set of rules followed to perform calculations or other problem solving operations.

Attribution

Credit given to an advertisement, website, or device for a successful conversion.

API

Application Programming Interface. APIs are a set of programming instructions and standards that let developers access your a Web-based software application or tools. Companies provide APIs to their customers so that other software developers can design products that are powered by its service. Tapad's publicly available API includes [Tapestry](#) and [TAJ](#).

Average Connections to Hardware IDs

Average number of connections from every primary ID to a hardware ID.

Average Connections to Cookies

Average number of connections from every primary ID to another primary ID.

B

A	B	C	D	E	F	G	H	I	J	K	Add term
L	M	N	O	P	Q	R	S	T	U	V	
W	X	Y	Z								

Banner Ads

Rectangular (horizontal or vertical) online ads. Width and height are usually measured in pixels. Some common types of banner ads include:

- Full banners (468x60)
- Leaderboards (728x90)
- Medium rectangle (300x250)
- Skyscraper (120x600)

See also, [web banner](#).

Bidder

Software that responds to opportunities to place an ad on a website or app in real time.

Bridging

Typically, bridging is a process that joins separate networks or network segments together. Tapad uses bridging to identify and connect devices (or nodes) on a network to help create the Device Graph. Variant terms include *bridge*, *device bridge*, or *network bridge*.

Behavioral Targeting

Online advertising delivered to consumers based on their current and past Web activity.

C

A	B	C	D	E	F	G	H	I	J	K	Add term
L	M	N	O	P	Q	R	S	T	U	V	
W	X	Y	Z								

Cookies

Small snippets of data stored in your computer's browser.

CMS

Campaign Management System.

CPA

Cost per action or acquisition is a formula lets you determine the costs for an creative action or conversion. The basic CPA equation is expressed as CPA Price = Revenue / Actions. See also [CPA: Cost per Action/Acquisition](#).

CPC

Cost per click is a formula lets you determine how much it costs to serve an ad per click. The basic CPC equation is expressed as CPC Price = Revenue / Clicks. See also [Cost per Click](#).

Conversion Rates

The % people who see an ad and take action (click, purchase, sign-up, opt-in/out, etc.) on it.

Confidence Score

See [Household Score](#).

CPI

Cost per Install is a formula that lets you calculate the cost of each app installation. The basic CPI equation is expressed as CPI = (Installs / Impressions) x 100.

CPM

Cost per Mill (or 1000 impressions) is a formula that lets you determine how much it costs to serve 1000 impressions. The basic CPM equation is expressed as CPM = Revenue / (Impressions / 1000).

CTR

Clickthrough rate is a formula that lets you calculate the number of users who click on a creative as a %. The basic CTR equation is expressed as CTR = (Clicks / Impressions) x 100.

Contextual Targeting

The practice of buying online advertising based on where a particular ad appears, rather than on the consumer's online behavior.

Connectivity Percentage

The % of IDs that connected or bridged to the total number of IDs sent in for matching/synchronization. This is expressed as connected primary IDs / Total Number of IDs.

Derivatives: *Percent Connected*

Conversion Rates

The % people who see an ad and take action (purchase, sign-up, opt-in/out, etc.) on it.

Cross Device

Refers to unified advertising across multiple devices (PCs, smartphones, tablets, game consoles, connected TVs.).

Also, *cross-platform*.

Connections to Hardware

The number of [primary IDs](#) that have 1 or more connection to a hardware ID.

Derivatives: *Cookie to hardware connections*

Connections to Cookies

The number of primary IDs that have 1 or more connection to another cookie ID.

Derivatives: *Cookie to cookie connections*

Client ID

An ID used to track a user activity on a partner site.

D

A	B	C	D	E	F	G	H	I	J	K
L	M	N	O	P	Q	R	S	T	U	V
W	X	Y	Z							

Add term

Data Management Platform

A Data Management Platform (DMP) is simply a large database of consumer data that's useful to online marketers. Usually, it contains first and third-party data about consumers' online behavior. Typically, DMPs help marketers store this data, create targetable audience segments, and provide reporting features that help clients monitor and manage segment performance. Access is usually provided through a web portal that lets DMP users manage their account. Some major DMPs include Adobe, Lotame, and BlueKai.

Demand Side Platform

A Demand Side Platform (or DSP) is a company or service that buys digital ad space in real time.

Data Provider

Businesses that provide user data for targeting.

dCPM

Dynamic Cost Per Mill (or dCPM) is a pricing model that ties bid price to impression value. This is different than CPM pricing because that model bids the same amount for every impression regardless of its value. With dCPM, the billing amount can vary along with the bidding amount (if you've set up a secondary goal). dCPM pricing enhances budget and campaign efficiency because it gives your campaigns the ability to selectively pick and bid on impressions based on their worth. See [dCPM Pricing: About](#).

Demand

Customers (usually advertisers) that want to buy ad space.

Demographic Targeting

The practice of serving ads to consumers based on such factors as age, sex, location, and ethnicity. See [Demographic Targeting](#).

Depth

In the Device Graph, *depth* (sometimes called *reach*) indicates how far each device in the graph is from the root device. The root device is the first device discovered by the Device Graph during the [bridging process](#). Depth has a range from 0 to 2. See [Understanding Depth](#).

Derived ID

Consists of the device IP address and browser [user agent](#). Tapad uses the derived ID to perform attribution for devices on wireless home networks with Safari browsers that do not accept third-party cookies.

Device Graph

Tapad's patent-pending, proprietary technology that helps marketers reach consumers and understand their behavior across devices while providing consumers with clear notice, persistent opt-out tools and privacy controls.

Device ID

IDs unique to each specific user device that can be used for targeting. See also, [Device ID Types](#).

DAA

Digital Advertising Alliance. An [online advertising association](#) that advocates industry self-regulation.

DMA

Designated Marketing Area (or *District Metered Areas*) Defined geographic regions of people who can receive the same content. It is also commonly specified in audience measurement and marketing tools. Applies to the US only. See [Designated Marketing Areas \(DMAs\)](#).

Do Not Track

Features that let web users to opt out of all cookie tracking. See [Ad Choices](#).

DSP

Demand Side Platform. A company that buys digital ad space in real time.

Dataflow

The movement of data in a given direction (e.g., from Tapad to a client or from a client to Tapad). See [Data Flow: Send Audiences To Tapad](#) and [Data Flow: Receive Connected IDs From Tapad](#).

DGA

Device Graph Access. A product that enables customers to license and use Tapad's Device Graph Data.

E

A	B	C	D	E	F	G	H	I	J	K	Add term
L	M	N	O	P	Q	R	S	T	U	V	
W	X	Y	Z								

eCPM

Estimated cost per 1000 impressions.

Edge

Represents an abstract connection between devices (or nodes) in the device graph. They consist of mathematical scores that define the closeness of the relationship among devices in the Device Graph. Edges are not physical links.

F

A	B	C	D	E	F	G	H	I	J	K
L	M	N	O	P	Q	R	S	T	U	V
W	X	Y	Z							

Add term

Fingerprinting

Describes techniques and/or companies that collect data for the purposes of identifying individuals. Fingerprinting technologies use:

- IDs to identify individual users.
- Invasive device inspections to help ensure identifier uniqueness.
- IDs that never change and that the user cannot control.

Generally, fingerprinting methodologies are undetectable and unaccountable to the individual user.

Tapad is not a fingerprinting company and does not use these techniques.

Flash Cookies

Small files that store data in your computer's browser. Flash cookies get created by websites that contain or use Adobe Flash.

Fragmentation

The breakdown of media audiences into smaller and smaller segments. The proliferation of media options such as Web and mobile display technologies are often cited as key sources of audience fragmentation.

Flat File

A file that doesn't have a structured format but contains relevant data.

Flight Date

The start and end dates for a campaign, line item, or tactic. Because these objects are organized in a hierarchy, dates set at a higher level take precedence over dates set at a lower level.

See also, [Budget Caps and Flight Dates for Campaigns, Line Items, and Tactics](#).

G

A	B	C	D	E	F	G	H	I	J	K	Add term
L	M	N	O	P	Q	R	S	T	U	V	
W	X	Y	Z								

Geofencing

A technique that delivers mobile messages to consumers based on how close they are to a physical store where they can purchase the advertised product or service. See [Geo Points Targeting](#).

Geotargeting

A technique that delivers ads to consumers based on geographic location. See [Time and Location Targeting](#).

GTM

Google Tag Manager. A service and technology that lets you add or update your website tags and mobile applications automatically.

GPS

Global Positioning System. Technology that helps advertisers locate consumers when they view an ad, message, or other online content.

H

A	B	C	D	E	F	G	H	I	J	K
L	M	N	O	P	Q	R	S	T	U	V
W	X	Y	Z							

Add term

Hardware ID

See [Device IDs](#).

History Sniffing

A practice that lets companies track an Internet user's web browsing history on an ongoing basis without the use of cookies. The latest versions of the most common browsers use technologies that help prevent this activity.

HTML 5 Storage

A mechanism that lets websites store data in the user's browser. Tapad's TAJ API [cache functions](#) give you the option to use local storage.

Household Score

Household scores measure the strength of the relationship between nodes (devices) in the Device Graph. Higher scores suggest a closer connection between devices in a household than lower scores. Keep in mind, however, that a household score is abstract representation of different data points. It is not a physical link between devices. Household scores work together with [Individual Scores](#) to help make accurate, cross-device targeting possible.

Derivatives: *Strength Score* (obsolete), *Confidence Score*.

I

A	B	C	D	E	F	G	H	I	J	K
L	M	N	O	P	Q	R	S	T	U	V
W	X	Y	Z							

Add term**IAB**

Interactive Advertising Bureau. See [About the IAB](#).

Imp

Shorthand for *impressions*, which is the number of times an ad has been served.

In App

Inventory in a mobile application.

In App Opt-out

The ability to opt-out of ads within an app. See [Ad choices](#).

IO

Insertion order. The contract between a marketer and a brand that specifies ad buy details.

IP Address

Internet Protocol address. A sets of numeric values that identify computers on a network. Typical IP addresses look like these: 172.16.254.1 or 10.10.10.10.

ID Mapping File

A file that maps/shows the relationship between primary IDs and related IDs. See [Understanding the ID Mapping File](#).

Individual Scores

Individual scores measure the strength of the relationship between devices (nodes) in the Device Graph. Higher scores suggest a closer connection between devices in a household than lower scores. Keep in mind, however, that a household score is abstract representation of different data points. It is not a physical link between devices. Individual scores work together with [Household Scores](#) to help make accurate, cross-device targeting possible.

Derivatives: *Strength Score* (obsolete), *Confidence Score*.

K

A	B	C	D	E	F	G	H	I	J	K	Add term
L	M	N	O	P	Q	R	S	T	U	V	
W	X	Y	Z								

Key-Value Pairs

Or name-value pairs. These are data elements organized into pairs that consist of:

- A key (or constant) that does not change. The key defines the category of data such as name or model.
- A value (or variable) that can change. Given the previous keys, sample key-value pairs could consist of data such as "name": "Test creative" and "model": "iPhone".
- Delimiters that consist of special characters that join/create a key-value pair. For example, this key-value pair is joined by a colon like this: "name": "Test creative".
- Separators that consist of special characters that separate one key-value pair from another key-value pair. For example, these key-value pairs are separated by a comma like this: "name": "Test creative", "model": "iPhone".

L

A	B	C	D	E	F	G	H	I	J	K	Add term
L	M	N	O	P	Q	R	S	T	U	V	
W	X	Y	Z								

Landing Page

The page a user is directed to after clicking an ad.

Location Targeting

See [Geotargeting](#).

Lookalike Modeling

A technique that helps build larger audiences from smaller audience segments. Audiences included in a lookalike model share similar characteristics of the original audience segment. Lookalike modeling helps advertisers expand reach, but sacrifice accuracy as segment size increases.

Lift

The % increase in performance due to advertising or marketing efforts. See [Surveys: About](#).

M

A	B	C	D	E	F	G	H	I	J	K	Add term
L	M	N	O	P	Q	R	S	T	U	V	
W	X	Y	Z								

MAC Address

Media Access Control. A unique ID that computer chip manufacturers assign to network interface cards during fabrication.

Media Buying

Online ad buying by ad agencies (or other entities) usually on behalf of brands, or directly by brands themselves.

Mobile Rich Media

Digital ads that contains video or interactive components. See [Expandable and Mobile Rich Media Ads](#) and [Technical Specifications for Creatives](#).

Merge Logic

A set of rules that evaluates IDs assigned to devices. These rules help determine if devices should be merged, separated, or updated with a new or additional ID. See [Merge Logic](#).

N

A	B	C	D	E	F	G	H	I	J	K	Add term
L	M	N	O	P	Q	R	S	T	U	V	
W	X	Y	Z								

Native Advertising

Paid advertising designed to blend in with editorial content.

NIC

Network Interface Card. A computer component that lets you connect to the Internet.

Node

A device (desktop, laptop, tablet, phone, etc.) in the Device Graph.

O

A	B	C	D	E	F	G	H	I	J	K	Add term
L	M	N	O	P	Q	R	S	T	U	V	
W	X	Y	Z								

Opt-Out

Tools that provide comprehensive opt-out coverage. Users who employ opt-out tools or who opt-out will not see ads targeted to their interests. See also [The Tapad User Opt-out Process: About](#).

OBA

Online Behavioral Advertising. See Behavioral Targeting.

Opt-in

A process that delivers specific ads to consumers who have asked to receive them.

Opt-out

A process that gives users the choice not to receive specific ads. See also [The Tapad User Opt-out Process: About](#).

P

A	B	C	D	E	F	G	H	I	J	K	Add term
L	M	N	O	P	Q	R	S	T	U	V	
W	X	Y	Z								

Post Click

A way to measure consumer behavior after they click an ad. Also used to link a conversion to a particular site.

Performance Advertising

A form of advertising based on campaign performance. The brand pays based on results, usually on a CPC basis.

PII

Personally Identifiable Information. Data that can be used to identify and target specific individuals.

Pixel

A shortened word for *picture element*. A pixel is a single point in an image or line of code, which can be used by a website or third party to assign online activities to a computer or browser. For example, pixels can record if a user has visited a particular web page or clicked on a particular ad.

Platforms

Devices used to deliver content (e.g., iPhone, Android, Computer, Connected TV, etc.).

Privacy

The ability of individuals to control the sharing of personal information. See also, the [Tapad Privacy Center](#).

Programmatic Buying

Automated media buying in real-time.

Post View

A way to measure consumer behavior after they view an ad. Also used to link a conversion to a particular site.

Primary ID

IDs sent in by partners for synchronization with related IDs in the Device Graph. See [Basic Text Data File](#).

Alternates: *Synched IDs, Direct Fires*

Antonyms: [Related ID](#)

Partner Cookie ID

A cookie ID used to track a user activity on a partner site.

R

A	B	C	D	E	F	G	H	I	J	K	Add term
L	M	N	O	P	Q	R	S	T	U	V	
W	X	Y	Z								

Responsive Design

Websites that adapt to multiple screens (computers, mobile phones, tablets, etc.).

Retargeting

Serving ads across the web and other platforms to consumers who have previously visited an advertiser's website.

RFI

Request for information or request for invoice.

RFP

Request for proposal.

Rich Media

Ads that contain video, audio, and interactive elements. See [Technical Specifications for Creatives](#).

RTB

Real Time Bidding. The process of buying and selling online advertising in real-time. During real-time bidding, advertisers compete for ad impressions and highest bidder wins.

Related ID

Device IDs that are bridged or linked to a Primary ID. See [Basic Text Data File](#).

Derivatives: *Connected IDs, Indirect Fires, Propagated IDs.*

Antonyms: *Primary ID*

Related Cookie IDs

Number of cookie IDs that are connected to a primary cookie ID.

Related Hardware IDs

Number of hardware IDs that are connected to a primary cookie ID.

S

A	B	C	D	E	F	G	H	I	J	K	Add term
L	M	N	O	P	Q	R	S	T	U	V	
W	X	Y	Z								

Strength Score

See [Household Scores](#).

Server-to-Server

A data transfer methodology that transfers data in real time or on a set schedule between servers. See [Server-to-server data collection](#). Often abbreviated as S2S.

SDK

Software Development Kit. Self-contained code that helps publishers and media partners collect data from mobile apps installed on iOS or Android platforms. See [SDK Data Collection](#).

Silos

The practice of brands and ad agencies creating and buying ads for one medium only, usually in isolation from other media outlets or platforms.

SSP

Supply-side Platform. A company or service that provides ad space to demand-side platforms in real time. SSPs often aggregate supply from publishers.

Supply

Publisher inventory or ad slots available for serving ads on web pages.

Supplier IDs

IDs received from an ID sync from a supplier.

T

A	B	C	D	E	F	G	H	I	J	K
L	M	N	O	P	Q	R	S	T	U	V
W	X	Y	Z							

Add term

Targeting

Delivering specific content to a selected audience.

Text Ads

Ads that don't contain images, video, or interactive features.

Third-party Cookies

Cookies placed on a user's computer by another site or service other than the website currently being visited. See [Cookies](#).

Time Stamp

A record of the date and time of a particular action or event. Examples include:

- UNIX timestamps (the number of seconds since 1 January 1970)
- Variations of the typical *mm-dd-yyyy hh:mm:ss* format.

Tokens

Identifiers contained in the User-agent string that provide specific details about the program making the request. Tokens vary among programs; for example, the tokens in the Internet Explorer User-agent string describe the browser, the operating system, and the current browser mode. See [User Agent](#).

Transparency

The practice of informing consumers about the methods used to deliver ads to them. See [Ad Choices](#).

