



Adobe® Marketing Cloud

DFA Data Connector for Adobe Analytics

Contents

- DFA Data Connector for Adobe Analytics.....3**
 - Key Benefits.....3
 - Ad Serving Integration Overview.....3
 - Adobe Integration: Realtime Data Collection.....4
 - Adobe Integration: Nightly Data Import.....6
 - Version Differences.....6
 - Prerequisites.....8
 - Integration Features.....8
 - Analytics Variables and Events.....10

- DFA Integration.....12**
 - Configuring the Integration.....12
 - Web Site Updates for the DFA Integration.....14
 - Verify Cookie Space on the Domain.....14
 - Update your DFA Query-String Parameter.....14
 - Update Your Web Site's Data Collection Code.....15
 - Confirming a Successful DFA Integration.....16
 - Tuning s.maxDelay.....17

- Analytics Reports Using DFA Data.....19**

- Reconciling Metric Discrepancies.....21**
 - Metric Definitions.....21
 - Possible Reasons for Discrepancies.....21

- Frequently Asked Questions.....23**

DFA Data Connector for Adobe Analytics

In today's increasingly complex and competitive online marketplace, on-line advertisers and agencies must continually improve their understanding of the on-line marketing environment, and their return on advertising spend. Although advertisers, agencies, and publishers all possess individual tools to help accomplish these objectives, manually aggregating data from disparate data systems and processes can severely hamper the effectiveness of on-line marketing campaigns, resulting in less-than-optimal campaign performance, data discrepancies and confusion.

The DoubleClick for Advertisers (DFA) integration solves this problem by using Adobe® Data Connectors™ to allow DoubleClick DFA to automatically pass data to Reports & Analytics.

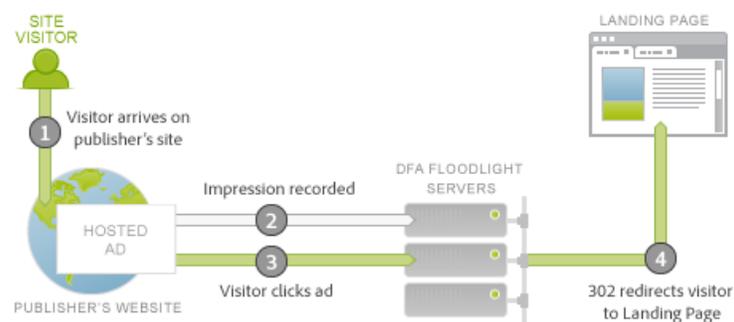
Key Benefits

Key Benefits of the Data Connector - DFA integration include:

- **Increased conversion:** Gain directional insight to optimize ad campaign placement and on-site conversion based on post-click visitor behavior and preferences.
- **Shared location for data:** Combine DoubleClick DFA click-through and view-through data with Reports & Analytics to improve cross-organizational collaboration and abilities to make objective decisions.
- **Value-added analysis:** Automated integration between DFA and Adobe Reports & Analytics allows advertisers and agencies to spend less time crunching data and more time analyzing reports and taking action.
- **Deeper customer insight:** Gain greater insight into where visitors are coming from and what they are doing on your site.
- **Lifetime success metrics:** Measure the effectiveness of your acquisition campaigns across the entire visitor life cycle.
- **Integrated Reporting:** Automatically synchronize data between DFA and Reports & Analytics for streamlined business processes and reporting.
- **Lifetime Visitor Analysis:** Measure campaign effectiveness by multiple user-defined success events and lifetime value.
- **Cost Metrics:** Optimize return on investment by comparing DFA cost figures and revenue generated from those costs in a single system.

Ad Serving Integration Overview

There are several ways in which this integration captures data about the ad-driven visitor. The first way is via clicking on an ad and arriving on a tagged landing page, called a click through:

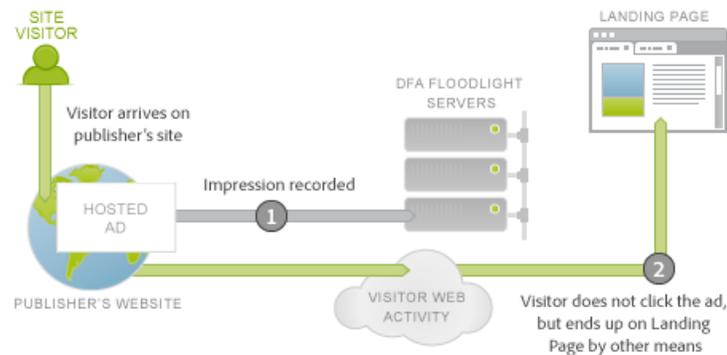


The visitor arrives upon a publisher's site, which hosts the Ad. This Ad has a unique identifier, called the Ad ID. Ads comprise a Placement plus a Creative, which describe where the Ad is on the Publisher's site and what content was shown to the visitor. When the visitor fetches this Ad/Placement/Creative from the DFA content servers, it tracks an Impression to the DFA Floodlight Servers for this visitor (1).

If the visitor clicks on the ad (2), the Floodlight Server is queried, which counts a click, then 302 redirects (3) the visitor to the Landing Page. When the visitor has arrived upon the Landing page, this is termed a click-through. This page contains Adobe tracking code which queries data from the DFA Floodlight Server.

If the visitor does not actually arrive on the Landing Page after the Floodlight Server has tracked a click, this is not termed a click-through. Some ads and implementations may not actually cause the visitor's browser to obey the 302 redirect. For further discussion on this topic, see .

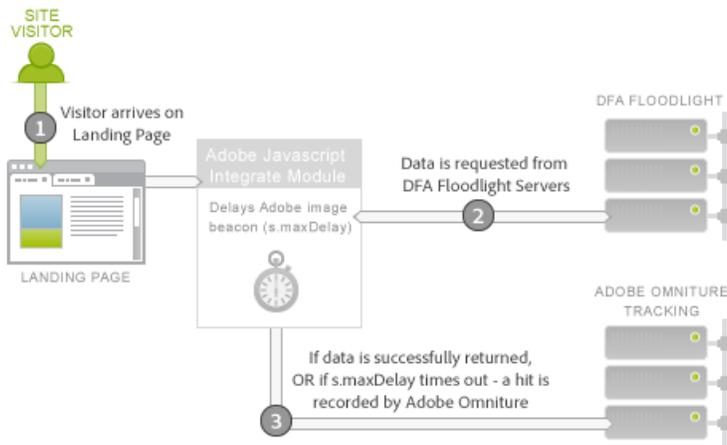
The next metric captured by this integration occurs when the visitor receives the Ad impression, does not click, yet sometime in the near future arrives upon the Landing page by another means.



