

Digital Services Innovation Center
Digital Analytics Program (DAP)

DAP GA v1.73 Code File Solution
Technical Implementation Guide

September 2013

Table of Contents

[Introduction](#)

[DAP GA v1.73 File Package](#)

[DAP GA v1.73 Script Block](#)

[Query String Parameters](#)

[Deployment Steps](#)

[Appendix](#)

[1. Script Block Examples by Scenario](#)

[2. Actual Path Examples](#)

[3. Youtube Tracking](#)

Introduction

DAP GA v1.73 code solution achieves three main goals: 1) preserves the main DAP GA file, which contains all of the core, standard functions in a single file, 2) lowers the number of DAP GA core code versions and inconsistencies across agencies, and 3) allows agencies to customize their DAP GA reporting through the use of custom parameters outside of the core DAP GA file.

In summary, DAP GA v1.73 offers a set of seven custom parameters:

1. **Agency Name**
2. **Sub-agency Name**
3. **Visitor Cookie Timeout**
4. **Parallel Google Analytics User Agent (UA)**. This parameter is for optional tracking to a second or multiple UA/Google Analytics implementations, where agencies have their own, existing GA implementation and want to leverage the DAP GA code for their intra-agency reporting
5. **Search Parameter(s)**. This parameter allows agencies to specify names used to designate on-site/internal search terms not already included in DAP GA core file
6. **File Extensions**. This parameter allows agencies to specify additional download extensions not mentioned in the core DAP GA file
7. **Youtube tracking**. v1.7 of the DAP GA core file implements tracking of Youtube videos embedded on agency sites. Since multiple events may be fired per video, agencies will need to use this parameter to expressly enable specific tracking

DAP GA v1.73 File Package

- ★ federated-analytics.js (44k)
- ★ federated-analytics.all.min.js (13k)

DAP GA v1.73 Script Block

The core of the DAP GA v1.73 solution is in the new script block, which looks similar to the following example:

```
<script id="_fed_an_js_tag" type="text/javascript"
  src="../../../federated-analytics.all.min.js?agency=GSA
  &sub-agency=OCSIT&pua=UA-55555-5&sp=find&yt=true">
</script>
```

Or, this (to better display the parameters, we set them out as in the following example):

```
<script
  id="_fed_an_js_tag"
  type="text/javascript"
  src="/path/federated-analytics.all.min.js"
  ?agency=GSA
  &sub-agency=OCSIT
  &vcto=6
  &pua=UA-8706900-7
  &sp=find
  &yt=true
">
</script>
```

The script block is a regular HTML script block with a URL pointing to the script file to be loaded. However, the inclusion of the following is rare:

- an id for the script block (id="_fed_an_js_tag")
 - query string parameters used within the script
- ❖ The id for the script block is:
- required so that the script can find the block efficiently.
 - Ids other than "_fed_an_js_tag" will not be recognized.
 - longer, begins with an underscore and refers to a js file so it is likely to be unique
- ❖ The query string parameters are:
- case-insensitive to be fault-tolerant
 - should be specified as provided
 - not required to be in the sequence specified but the first query string parameter
 - needs to be preceded by the ? in accordance with HTML & HTTP standards

With the exception of Agency Name, all of the custom parameters are optional and not required. To customize the parameters, they must be specified and placed within the DAP GA script block. For example:

```
<script id="_fed_an_js_tag" type="text/javascript"
  src="/path/federated-analytics.all.min.js"
  ?agency=GSA&sub-agency=OCSIT
  &pua=UA-555555-1
  &sp=find&yt=true">
</script>
```

Query String Parameters

- *agency*
 - Agency
 - A short, uppercase abbreviation uniquely identifying the Agency. E.g. DHS, EPA, GSA, etc
 - Syntax: `agency=DHS`
 - Values are uppercased in the code
 - The value is reported in Custom Variable 33.
 - The value should always be present and populated
 - A value is required
 - Default = "unspecified: <top domain name>" e.g. unspecified: treasury.gov
→ *"unspecified: <top domain name>" will indicate the Agency value is missing from the query string. The top domain name (without subdomains) will identify the site with the missing value without the need for additional reporting, which allows for some tracking even without the parameter being set. The value of the data is related to how closely the top domain name reflects the agency name does not compromise the data.*

- *sub-agency*
 - Sub-agency
 - A short, upper-case abbreviation uniquely identifying the Agency's sub-agency
 - Syntax: `sub-agency=FEMA`
 - Values are upper-cased in the code
 - The value is reported in Custom Variable 34 in the form: AGENCY - SUBAGENCY, E.g. DHS - FEMA
 - The value should always be present and populated
 - The code is fault tolerant, accepting the parameter names sub, subagency and sub-agency
 - A value is recommended
 - Default = site's hostname. E.g. DHS - www.fema.gov
 - To date, the default value has been used to prevent the need for a different file version for each sub-agency