U

A	B	C	D	E	F	G	H	I	J	K
L	M	N	O	P	Q	R	S	T	U	V
W	X	Y	Z							

Add term

User Agent

Identification data sent by your browser to a web server. A basic user agent, or user-agent string, contains information such as browser type and version, operating system information, and other data such as tokens, which provide specific details about the program making the request (see [Tokens](#)). A sample user agent string looks similar to the following:

```
User-Agent: Mozilla/5.0 (Windows NT 6.3; Trident/7.0; rv:11.0) like Gecko
```

For more information, see [Understanding User Agents](#).

UI

User Interface. It's junction between a user and a computer program.

Unique Device Identifier

A 40-character string assigned to certain devices and stored in a web or mobile cookie. Each ID character is a numeral or a letter of the alphabet and can be used to monitor subscriber behavior. Abbreviated as UDID.

User Data

Information about users based on behavior, demography, income, age, gender, etc. User data is usually associated with a *unique user ID* (UUID).

V

A	B	C	D	E	F	G	H	I	J	K	Add term
L	M	N	O	P	Q	R	S	T	U	V	
W	X	Y	Z								

Video Ads

Ads that contain video. See [Technical Specifications for Creatives](#).

W

A	B	C	D	E	F	G	H	I	J	K	Add term
L	M	N	O	P	Q	R	S	T	U	V	
W	X	Y	Z								

WAP

Wireless Application Protocol. A set of standards and web browsers for mobile devices.

Z

A	B	C	D	E	F	G	H	I	J	K	Add term
L	M	N	O	P	Q	R	S	T	U	V	
W	X	Y	Z								

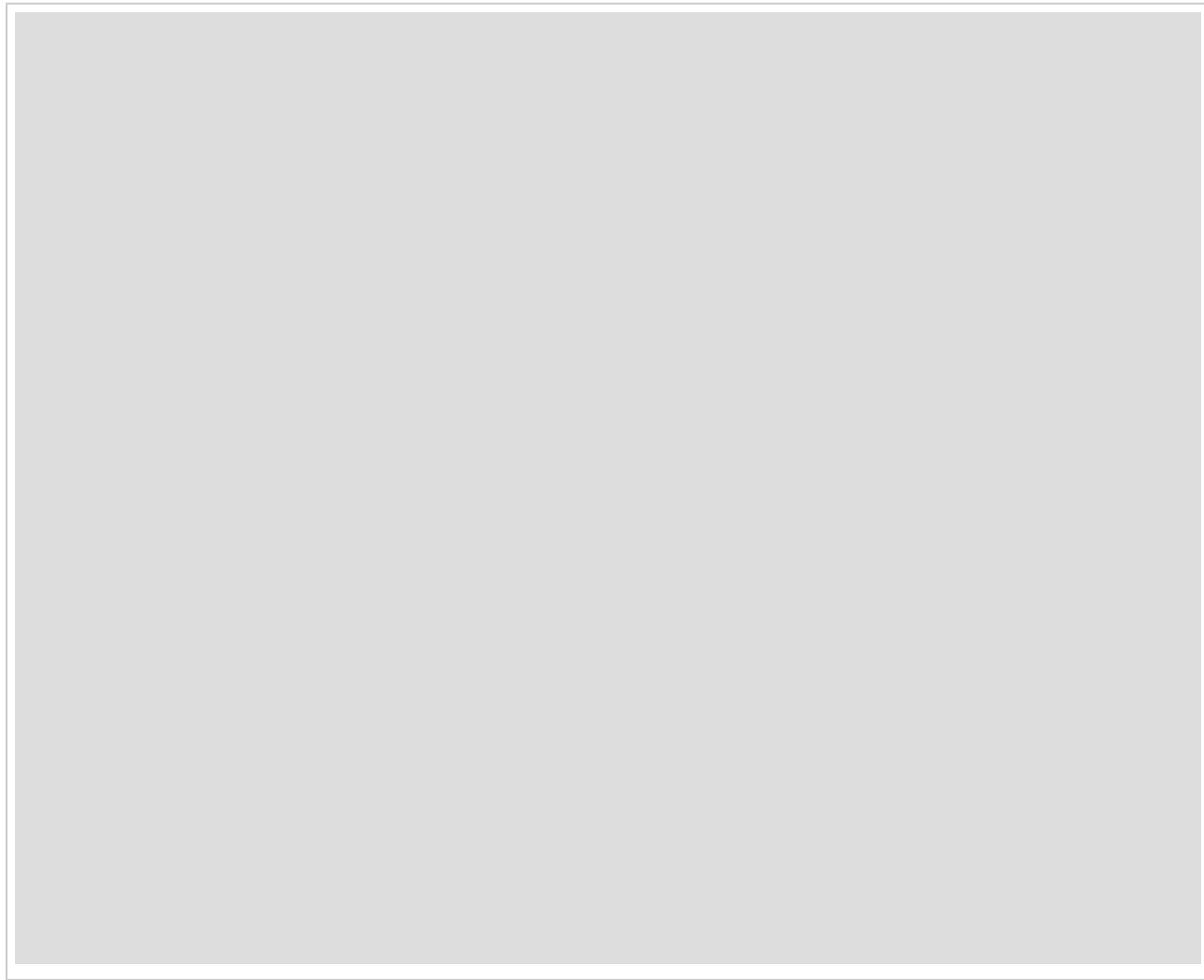
Zombie Cookies

Cookies that return or persist even after being deleted.

IAB Real-time Bidding Specifications

Open RTB Specifications v2.1 in PDF format.

[View](#) or [download](#) the PDF.



JIRA - Confluence Linking

Instructions for setting up links between JIRA and Confluence. Includes examples from both systems.

- [Add Confluence Page Links to a JIRA Issue](#) — Follow these instructions to link one or more Confluence pages to a JIRA issue.
- [Add JIRA Issues to a Confluence Page](#) — Follow these instructions to add one or more JIRA issues to a Confluence Page.
- [Linked JIRA Issues: Examples](#) — Samples of linked JIRA issues in Confluence.

Add Confluence Page Links to a JIRA Issue

Follow these instructions to link one or more Confluence pages to a JIRA issue.

To add a Confluence page link to a JIRA ticket:

1. Open the JIRA ticket you want to work with.
2. Click **More** and then click **Link**.
3. Click **Confluence Page** and enter the page URL in the "Page URL" field. If you do not know the Confluence URL, click **search for a page** and use the dynamic search feature to find and add it.
4. Click **Link**. A link to your selected Confluence page appears in the "Issue Links" section of your JIRA ticket.

Your Confluence link should look similar to the circled example below:

The screenshot shows a JIRA ticket for 'Documentation / DOC-55' titled 'Document attribution reporting process'. The ticket has standard action buttons: Edit, Comment, Assign, More, Start Progress, Resolve Issue, and Workflow. Below these are two sections: 'Description' and 'Issue Links'. The 'Description' section contains the text: 'Similar to conversion reporting, we need docs around how users set up and run attribution reports.' The 'Issue Links' section contains a single link: 'Wiki Page' followed by '[JIRA Linking Test Page](#)'. A red oval highlights the 'JIRA Linking Test Page' link.

Add JIRA Issues to a Confluence Page

Follow these instructions to add one or more JIRA issues to a Confluence Page.

Add a single JIRA issue

To add a single JIRA issue to a Confluence page:

1. From your JIRA issue, click the **Export** menu and choose **XML**.
2. Copy the URL that appears in the comment block at the top of the page. In the code block, the URL appears before the closing comment tag as shown:

```
<!--  
RSS generated by JIRA (6.0-rc1#6085-sha1:c75e7300f4c262d9e27cea8c81b31271cb2eb431) at Mon  
Sep 09 21:12:07 UTC 2013  
It is possible to restrict the fields that are returned in this document by specifying the 'field' parameter in  
your request.  
For example, to request only the issue key and summary add field=key&field=summary to the URL of  
your request.  
For example:  
https://jira.tapad.com/jira/sr/jira.issueviews:searchrequest-xml/12101/SearchRequest-12101.xml?tempMax=1000&field=key&field=summary  
-->
```

3. From your Confluence page, open the **macro browser** and select **Other macros**.
4. Type "JIRA" in the search field and click the JIRA macro when it appears. This automatically places the JIRA macro on your Confluence page.
5. Click the macro placeholder and click **Edit**.
6. Put the URL for the JIRA ticket in the "Issues URL" field and click **Save**.

To customize the fields returned by the JIRA macro, see the Parameters section of the [JIRA Issues Macro documentation](#). For examples, see the [Linked JIRA Issues: Examples](#).

Add multiple JIRA issues

To add multiple JIRA issues to a Confluence page:

Part 1: Create a new JIRA filter

1. From your project workspace select **All issues**.
2. Click **New filter** in the upper left corner.
3. Choose options from the **Project**, **Type**, **Status**, and **Assignee** menus as required.
4. Click **Save as**, name your filter, and click **Submit**.

Part 2: Find and copy the issue URL

1. From your saved filter, click the **Export** menu and click **XML**.
2. Copy the URL from comment block at the top of the page. In the code block, the URL appears before the closing comment tag as shown:

```
<!--  
RSS generated by JIRA (6.0-rc1#6085-sha1:c75e7300f4c262d9e27cea8c81b31271cb2eb431) at Mon  
Sep 09 21:12:07 UTC 2013  
It is possible to restrict the fields that are returned in this document by specifying the 'field' parameter in  
your request.  
For example, to request only the issue key and summary add field=key&field=summary to the URL of  
your request.  
For example:  
https://jira.tapad.com/jira/sr/jira.issueviews:searchrequest-xml/12101/SearchRequest-12101.xml?tempMax=1000&field=key&field=summary  
-->
```

3. From your Confluence page, open the **macro browser** and choose **Other macros**.
4. Type "JIRA" in the search field and click the JIRA macro when it appears. This automatically places the JIRA macro on your Confluence page.

5. Click the macro placeholder and click **Edit**.
6. Put the URL for the JIRA ticket in the "Issues URL" field and click **Save**.

To customize the fields returned by the JIRA macro, see the Parameters section of the [JIRA Issues Macro documentation](#). For examples, see the [Linked JIRA Issues: Examples](#).

Linked JIRA Issues: Examples

Samples of linked JIRA issues in Confluence.

To view the macro settings:

1. Click **Edit**.
2. Click the macro placeholder and click **Edit**.
3. Review the macro settings and click **Cancel** to return to the page.
4. Click **Cancel** to return the page to read-only mode.

Examples:

- Individual JIRA Issue
- Issues summary (default view)
- Custom views
 - Example 1
 - Example 2

Individual JIRA Issue

 **DOC-55** - Document attribution reporting process CLOSED

Issues summary (default view)

JIRA Issues Macro: The URL filter is not available to you, perhaps it has been deleted or had its permissions changed

Custom views

Example 1

Test link customized to show type, key, and summary only. Edit the macro to show only those columns you want to see.

JIRA Issues Macro: The URL filter is not available to you, perhaps it has been deleted or had its permissions changed

Example 2

Customized to show issue count only. Edit the macro and select the **Show issues count** checkbox.

JIRA Issues Macro: The URL filter is not available to you, perhaps it has been deleted or had its permissions changed

Key-Value Pairs: About

A key-value pair (or key-value data), consists of individual data elements organized into pairs. The *key* is a constant that does not change. It defines a category of data such as name, color, or model. The *value* is a variable that can change in each key-value set. For example, sample key-value pair could contain data such as name:my_creative, color:blue, or model:new.

Mobile Device Bid Request

Includes sample bid request, definitions, and the IAB specifications for mobile, real-time bidding.

Request example

A mobile device bid request contains a [JSON object](#) with individual data elements that consist of:

- A key (or constant) that does not change. The key defines the category of data such as name or model.
- A value (or variable) that can change on each bid request. Given the previous keys, a bid request could contain key-value pairs such as "name": "Test creative" and "model": "iPhone".

A sample bid request could look similar to the following. For more details about other bid request object types and variables, see the [OpenRTB specs](#).

Bid request code

[Expand source](#)

```
//Start JSON object
{
    "imp": [
        {
            "h": 250,
            "w": 300,
            "battr": ["9", "10", "12"],
            "instl": 0,
            "impid": "agltb3B1Yi1pbmNyDAsSA0FwcBiLo_8DDA",
            "api": 3
        }
    ],
    "app": {
        "name": "Test App",
        "pid": "agltb3B1Yi1pbmNyCwsSA0FwcBiowXkM",
        "pub": "Test Pub",
        "cat": ["Games", "Education"],
        "paid": 0,
        "global_aid": "agltb3B1Yi1pbmNyCwsSA0FwcBiowXkM",
        "aid": "agltb3B1Yi1pbmNyCwsSA0FwcBiowXkM"
    },
    "pf": 0.05,
    "at": 2,
    "restrictions": {
        "badv": [],
        "bcat": ["IAB25", "IAB7-39", "IAB8-5", "IAB8-18", "IAB9-9", "IAB14-1"]
    },
    "device": {
        "os": "iPhone OS",
        "ip": "98.116.12.94",
        "js": 1,
        "dpid": "agltb3B1Yi1pbmNyDAsSA0FwcBiLo_8DDA",
        "osv": "4.2.1",
        "loc": "37.7,-122.4",
        "country": "USA",
        "make": "Apple",
        "carrier": "Wi-Fi",
        "model": "iPhone",
        "ua": "Mozilla/5.0 (iPhone; U; CPU iPhone OS 4_2_1 like Mac OS X; en-us) AppleWebKit/532.9 (KHTML, like Gecko) Version/4.0.5 Mobile/8A293 Safari/6531.22.7"
    },
    "tmax": 200,
    "id": "agltb3B1Yi1pbmNyDAsSA0FwcBiLo_8DDA"
},
"user": {
    "uid": "agltb3B1Yi1pbmNyDAsSA0FwcBiLo_8DDA",
    "zip": "94110",
    "gender": "F",
    "yob": 1990,
    "country": "USA",
    "keywords": "m_age:22,startups"
}
}
//End JSON object
```

Common bid request variables defined

The following table defines some of the common keys found in a typical mobile bid request. For additional details see the [OpenRTB specs](#) below.

Variable	Description
api	A list of supported video APIs. IDs and and video specs include: <ul style="list-style-type: none"> • ID 1, VPAID 1.0 • ID 2, VPAID 2.0 • ID 3, MRAID • ID 4, ORMMA See Table 6.4 in the OpenRTB specs .
app	The <code>app</code> object is included if content is part of a mobile app. The <code>app</code> object and its parameters are optional.
at	Auction type: <ul style="list-style-type: none"> • 1: First price auction • 2: Second price auction
badv	<i>Blocked advertiser domain.</i> An array of blocked, top-level advertiser domains (e.g., <code>companyA.com</code> , <code>companyB.com</code>).
battr	<i>Blocked creative attributes.</i> IDs and descriptions for creative attributes (e.g., "audio ad," "expandable," "surveys", etc.). An empty field indicates all creative types are allowed. See Table 6.3 in the OpenRTB specs .
bcat	<i>Blocked advertiser categories.</i> The OpenRTB specs do not define or classify advertiser categories. However, IAB content categories can be used instead. See Table 6.1 in the OpenRTB specs .
carrier	The carrier or ISP as specified by the IP address.
cat	An array of IAB content categories. See Table 6.1 in the OpenRTB specs .
country	3-letter country codes as specified by ISO 3166-1 alpha-3 .
device	The <code>device</code> object contains device information such as hardware, platform, location, and carrier. The <code>device</code> object and its parameters are optional.
dpid	Platform specific ID (e.g., Android ID for Android or IDFA for iOS). Hashed variants include: <ul style="list-style-type: none"> • <code>dipdsha1</code> (preferred) • <code>dpidmd5</code> (case insensitive)
gender	<ul style="list-style-type: none"> • M: Male • F: Female • O: Other • Null: Unknown
h	Impression height or player height in pixels.
id	Unique ID for the object. The OpenRTB specs contain different object-specific <code>id</code> parameters. See the OpenRTB specs .
imp	The <code>imp</code> object contains information about the ad position or impression.
impid	The ID of the impression object for a bid.
instl	<ul style="list-style-type: none"> • 0: Not an interstitial or full screen ad. • 1: Interstitial or full screen ad.
ip	IP address of the device.
js	<ul style="list-style-type: none"> • 0: The device does not support JavaScript. • 1: The device does support JavaScript.
keywords	A list of keywords for the object. The OpenRTB specs contain different object-specific <code>keywords</code> parameters. See the OpenRTB specs .

<code>loc</code>	Latitude and longitude coordinates.
<code>make</code>	Device manufacturer (e.g., Apple).
<code>model</code>	Device model.
<code>name</code>	A site, application, publisher, etc. name. The OpenRTB specs contain different object-specific <code>name</code> parameters. See the OpenRTB B specs .
<code>os</code>	Device operating system.
<code>osv</code>	Device operating system versions.
<code>paid</code>	<ul style="list-style-type: none"> • 0: Free version of an application. • 1: Paid version of an application.
<code>pf</code>	Price floor.
<code>pub</code>	Publisher name.
<code>tmax</code>	Timeout interval in which to make a bid request.
<code>ua</code>	See user agent .
<code>uid</code>	User ID.
<code>w</code>	Impression width or player width in pixels.
<code>yob</code>	<i>Year of birth</i> . Specified as 4-digit integer.
<code>zip</code>	ZIP code or other postal code.

Related concepts

[Understanding User Agents](#)

Publisher ID: How to find it

Each publisher in our network has a unique numeric ID. Enter this ID in the blacklist or whitelist targeting field to exclude or include a specific publisher in your targeting.

To find the publisher ID:

1. Click the suppliers tab.
2. Click the name of your supply-side partner in the Supplier list.
3. Type a search term in **Search** field. Results display a list of publisher IDs and names based on your search terms.

A search returns results based on a publisher URL, friendly name, or description. Click the publisher name to view these settings.