This scenario is termed a view-through. The difference in this scenario with the click-through scenario is that the visitor does not click on the Ad, but instead continues to other activities before coming to the Landing page (2). In the most simple case, the visitor types in the landing page's URL in the browser. In other cases, the visitor continues browsing but later uses a search engine, which drives the visitor to the landing page. In any case, the user arrives upon the landing page.

Adobe Integration: Realtime Data Collection

The following figure shows how data collection works.



The data collection portion of the Adobe integration begins when the visitor arrives to the landing page (1). The Adobe data collection code running on the landing page has no knowledge of the history the visitor has had with served ads. The Google DFA team has coordinated a service running on the DFA Floodlight Server to allow the Adobe code to query ad information about the visitor currently on the site (2). To obtain this data, it temporarily delays the Adobe image beacon, and requests the data from the Floodlight Server.

Once the data arrives, or takes too long, it fires the hit to the Adobe tracking servers (3).

The Integrate module is a special core Adobe JavaScript module which causes the Adobe image beacon to delay, waiting on a 3rd party request for a specific amount of time (`s.maxDelay`). `s.maxDelay` defines how long the Integrate module will wait for data from the DFA Floodlight Server before firing the image tag to the visitor's browser. This behavior is important so that basic visitor data is still collected, even when the DFA Floodlight Servers are down or heavily loaded. If the Floodlight data arrives before `s.maxDelay` has expired, the Adobe tracking data will be fired immediately, and will contain the additional DFA data.

When a timeout occurs, the page code can specify an Adobe Reports & Analytics Event to be used as a Timeout Event. This event is useful when diagnosing problems with the integration, or when adjusting `s.maxDelay`. In cases where there are excessive timeouts, increase `s.maxDelay`. `s.maxDelay` can be set too high, however, in which cases visitors could have the potential of leaving the site prior to the `s.maxDelay` timer expiring. For more discussion on this topic, see [Tuning s.maxDelay](#).

Sometimes the Floodlight Server responds with errors about the visitor. This usually occurs when the Floodlight Server does not know anything about the visitor, because the visitor has not yet seen any ads, or does not have a DFA visitor cookie. The page code can specify a Custom Conversion variable (eVar) which will collect these errors, and can aid in troubleshooting implementation problems or point out issues with the Google transaction. The most common errors are No History, No Cookie, Query Error, and Opted Out, as described in the following table:

Error	Name	Description
nh	No History	The visitor has not viewed or clicked any ads.
nc	No Cookie	The visitor does not have a DFA visitor cookie.
qe	Query Error	There was an error querying data for the Floodlight Server.
oo	Opted Out	The visitor opted-out of Google impression/click tracking.

Adobe Integration: Nightly Data Import

The data collection portion of the integration collects click-through and view-through data about site visitors. To obtain DFA click, impression, and cost metrics, there is a nightly process coordinated by Google and Adobe to import this additional data to the integrated report suite. These metrics are imported through Data Sources, meaning they are available in aggregate only, and are not at the visit level. See [Integration Features](#) for more details.

Version Differences

There are currently three versions of the DFA integration, versions 1.0, 1.5, and 2.0.

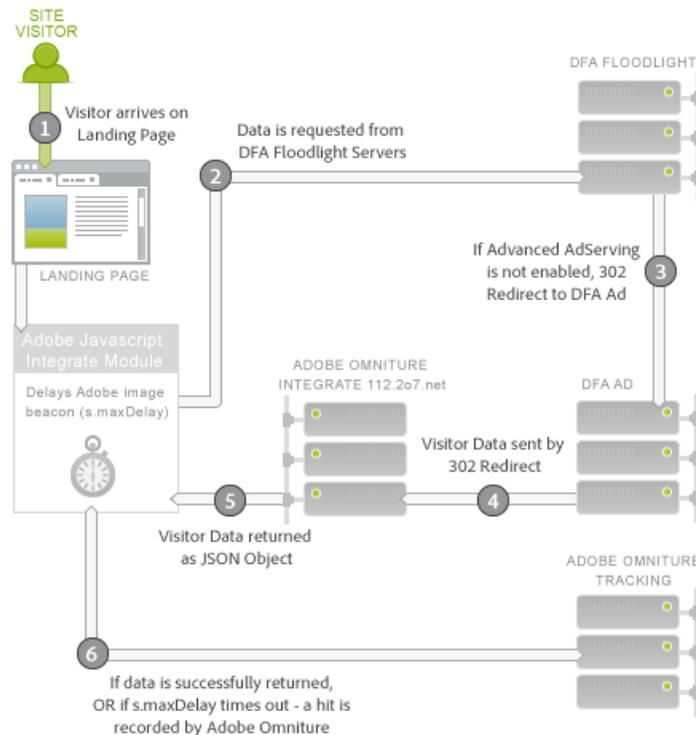
The following table summarizes the features in each version of the integration.

Feature	Version 1.0	Version 1.5	Version 2.0
DFA Click and Impression Nightly Metrics	Yes	Yes	Yes
Click-through and View-through Tracking	Yes	Yes	Yes
Integration receives data on an Advertiser level	No	Yes	Yes
Integration receives data on a Floodlight Configuration level	No	No	Yes
Cost Metrics	No	No	Yes
Creative Metrics	No	No	Yes
Query strings beyond 2k bytes	No	Yes	Yes
Uses Integrate module for optimal 3rd party data collection	No	Yes	Yes
Timeout and Error tracking	No	Yes	Yes
No need for negotiated Client Side ID	No	No	Yes

About Version 1.5

Version 1.5 of the integration introduces the Integrate module to the landing page Java Script. The Integrate module allows for fixed sized requests to the DFA ad server (ad.doubleclick.net) which overcomes the 2K request limits of the previous integration. It also introduces a configurable timeout, s.maxDelay, to still collect Adobe visitor data when network outages occur. Errors and timeouts can also be captured into Analytics variables.

The following illustration shows network interactions on the landing page in version 1.5.



In version 1.5, the Integrate module (2) requests data from the Floodlight Server (3). The Floodlight Server will redirect to the DFA ad server, which will return data about the visitor in the same way as version 1.0. It will 302 redirect (4) to a special translator service on integrate.112.2o7.net, which will turn the response structure into a JSON object. The Integrate module consumes this JSON object and passes the information along to the Adobe Tracking (5).

Moving from Version 1.0 of the integration to 1.5 involves a JavaScript change. To obtain this JavaScript, log into your Adobe Online Marketing Suite account, choose the Genesis product, click Edit on your DFA integration, and proceed through the wizard. Provided a Client Site ID has previously been assigned, once you have saved the Integration you will immediately receive the new JavaScript code via email. Once you have this code, you will also need a new version of the core s_code which has the Integrate module. This code can be requested from your Account Manager or Implementation Consultant.

An important feature of the new JavaScript code is that there is no implementation change necessary between version 1.5 and version 2.0.