Related concepts:

- [Whitelist and Blacklist](#)

Request a Deal ID

A Deal ID is a universal identifier that links private exchange inventory to specific buyers through programmatic real time bidding. This ID is carried into every bid request a private exchange transmits to DSPs and indicates when a private deal impression is available. For instance, a Deal ID can grant access to differentiated or unique placements such as first-look inventory or more premium inventory that is not available on the open exchange. When creating a Deal ID, buyers negotiate directly with sellers to strike deals. Negotiations cover topics such as pricing, available inventory, specific publishers and advertisers, and budgets. Upon reaching an agreement, the publisher or SSP creates a Deal ID. The DSP can then target this ID on a bid request. Review this material to request a Deal ID.

To request a deal ID:

1. Identify and list your campaign goals (see examples below).
2. Collect this material and email it to Sashi Ramani. Sashi will contact our suppliers and negotiate terms.
3. The supplier creates a deal ID and sets a price floor amount. Sashi will provide you with this information. You can target the Deal ID in *Tactics > Targeting > Private Marketplace Deals*. See [Private Marketplace Deals](#).

Pro tips

When targeting by deal ID

- Vendor fees apply so make sure the tactic bid price is => the price floor.
- Check the creative to ensure it clicks through to the correct landing page and that the landing page accurately reflects the creative content.

Campaign goal information

The following list contains some of the common campaign items you should try to include in your email to Sashi. This goal information will vary depending on campaign type and other factors. Try to be as thorough as possible when requesting a Deal ID. In this case, more really is more.

- Campaign type
- Flight dates
- Budget
- Verticals, white lists, black lists
- Platform types (e.g., desktop, mobile, tablet)
- Mobile device type (e.g., tablet, IOS, Android, all devices, etc.)
- Creative type
- Size
- Rich Media type
- Video
- Any due dates or other time constraints

Tapad Rate Cards

Prices and descriptions for the various ad placement options available from Tapad.

Pro tip

If you need creative specs, see [Creative Specification PDFs](#).

- [Tapad Rate Card I Q3 2014](#) — Ad placement rates for North America.
- [Tapad Rate Card I Q2 2014](#) — Ad placement rates for North America.
- [Tapad Rate Card I Q1 2014](#) — Ad placement rates for North America.
- [Tapad Rate Card I Q4 2013](#) — Ad placement rates for North America.

Tapad Rate Card I Q3 2014

Ad placement rates for North America.

Pro tip

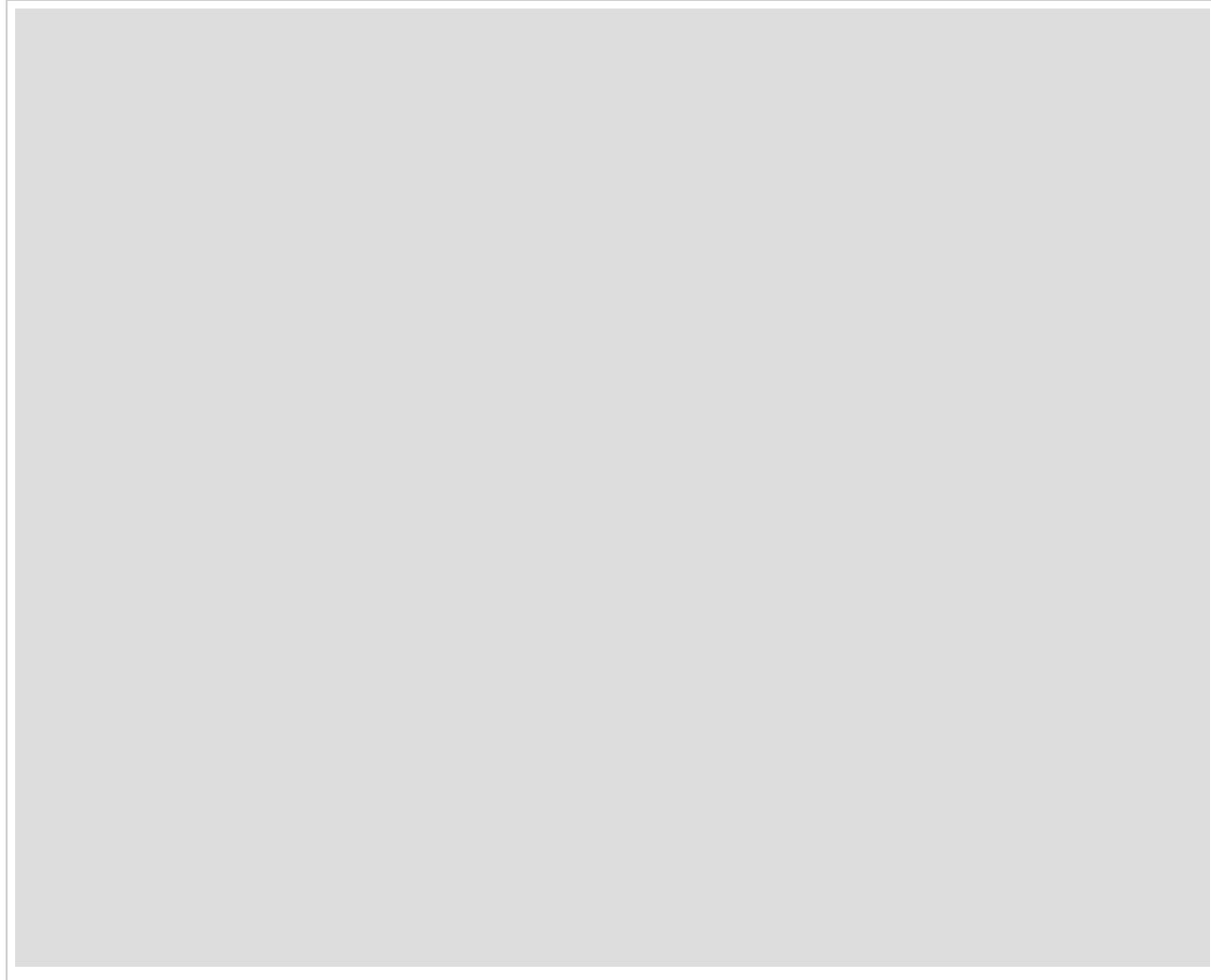
If you need creative specs, see Creative Specification PDFs.

Contents

- 0 to 5MM avails
- 5MM to 15MM avails
- < 15MM avails
- Rate card by audience

To download, click in the PDF frame. [Click here](#) to download the full Excel file if you need it.

0 to 5MM avails



5MM to 15MM avails



< 15MM avails



Rate card by audience



Tapad Rate Card I Q2 2014

Ad placement rates for North America.

Pro tip

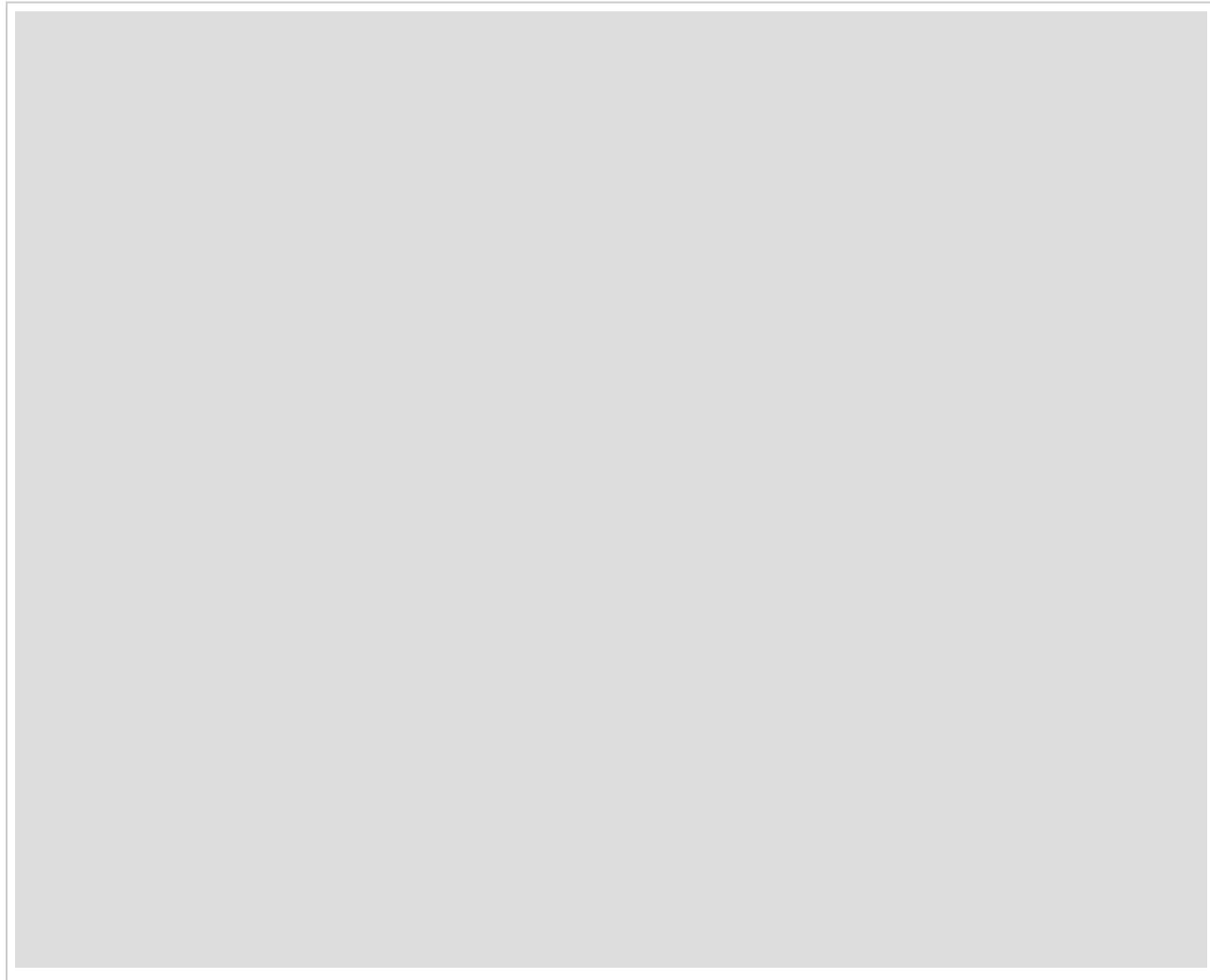
If you need creative specs, see Creative Specification PDFs.

Contents

- 0 to 5MM avails
- 5MM to 15MM avails
- < 15MM avails
- Rate card by audience

To download, click in the PDF frame. [Click here](#) to download the full Excel file if you need it.

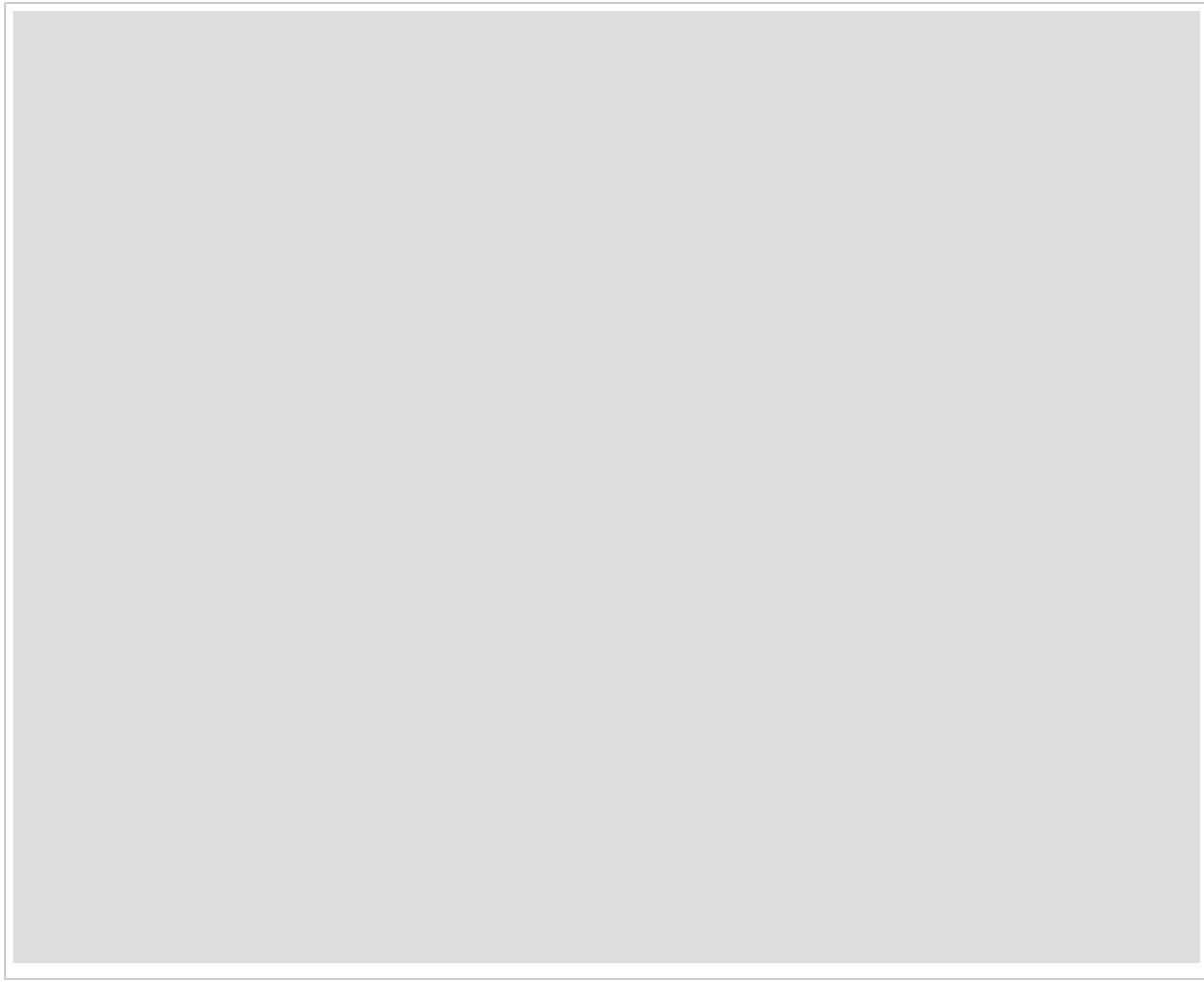
0 to 5MM avails



5MM to 15MM avails



< 15MM avails



Rate card by audience



Tapad Rate Card I Q1 2014

Ad placement rates for North America.

Pro tip

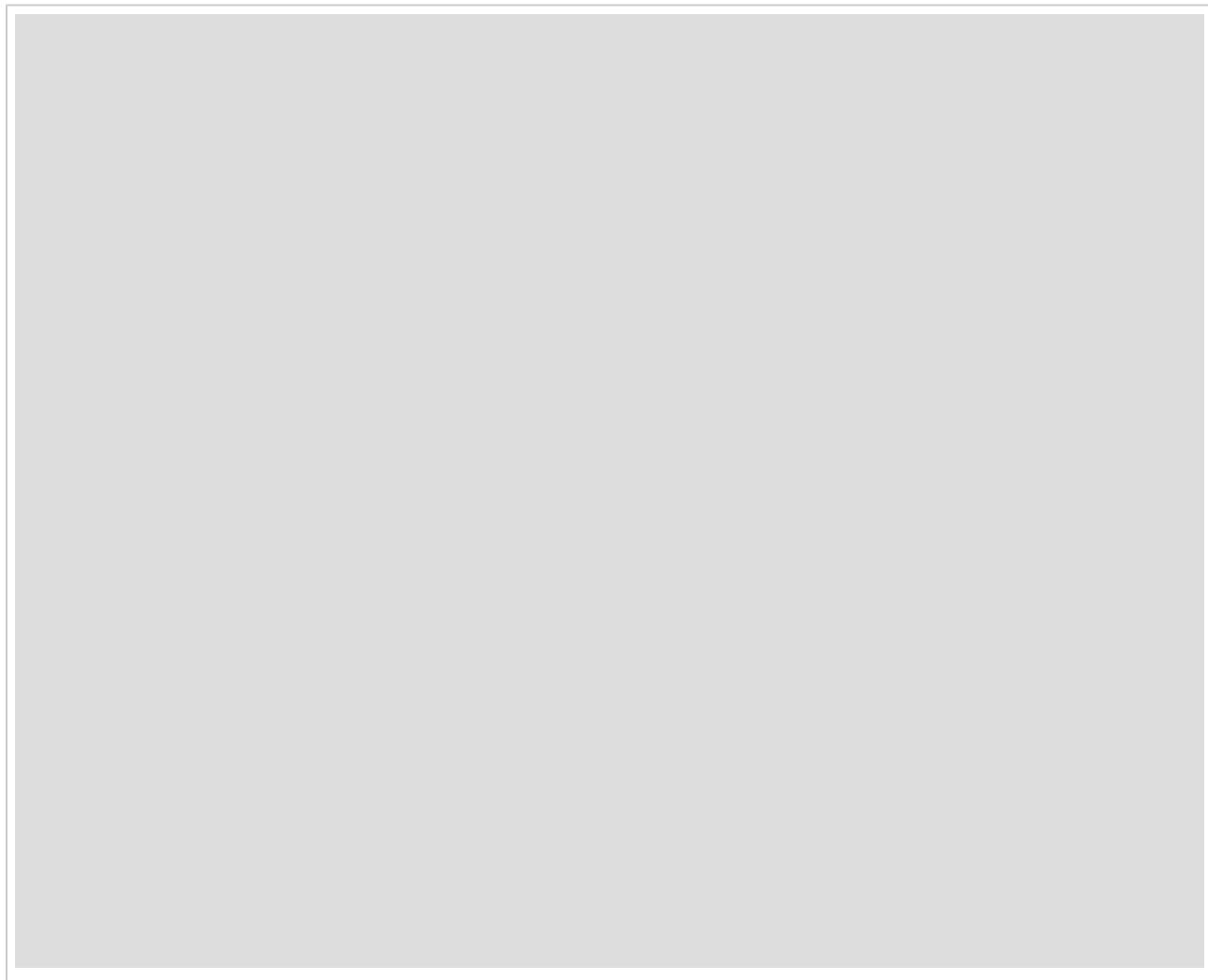
If you need creative specs, see Creative Specification PDFs.

Contents

- 0 to 5MM avails
- 5MM to 15MM avails
- < 15MM avails
- Rate card by audience

To download, click in the PDF frame. [Click here](#) to download the full Excel file if you need it.

0 to 5MM avails



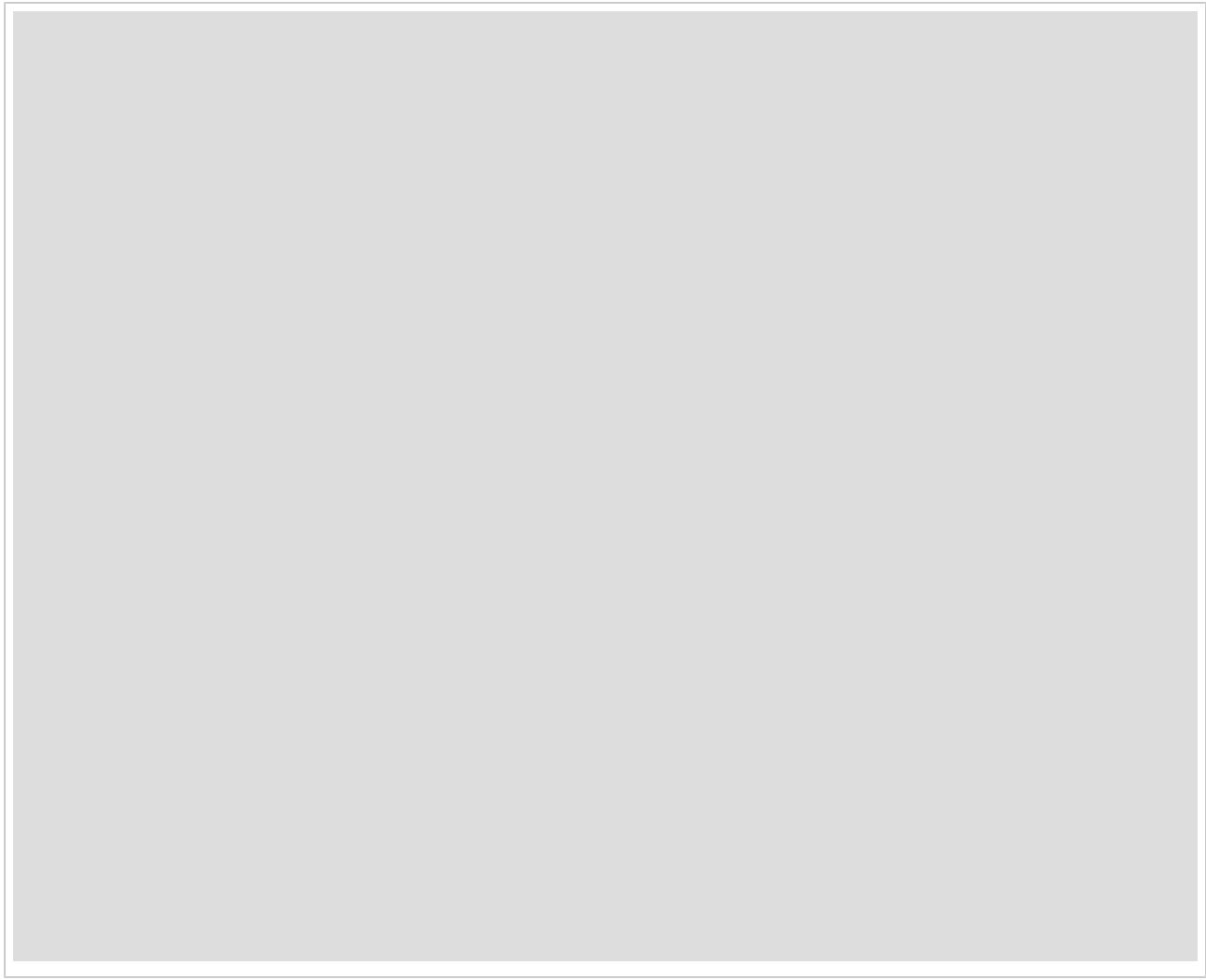
5MM to 15MM avails



< 15MM avails



Rate card by audience



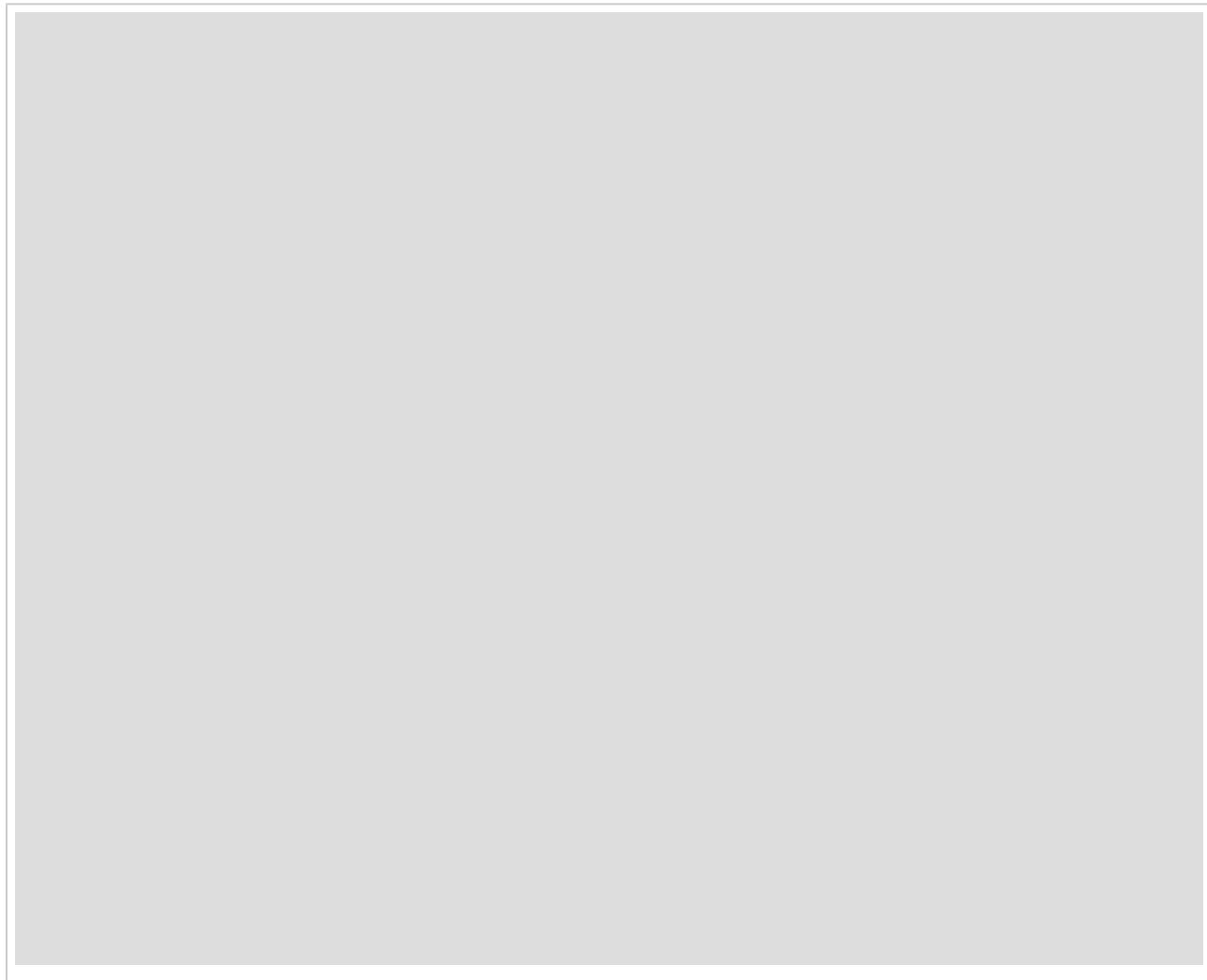
Tapad Rate Card I Q4 2013

Ad placement rates for North America.

Pro tip

If you need creative specs, see Creative Specification PDFs.

Download the rate card.



Understanding User Agents

A user agent is an HTTP header sent by your browser to a web server. A basic user agent, or *user-agent string*, contains *tokens* that provide details about the application you're using, browser type, operating system, and version information for those variables. User agents can include language information (optional) and additional details about browser enhancements, add ons, or the application making an HTTP request.

Anatomy of a user agent

RFC 2616 describes the formal user agent specifications. A typical user agent string consists of tokens that can look like this:

Mozilla/5.0 (Macintosh; Intel Mac OS X 10_8_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/30.0.1599.101 Safari/537.36

The following table defines the common tokens in a typical user agent string.

Token	Description
Mozilla/5.0	Tells web servers that this browser works like a Mozilla (Firefox) browser. It was originally developed by Microsoft to help early versions of Internet Explorer compete with Netscape. See A History of the Browser User Agent String for a serious, yet sly, discussion about this item.
(Macintosh; Intel Mac OS X 10_8_5)	Provides hardware and operating system information. In this case, the user has a Macintosh computer running a version of OS X with an Intel processor.
AppleWebKit/537.36	Defines the web browser engine and version. The browser engine is code that reads HTML (and other instructions) and formats that information on the screen. The AppleWebKit is used by Safari and Chrome browsers.
(KHTML, like Gecko)	Similar to the Mozilla token, this is used to indicate cross-browser compatibility.
Chrome/30.0.1599.101 Safari/537.36	Browser version details.

Other user agent examples

You can see examples of other common user agents here:

- [Chrome examples](#)
- [Firefox examples](#)
- [Internet Explorer examples](#)
- [iPhone examples](#)
- [Android examples](#)

For an extensive list of user agents on different devices, see the [categorizr](#) site. It takes some time to load, so be patient.

Language indicators in user agent strings

Language tokens or indicators are not required by the user agent specifications.. A user agent can include a language token, but this is optional. Common language codes include (but are not limited to) the following:

- **Chinese:** zh
- **English:** en or other variations such as en-US, en-AU (Australia), en-CA (Canada), en-UK (UK), etc.
- **French:** fr
- **German:** de
- **Italian:** it
- **Japanese:** ja
- **Spanish:** es
- **Russian:** ru

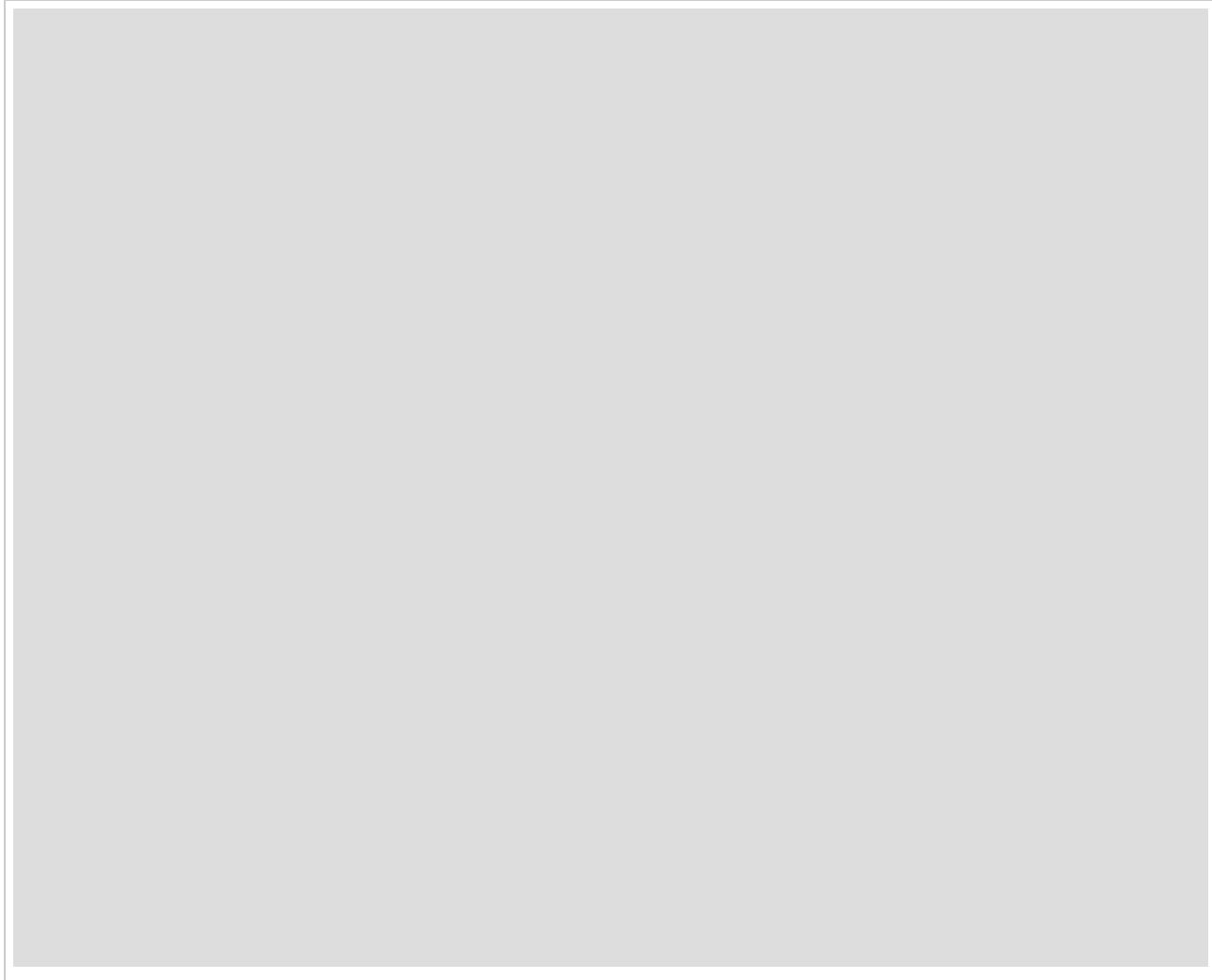
See [ISO 639.1](#) for a complete list of all the language codes.

Check your own user agent

Visit the [what's my user agent](#) website to see your current user agent.

Tapad and user agents

For more information about how Tapad works with user agents, review or download the Powerpoint presentation. To download, click [in the PDF frame](#).



Related concepts

[Mobile Device Bid Request](#)

Whitelisting Steps and Site/App Evaluation Guide

A whitelist is a list of websites and apps that have been reviewed and approved for ad serving. Sites and apps must be approved because Tapad will not respond to bid requests or serve ads from unreviewed or blacklisted publisher inventory. The whitelisting process involves researching the publisher's website or app, evaluating those objects according to our whitelisting criteria, and applying those results in Backoffice to the Display Context settings for those sites and apps.

[Download the guide \(PDF\).](#)

- [How to Whitelist a Website or App](#) — Refer to these instructions for steps on how to whitelist a website or app.
- [Whitelist/Blacklist Evaluation Criteria for Sites and Apps](#) — Evaluating websites and apps is a subjective process. Sometimes it's hard to tell if a site should be approved or added to the blacklist. As a result, we've developed these guidelines to help you make informed decisions about the sites and apps you're reviewing and apply your best judgement based on this material.
- [Maturity Rating](#) — Classify and deliver ads to publishers based on the IAB classification (gold, silver, bronze, etc.) scheme.

How to Whitelist a Website or App

Refer to these instructions for steps on how to whitelist a website or app.

Part 1: Evaluate the website or app

To evaluate a site or app:

1. Open the whitelist spreadsheet and go to the ID column. These are the display context IDs. (Note: Dave Austria generates and distributes the whitelist spreadsheet. Contact Dave to get on the distribution for this list.)
2. Find the site or app for the selected ID and evaluate it according to the [evaluation criteria](#).
 - To find a site, use the URL in the spreadsheet.
 - To find an app, search for it online (iTunes, Google Play, web search) to review its rating, description, or purpose/functionality.

Part 2: Modify Display Context settings

After you've completed researching a site or app, apply your findings to the Display Context section in Backoffice. To modify the display context settings:

1. Log in to Backoffice and change the default URL to <https://backoffice.tapad.com/displaycontexts/ID> (replace *ID* with the ID from the spreadsheet). This opens the display context page for the site or app ID.
2. Click **Display context** and:
 - Select the **Audited** check box.
 - Choose a rating from the **Maturity rating** list (see [Maturity Rating](#) for details).
 - In the **IAB categories** list, click or type the category you want to apply to the site or app. A site or app can belong to multiple categories. For a full list of categories, [download](#) the IAB Open RTB specifications and see section 6.1 Content Categories.
3. Review your settings, click **Save**, and repeat this process for the next ID in the whitelist spreadsheet.

Your Display Context settings may look similar to the following example:

The screenshot shows a 'Display context' form with the following fields and values:

Display Context		Performance
Display context		
Tapad Id:	23063504	
Supplier Id:	65837002	
First seen:	06/24/2014	
Url:	www.todacell.com	
Friendly name:	International Android (2)	
Type:	In-App	<input type="button" value="▼"/>
Description:	<input type="text"/>	
Audited:	<input checked="" type="checkbox"/>	
Maturity rating:	<input type="button" value="None-classified"/> <input type="button" value="▼"/>	
IAB categories:	<input type="button" value="Blind/no site list"/> <input type="button" value="x"/>	
<input type="button" value="Save"/> <input type="button" value="Cancel"/>		

Whitelist/Blacklist Evaluation Criteria for Sites and Apps

Evaluating websites and apps is a subjective process. Sometimes it's hard to tell if a site should be approved or added to the blacklist. As a result, we've developed these guidelines to help you make informed decisions about the sites and apps you're reviewing and apply your best judgement based on this material.