About Version 2.0

The latest version of the DFA integration brings data for an entire Floodlight Configuration into the integration. Previous to version 2.0, individual integrations were tied to a single DFA Advertiser. With this change, Clicks, Impression, and Cost metrics for the entire Floodlight Configuration will be included in the integrated report suite. It is also possible to track cross-site view throughs, when those two sites are inside the same Floodlight Configuration.

Media Cost metrics are also available as of version 2.0 of the integration. To enable media cost metrics for an integration, you must choose a Reports & Analytics event for Media Cost in the Genesis wizard, as well as specify what currency the metric figures are in the DFA interface.

Timeouts are expected to decrease with the 2.0 integration, since the 302 redirects have been eliminated. Eliminating these hops should decrease timeouts, and increase the amount of DFA data you can integrate.

If a Floodlight Configuration is a shared configuration in DFA, the upgrade from version 1.5 to 2.0 causes conversion data for all shared advertisers within the Floodlight Configuration to be included in the report suite.

Upgrading to Version 2.0

The following table outlines the owners for migration to newer versions of the integration:

Migration	Owner	Tasks
Version 1.0 to 1.5	Client	Implement Version 1.5 JavaScript with Integrate Module
Version 1.5 to 2.0	Client	Client begins discussion with Google about time frames for upgrading. After approval, Google enables Advanced Ad Serving.

Prerequisites

Before starting the Adobe Data Connectors integration for DFA, do the following:

- Decide whether to integrate against version 1.5 of the integration, or to wait for version 2.0. This decision is dependent upon which features are utilized in your DFA account, and the time frame in which you want to integrate. See [About Version 2.0](#) for more information.
- Decide how DFA Advertisers will map to Adobe Analytics report suites. For example, if you have multiple DFA Advertisers and multiple report suites, you will need to decide which Advertisers pair with which report suites.
- Implement Adobe data collection code on all the pages you want to track, using version H.22 or later of the data collection code.
- Know the Advertiser ID for a DFA account that is part of the Floodlight Configuration you want to integrate. The integration automatically imports all of the Advertisers that lie within the Floodlight Configuration.
- Know your DFA account User Name and Password.
- Associate each DFA Advertiser to only one Adobe Analytics report suite. Tagging to multiple report suites is not supported with the default Genesis integration for DFA.
- Add a click-through query string parameter to the landing page for each DFA Placement which will be part of the integration. This query string parameter is necessary to properly count Click-throughs.
- Configure your DFA Placements so that they do not redirect visitors through multiple domains. For example, a Placement should not direct respondents to a micro-site hosted under [www.xyz.com](#) if the micro-site then redirects them to another site, [www.fgh.com](#). If the campaign response spans multiple domains, click-through and view-through data can be inflated and misleading.
- Identify a custom variable in Reports & Analytics to hold your campaign information.
- Identify a Reports & Analytics eVar to store DFA view-through information. Use this eVar only for this DFA integration.
- Identify the Reports & Analytics events where you want to store Impressions and Click data. You might want to rename these events appropriately.
- (Optional) Identify the Reports & Analytics events which will store DFA Cost data. You might want to rename these events appropriately.
- (Optional) Identify a Reports & Analytics Custom Variable and Success Event which will store DFA errors and timeouts. These variables help to diagnose problems that might arise with the integration.
- (Optional) Create a special Email account for receiving information and notifications related to the Data Connectors integration for DFA.

Integration Features

Once activated, the Data Connectors DFA integration provides the following metrics for your Adobe Analytics reports:

- View-throughs
- DFA Clicks
- Impressions
- (optional) DFA Cost Data
- (optional) DFA Query errors, Timeouts



Note: This integration does not provide support for click trackers (formerly click commands). Click trackers are used to record the number of clicks on text links, links in email messages, or on other elements that are hard-coded on a Web site.

The Data Connectors DFA integration automatically constructs DFA tracking codes from the data returned by DFA. These tracking codes are constructed to uniquely identify an Ad along with its associated Placement and Creative. The following outlines the structure of the tracking code, depending on version of the integration. Version 1.5 looks like this:

DFA: 123456 : 78901234 : 56789

Site ID
Placement ID
Ad ID

Version 2.0 looks like this:

DFA: 123456 : 78901234 : 56789 : 012345678

Site ID
Placement ID
Ad ID
Creative ID

These IDs serve as a shared key between Genesis and DFA to associate the correct classifications and metrics.

Site ID	The 3rd party site on which the ad was hosted. The Site Name classification provides a descriptive name of this Site ID.
Ad ID	An ID for the commercial message that is delivered to a user. The Ad Name classification contains the name of the ad as defined by your organization in the DFA system. For example: Hybrid Coup Textlink - Build.
Placement ID	A representation in your DFA account of a Web site, portion of a Web site, or group of Web sites where you have purchased ad space.
Creative ID	The image, Flash SWF, or other resource that is intended to be shown to the visitor. The Creative Name classification contains the name you have provided to this creative in the DFA interface.

The other two classifications, Delivery Tool (DoubleClick for Advertisers) and Channel (Banner Ad) have the same values for any DFA campaign and help to distinguish DFA imported data.

SearchCenter De-Duplication

The DFA integration is now Adobe SearchCenter aware. By enabling SearchCenter de-duplication via the Data Connectors wizard, search driven visitors will not cause data to be pulled from DFA's Floodlight Server, and s.campaign will not be filled

in by DFA, thus allowing SearchCenter to populate it. Also, DFA and SearchCenter now populate de-duplication values into the variables for each product.

The list below outlines the logic that is enabled when SearchCenter de-duplication is enabled:

If **DFA > SearchCenter deduplication** is selected in the wizard:

- In the case of a DFA click-through, the integration will populate the string “DFA Clickthrough” to the configured SCM eVar.
- In the case of a DFA view-through, the integration will populate the string “DFA Viewthrough” to the SCM eVar.

If **SearchCenter > DFA deduplication** is selected in the wizard:

- In the case of a DFA view-through, the integration will populate the string “DFA Viewthrough” to the SCM eVar.



Note: If SearchCenter > DFA de-duplication is enabled and the SearchCenter query string parameter is set, the visit will not be considered for DFA processing. This means that the SearchCenter query string parameter should be different from the DFA click-through parameter, and no Display ads should set the SearchCenter query string parameter.

Analytics Variables and Events

The Data Connectors integration for DFA uses Analytics variables to track DFA campaign results.

Other than the campaign variable, you can use the Analytics Events and eVars that make sense to you. Once you have identified the Event and eVars to use with this DFA integration, use the Analytics Admin Console to enable them. The integration variables must be enabled before activating the DFA integration. The following table describes the Analytics variables needed for the DFA integration.