- Blacklisting criteria
- Whitelisting criteria
 - Gold level
 - Silver level
 - Bronze level
 - None-Classified
- iTunes app ratings
- Google Play app ratings

Blacklisting criteria

Blacklist a site or app if it falls in to or exhibits any of the following:

- Profanity, hate speech, racist, extreme political views
- Nudity, pornography, sexual products, sexually provocative material
- File hosting, bit torrent, photo hosting
- Alcohol, illegal drug use, guns
- Targeting children under 13
- More than 6 ads on page
- Multiple pop-ups
- Auto-refreshing ads in less than 2 minutes
- iTunes maturity rating 12+ (or greater) or if the app exhibits any of the following:
 - Cartoon or Fantasy Violence
 - Realistic Violence
 - Sexual Content or Nudity Profanity
 - Crude Humor
 - Alcohol, Tobacco, or Drug Use
 - References Mature/Suggestive Themes
 - Simulated Gambling Horror/Fear Themes
 - Prolonged graphic or sadistic violence
 - Graphic sexual content and nudity
- Google Play rating of High Maturity or any of the bulleted items listed under the iTunes bullet above.

Whitelisting criteria

If site or app does not exhibit any characteristics, traits, or behaviors described in the blacklisting criteria then you should apply an appropriate maturity rating. Refer to these guidelines for examples:

Gold level

Gold level sites or apps consist of/exhibit/contain:

- No user generated content
- No social networks or gossip sites. Note, *exclude* Facebook from this requirement. It is a gold level site.
- No more than 3 ads above fold
- Google and Microsoft are gold-level sites.
- Only highbrow gaming sites (e.g., Angry Birds, Words with Friends, Scrabble, Chess, Solitaire)
- No Dating sites
- Highbrow, mainstream news sites (e.g., *New York Times*, MSNBC, CBS, *Financial Times*, BBC, etc.)
- Highbrow entertainment sites (*entertainmentweekly.com*, A&E, etc.)
- Highbrow sports sites (*espn.com*, *247sports.com*, *mlb.com*, *nfl.com*, etc...)
- Weather sites (*weather.com*, *acuweather.com*, *weatherbug.com*, etc...)
- Travel sites
- Premium, well known sites
- iTunes application ratings for a gold-level site are rated as:
 - 4+
 - 9+
- Google Play ratings for a gold-level are rated as:
 - Everyone
 - Low maturity

Silver level

Silver level sites or apps consist of/exhibit/contain:

- Highbrow gossip sites
- Lowbrow gaming sites
- Highbrow social networks
- [iTunes application ratings](#) for a gold-level site are rated as 12+.
- [Google Play ratings](#) for a gold-level are rated as medium maturity.

Bronze level

Bronze level sites or apps consist of/exhibit/contain:

- Lowbrow social networks
- Lowbrow gossip sites
- Dating sites
- Sites with many ads

None-Classified

These types of sites or apps consist of ad networks where the site list is blind, that contain a lot of user-generated content, anonymous content, and/or all other non-blacklisted sites.

iTunes app ratings

iTunes provides [formal ratings](#) for each app. Usually, the iTunes rating appears on the left side of the page for the app you're reviewing. iTunes uses the following rating levels:

Rating type	Description
4+	Apps in this category contain no objectionable material.
9+	Apps in this category may contain mild or infrequent occurrences of cartoon, fantasy, or realistic violence, and infrequent or mild mature, suggestive, or horror-themed content which may not be suitable for children under the age of 9.
12+	Apps in this category may also contain infrequent mild language, frequent or intense cartoon, fantasy, or realistic violence, and mild or infrequent mature or suggestive themes, and simulated gambling, which may not be suitable for children under the age of 12.
17+	Apps in this category may also contain frequent and intense offensive language; frequent and intense cartoon, fantasy, or realistic violence; and frequent and intense mature, horror, and suggestive themes; plus sexual content, nudity, alcohol, tobacco, and drugs which may not be suitable for children under the age of 17.

Google Play app ratings

The Google Play store provides [formal ratings](#) for each app. Google Play uses the following rating levels:

Rating type	Description
Everyone	Applications in this category should not collect user's location data or contain objectionable material. Applications should not share user content or include social features.
Low maturity	Applications in this category may include instances of mild cartoon or fantasy violence or other potentially offensive content. Applications may collect user location data for the purpose of providing location specific information or otherwise improving the user experience, but should not share the data with other users. Applications may include some social features but should not focus on allowing users to find and communicate with each other.
Medium maturity	Applications in this category may include sexual references; intense fantasy or realistic violence; profanity or crude humor; references to drug, alcohol and tobacco use; social features and simulated gambling. Applications may collect user location data for the purpose of sharing or publishing with the user's consent.
High maturity	Applications in this category may focus on or include frequent instances of sexual and suggestive content; graphic violence; social features; simulated gambling; and strong alcohol, tobacco and drug references. Applications may collect user location data for the purpose of sharing or publishing with the user's consent.

Maturity Rating

Classify and deliver ads to publishers based on the IAB classification (gold, silver, bronze, etc.) scheme.

The following table defines the IAB ratings by type:

Rating	Description
Gold	The highest site rating. Ideal for brand and context-sensitive advertisers. Excludes sites that contain: <ul style="list-style-type: none">• User generated content• Gossip• An excessive amount ads• Most types of gaming (e.g., Angry Birds, World of Warcraft) content
Silver	Includes sites with gold classifications and sites that contain: <ul style="list-style-type: none">• Highbrow gossip• Highbrow social networks• Online or in-app games.
Bronze	Includes sites with gold and silver classifications and sites that contain: <ul style="list-style-type: none">• User generated content• Lowbrow gossip• Lowbrow social networks
None-classified	Includes sites with gold, silver, and bronze classifications and all ad networks where Tapad cannot identify the publisher.
Any	Any publisher in our network that is not blacklisted.

Index

An alphabetical list of articles by title.

Space Index

0-9 ... 0	A ... 68	B ... 25	C ... 69	D ... 65	E ... 19
F ... 13	G ... 8	H ... 11	I ... 37	J ... 6	K ... 2
L ... 10	M ... 20	N ... 10	O ... 9	P ... 35	Q ... 0
R ... 43	S ... 63	T ... 96	U ... 27	V ... 6	W ... 8
X ... 0	Y ... 0	Z ... 3	!@#\$... 0		

[0-9](#)

B

B

Rectangular (horizontal or vertical) online ads. Width and height are usually measured in pixels. Some common types of banners include Leaderboards (728x90), Medium rectangle (300x250), and Skyscraper (120x600). See also, web banner.

Backoffice UI

Concepts, tasks, and reference material for our internal, managed service UI.

Banner Object

The "banner" object is included in the impression object for creative unit specifications. All attributes are made available for creative constraints. Field Scope Type Example value Description w recommended in

Basic JSON Data

Syntax, definitions, and examples for synchronized ID data in a JSON object. File syntax and descriptions A basic JSON that have been matched to other, corresponding devices in the Device Graph. The file wi

Basic Text Data File

Syntax, definitions, and examples for synchronized ID data in a tab-separated (.tsv) text file. Syntax and contents Tapad tab-delimited flat file which uses this syntax (italic type indicates variable data): PRIMAR

Batch File ID Bridging

DRAFT Tapad partners can collect authenticated user IDs and send that data to Tapad in a batch file. Batch file bridging d Instead, you can collect data from your own servers, format it according to these sp

Batch File Opt-out Process: About

The batch file option uses FTP to deliver opt-out data to your dropbox on a nightly schedule. Batch file opt-out sends data timestamp and a device ID for the user who elected to opt-out from data collection. Vi

Best Practices for Campaigns

DRAFT Reference is We've found these guidelines helpful when planning a campaign and during its active lifecycle. The tactic should be removed from the campaign if it's not contributing to the campaign's success. This is a simple rule, and it

Best Practices for Demographic Targeting

Applying too many demographic targeting elements to a tactic can limit the scale of a brand awareness campaign. To avoid users based on demographic settings, consider these best practices: Try to limit

Bid Object

Object
Array of bid objects; each bid object relates to an imp object in the bid request. Note: Tapad only supports one bid for each contains the win notice URL. If the bidder wins the impression, the exchange call

Bid Optimization

Tapad provides optimization features that let you algorithmically leverage historical learning data to help improve bidding, amounts and types, include/exclude learning from other campaigns and tactics, and set

Bid Request Format Example

This is an example JSON format for a bid request sample. Please note: formatting may be incorrect due to PDF so please

Bid Request Object

The top-level bid request object contains a globally unique bid request or auction ID. This "id" attribute is required as is at line 1. Example value Description id required string "2cfb3f10-f6cc-11e1-a108-123139314d8"

Bid Response Format Example

This is an example JSON format for a bid response sample. Please note: formatting may be incorrect due to PDF so please use a code editor to view the original JSON. { "id": "d5a3c7c0-f2d2-11e1-8998-0a0027000000"

Bid Response Object

The top-level bid response object. No-Bids on all impressions should be indicated as a HTTP 204 response. For no-bids or these from the bid response. Field Scope Type Example value Description id Required String

Bidding and Overdelivery

Troubleshoot overdelivery when your bid price is too high. REVIEW INTRO Troubleshooting steps Key indicators Recommended performance report. In the results, check the win rate for your tactics. High win rates indicate delivery issues.

Blacklist Optimization and Targeting

Reference is Also known as context ID targeting, these procedures help you find and exclude sites from a campaign. Overview: Blacklist publisher targeting lets you exclude them from a campaign. Follow these steps:

BlueKai Integration Details

BlueKai Type DMP Tapestry Data Activation Delivery File Exporter Product Requirements Business Owner ryan.hotaling Objective Development Work Documentation BlueKai Integration Guide Notes from 7/17 meeting: 1:

BlueKai Integration Guide

Brand Awareness Campaign Settings

DRAFT Reference is A brand awareness campaign is designed to raise the profile of your product or service through ubiquitous reach. Use these recommended settings to set up various types of line items that deliver ads effectively.

Browser Supply Targeting

Reference is Target users on web browsers based on Tapad (IAB) or Integral Ad Science (IAS) classification and rating targeting and in-app supply targeting to the same tactic. Browser and in-app targeting

Budget Cap Enum

Required parameters that correspond to the budget allocation settings you can apply to tactics in the UI. The following table lists the equivalents: Caps for API Tactics Data type Caps for UI Tactics PERCENTAGE_OF_CAMPAIGN

Budget Caps and Flight Dates for Campaigns, Line Items, and Tactics

Overview and use cases of how budgets and flight dates work at the campaign, line item, and tactic levels. Tapad campaign, line items, and tactics. Within this structure, the budget and flight dates set at the campaign level.

Budget Distribution Between and Within Data Centers

Each Tapad data center receives campaign budget in direct proportion to the bid requests matched by their bidders. Within each server affects how much budget each bidder receives. Match rates determine budget distribution.

Budgets, Spending, and Pacing

A detailed examination of how Tapad systems and software manage budgets, spending, and pacing processes.

D

D

A Data Management Platform (DMP) is simply a large database of consumer data that's useful to online marketers. Usually consumers' online behavior. Typically, DMPs help marketers store this data, create target

Dashboard Filters, Devices, and Data

Audience Pulse dashboard filters let you limit results by audience, Confidence Score, and device types. Data filters Dashboard the Audience and Confidence Score options. These filters help you control the amount of infor

Dashboard Reports

The dashboard displays management metrics at the campaign level and for individual line items and tactics. Data is updated previous 24-hour period. Dashboard access Click the Dashboard tab to search for a campa

Data Collection With Google's Classic Analytics

This code uses Tapad's JavaScript library (TAJ) to collect data and send it to Google Analytics. Use this code if you're wor Analytics. Placement We recommend that you place this analytics code within the <head>

Data Collection With Google's Universal Analytics

This code uses Tapad's JavaScript library (TAJ) to collect data and send it to Google Analytics. Use this code if you're wor Analytics. See About Universal Analytics <https://support.google.com/analytics/answer/27>

Data File Delivery Methods and Filter Options

Basic and enhanced delivery options.

Data File Formats for Audience Amplify

An Audience Amplify data file contains device IDs, audience IDs, and device types. There are 2 types of data files, each w
is for clients who do not use the AppNexus DSP. An AppNexus data file is for clien

Data Flow Appendix

Supplemental reference material related to Tapad Data Flow processes.

Data Flow PDFs

PDF formatted content that describes options for sending synchronized IDs to Tapad partners. Updated March, 2015. exc
options Includes basic JSON and text data, content separators/delimiters, and standard deli

Data Flow: Receive Connected IDs From Tapad

ID matching is a process that takes device IDs you've sent to Tapad and matches them to other device IDs. As determinec
devices that share a connection with your original ID. When the matching process is comple

Data Flow: Send Audiences To Tapad

Tapad needs an audience data file and an audience taxonomy to match your audience IDs with related IDs in the Device C
and audience IDs in a JSON file. A taxonomy file consists of metadata related to the audie

Data Integrations

Send data to Tapad from other products for cross-device targeting and retargeting.

Data Object

An array of objects with data information about the user. This data may be sourced from Tapad proprietary data along with
can include information from multiple data providers. The data object includes th

Data Visualization in the Supplier Health Dashboard

The Supplier Health Dashboard displays supply partner data with sparklines and as percentages. Scroll through the list for
or click on each data point to view detailed results in a pop-up window. Data is upd

Dataflow Management

Dataflow Management provides features and workflows that let you set up and manage your data flows. Refer to this secti
for feature settings, descriptions, and definitions. A dataflow is part of an ID s

Dataflow Metrics

Evaluate your data flows with these dashboard tiles that return information about devices and their related levels of connec

Date and Time Formats

Preferred styles and standards for dates and times. Date and time formats: Rules and guidelines Try to follow these: Prefe
DD-MMM-YYYY, e.g., 28-Nov-2005 (least ambiguous for a global audience). Time 12 hour clock: 4:00 p

Day of Week

Deliver ads during selected days of the week. To target by day, click the Day of week field, select a day and click Save. Y
of week.png

Daypart

Deliver ads during times most likely to reach your intended audience. To target by time, click the Day part dropdown, sele
or the specific hours you require. Click Save to apply your daypart targeting.

dCPM Pricing: About

Dynamic Cost Per Mill (or dCPM) is a pricing model that ties bid price to impression value. This is different than fixed CPM
amount for every impression regardless of its value. With dCPM, the billing amount can va

Deactivate a Tapad - Adobe Data Connection

Follow these steps to deactivate and remove the Tapad data connector from your Adobe Analytics account. To delete a c
to Adobe Marketing Cloud > Analytics > Data Connectors. Select the Configure tab. In General Set

Debugging

And this: <https://jira.tapad.com/confluence/x/BIAuAQ> <https://jira.tapad.com/confluence/x/BIAuAQ> BlueKai Third Party Seg
with "BK-" (e.g. BK-12345) If tracking tag does not exist in the UI/db check that it is

Delete (Remove) Audiences

Parameters, definitions, and code samples. Audiences: About An audience consists of set of devices or users that exhibit
traits. Tapad API or SDK audience data usually consists of numeric IDs (integers) tha

Delete a Campaign

A DELETE method that lets you remove a campaign. Syntax DELETE <https://backoffice.tapad.com/api/campaign/campaignID> <https://backoffice.tapad.com/api/campaign/1234> Sample response A successful request does not return any data.

Delete a Creative

A DELETE method that lets you remove a creative. Syntax DELETE <https://backoffice.tapad.com/api/creative/creativeID> <https://backoffice.tapad.com/api/creative/123> Sample response A successful request does not return any data.

Delete a Line Item

A DELETE method that lets you remove a line item. Syntax DELETE https://backoffice.tapad.com/api/line_item/lineItemID
Make separate calls to delete additional line items. Deleting a line item deactivates al

Delete a Property

A DELETE method that lets you remove a property. Syntax DELETE <https://backoffice.tapad.com/api/property/propertyID>
<https://backoffice.tapad.com/api/property/1234> Sample response A successful request does not return any data.

Delete a Tactic

A DELETE method that lets you remove a tactic. Syntax DELETE <https://backoffice.tapad.com/api/tactics/tacticID> Sample <https://backoffice.tapad.com/api/tactics/1234> Sample response A successful response returns status code 204 No content

Delete an Account

A DELETE method that lets you remove an advertiser. Syntax DELETE <https://backoffice.tapad.com/api/account/accountID> <https://backoffice.tapad.com/api/account/123> Sample response A successful request does not return any data.

Delete an Advertiser

A DELETE method that lets you remove an advertiser. Syntax DELTE <https://backoffice.tapad.com/api/advertiser/advertiserID> <https://backoffice.tapad.com/api/advertiser/1234> Sample response A successful request does not return any data.

Delete Data

Parameters, definitions, and code samples. Data deletion methods The remove and clear methods belong to a family of devices by the various add_data and sadd_data methods. Like add and sadd, these deletion

Delivery

The Delivery tile returns household impression data by device in a Venn diagram. The overlap between circles in the diag impressions were delivered to consumers across multiple devices. Delivery tile platforms include

Delivery Methods

Information about available transfer methods, naming conventions, and delivery schedule. Standard delivery methods inc partner (partner provides login credentials to Tapad) Update type: Incremental Delivery

Delivery Methods for Audience Data and Taxonomy Files

You can send audience and taxonomy data to Tapad by batch file or real-time delivery methods. This section includes info schedules, and file formatting. Batch file delivery Batch file delivery method relies o

Demographic Targeting

Deliver content based on Semcasting <http://semcasting.com/> demographic data like area, ethnicity, income, age, net worth Semcasting demographic targeting features apply a \$0.13 fee against your bid price. The defau

Demographic Targeting and the Targeting Slider

Demographic targeting relies on census data tied to groups of IP addresses. The targeting slider lets you reach a demogra average. Data sources used by demographic targeting Demographic targeting relies o

Demographic Targeting Tactics for a Brand Awareness Campaign on Mobile, Tablet, and Desktop Devices

Reference is Follow these instructions to create separate tactics optimized for mobile, tablet, and desktop devices that i campaign. The goal of each tactic is to reach a specific demographic audience and

Deprecated Content Archives

Older or unused material from the documentation space.

Deprecated Creative Specs

Creative specs have been redesigned. This is the old layout. For the latest, see the redesigned specs in the DSP Self-Ser

Deprecated Tapestry Methods

A dusty archive and final resting place for unused or obsolete API methods.

Designated Marketing Areas (DMAs)

Deliver ads to specific designated marketing areas. To target by marketing area, click the DMA field, select a name from th excerpt_digitalEnvoy DMA targeting is not available when targeting: Countries outside

Device Graph

Overview and functionality.

Device Graph Basics

The Device Graph (or graph) is Tapad technology that lets our systems target and deliver content to the same user across internet-enabled devices seen on a network and the connections (or edges) shared betwee

Device Graph Data Structures

Definitions and Data Structures Device = storage_key = canonical ID DeviceId (array) = value = ID of device Sighting (ma "HARDWARE" are hardware identifiers and can come either from an SDK/S2S or come from a

Device Graph Methodology

Purpose of the Device Graph The Device Graph™ identifies and maps the relationships among and between multiple devi household. It uses various rules, data sets, algorithms, and other metrics to measure these r

Device ID

Deliver ads to devices based on device ID. To target by ID, click Device ID field, select an ID, and click Save. This targetin similar to this: Screen Shot 2013-10-03 at 6.21.11 PM.png The following table l

Device ID Collection with typed_device_id

In a server-to-server (S2S) data transfer, the typed_device_id parameter sends iOS and Android IDs to Tapad. typed_devi capture device IDs and values related to the ID. These IDs are useful because they help yo

Device ID Superclusters

Device ID superclusters are unusually large groups of device IDs connected to a single device ID in the Device Graph. Un that define or identify a supercluster. For example, a typical household might have a doz

Device ID Synchronization

Parameters, definitions, and code samples. Purpose of device ID synchronization All of our APIs provide methods that le used used to help: Identify devices in the Device Graph. Extend the Device Graph by add

Device ID Types

Tapad's Device Graph Access (DGA), and other systems, identifies devices and cookies with the IDs contained in this list. cookie type id_type Description 3 Partner Cookie ID 2-4 character abbreviation In t

Device Identifiers

ID types, adding IDs to the Device Graph, and detecting browser support for third-party cookies. Types of identifiers The i Tapad: ID types Description First-party cookies Tapad can store a unique ID in

Device IDs

A list of cookie and hardware IDs. Cookie IDs Tapad and the Tapestry APIs and SDKs identify cookies as shown in the fc TAPAD_COOKIE The Tapad cookie ID. TAPAD_FIRST_PARTY_COOKIE The ID of a first-party cookie set by

Device Object

Root object for device / browser information. The "device" object provides information for the device including its hardware mobile handset, tablet, or desktop computer. The device object includes the geo o

Device Targeting

Deliver content to specific devices, wireless carriers, networks, user agents, and IP addresses.

Devices and Audiences

DRAFT Click to see how many devices Tapad can add to an on-site or off-site audience at a selected confidence score. Th chart. Device types include mobile, tablets, desktops. To help display large numbers, th

Direct Conversions and IP Conversions: About

When creating a report, select the Attribution Connection checkbox to return results by direct conversions and IP conversions takes place on the same device that initially receives the impression. For example, say

Direct Response Campaign Settings

foo

Display Expandable Rich Media Creatives: About

Overview and supported ad servers for non-mobile Rich Media creatives. Overview Display expandable Rich Media creat desktop or laptop computer. These creatives contain standard ad elements like text, imag

Display Position

Reference is Target display, mobile, and tablet inventory by page/display position. To target by display position, click the location. Your display position targeting could look similar to this:

DMP Code Placement and Configuration

Follow these instructions to set up our JavaScript data collection code. Placement Put the API code in the <head><script> collection. <head> ... <script>!-- Tapad code here --> </script> ... </head> Configur

Documentation Home

Tapad documentation Documentation contains concept, task, and reference material on how Tapad products, systems, an browse the contents of this space for help with concepts, tasks, and reference material for Bac

Documentation Processes for IAB Certification

Reference is Documentation creation and revision process Requests for documentation come from formal and informal Project managers may include a documentation requirement when they start to create a new produ

Domain Names

Domain targeting helps you deliver ads to users on business networks with specific domain names like microsoft.com, app name targeting also accepts optional wildcard characters like an asterisk. To target b

Download the CTR Report Templates

Click these links to download the CTR template. CTR template with conversions CTR template without conversions Relat with the CTR Spreadsheet Related reference: CTR Spreadsheet Tabs Described

Duplicate a Campaign

A POST method that lets you clone a campaign and its related elements. Syntax POST https://backoffice.tapad.com/api/campaigns/ body of a duplicate request looks like this: { "campaign":1234, //ID of the campaign you want t

F

F

Describes techniques and/or companies that collect data for the purposes of identifying individuals. Fingerprinting techn Invasive device inspections to help ensure identifier uniqueness. IDs that never cha

Feature Phones

Feature phones are uncommon or specialized mobile devices, software, or communication protocols. As defined by Tap http://en.wikipedia.org/wiki/Mobile_Information_Device_Profile: Mobile Information Device Profile MOBI h

File Format Requirements for Audience Device IDs and IP Addresses

Upload device IDs and IP addresses in separate, single-column, plain-text files after you create an audience. File format i addresses to an audience as you create it, but it's better to set up the audience first

Filter By Synced Devices

By default, Tapad returns all synchronized data for every device mapped to the IDs you send us. This includes data for dev about. These "unknown" devices get included because they're connected to your known IDs in

Filter Options

Filters let you choose how you would like Tapad to return synchronized ID data. Filter options include return all data (default), filter by device IDs, filter by platforms, and filter by Confidence S

Filtering ID Superclusters

To remove superclusters from reports, Tapad sorts device IDs into a series of equal buckets that group clusters by size percentage larger than those in the 99.99th percentile. You can see this filtering in the cluster report.

First Visited Platform

First Visited Platform counts the devices a visitor uses to access your site for the first time. More importantly, subsequent visitors will not add a first visit record for that new device. Instead of adding a device, FV

Flight Dates, Budgets, and Overdelivery

DRAFT Troubleshoot overdelivery based on budget and flight date problems. Troubleshooting steps Key indicators Recommended following budgets are susceptible to overdelivery: % of line item % of campaign Fixed \$ amount

Formatted example for JSON data files

Putting this here as a temporary measure. Will be restored to Sample JSON Audience Data File doc once JSON parsing has been completed.

```
[{"device_id": "fb725e65-53d2-7d37-589e-7c2e159b65c0", "device_id_type": "THIRDPARTY", "au...
```

Frequency Cap Hierarchy

Tapad organizes frequency cap delivery options into a hierarchy of household, individual, and device-level groups. Frequency caps are applied to items, and tactics. Within the frequency cap/campaign structure, caps set at parent (higher)

Frequency capping

Frequency capping lets you determine how often an ad appears within a set amount of time. Impression caps help you control delivery and prevent overexposure. You set up frequency capping at the campaign level. Tapad lets you control delivery

Frequency Capping for Campaigns, Line Items, and Tactics

A frequency cap lets you control the maximum number of impressions that can be shown on a device within a specific time period. Frequency caps are applied to the household, individual, or device level. However, only the device-level option is available.

Frequency Caps and Overdelivery

Troubleshoot overdelivery by applying a frequency cap to your tactics. Troubleshooting steps Key indicators Recommended

H

H

See Device IDs. A practice that lets companies track an Internet user's web browsing history on an ongoing basis without their knowledge. Most common browsers use technologies that help prevent this activity. A mechanism that allows websites to store information about users' browsing history.

Home Value

Deliver ads to selected users based on home value. To target by home value, click the Home Value field, provide a minimum value, and apply. Your target settings could look similar to this: home value.png

Household Scores

Household scores measure the strength of the relationship between nodes (devices) in the Device Graph. Higher scores suggest stronger relationships between devices in the same household than lower scores. High scores improve accuracy, but reduce reach. Lower scores suggest weaker relationships.

Households and Individuals

Click to see the number of household and individual groups for your selected audience and confidence score. A single household can contain multiple individuals and multiple households can contain more devices depending on how you limit data with the Audience and Confidence Score filters.

How Confidence Scores Affect Report Results

Most devices in the Device Graph share a "connection" with another device. The data linking groups of devices together is represented by a confidence score. This component data includes a household score (H) and an individual score (I). Collectively, these scores are used to calculate the overall confidence score for each device.

How Optimization Affects Bid Amounts on dCPM Tactics

Adding optimization goals to a dCPM tactic lets you set various CPx performance targets and apply learning algorithms to the bidding process. dCPM, bidding can be static or dynamic depending on if you're using optimization, but it's always based on historical data and current market conditions.

How Site List Targeting Affects Managed and Self-Service Clients

Site list targeting is available to managed clients only. It is not available to self-service customers. Site list targeting lets you target specific websites or domains within a campaign or on specific line items. Managed clients and site lists For managed clients, site list targeting is available.

How the Budget Update Process Can Delay Tactic Delivery

Asynchronous budget updates can delay tactic delivery from 10 minutes to an hour. However, you can force a budget update immediately by using the Force Update button. Budget allocator and bidder update cycles are not synchronized. Budget all tactic delivery.

How to Make TAJ API Calls

Sample TAJ code module, authentication code, variable syntax, and request/response examples. TAJ code module TAJ is a Java library that provides a simple interface for interacting with the Tapad Audience API. To get started with using the TAJ API, put the following code snippet on every page that needs to use the TAJ API.

How to Make Tapestry API Calls

Base URL endpoints, authentication, versions, variable syntax, and sample request/response code. Getting started Talk to the Tapestry API. They can help you develop a strategy for working effectively with the Tapestry API.

How to Whitelist a Website or App

Refer to these instructions for steps on how to whitelist a website or app. Part 1: Evaluate the website or app To evaluate a website or app, go to the ID column. These are the display context IDs. (Note: Dave Austria generated these IDs.)

J

[JIRA - Confluence Linking](#)

Instructions for setting up links between JIRA and Confluence. Includes examples from both systems.

[JSON Data Files: Syntax and Content Options](#)

Syntax, definitions, and examples for synchronized IDs in a JSON object. Content options include basic ID data, IDs with s

[JSON Data With Platform Types](#)

Syntax, definitions, and examples for synchronized ID data in a JSON object, including optional platform types. The platfor
the device itself and can include results like iPad, Android, Kindle, iPhone, etc.

[JSON Data With Scores](#)

Syntax, definitions, and examples for synchronized ID data in JSON data, including optional scores that quantify the relatic
score suggests a closer connection between devices than a lower score. These scores help

[JSON Format for Audience Data Files](#)

An audience data file contains strings, integers, arrays, and mapped data in a JSON object <http://www.json.org/>. Format th
specifications. All parameters are required unless indicated otherwise. The follo

[JSON Format for Audience Taxonomy Files](#)

An audience taxonomy file contains strings and integers in JSON objects <http://www.json.org/>. Format the parameters in yo

specifications. All parameters are required unless indicated otherwise. The following table list

L

L

The page a user is directed to after clicking an ad. See Geotargeting. A technique that helps build larger audiences from similar lookalike model share similar characteristics of the original audience segment. Lookalike audiences.

[Line Item API](#)

Available methods, syntax, and code samples.

[Line Item Features](#)

Purpose, structure, CMS comparison, and UI feature summary. Purpose of line items Line items let you: Link delivery to ads to creative assets, creatives to line items and campaigns. Manage tactics and budgets at the line item level.

[Line Item Metrics](#)

Landing page metrics defined. Clicking the Line Items tab shows line item performance details by default. To create a new button Campaign roll-up The roll-up section provides a budget and duration.

[Line items](#)

Overview, set up, and features.

[Linked JIRA Issues: Examples](#)

Samples of linked JIRA issues in Confluence. To view the macro settings: Click Edit. Click the macro placeholder and click Cancel to return to the page. Click Cancel to return the page to read-only mode. Example

[List of Creative Size IDs and Sizes](#)

ID and size values you can use to specify creative size. The creative size enum includes the following IDs and creative sizes: 320x50 2 300x50 3 216x36 4 168x28 5 120x20 6 728x90 7 234x60 8 160x600 9 236x60 10 468x60

[Lists and Tables](#)

Styles and format for bullet lists, numbered lists, and tables. List basics Place a colon at the end of the text that introduce available:..." Avoid creating lists that contain both sentence fragments and

[Lotame Data Integration Guide](#)

A Tapad - Lotame data integration helps you reach audiences across multiple devices such as desktop and laptop computers and other platforms. Overview User ID sharing and synchronization are the key components in

[Lotame Integration Details](#)

Data synchronization methods and examples. Lotame Overview Lotame is a data management platform (DMP) that provides on-line data and build audiences. A Tapad-Lotame data integration lets Lotame share and synchronize data between the two platforms.

N

N

Paid advertising designed to blend in with editorial content. Network Interface Card. A computer component that lets you connect a tablet, phone, etc.) in the Device Graph.

Name and Update Your Audience Data Files

Name and update an audience data file according to these specifications. Naming conventions The following syntax defines a name. Note, colored italics indicates a variable placeholder. The other elements are required.

Name and Update Your Audience Taxonomy File

Name and update an audience taxonomy file according to these specifications. Naming conventions The following syntax defines a name. Note, colored italics indicates a variable placeholder. The other elements are required.

Naming Conventions for Campaigns

Format and syntax for campaign names. Format To format a campaign name, include: A reference number specified as F ID (Account Team do this in the UI only). A 3 letter abbreviation for the agency name. The adve

Naming Conventions for Tactics

Format and syntax for tactic names. Format A properly formatted tactic name contains a prefix, base, and suffix underscore as shown in the example below: Syntax: PREFIX_BASE_SUFFIX Example: NULL_GEO_CrossRT

Net Worth

Deliver ads to selected users based on net worth. To target by net worth, click the Net Worth field, provide a minimum/max apply. Your target settings could look similar to this: net worth.png

Network

Deliver ads to devices that appear on a selected network type. To target by network, click the Network field, select a network identifies networks from the Device Graph, IP addresses, domain extensions (e.g.

Networks and Overdelivery

DRAFT foo Troubleshooting steps Key indicators Recommendations Expectations

Non-merging Conditions

DRAFT In most cases, 2 separate devices cannot merge if they have already been identified with a Tapad hardware or condition in a 2-device set. excerpt_merge_symbology Case 1 In this example we have 2 devic

Numbers and Currency

When to user numbers, numerals, and how to write about money. Numbers In general, use numbers instead of numerals <http://www.nngroup.com/articles/web-writing-show-numbers-as-numerals/> that it is easier to rea

P

P

A way to measure consumer behavior after they click an ad. Also used to link a conversion to a particular site. A form of advertising where a brand pays based on results, usually on a CPC basis. Personally Identifiable Information

Pacing: Everything You Wanted to Know but Were Afraid to Ask

An overview of how Tapad manages pacing to help ensure smooth, even spending and delivery during the lifetime of a campaign spends budget during its lifetime. Tapad controls pacing at the tactic level. Campaign

Partner Initiated ID Synchronization Code: Syntax, Parameters, and Sample Request

ID synchronization code and variables defined. Put this code in the <head> section of your client's website when you set up the

Our ID synchronization code can send data to secure (HTTPS) and un-secure (HTTP) website

Path Splitting Rules for Campaign Pulse

In Campaign Pulse, path splitting rules help the Pathing tile <https://confluence.tapad.com/display/DOC/Conversion+Path> or [here](#). This logic provides rules that filter by selected conversions, for merging events, and so on.

Path Truncation Rules for Campaign Pulse

In Campaign Pulse, the Pathing tile follows several rules that define the path length, qualifying initial events, and how to truncate paths. These rules also define what to do when a path does not contain any impressions or conversions.

Pathing

Reference is [here](#). The Pathing (conversion path) tile uses a starburst graph composed of concentric rings to show you the direct path from an impression to a conversion. By default, the interaction path starts with an impression in the innermost ring.

PDF workspace

Placement Type Enum

IDs you can use to set placement type (e.g., Rich Media, video, image). The placement type enum includes the following values:

- 1 Non-expandable Rich Media
- 2 Expandable Rich Media
- 4 Video
- 5 Facebook creative

Platform and Platform-in-graph Targeting

Describes how these targeting combinations work together. Targeting review To start, let's review how platform and Platform-in-graph targeting work together. Platform targeting delivers ads to selected devices in real time. For example, choosing "Computer" as the targeting lets you exclude them from a campaign. Follow these steps to get started. Create a platform targeting rule.

Platform IDs

Click to see audience composition by device hardware or cookie type for the selected platform. Platform options include mobile, desktop, and tablet. This illustration shows the Audience Platform tile in the Ask Tapad dashboard.

Platform Mix

Click to see the % of devices and device combinations for the selected audience or confidence score. This tile displays results for each device combination. The circles show you how many household or individual groups share connections between multiple devices.

Platform Optimization and Targeting

Reference is [here](#). These procedures help you find and remove ineffective platforms from a campaign. Once you've identified ineffective platforms, targeting lets you exclude them from a campaign. Follow these steps to get started. Create a platform targeting rule.