Variable	Friendly Name	Population Method	Description
s.campaign or eVar	Ad Tracking Code	Automatically populated by Data Connector for DFA campaigns.	Tracks click-through conversions for all campaigns.
eVar*	View-Through	Automatically populated through VISTA and DFA for DFA campaigns.	Tracks view-through data for DFA IDs. This eVar should have the same expiration as the s.campaign variable. Must be the same conversion variable as identified in the Variable Provider ID (See Update Your Website's Data Collection Code). Make sure the eVar has full subrelations enabled. The cost for enabling this feature is part of the Data Connectors integration fee
eVar*	DFA Query Errors	(Optional) Populated through Javascript collection code.	Contains one of several error codes returned from DFA (see the table in Configuring the Integration for a listing of these error codes)
event*	View-Through Count	Automatically populated by Data Connector for DFA campaigns.	Captures the number of times users viewed an ad, did not click-through, but arrived at your site.
event*	Impressions	Automatically populated through a data feed from DFA.	Tracks the number of time a specific DFA ad was served.

Variable	Friendly Name	Population Method	Description
event*	Clicks	Automatically populated through a data feed from DFA.	Tracks the number of times users clicked a specific DFA banner ad. This metric can yield different numbers compared to the native Analytics click-throughs metric for one of several reasons. See Reconciling Metric Discrepancies for more information.
event*	DFA Timeouts	(Optional) Populated through Javascript collection code.	Counts the number of times DFA fails to respond before the <code>s.maxDelay</code> time out. This can help you determine if there is a DFA implementation issue.
event*	DFA Media Cost	(Optional) Populated through Javascript collection code.	Contains the media cost metrics that have been input in the DFA interface. This feature must be enabled on the DFA side in order for these metrics to appear.

*Select an unused eVar or Custom Event.

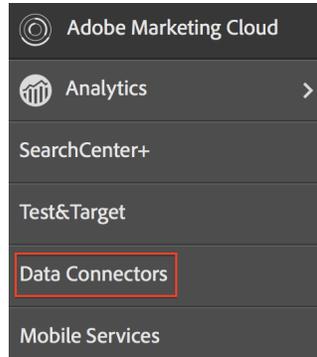
DFA Integration

Configuring the DFA integration involves the following tasks:

Configuring the Integration

The Data Connectors Integration Wizard steps you through the DFA integration.

1. Log in to the Adobe Marketing Cloud.
2. Click **Data Connectors**.



3. Click **Add New**.
4. Launch the Integration Wizard by dragging the DFA icon to an empty plug-in slot.
5. On the DFA integration introduction page, select the checkbox to accept the fees associated with the DFA integration, then click **Next**.

This page provides an overview of the integration, along with helpful links for more information. There are both Adobe and DoubleClick fees associated with this integration. Contact your appropriate Sales Representatives for both organizations and make sure you understand the fee structure.

6. On each page of the Integration Wizard, provide the required information, then click **Next**. The following table explains the information you need to complete the integration through the wizard.

Wizard Page #	Field	Description
1	Integration Name	The integration name that Genesis displays in the report suite's Active Integration List.
1	Integration Email Address	The email address that receives all notifications related to this integration.
2	User name	The DFA API username to use with this integration. To enable a user for API login, check the API attribute in the DFA interface. After you enable API login, a password field appears to provide a password for the user. This password is entered along with the username into the wizard to authenticate.
2	Password	The DFA API password.
2	Advertiser ID	The DFA advertiser ID or the parent Floodlight Configuration ID. Data Connectors uses this ID to identify the DFA Advertiser to track (version 1.5 of the integration). This Advertiser ID is not used in version 2.0 of the integration.

Wizard Page #	Field	Description
		– the parent Floodlight Configuration ID will be looked up and used. See the instructions on the screen
3	DFA Ad Variable	<p>The Analytics eVar that receives DFA campaign attribute, impressions, and clicks data. Typically, this is the Tracking Code eVar (s.campaign), but you can choose any available eVar. Data Connectors also adds the following DFA-related classifications to the selected eVar:</p> <p>Campaigns: A collection of ads served to multiple sites that carry common messaging.</p> <p>Site Name: The site where the ad was served.</p> <p>Ad Name: The Ad name, as defined in your DFA account.</p> <p>Site Placement Name: The Web site and page where the Ad was served.</p> <p>Delivery Tool: DoubleClick for Advertisers.</p> <p>Channel: Banner Ad.</p> <p>Cost Structure: CPM, CPC, or Fixed, based on the cost structure of the ad.</p> <p>Creative Name: The name of the creative associated with an ad/placement/creative ID.</p> <p>DFA > SearchCenter Deduplication: Specifies that DFA should place values in Searchcenter variables when DFA Click-throughs or View-throughs occur. For more information, see SearchCenter Deduplication .</p>
4	Impressions	The Custom Event that receives DFA Impressions metric data. Impressions indicates the number of times the ad was served.
4	Clicks	<p>Select the custom Event that receives DFA Clicks metric data. Clicks indicates the number of times visitors clicked on the ad as measured by DFA's redirect. The Clicks metric correlates with the Analytics Click-throughs metric.</p> <p> Note: DFA Clicks and Analytics Click-throughs might not match exactly due to differences in the way data is collected. For more information, see Metric Definitions.</p>
5	View-Through Variable	<p>The Analytics eVar that receives DFA View-Through data. The View-Through variable helps you see how view-throughs affect conversion rates on your site.</p> <p>Data Connectors adds the same DFA-related classifications to this eVar as it does to the DFA Ad Variable (see above).</p>
5	Time Since Last View (view-through time bucket variable)	The Analytics eVar that receives DFA Time Since Last View data. The Time Since Last View indicates the amount of time that has lapsed since the last ad view-through.
5	View-Throughs	The Custom Event that receives DFA View-Throughs metric data. Use the View-Throughs event with the View-Through Variable to see which campaigns

Wizard Page #	Field	Description
		<p>didn't influence a direct click-through, but may have played a role in driving traffic to the site at some subsequent time.</p> <p>Data Connectors renames the selected Custom Event to "View Through".</p>
6	DFA Query Failure	<p>(Optional) The Analytics eVar that receives any reported DFA query failure message codes. Possible DFA message codes include:</p> <ul style="list-style-type: none"> • nc: No DoubleClick cookie. • oo: User has opted out. • nh: No campaign history. • qe: Query error (time out, server down, etc.)
6	Timeout Event	<p>The Analytics Counter Event that increments each time the s.maxDelay timer expires, and not response was received from the DFA servers. Use this event to configure the s.maxDelay variable (see Tuning s.maxDelay.)</p>

Web Site Updates for the DFA Integration

Once Genesis has configured your Analytics report suite for the DFA integration, you must do the following to configure your Web site and DFA environment to support the integration:

Verify Cookie Space on the Domain

The Data Connectors integration for DFA requires you to set a cookie on the domain of the page.