Platform Targeting

Platform targeting delivers ads to the selected device in real time. To target by device, click the Platform field, select a device type, and then click Save. The following table lists the predefined device groups, data sources, and costs.

Platform Types

Platform Types counts user visits by groups of named platform types (e.g., iPhone, computer, iPad, etc.). Over time, this data helps you understand which specific devices and device combinations site visitors are using to interact with your site.

Platform Types List

Platform types are the commercial or commonly used names for various devices (e.g., Android tablet, iPhone, Kindle, etc.) other than Device ID types, which are Tapad-specific descriptors for device hardware or cookie data.

Platform-in-Graph

Platform-in-graph targeting lets you deliver ads to other devices in the Device Graph that are connected to the selected device. To target by device, click the Platform in graph field, select the required devices and click Save. How it works

Platforms and Overdelivery

DRAFT foo Troubleshooting steps Key indicators Recommendations Expectations

Platforms Associated

Platforms Associated counts visits for devices grouped together as a collection of identical platforms. By counting visits by platform, you understand the levels of connectivity among your site visitors. For example, data from the Platforms Associated tile shows the number of unique visitors per platform.

Platforms Associated and Visited Platforms Compared

An overview of how Platforms Associated and Visited Platforms count site visits and associate visits to a ranked platform group. Platforms Associated and Visited Platforms organize devices into collections called a platform group.

Political Affiliation

Deliver ads based on political preference. To target by political affiliation, click the Political affiliation field, select an affiliation, and then click Save. Political affiliations include: Registered Democrat, Registered Republican, Independent, and so on. Fees apply. Political affiliations include: Registered Democrat, Registered Republican, Independent, and so on.

Post-campaign Wrap Up Report

DRAFT TEST WORKSPACE Holding space for work on that really big report.

Prepare Cobalt Tags for Trafficking

Cobalt tags require substantial modifications before you can traffic them in the Tapad UI. These changes are repetitive and involve bulk find/replace operations available in any text editor. Once you've made the necessary changes, save the file and upload it to the Tapad UI.

Price Model Enum

These line item API parameters correspond to the available pricing types in the UI. In an API request, format the corresponding key-value pair. A formatted key-value could look similar to this: "pricingModel": "FIXED_CPM". At this point, you can add the rest of the line item parameters.

Primary ID Analysis

For all the IDs you send to Tapad, the Primary ID Analysis shows how many of them have at least 1 connection to another ID. It also shows the total number of IDs sent, total count by hardware and cookie IDs, and a bar graph.

Private Marketplace Deals

Deliver ads directly to publishers by targeting a specific deal ID. To target by deal ID click Private Market Place Deals, type multiple deal IDs if they're from the same supplier only. Separate multiple deal

Processing and Delivery Times for Scheduled Reports

Scheduled reports (including conversion attribution data) are processed and delivered according to UTC time <http://en.wiki> users in time zones that trail UTC, it may seem as if your reports arrive unusually ea

Product Model Enums

Optional parameters that correspond to the product type options you can select when creating a line item in the UI. In an corresponding products parameter with the ID only. A formatted key-value could look similar to this: "produ

Programmatic Real Time Bidding Access

Tapad's Programmatic Real Time Bidding Access product enables ad networks, demand side platforms and other program on advertising media while also gaining valuable access to Tapad's Device Graph™ in real t

Programmatic RTB Access Introduction

Tapad's Real Time Bidding API enables ad networks, demand side platforms and other programmatic buying entities (coll unique offerings to bid on advertising media while also gaining valuable access to Tapad's

Programmatic RTB Access Version and Revisions

This page will designate the current version of the spec and any revisions made to prior specs. Version 1.2 Revisions Da bidfloor to the imp object Added btype and btr to the banner object Added audited, matur

Properties API

Available methods, syntax, and code samples.

Property Type and Platform Enums

Values accepted by the property API type and platform keys. Type The type key corresponds to the trackingImplementation type values include: "PIXEL" "S2S" (server-to-server) "SDK" (software development kit) Pla

Publisher ID: How to find it

Each publisher in our network has a unique numeric ID. Enter this ID in the blacklist or whitelist targeting field to exclude o To find the publisher ID: Click the suppliers tab. Click the name of your su

Punctuation

A few rules and examples, not a comprehensive guide. Punctuation: Rules and guidelines Try to follow these: Style Desc in the grammar section. Plural acronyms No apostrophe before the "s"! This: IDs, CPMs, URL

Purpose, Style, Voice, and Tone

Writing can be challenging. Oftentimes, a blank screen and a flashing cursor are among the most intimidating things that c rules, and advice in this style guide can help you break into that blank page and c

R

R

Websites that adapt to multiple screens (computers, mobile phones, tablets, etc.). Serving ads across the web and other p visited an advertiser's website. Request for information or request for invoice. Request for

Reach and Frequency

Reach is the number of unique platforms that have seen an ad for your brand, product, or service. Frequency describes hc platforms. The Reach and Frequency tile summarizes unique impressions for your selected campa

Real-time Opt-out Process: About

This method uses API methods to transfer opt-out information via direct server-to-server data calls. As a Tapad partner, yc to receive this data. Real-time transfers send data in a URL string to the partner's

Reallocate Budgets to Efficient Tactics

Reference is Budget reallocation is an important part of the campaign optimization process. This procedure helps you budget from less to more efficient tactics. UI reporting features let you find underp

Reference

General information, terms, useful facts, tables, and other data.

Reference Material for Technical Writing

Consult these resources when you have a style or usage question that's not covered in this guide or need help with writing dropdown, etc. Helpful guides: Merriam-Webster's Collegiate Dictionary <http://www>.

Related Buyer User ID Object

Array of objects with data about the buyer's related users. This data is sourced from Tapad's proprietary Device Graph. Fi optional string "12345" Programmatic Partner's ID for this related user as mapped by Tapa

Related ID Analysis

The Related ID Analysis tile returns a count of all the related IDs that Tapad has matched to the IDs you send us. This tile devices, iOS devices, and cookies that have a connection to a related ID. In addition

Replacing and Appending IDs

DRAFT During an ID sync, Tapad and cookie IDs get replaced while hardware IDs get appended to existing IDs for a devic cookie ID replacement In this example, Device A has been identified with Tapad cookie ID T1 an

Reporting

A well-designed Tapad report gives you data that provides actionable insights to campaign performance and helps you manage and run a report.

Reporting Features Overview

Tapad reports give you an extensive set of controls, features, and filters you can use to return data about campaigns, line items, and reporting to set up and run a report. With standard reports you can: Select only the relevant data, filter by date range, sort by revenue or impressions, and export to CSV or PDF.

Request a Deal ID

A Deal ID is a universal identifier that links private exchange inventory to specific buyers through programmatic real time bidding. It's used to identify private exchange transmits to DSPs and indicates when a private deal impression was served.

Requirements and Implementation Questionnaire for Adobe Analytics

To use this service you must have an Adobe Analytics account and a Tapad customer ID. Contact your Tapad consultant to complete the implementation questionnaire in this section. Please let us know about the following: eVar av

Resize Creatives with Pixlr

Follow these instructions to use Pixlr, a Web-based tool that lets you resize creatives. Useful when you need to change the size of a creative or add a new one. This tool works well with creatives that contain small fonts or typefaces. Does not work for creative assets with complex backgrounds.

Restrict Attribution to a Campaign

It's common to run multiple campaigns that use the same conversion event (pixel) and identical or overlapping flight dates. In these cases, the reporting system credits the campaign that was responsible for the last conversion.

Retargeting Tactics for Brand Awareness Campaigns

Reference is Follow these instructions to create optimized tactics for mobile, tablet, and desktop devices that use third-party ad networks. Retargeting is an ad delivery strategy designed to help you reach users who don't convert.

Return Data by Depth and Strength

Reference is JIRA DOC-266 <https://jira.tapad.com/browse/DOC-266>. Depth and Strength: About Depth measures the distance between devices in the Device Graph. Strength scores measure the relationship between bridged devices in the Device Graph. Set depth and strength values to get the most accurate results.

Return Data Sorted by Device ID

DEPRECATED Reference is JIRA DOC-266 <https://jira.tapad.com/browse/DOC-266>. Return data sorted by device Method: GET. Parameters: deviceID. Returns a list of devices grouped by device. Get device API and SDK methods The following methods let you exchange data between devices.

Return Properties for a Campaign

A GET method that returns details for a campaign. Syntax GET <https://backoffice.tapad.com/api/campaigns/campaignID> Sample response GET <https://backoffice.tapad.com/api/tactics/1234,5>

Return Properties for a Creative

A GET method that returns details for a creative. Syntax GET <https://backoffice.tapad.com/api/creatives/creativeID> Sample response GET <https://backoffice.tapad.com/api/creatives/1234> Sample response { "id": 9876, "name": "Sample creative", "campaignId": 1234 }

Return Properties for a Line Item

A GET method that returns details for a line item. Syntax GET <https://backoffice.tapad.com/api/line-items/lineItemID> Sample response GET <https://backoffice.tapad.com/api/line-items/1234> Sample response { "id": 1234, "name": "Sample_line_item", "campaignId": 1234 }

Return Properties for a Line Item by Campaign

A GET method that returns details for all the line items in a campaign. Syntax GET <https://backoffice.tapad.com/api/line-items?campaignIds=1234> Sample response Returns a list of line items.

Return Properties for a Tactic

A GET method that returns details for a tactic. Syntax GET <https://backoffice.tapad.com/api/tactics/tacticID> Separate tactic from creative. Sample request GET <https://backoffice.tapad.com/api/tactics/1234,5678> Sample response { "id": 1234, "name": "Sample_tactic", "campaignId": 1234 }

Return Properties for an Account

A GET method that returns details for an account. Syntax GET <https://backoffice.tapad.com/api/advertiser/advertiserID> Sample response GET <https://backoffice.tapad.com/api/advertiser/123,456> Sample response A successful request returns account details in a JSON object.

Return Properties for an Advertiser

A GET method that returns details for an advertiser. Syntax GET <https://backoffice.tapad.com/api/advertiser/advertiserID> Sample response GET <https://backoffice.tapad.com/api/advertiser/1234,5678> Sample response A successful request returns advertiser details in a JSON object.

Return Property Details

A GET method that returns details for your property. Syntax GET <https://backoffice.tapad.com/api/property/propertyID> Sample response GET <https://backoffice.tapad.com/api/property/1234> Sample response { "id": 1234, "name": "test_property", "advertiserId": 1234 }

Return Supplier Metrics

A GET method that returns detailed bidding and pricing data for a supplier. Includes URL endpoints, syntax, and parameters. This GET method uses an internal and external endpoint to make API calls. The internal endpoint is <https://backoffice.tapad.com/api/suppliers/metrics>.

Return Supplier Metrics: Sample Requests and Responses

Sample GET requests that let you return real or predicted data for a supplier. Includes data points such as bid amounts, CPM, and impressions. List supplier calls include the real and predict parameters that let you return real or predicted data.

Return Supplier Properties

A GET method that returns properties like ID, name, and a Boolean flag that indicates if we're tracking real and predicted data. This GET method uses an internal and external endpoint to make API calls. The internal endpoint is <https://backoffice.tapad.com/api/suppliers/properties>.

Revenue Opportunity

Click to see potential revenue (before and after margin) for an on-site or off-site audience. This report lets you adjust CPM, targeting, and other variables to see how they impact your bottom line.

margin % to help you discover the optimum revenue settings. Calculations are based on to

Rich Media Reports

Formulas, metrics, and reports related to Rich Media creatives.

RTB Access Data Files: Syntax and Variables

Reference is In an RTB data file, the JSON user and ext objects contain the IDs you can match for cross-device attributi definitions used by these objects in an RTB data file. Syntax An RTB data file confor

RTB Access Integration Process

The integration process with Tapad's Programmatic RTB Access product includes ID synchronization, audience ingestion, RTB 2.1 spec with Tapad's extensions. A Tapad integration expert can help answer questions

RTB Access Overview

Reference is RTB Access is a service that helps you attribute conversions across multiple devices. RTB works in real tir a bid request. For each bid, Tapad attempts to match the original ID to other, rela

RTB Access PDFs

Tapad's RTB Access Requirements and specifications and the IAB Open RTB Specifications in PDF format. Tapad RTB TBD IAB Open RTB Specifications excerpt_pdf_download

RTB Macros

The macros object will allow you to dynamically replace macros in the markup of the creative that you previously submitted descriptions Field Scope Type Example value Description CUSTOM_MACRO Optional String "CU

Run a CTR Report by Day and Hour

Returns data for impressions, clicks, and CTR, grouped by day of week, hour, platform, and tactic. This is one of 3 required spreadsheet. To generate all the required data you must also Run a CTR Report by Tactic, Cr

Run a CTR Report by IAB Category

Returns returns data for impression, clicks, and CTR, grouped by IAB category. This is one of 3 required reports that gene all the required data you must also Run a CTR Report by Tactic, Creative, Platform, an

Run a CTR Report by Tactic, Creative, Platform, and Inventory Type

Returns data for impressions, clicks, and click-through rates grouped by tactic, creative, platform, and inventory type. This the CTR spreadsheet. To generate all the required data you must also Run a CT

Run a Data Tab Report

DRAFT Returns results you paste into the "Data" tab in the External Metrics Template. This is 1 of 2 required reports that g Report spreadsheet. To generate all the required data you must also run the XPlatform Da

Run a Video Completion Rate Report

Follow these instructions to create a report that returns data for video completion rates. To run a video completion report other groupings as required): Select a date range. In the "Group by" section, click

Run a XPlatform Data Tab Report

Returns results you paste into the "XPlatform Data" tab in the External Metrics Template. This is 1 of 2 required reports tha Report spreadsheet. To generate all the required data you must also run the Data Tab

Run-of-Network Tactics for a Brand Awareness Campaign

Reference is Follow these instructions to create optimized tactics for mobile, tablet, and desktop devices in a brand awar campaign is a high-reach/low-cost campaign strategy. It sacrifices control over ad pla

T

T

Delivering specific content to a selected audience. Ads that don't contain images, video, or interactive features. Cookies placed by a service other than the website currently being visited. See Cookies. A record of the

Tactic API

Available methods, syntax, and code samples.

Tactics

Tactics are the rules for ad delivery that you apply or set on campaign line items. For example, you can use tactics to set restrictions like frequency caps, pricing types (CPC, CPM), etc. Refer to the material in this section for more information.

TAJ Browser Cache and Storage Options

Optional methods that let you control how TAJ works with the browser cache and stores data, including storage limits and how TAJ controls how TAJ stores data in the browser cache. Option Syntax Description on_hit_taj

TAJ ID Synchronization Parameters

Pass in device IDs with identify and typed_did. Includes syntax and code samples. Parameters and description TAJ device ID synchronization. Parameter Description identify Accepts alphanumeric device ID strings and cookie

TAJ: Add Audience IDs With add_audiences

Includes syntax and code samples for add_audiences. Syntax and description TAJ parameters that let you add audience IDs to a device. Description 'add_audiences':['audience_ID'] Adds audience IDs to a device and adjacent devices

TAJ: Append Data and Remove Duplicate Records With sadd_data

Includes syntax and code samples for sadd_data. Syntax and description TAJ parameters to add data and remove duplicate records. Description 'sadd_data':{'key':'value'} Appends data to a device and eliminates duplicate record

TAJ: Append Data With add_data

Includes syntax and code samples for add_data. Syntax and description TAJ parameters that let you append data to a device. Description 'add_data':{'key':'value'} When working with add_data: Form data into key-value pairs like 'key':value

TAJ: Assign User IDs to Devices With user_ids

Syntax and code samples for user_ids. Syntax and description TAJ parameters that let you add audience IDs to a device. Description 'user_ids':{'key':'value'} When working with user_ids: Form data into key-value pairs like 'key':value

TAJ: Authenticate With initialize

Syntax and code samples for initialize. Syntax and description Tapestry parameters that let you make authenticated requests. Description initialize Accepts your Tapad provided partner ID. Provides authenticated access to Tap

TAJ: Delete Audiences With remove_audiences

Syntax and code samples for remove_audiences. Syntax and description TAJ parameters that let you remove audience | Description 'remove_audiences':['audience_id'] Removes audience IDs from a device and adjacent

TAJ: Delete Data With remove_data and clear_data

Syntax and code samples for remove_data and clear_data. Syntax and description TAJ parameters that let you remove k following: Syntax Description 'remove_data':{'key':'value'} Removes all specified value data in a

TAJ: Overwrite Device Data With set_data

Includes syntax and code samples for set_data. Syntax and description Tapestry parameters that let you overwrite/replace Description 'set_data':{'key':'value'} When working with set_data: Form data into key

TAJ: Return and Process Data With callback

Syntax and code samples for callback. Syntax and description To process API data with your own code functions, you need The TAJ JavaScript parameters that let you execute a function on returned data include

TAJ: Return Data by Depth and Strength

Syntax and code samples for depth and strength. Syntax and description TAJ parameters used to limit the results generated following: Return/limit data by Syntax Description Depth 'depth':depth_value' Returns

TAJ: Return Data Sorted by Device ID With get_devices

Syntax and code samples for get_devices. Syntax and description TAJ parameters that let you return data sorted by device 'get_devices':true/false Used in conjunction with other request parameters to return data

Tapad (IAB) Browser Supply Targeting

Deliver ads to browsers according to your selected content category. To target by channel, click the Channel field, select a channel. Don't combine Tapad and IAS browser supply targeting on the same tactic. When used

Tapad - Adobe Analytics (SiteCatalyst) Data Integration

Bring Tapad data into your Adobe Analytics dashboard. Includes instructions and code samples.

Tapad - Adobe Analytics Data Integration: About

Purpose, how it works, getting started requirements, and implementation questionnaire. Purpose Tapestry Web Analytics devices visitors use when they interact with your website or mobile app. For example, it

Tapad - Adobe AudienceManager Integration Details

Data synchronization methods and examples. ID synchronization Both Tapad and Adobe initiate the ID synchronization process. Tapad synchronization: Sample call When Tapad makes the ID synchronization call, it looks

Tapad - Google Analytics Data Integration

Bring Tapad data into a custom report in Google Analytics. This material applies to classic analytics and Universal Analytics

Tapad - Google Analytics Integration Guide

Reference material in PDF format. Updated 8/2014. Download or view the PDF.

Tapad - Google Analytics Integration: About

The Tapestry Web Analytics service collects data on the types of devices visitors use when they interact with your website. Tapad sends this data to your Google Analytics dashboard. You can use this data to better understand

Tapad and Adobe Analytics Integration Guides

Reference material in PDF format and video (.mp4) format. Tapad - Adobe Analytics integration guide excerpt_pdf_download reference excerpt_pdf_download Video Setup steps in .mp4 format. With narration (4 minutes)

Tapad as a DMP: Overview and Requirements

Tapad's JavaScript API lets you use our service as a data management platform (DMP). This can be useful if your current or other technologies and platforms. How it works TAJ, Tapad's JavaScript API, provide

Tapad DMP Code Sample

Put this JavaScript code on the pages you want to use for meta tag data collection. This code sends meta tag data to Tapad on your page. You can make function calls against the returned data or send it to an ad server

Tapad Initiated ID Synchronization Code: Syntax, Parameters, and Sample Request

Tapad will send a user it sees to the URL specified. Syntax Our ID synchronization code can send data to secure (HTTPS) example uses the HTTP version. http://<hostname><:port><redirect_uri_prefix>\${TA_DEVICE_ID}

Tapad Opt-out Overview

Tapad's opt-out policies and procedures support industry-standard consumer privacy initiatives and we expect our partners to follow. Tapad partner, you need to be aware that: Tapad process opt-outs in real-time. The ID manager

Tapad Parameters Available in Adobe Analytics

Defines the available Tapad report metrics you can bring into Adobe Analytics. This integration uses a single Analytics event type. http://microsite.omniture.com/t2/help/en_US/reference/conversion_var_admin.html to handle all your Tapad data. The following variables

Tapad Parameters in Google Analytics: Definitions and Syntax

A Tapad - Google Analytics integration provides 3 custom variables you can use for data collection. Tapad data collector defines custom variables that are available for data collection and integration with Google Analytics

Tapad Rate Card | Q1 2014

Ad placement rates for North America. excerpt_creatives_pdfs Contents excerpt_pdf_download Click here to download Tapad Rate Card | Q1 2014. 5MM to 15MM avails < 15MM avails Rate card by audience

Tapad Rate Card | Q2 2014

Ad placement rates for North America. excerpt_creatives_pdfs Contents excerpt_pdf_download Click here to download 5MM to 15MM availss Rate card by audience

Tapad Rate Card | Q3 2014

Ad placement rates for North America. excerpt_creatives_pdfs Contents excerpt_pdf_download Click here to download 5MM to 15MM availss Rate card by audience

Tapad Rate Card | Q4 2013

Ad placement rates for North America. excerpt_creatives_pdfs Download the rate card.

Tapad Rate Cards

Prices and descriptions for the various ad placement options available from Tapad. excerpt_creatives_pdfs

Tapad's Device Graph

Tapad's proprietary Device Graph technology unifies all of the devices linked to a single consumer.

<https://jira.tapad.com/confluence/download/attachments/32670440/image2014-2-20%202012%3A57%3A47.png?version=1&>

Tapestry \${DATA} or \${IDS} Macro

Macros and optional parameters that let you return data from Tapestry. If you're looking for macros you can use in the m Available macros Tapestry accepts the following macros. Macro Description Examples

Tapestry (Deprecated)

Available as PDF only. See the new Tapestry API section for the most current documentation. Obsolete material. Dow documentation.

Tapestry - Android Configuration Guide

Follow these instructions to set up the Tapestry - Android SDK. Code and setup instructions are duplicated on our public G <http://engineering.tapad.com/tapestry/>. To set up the Tapestry SDK for Android: Download <https://github.com/Tapad/tap>

Tapestry - iOS Configuration Guide

Follow these instructions to set up the Tapestry - iOS SDK. Code and setup instructions are duplicated on our public GitHub <http://engineering.tapad.com/tapestry/>. To set up the Tapestry SDK for iOS: Download <https://github.com/Tapad/tap>

Tapestry API Format Method

Syntax and code samples for ta_format. Syntax and description If you don't need to return data in a request with ta_get, 1 default. Sometimes, returning an empty object can cause problems on the page. To

Tapestry API Methods

Includes methods, syntax, and code samples for the Tapestry API family.

Tapestry API Overview

Welcome to the Tapestry API. By itself, Tapestry is a standard API that lets you interact programmatically with the Device defined methods and parameters. It encompasses a larger, related family of API services.

Tapestry API Redirect Method

Syntax and code samples for ta_redirect. Syntax and description Use the Tapestry parameter ta_redirect when you need through an API call. This parameter is available to Tapestry only. Syntax Description ta

Tapestry APIs

Tapestry, TAJ, and the Android and iOS SDKs are a related family of APIs and software development kits (SDKs) that let you and services. This includes HTTP and Javascript methods along with separate SDK code for A

Tapestry Error Codes

Tapestry returns the following IDs and messages when it encounters an error. Response ID Response text Description () encountered an unexpected error that prevented it from fulfilling a request. 1 JsonParseError Tape

Tapestry ID Synchronization Parameters

Pass in device IDs with ta_partner_id and ta_typed_id. Includes syntax and code samples. Parameters and description include the following. Parameter Description ta_partner_id=device ID Accepts alphanumeric

Tapestry PDFs

Tapestry content in PDF format. Tapestry and TAJ API methods Tapestry and TAJ only. No SDK methods. excerpt_pdf_0 SDK code samples and links to the public GitHub site. excerpt_pdf_download All Tapestry API me

Tapestry Reference

Terms, useful facts, tables, or other types of related information.

Tapestry Server-to-Server Parameters

Parameters definitions, and code samples for Tapestry server-to-server (S2S) data transfer methods. S2S data transfers server-to-server data transfers. All the other parameters are optional, but recommended for

Tapestry: Add Audience IDs With ta_add_audiences

Syntax and code samples for ta_add_audiences. Syntax and description Tapestry parameters that let you add audience l Description ta_add_audiences=[{"audience_id"}] Adds audience IDs to a device and adjacent devi

Tapestry: Append Data and Remove Duplicate Records With ta_sadd_data.

Includes syntax and code samples for ta_sadd_data. Syntax and description Tapestry parameters that let you append da include the following: Syntax Description ta_sadd_data={"key":"value"} Appends data to a devi

Tapestry: Append Data With ta_add_data

Syntax and code samples for ta_add_data. Syntax and description Tapestry parameters that let you append data to a devi ta_add_data={"key":"value"} When working with ta_add_data: Form data into key-value pairs

Tapestry: Assign User IDs to Devices With ta_partner_user_id

Syntax and code samples for ta_partner_user_id. Syntax and description Tapestry parameters that let you assign a user Graph include the following: Syntax Description ta_partner_user_id={"key":"value"} Wh

Tapestry: Authenticate With ta_partner_id

Syntax and code samples for ta_partner_id. Syntax and description Tapestry parameters that let you make authenticated Description ta_partner_id=partner_id Accepts your Tapad provided partner ID. Provides authentica

Tapestry: Delete Audience IDs With ta_remove_audiences

Syntax and code samples for ta_remove_audiences. Syntax and description Tapestry parameters that let you remove au Syntax Description ta_remove_audiences=["audience_id"] Removes audience IDs from a device a

Tapestry: Delete Data With ta_remove_data and ta_clear_data

Syntax and code samples for ta_remove_data and ta_clear_data. Syntax and description Tapestry parameters that let yo the following: Syntax Description ta_remove_data={"key":"value"} Removes all specified valu

Tapestry: Return and Process Data with ta_callback

Syntax and code samples for ta_callback. Syntax and description To process returned data with your own code functions information in a JSON wrapper. Tapestry parameters that let you return data in a JSON wrapper

Tapestry: Return Data by Depth and Strength

Syntax and code samples for ta_depth and ta_strength. Syntax and description Tapestry parameters used to limit the res include the following: Return/limit data by Syntax Description Depth ta_depth=depth_value R

Tapestry: Return Data by Device ID With ta_get_Devices

Syntax and code samples for ta_get_devices. Syntax and description Tapestry parameters that let you return data sorted Description ta_get_devices=truefalse Used in conjunction with other request parameters to

Tapestry: Set or Overwrite Device Data with ta_set_data

Syntax and code samples for ta_set_data. Syntax and description Tapestry parameters that let you overwrite/replace dat Description ta_set_data={"key":"value"} When working with ta_set_data: Form data into key-v

Target by Expiring Contracts

Follow these instructions to target with the expiring contracts settings. To target by expiring contracts: Go to the tactic you Click Audience Assignments. In the "Include" or "Exclude" fields, start typing

Targeting

Control delivery with targeting rules and features. Includes setting definitions and requirements.

Targeting Constraints and Overdelivery

Control overdelivery with targeting requirements. These rules force your tactics to make bids only when requests meet spe targeting constraints. Troubleshooting steps Key indicators Recommendations Expectations

Targeting Fees

Many targeting features are free but others apply a sliding or flat-rate fee against your bid price. When you choose targetin each related tactic. Download the fee instruction sheet for more information. Note

Technical Specifications for Creatives (deprecated)

The Technical Specifications for Creatives has moved to the Self-Service DSP space. Old specs are available in PDF form

Technical Specifications for ID Bridging

DRAFT Reference is JIRA DOC-165 <https://jira.tapad.com/browse/DOC-165> and this Google doc <https://docs.google.com/a/tapad.com/document/d/1v3ch3DSOiaHM1Ynee0k-bQQRoY8hLHsmMw7FBqHD5Do/edit>. ID br hashed use

Technical Specifications for Programmatic RTB Access

Tapad Exchange is a way for programmatic buyers to access Tapad's proprietary cross-platform data while also accessing specifications are largely based on the OpenRTB 2.1 <http://www.iab.net/media/file/OpenRTB-API-Specif>

Technical Specifications for RTB Access Data Files

Reference is An overview of RTB Access and information about file formats, syntax, examples, and delivery options.

Technical Specifications for the Impression Log File

The Impression Log File provides data about impressions and clicks across bridged devices in the Tapad Device Graph. It your first or third-party audiences. Tapad can deliver the report to you in a compressed,

Technical Writing Standards

Rules and guidance for grammar, page format, UI elements, punctuation, tables, lists, numbers, problematic terms, and to DITA standards for technical writing http://en.wikipedia.org/wiki/Darwin_Informati

Technology Vendor IDs and Names Enumerated

IDs you can use to set the technology vendor for a creative. The technology vendor enum includes the following IDs and EyeWonder 5 eyeReturn 6 Pictela 7 Flashtalking 8 Doubleclick 9 Unicast 10 Spongcell 11 Adrime

Text Data File With Platform Types

Syntax, definitions, and examples for synchronized ID data in a tab-separated (.tsv) text file, including optional platform typ that describes the device itself and can include results like iPad, Android, Kind

Text Data File With Scores

Syntax, definitions, and examples for synchronized ID data in a text file, including optional scores that quantify the relation suggests a closer connection between devices than a lower score. These scores he

Text Data Files: Syntax and Content Options

Syntax for receiving synchronized IDs in tab-separated (.tsv) text file. Content options include basic ID data, IDs with score

Text Format for Audience Data Files (Not Used)

Tapad does not use this text file format. Documentation retained for legacy purposes only. See . A text-formatted audience segment IDs in a simple text file. You must format the parameters in the audience dat

Text Format for Audience Taxonomy Files (Not Used)

Tapad does not use this text file format. Documentation retained for legacy purposes only. See . Format the taxonomy to the following table lists and defines the data elements in a properly formatted audience

The Shoals of Grammar

A list of common elements, descriptions, and examples. Contents Acronyms and abbreviations Acronyms On first use in parentheses. Use the acronym on any subsequent mention. Consider a glossary link to a

The Tapad User Opt-out Process: About

An overview of how Tapad opt-out processes work and how we send consumer opt-out data to our partners. See the Tapad privacy policy <http://www.tapad.com/privacy-policy/>.

Third-party Ad Server (3PAS) Attribution

Reporting discrepancies between different ad serving systems are common in digital advertising. Mobile devices and apps these devices don't accept cookies, which ad servers use for conversion and attribution r

Third-party Audience Targeting Tactics With a Brand Awareness Campaign

Reference is Follow these instructions to create optimized tactics for mobile, tablet, and desktop devices that use third-party targeting. The goal of this type of targeting is to reach a specific target audience and gen

Tiger Pricing With Optimization

Add optimization and learning to a tactic to help improve bidding behavior. By itself, Tiger Pricing maximizes margin and profit. If bids the lowest amount possible, it often wins on lower-price, lower-performance inv

Tiger Pricing: About

Tiger Pricing is a bidding feature for CPM tactics. This pricing model attempts to spend least amount required to win an impression. Pricing features let you apply historical learning when determining a bid amount, which

Time and Device Options for Frequency Caps

These required frequency settings let you control impressions by time (minutes, hours, days) and device groups (household, device). Control ad exposure by time. Time intervals include: Minutes Hours Days Device options Co

Time and Location Targeting

Deliver content according to time, day, location, domain name, or with custom settings.

Traffic Click-to-call Creatives

Follow these instructions to traffic and set up click-to-call creatives. Click-to-call ads are image creatives that use a phone number as a clickthrough URL. Clicking the creative opens a dialog that prompts the user to call t

Traffic Cobalt Ad Tags

Follow these instructions to traffic Cobalt ad tags. You must modify Cobalt ad tags before trafficking them in the Tapad UI. Add a new Cobalt creative. Browse to the campaign you want to work with, click the

Traffic display expandable Rich Media creatives

Follow these instructions to traffic display expandable Rich Media creatives. To traffic display expandable Rich Media creatives: Step 1: Go to the Creative tab, click New. In the Creative pop, use the current campaign or select a campai

Traffic mobile Rich Media creatives

Follow these instructions to traffic mobile Rich Media creatives. All mobile Rich Media creatives must be MRAID1 compliant. To traffic mobile Rich Media creatives: Step 1: Add new creative From the Creative tab, click New. In the

Traffic Site-Served Creatives With the Bulk Creative Uploader

Follow these instructions to upload multiple creatives and assign placements with the Bulk Creative Uploader. For site-served banner creatives: Step 1: Upload multiple creatives and assign placements with the Bulk Creative Uploader. For banner creative data file A creative data file contains creative assets and a spreadsheet that

Traffic Site-Served, Third-Party Tracking Creatives With the Bulk Creative Uploader

Follow these instructions to upload multiple creatives and add third-party click tracking with the Bulk Creative Uploader. For site-served, third-party tracking creatives: Step 1: Upload multiple creatives and add third-party click tracking with the Bulk Creative Uploader. For site-serv

Traffic Tapad Hosted Video Creatives

Follow these instructions to traffic Tapad hosted video creatives. To traffic third-party video creatives see Traffic Third-party. To traffic Tapad hosted video creatives: Step 1: Browse to campaign you want to work with and click the Creatives tab. Click New.

Traffic Third-Party Creatives With the Bulk Creative Uploader

Follow these instructions to upload third-party creatives with the Bulk Creative Uploader. For banner, expandable, and interstitial creatives: Step 1: Upload third-party creatives with the Bulk Creative Uploader. For banner, expandable, and interstitial creatives

Traffic Third-party Video Creatives

Follow these instructions to traffic video creatives hosted by third-party ad servers. To traffic Tapad-hosted video creatives: Step 1: Add a new creative. Browse to campaign you want to work with and click the Creatives tab. Click New.

Trafficking Creatives

Instructions to help you upload creatives in the Backoffice UI.

Turn Integration Overview

The Turn Audience Suite is a platform that provides its customers with a system to manage and build audiences with online audiences. Tapad share synchronized audience segments with Turn and make and make that data available.

V

V

Ads that contain video. See Technical Specifications for Creatives <https://confluence.tapad.com/display/DOC/Technical+Specifications>

Vehicle Preference

Deliver ads based on vehicle likes and desires. To target by vehicle, click the Vehicle preference field, select a vehicle type and fees apply. Vehicle preferences include: Compact car Coupe Luxury Crossover Exo

Video Report Metrics Defined

Describes the available report metrics for video creatives. Report metrics for video creatives include: Report metric Count inside the video player and the player loads on the page. Video start The video player starts

Video Reports

Formulas, metrics, and reports related to video creatives.

Viewing Report Data in Google Analytics

To see your custom report in Google Analytics, go to the Standard Reports sidebar and select Audience > Custom > Custom Reports. For more information, see [Customizing Standard Reports](https://support.google.com/analytics/answer/1033013?hl=en&ref_topic=1) https://support.google.com/analytics/answer/1033013?hl=en&ref_topic=1

Visited Platforms

Visited Platforms counts user visits by platform group. Over time, this data can help you understand how many platform types your users visit. For more information, see [Understanding platform groups](#) For Visited Platforms, a platform

X

Z

Z

Cookies that return or persist even after being deleted.

ZIP Codes

Deliver ads to specific postal code areas. To target by postal code, click the ZIP codes field, type a ZIP code and click Save. For example, if you target by ZIP code, your target settings could look similar to this: 10001,10025,100

zz_inclusions library

Workspace for draft content, testing, excerpts, and content that is not ready for use or circulation. These are not the docs you