Although it is rare, some domains have reached the maximum cookie capacity for some Web browsers. To avoid affecting a visitor's browsing experience on your Web site, consult with your network operations, development team, or engineering group to verify that adding another cookie to the domain of the pages used for the DFA integration will not affect the user experience. You will also need to select a name for the cookie.

Update your DFA Query-String Parameter

If you have already been tracking Ad campaigns with Adobe Analytics prior to the DFA integration, it is possible that all campaigns (email, search, or banner) use the same query string parameter to identify the referring campaign ID on the landing page.

To understand when to request view-through and click-through data from DFA data for your DFA Ad campaigns, Data Connectors needs to identify when a visitor has clicked on a DFA campaign banner ad. To make this possible, you must add a differentiated query-string parameter to the DFA Ad campaign's landing page URL so Data Connectors can distinguish between DFA Ad campaign pages and other ad campaign pages that you might have on your Web site. The `dfa_overrideParam` in the JavaScript plug-in (see [Update Your Web Site's Data Collection Code](#)) determines the query string parameter used for DFA.

: Although the Campaign variable can be used for other campaigns, do not use it for DFA campaigns. If you set the Campaign variable to a DFA campaign landing page, Adobe cannot tie impressions and clicks to DFA campaign click-throughs. Once per visit, Adobe collection servers check DFA servers for a previous click- or view-through. Because of this, include the DFA plug-in code (see [Update Your Web Site's Data Collection Code](#)) only on common landing pages to avoid unnecessary redirects that can slow page-load times, particularly for users with slower Internet connections.

Update Your Web Site's Data Collection Code

The Genesis integration for DFA leverages the DFA Floodlight Configuration ID (`dfa_SPOTID`), which improves report consistency between DFA and Adobe data collection system.



Note: The term *Spotlight* was changed to *Floodlight* in a recent release of Google DFA. The JavaScript parameter `dfa_SPOTID` was named based on the *Spotlight* terminology but is used for both versions.

To enable the DFA integration on your Web site, you must update your JavaScript data collection code by adding the following:

- Integrate Module for DFA
- Addition to your Collection Code

Integrate Module for DFA

The DFA integration leverages the Adobe Marketing Cloud Integrate Module, which adds functionality to your core JavaScript data collection code (`s_code.js`). Contact your Implementation Consultant, or Account Manager, to get a copy of the Integrate Module.

Insert the Integrate Module code in the `Modules` section of your website's `s_code.js` file.

Addition to Your Collection Code

Based on your selections when activating the DFA integration in the Integration Wizard, Data Connectors generates and emails you a customized addition to your JavaScript data collection code. Insert this code in the main section of the `s_code.js` file (not in the `doPlugins` function or any other function).

The sample code shown below is for illustration only; use the code that was emailed to you after you completed the Data Connectors Integration Wizard.

The collection code consists of the following components:

- DFA Integrate Settings
- Integration-Required Plug-ins

DFA Integrate Settings

```

/***** DFA VARIABLES *****/
var dfaConfig = {
  CSID:          "1234567",
  SPOTID:       "29876543",
  tEvar:        "eVar17",
  errorEvar:    "eVar59",
  timeoutEvent: "event76",
  requestURL:   "http://fls.doubleclick.net/
json?spot=[SPOTID]&src=[CSID]&var=[VAR]&host=integrate.112.2o7.net%2
Fdfa_echo%3Fvar%3D[VAR]%26AQE%3D1%26A2S%3D1&ord=[RAND]",

  maxDelay:     "1500",
  visitCookie:  "s_dfa",
  clickThroughParam: "CID",
  searchCenterParam: "s_kwcid",
  newRsidsProp: undefined
};
/***** END DFA Variables *****/

```

The DFA Integrate Settings Block sets variables required by the DFA integration. The values for each of these variables come from the following sources:

CSID: Client Side ID. Generated by DFA once you complete the Integration Wizard. Data Connectors pre-populates this variable with your DFA CS ID, and also sends you this value in the setup email after you complete the Integration Wizard. This variable is not required if Advanced Ad Serving is enabled on your account.

SPOTID: Floodlight Configuration (previously called Spotlight ID). Data Connectors pre-populates this variable with your DFA Floodlight Configuration ID, based on the DFA account information you specified in the Integration Wizard.

tEvar: Transfer Variable. Data Connectors pre-populates this variable with the Analytics variable name you specified for the View-Through variable in the Integration Wizard. Do not change this value without careful coordination with Adobe Engineering or Engineering Services.

errorEvar: Error Variable. Data Connectors pre-populates this variable with the Analytics variable name you specified for DFA Query Failure variable in the Integration Wizard.

timeoutEvent: Timeout Event. Data Connectors pre-populates this variable with the Analytics variable name you specified for the Timeout Event variable in the Integration Wizard.

requestURL: The remote DFA host to query for ad information. Do not change this value unless so instructed by Adobe.

maxDelay: Specifies the amount of time the JavaScript data collection code waits for a response from the DFA Floodlight server, in milliseconds. Adobe recommends experimenting with this value to find the optimum value based on your site's traffic. For example, increasing this value generally collects more DFA data, but increases the risk of losing the base visitor data if the visitor leaves the site during the delay period. Reducing this value lowers the risk of losing hit data, but can reduce the amount of DFA data sent with the Adobe hit data.

visitCookie: The name of the cookie used to restrict DFA calls to once per visit.

clickThroughParam: A query string, typically included on all ads, that notifies the Integrate module that a click has just occurred. The presence of this parameter in the query string causes the request to happen to DFA Floodlight servers no matter if the visitor had already been queried in the last 30 minutes.

newRsidsProp: (Optional) Mapped to an unused Traffic property variable. The DFA integration collects and stores this value in the visit cookie to identify the report suites that collected data for a particular visitor. This property is only needed with custom implementations with Adobe Engineering services.

Integration-Required Plug-Ins

The Collection Code addition incorporates additional plug-ins that improve the operation of the DFA integration:

- Limits DFA queries to once per visit
- Provides cookie name flexibility. Although most organizations use `s_dfa`, you can use any valid cookie name for the DFA integration.
- Eliminates unnecessary redirects. Because view-through data is collected in real-time, Adobe collection servers and DFA could potentially exchange data on every page view. The plug-in blocks these data exchanges when the information is not necessary.

: One of the mechanisms the plug-in uses to eliminate unnecessary DFA queries is a domain-based visit cookie. An integration report suite that spans multiple domains inflates click-through and view-through data when visitors cross domains after a DFA-influenced view-through or click-through.

Confirming a Successful DFA Integration

After you have made all the necessary Web site updates, you can use a network traffic viewer, such as Charles*, Chrome Developer Tools, or Firebug*, to confirm DFA is communicating with Adobe collection servers.

After you have deployed the DFA-enabled `s_code.js` file, use the network traffic viewer to view the requests between DFA and Adobe data collection servers, looking for the following:

- A request to DFA's `fls.doubleclick.net/json` service. This service can respond differently depending on the version of DFA you are using. With the DFA Integration version 1.5:
 - An HTTP 302 redirect to `ad.doubleclick.net`. This will send a `Location: tag` in the response which contains information about the ad visitor.
 - This `Location` tag causes a redirect to `integrate.112.2o7.net/dfa_echo`. This service translates the information about the ad visitor into JSON (JavaScript Object Notation) encoded string. This data is returned with a 200 OK HTTP response.
- With DFA Integration version 2.0 (Advanced Ad Serving enabled):
 - `fls.doubleclick.net` will directly respond with a 200 OK.

In either case, a successful request will result in a request to the Adobe data collection servers that contains the parameter `vX`, where `X` is your View-Through eVar number. This parameter value takes the form: `DFA-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX-XXXX`. This string contains data about the last click and the last impression for the current visitor.

Tuning `s.maxDelay`

Achieving a successful DFA implementation involves optimizing `s.maxDelay` for your particular site.

In general, the decision to raise or lower `s.maxDelay` involves a tradeoff between obtaining more DFA visitor data and endangering collecting Adobe visitor data. Increasing `s.maxDelay` obtains more DFA visitor data, but (placed excessively high) could endanger the collection of Adobe visitor data. Decreasing `s.maxDelay` ensures the collection of Adobe visitor data, but could lose DFA visitor data.

`s.maxDelay` encapsulates more than just the time in network communication to contact DFA; it also represents browser delays to fire and evaluate the JavaScript from which these communications are based. This is because the Integrate module begins the `s.maxDelay` timer after it has inserted the HTML element into the DOM which pulls the data from the DFA Floodlight server. The amount of time it takes for the browser to actually initiate the HTTP request based on this new HTML element varies based upon other images or JavaScript files that are loading simultaneously, speed of the visitor's computer, and specific browser implementations. Furthermore, when the JSON data is retrieved from the DFA Floodlight server, the JavaScript must be evaluated by the browser. This again is controlled completely by the browser and can be delayed if there are large amounts of JavaScript code running simultaneously or many asynchronous JavaScript requests.

With this in mind, `s.maxDelay` needs to be set dependent upon the complexity of the landing page plus the amount of network delay with DFA. On some sites, one possible way to decrease complexity is to fire the Adobe collection code early in page loading, so that there is less going on in the browser at the time the Floodlight server is being requested.

The `Timeout` variable is absolutely required when tuning `s.maxDelay`, because it is incremented every time the `s.maxDelay` timeout is reached. When deciding whether to increase or decrease `s.maxDelay` we recommend following this process:

1. Collect several days of data with `s.maxDelay` set to a particular value.
2. Run a Daily Unique Visitors report for the time range.
3. Run the Timeout event report to check the number of timeouts that are coming through. Remember that a timeout is only collected once per visitor.

Now having the figures in hand, compute

```
Timeout Percentage = [Step 3] / [Step 2] * 100
```

Note that the Timeout Percentage is actually considering all visitors to the site. Some of those visitors would not have been tied to DFA at all, and so the timeout is misleading. To improve this computation, another analysis could consider only unique visitors to pages with the `clickThroughParam` set (for example, `?CID=1`). This will show more accuracy.

If the Timeout Percentage is very low, consider decreasing `s.maxDelay`. If it is very high, increase `s.maxDelay`. When decreasing `s.maxDelay`, you will want to rerun the Timeout report to ensure that timeouts haven't dramatically increased. When increasing `s.maxDelay`, you will want to run a Page Views report to make sure page views aren't falling out due to lost data. Each time `s.maxDelay` is changed observe the data for several days in order to ensure that the data represents a trend, and not just a day-to-day fluctuation.

The optimal setting for `s.maxDelay` is the point at which the Timeout Percentage is minimized while Page Views do not drop off.

Timeouts are expected to decrease when you move to version 2.0 of the integration, because of the eliminations of 302 redirects. Initial findings with beta clients have shown consistent reduction in timeouts, and thus more DFA data being collected

Analytics Reports Using DFA Data

Once DFA and Adobe® collection servers can communicate, you can generate reports in Reports & Analytics that include DFA data.

Pre-packaged DFA reports include the following:

Channel: Displays data comparing Banner ads against other on-line ad options, such as Paid Search and Email.

Delivery Tool: Displays data comparing DoubleClick for Advertisers against other on-line ad services.

Site Name: Displays data comparing Web sites where DFA banner ads appeared.

Page Name: Displays data comparing individual Web pages where DFA banner ads appeared.

Ad Name: Displays data comparing specific DFA banner ads.

Campaigns: Displays data comparing different DFA ad campaigns

To generate a DFA report:

1. Log in to Adobe Marketing Cloud.
2. Go to **Analytics > Reports & Analytics**. Make sure you have selected the report suite where you installed the DFA integration.
3. In the left-side navigation, select the Conversion variable that captures the DFA click-through data, then select the desired DFA report.
4. Click **Add Metrics**. This opens the Metric Selector dialog box.
5. Check the Impressions and Clicks metrics in the **Available Metrics** list and click **Apply**.

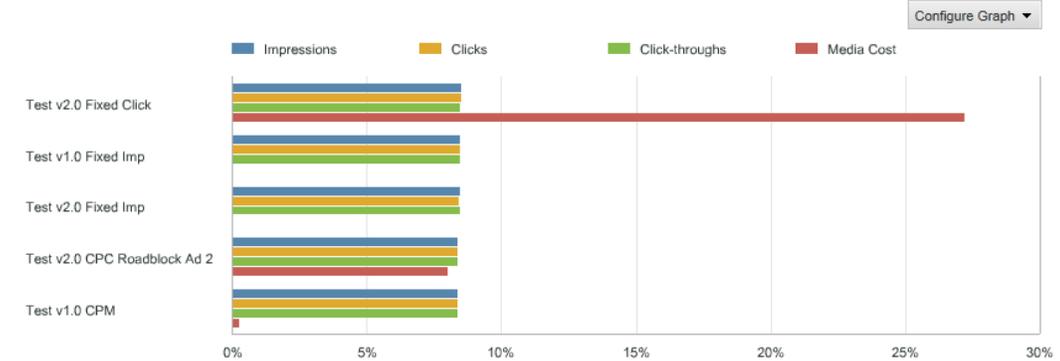
The resulting report displays Impressions, Clicks, and Revenue data so you can see how your DFA services translate into bottom-line revenue.

For example, a DFA Ad Name report displays the impressions, clicks, and revenue data directly attributable to specific banner ads in your current DFA ad campaign:

Ad Name Report

Sep 2011

Report Type: **Trended | Ranked**
 Selected Metrics: **Impressions, Clicks, Click-throughs and Media Cost**
 Data Filter: **None**
 Compare to Site: **None**
 Broken Down by: **None**
 Percent Shown as: **Number | Graph**



Ad Name Report | All Visits (No Segment) | September 2011 | Graph generated by SiteCatalyst using Report Accelerator at 10:5...

Ad Name	Impressions	Clicks	Click-throughs	Media Cost
1. Test v2.0 Fixed Click	5,911 8.5%	5,910 8.5%	5,910 8.5%	\$8,865.00 27.2%
2. Test v1.0 Fixed Imp	5,877 8.5%	5,870 8.5%	5,903 8.5%	\$5.90 0.0%
3. Test v2.0 Fixed Imp	5,871 8.5%	5,853 8.4%	5,918 8.5%	\$5.85 0.0%
4. Test v2.0 CPC Roadblock Ad 2	5,810 8.4%	5,816 8.4%	5,839 8.4%	\$2,617.20 8.0%
5. Test v1.0 CPM	5,817 8.4%	5,806 8.4%	5,834 8.4%	\$87.34 0.3%
6. Test v2.0 CPC Package Ad 1	5,777 8.3%	5,770 8.3%	5,792 8.3%	\$2,019.50 6.2%
7. Test v1.0 Fixed Click	5,766 8.3%	5,757 8.3%	5,809 8.3%	\$8,635.50 26.5%
8. Test v2.0 CPM	5,763 8.3%	5,755 8.3%	5,786 8.3%	\$86.51 0.3%
9. Test v2.0 CPC Roadblock Ad 1	5,757 8.3%	5,746 8.3%	5,792 8.3%	\$2,585.70 7.9%
10. Test v1.0 CPC	5,747 8.3%	5,740 8.3%	5,765 8.3%	\$2,870.00 8.8%
11. Test v2.0 CPC Package Ad 2	5,717 8.2%	5,711 8.2%	5,753 8.3%	\$1,998.85 6.1%
12. Test v2.0 CPC	5,570 8.0%	5,568 8.0%	5,586 8.0%	\$2,784.00 8.6%
13. 468 x 60 Default Ad	1 0.0%	0 0.0%	0 0.0%	\$0.00 0.0%
TOTAL	69,384	69,302	69,687	\$32,561.35

This report has the following important features:

- The report header, labeled DFA_Ad Name Report, displays the report time period (February 2009), and indicates that this is a Ranked report using a Horizontal Bar format.
- The Graph itself displays three metrics (Impressions, Clicks, and Revenue) for each DFA Ad in the report.
- The graph displays data for each metric as a percentage of the metric total, rather than using the actual numbers for each metric. You can change this in the **Configure Report** dialog box.
- Below the graph, the DFA_Ad Name report lists the report details for each of the DFA Ads, and displays both numeric and percentage data for the Ad performance.
- Select a specific DFA Ad Name to open a menu of options for getting more information about that particular DFA ad.
- Each metric column includes a color key that corresponds to the metric's assigned color in the graph.
- The report is ordered by the Revenue metric with highest revenue Ad at the top. You can change the report ordering clicking on a different column (metric) label.

Reconciling Metric Discrepancies

Occasionally, some metrics might not fall within an acceptable difference when comparing Adobe Analytics metrics to DFA metrics. Below is a list of metric definitions and possible reasons for variances.

This section includes the following topics:

Metric Definitions

Adobe uses the following terms when talking about metrics related to the DFA integration:

Impressions: Impressions refer to the number of times an ad was viewed. Impressions are reported on an ad-by-ad basis, but can also be aggregated into ad groups or other multi-ad groupings. The impressions metric in Analytics is imported from DFA via a nightly data sources import.

Clicks: Clicks refer to the number of times an ad was clicked, as report by DFA. Clicks are registered on the DFA redirect page prior to the visitor landing on the customer's website. Like impressions, the clicks metric in Analytics is imported from DFA via a nightly data sources import.

Click-Throughs: Click-Throughs refer to the number of times the user arrived to the landing page, after clicking on an ad. This metric can be subtly different from Clicks.

View-Throughs: View-Throughs refer to the number of times a visitor came to the customer's Web site after viewing an ad, but having NOT clicked the ad. The visitor must come to the site within the view-through window, which by default, is set to 30 days. The impression must have happened more recently than the last click. View-throughs are registered once per campaign, per visit and are counted when the integration populates the view-through eVar with the DFA campaign ID, and the view-through event is set.

Possible Reasons for Discrepancies

Lists a number of reasons why data discrepancies can occur between Adobe Analytics and DFA reports.

Users on Safari Browser and other Browsers that block 3rd-party cookies

3rd party cookie acceptance is typically the largest cause of discrepancy between Adobe Analytics and DFA. Safari and some other browsers block third-party cookies by default. This means that by default, Safari accepts the first-party cookie used by most Analytics implementations, but rejects the third-party cookie used by DFA.

A sample of data from our Analytics 15 beta customers showed less than 0.5% of users typically reject first-party cookies, while 5–12% reject third-party cookies (many of these rejections are likely due to default browser settings).

This discrepancy can result in large difference in the data collected by Analytics and DFA.

Why might Impressions Reported in DFA be Higher than Impressions Reported in Adobe Analytics?

- DFA sends data to Adobe data collection servers in a nightly batch, so impression data in Analytics can be up to 2 days behind the DFA reports.
- Adobe uses SAINT classifications to classify imported DFA tracking codes into various levels of aggregations (campaign name, placement name, ad name, etc.) If the discrepancy appears when running a classification report, perform a simple test to see if the classifications have not yet caught up with imported metrics:
 - In the classification report, locate a line item called "None".
 - Do a breakdown of this line item by the same variable, using the raw DFA ID report (such as Campaign ID).

- In this report, take note of any unclassified DFA tracking codes which are in the form DFA:XXXXX:XXXXX ...
- If many of these tracking codes exist, investigate the nightly SAINT classification process.

Why Might DFA Clicks be Higher than Adobe Analytics Click-Throughs?

- DFA records a click before the visitor lands on the customer website. Analytics records Click-Throughs after the landing page loads and executes the Adobe JavaScript beacon. Typically discrepancies are had because either the visitor is not arriving to the landing page after DFA tracks a click, or the `s.maxDelay` timer is being hit.
- Ensure all placements and creatives in the Floodlight Configuration include the `clickThroughParam` in the landing page URL (for example “?CID=1”). Failing to set this parameter will cause the Adobe Analytics JavaScript to miss any click-throughs which happen after the first hit of the visit.
- Ensure all placements and creatives have a landing page which is tagged with the Omniture JavaScript and has the DFA Integrate module, and that the Floodlight Configuration ID on that landing page matches the Floodlight configuration ID of the served ads. Often, discrepancies come because the landing page for ads are set to third party site, or the served ads.
- If you use Rich Media ads or Flash (swf) ads, make sure that whenever the DFA clicktracker is hit, that the visitor's browser is also being redirected to the Landing page with the `clickThroughParam` included in the query string. Failure to redirect the browser will not cause a click-through to be recorded.
- Timeouts represent instances where DFA data might have been available but the JavaScript did not receive a response in time. When a visitor arrives on the landing page, Adobe JavaScript requests the visitor's information from DFA's `fls.doubleclick.net` service. The `s.maxDelay` parameter determines how long the JavaScript waits for the Floodlight Service (FLS) data. If `s.maxDelay` is too high, visitors can leave the site before Adobe collects the hit data; meaning that no click data is recorded. If `s.maxDelay` is set too low, the visitor's Internet connection cannot retrieve the FLS data in time; meaning that the hit is sent to Adobe without DFA click information. See [Tuning s.maxDelay](#) for a further discussion on this subject.
- Bot traffic might inflate the DFA click numbers. Bots may have functionality to click on an ad but may not have the complexity to execute an Analytics beacon or fire the synchronous script tag to load the Floodlight servers request data. If these bots are not removed from the Clicks figure, this might be a source of discrepancy.
- Visitors which leave the page prior to `s.maxDelay` expiring, and DFA data returning, will be lost; no DFA or visitor data will be collected for them.
- Analytics attempts to identify and remove duplicate Click-throughs so they are counted only once per campaign per visit. DFA counts visitors who click “Back” and pass through the ad redirect multiple times as additional ACM Clicks, while Analytics does not count these as multiple Click-throughs.
- DFA Floodlight tags do not rely on JavaScript enabled, while Analytics does. Because of this, there might be some cases where DFA records a hit when Analytics does not. To identify if this might be a problem, use the Analytics JavaScript report in the Visitor Profile menu.

Why Might DFA Post-Impression Activities be Higher than Adobe Analytics View-Throughs?

- Analytics attempts to identify and remove duplicate Click-throughs so they are counted only once per campaign per visit. DFA counts visitors who click “Back” and pass through the ad redirect multiple times as additional ACM Clicks, while Analytics does not count these as multiple Click-throughs.
- DFA Floodlight tags do not rely on JavaScript disabled, while Analytics does. Because of this, there might be some cases where DFA records a hit when Analytics does not.
- DFA counts Post-impession activities when using Floodlight tags, which can be placed on the client Web site. Analytics counts View-throughs after the JavaScript beacon (image request) executes. Code placement on the Web page can determine if an aborted page load counts as a Post-impession activity or a View-through.

What if Discrepancies are Far Outside an Acceptable Range and the Possible Reasons Above do not Apply?

Consult your Integration Consultant, or Adobe Client Care, to document the discrepancies and report them to the Data Connectors engineering team. To expedite your request, have 2 - 3 days of data comparing the metrics in question (at a campaign code level). In your request, identify all actions you have already taken to reconcile the discrepancy.

Frequently Asked Questions

Why won't the Data Connectors wizard accept my login credentials?

If you have verified that the login credentials are valid, next verify that the username provided to the integration is enabled for API access. The Data Connectors wizard uses the DFA API to validate login credentials, as well as turn off and on Adobe-specific settings in the DFA API. API Access is a setting that must be turned on from within the DFA interface by an administrator. Next, make sure that you have permission to access the Advertiser ID or the Floodlight Configuration ID selected in the wizard.

Why am I not seeing any data from the nightly uploaded metrics (DFA Impressions, DFA Clicks, etc)?

If you are using version 1.5 of the integration, this could be because your integration has not yet been assigned a Client Site ID. A Client Site ID (CSID) is required for the nightly exchange to occur, and to request data from the DFA ad server. CSIDs can take up to 3 days from the date of integration to be exchanged with Google. Once the CSID has been received from Google, you will be notified via the integration's email address of the new CSID, along with the latest JavaScript code.

If it has been more than 3 days and you have not received the setup email and metrics are not flowing, then the most likely problem is that the CSID has already been assigned to another integration. Google maintains a 1 to 1 mapping between CSID and Report Suite, meaning that if an integration in one report suite uses same Advertiser ID as another integration in another report suite, only the first one will be assigned a CS ID. To change which report suite or Advertiser ID a CSID is mapped to, a ticket must be opened with Google Support.

For example, say there is an integration in Report Suite A with Advertiser ID Z which gets assigned a CSID. If another integration is later setup in Report Suite B with Advertiser Z, this newer integration will NOT be re-assigned the CSID. This would require a Google ticket. On the other hand take the example of an integration in Report Suite A, with Advertiser ID Z, and later another integration in Report Suite A, Advertiser Z is setup. Only the first Integration will receive data for the integration; however, in this case, the first integration can be deactivated and data will flow to the second integration.



Note: CSIDs are not used in version 2.0 of the integration and so the CSID negotiation process does not apply.

I am using version 2.0 of the integration and cost metrics are not appearing for my DFA ads. Why might this be?

Cost metrics have to be both turned on from both the Google DFA side, and supplied in the DFA interface, as well as turned on in the Genesis wizard. The first place to verify is that you have mapped a SiteCatalyst event for DFA Media Cost, and supplied a currency code. If you have mapped the Media Cost event, and finish and save the wizard, the DFA omnitureCostData flag will be turned on in the DFA API. This will signal to Google that the metrics should be sent in the nightly file. You can double check via the DFA interface that omnitureCostData is enabled by looking at properties on the integrated Floodlight. Finally, after checking these two places, ensure that the ads that are part of the integrated Floodlight specify cost data and cost structures. If cost data is not provided in the DFA interface, it will not show up in Analytics.

Why do certain ads not show any DFA Impressions or View Throughs, but they do show Clicks and Click-throughs?

There are some ads which only record click data, called clicktrackers. These type of ads do not return last impression data from when the Floodlight server is queried. To verify if a certain ad is a clicktracker or click-only ad, contact either your DFA agency or your Google Support representative.

Why aren't there any click-throughs for ads that show DFA Clicks?

There can be one of many answers to this.

First, check that the ad in question has a landing page URL which is both (a) tagged with Adobe code for the same report suite you are viewing the discrepancy in and (b) contains the clickThroughParam query string parameter.

Second, verify that you have a working integration by following through the steps in [Confirming a Successful DFA Integration](#). If you see a DFA tracking code come through with the Adobe hit on the landing page, then you should see that Click-through come in the DFA Campaigns report. If you are not seeing it come through, verify that the report suites match between the landing page's s.account variable, and the report suite being viewed in Reports & Analytics. If these match, check for tracking codes in the View Through evar report that look like DFA:XXX:XXX:XXX:llXXX:XXX:XXX:XXX:XXX.

These indicate failures of the DFA VISTA rule to digest the raw data from DFA. This problem can be remedied by opening a support ticket through your Adobe Account Representative.

If none of the solutions above explain the problem, see [Reconciling Metric Discrepancies](#) to explore other possibilities