- *vcto*
 - Visitor Cookie Timeout
 - An integer value specifying period in months.
 - Syntax: `vcto=6`
 - Parameter and value not required
 - Default = 24
 - Those agencies that have specified a value in their customized script files

have used values of either 6 or 12 months

○ *pua*

- Parallel UA id for optional tracking to a second, alternative GA account (UA-XXXX-YY) belonging to and managed by the Agency
- Multiple UA ids can be specified in a comma-separated list
- Should only be presented in the form if the user first checks to enable parallel tracking to an alternative GA account.
- One or more single values each in the form UA-XXXX-YY
- Each UA id validated on the form as `/^UA-\d{5,8}-([1-4])d|([1-9])50$/`
- Syntax: `pua=UA-1234567-8`
or
`pua=UA-1234567-8,UA-76543-2,UA-76544-1, ...`
- Values are upper-cased in the code
- Parameter and value not required but if provided must be a valid UA ID.
- There is no Default. If not specified, no parallel tracking will occur
- Naturally, if the Web Property is not enabled for Premium, it will not get the Custom Variable data/
- Purposes:
 - Provide an Agency the facility to send the same GA DAP data to 1 or more Web Properties controlled by the Agency
 - This provides a form of transparency
 - The principal purposes of supporting more than 1 Web Property include:
 - Avoid sampling by sending data from individual Sub-domains (individual sites) to their own Web Properties
 - Avoid the limitation of 50 profiles per account.
In order to achieve this purpose, all the UA ids used by an Agency should be in different GA Accounts as indicated by the different UA-XXXXX portions of the Web Property Ids:
`pua=UA-1234567-8,UA-76543-2,UA-76544-1`
 - An additional benefit allows Agencies to create a roll-up account for all sites while tracking sites separately

○ *sp*

- Search Parameter(s). A single value or a list of comma-separated search parameter names
- Allows Agencies to specify parameter names used to designate internal search terms not already included in federated-analytics.js, being `q, query, nasaInclude, k, querytext`
- Syntax:
`sp=find`
or
`sp=find,pregunta,term`

- The syntax example allows a site to use find as a search parameter for GA search tracking. This is required since GA profiles allow a maximum of 5 possible search parameter names.
 - In this example:
 - `http://www.examplesite.gov?find=search+term`
 - would be reported to GA as
 - `http://www.examplesite.gov?query=search+term`
 - allowing GA to recognize find as one of its 5 search parameter names
 - It also has the effect of not having to change GA Profile configurations for different Agencies
 - Parameter and value not required
 - There is no Default although the pre-existing parameter names `q, query, nasaInclude, k, querytext` serve as defaults. Sites don't need to specify their search parameter names if already included in the list of pre-existing ones.
 - *exts*
 - Additional Download Extensions
 - A comma separated list of 1 or more extensions
 - The DAP code uses strips spaces from the string and uses it as case insensitive.
 - The current list of supports the following extensions :
 - `xls,xlsx,doc,docx,ppt,pptx,pdf,txt,zip,rar,7z,gz,tgz,exe,wma,mov,avi,wmv,mp3,mp4,cs,tsv,mobi,epub,swf`
 - *yt*
 - Youtube tracking
 - ver 1.7 of the DAP code implements tracking of Youtube videos embedded on Agency sites
 - Since multiple events may be fired per video, tracking needs to be expressly enabled via this parameter
 - A value of true or false
 - Syntax: `yt=true`
 - Parameter and value not required
 - There is no Default. If not specified, embedded Youtube videos will not be tracked

Deployment Steps

1. Download the federated-analytics.js or federated-analytics.all.min.js file
2. Save the file to a convenient location locally

3. Copy the file to the same location as your agency site's other javascript files (typically a folder called /scripts or /javascript, etc.). The actual location is not critical it should simply be on the same server as the site's other files
4. Insert the Federated Analytics Script Block in the <head> of all pages on your site.
 - a. The best location is directly before the closing </head> tag
 - b. It is expected that your agency site uses templates and that the DAP GA v1.73 script block can be inserted into the templates at that location
 - c. It needs to be inserted so that it appears on every page across the entire site, including subdomains (eg: ed.gov and www.ed.gov and www2.ed.gov, etc) and search ubdomains (eg: find.ed.gov, searchjustice.usdoj.gov, search.nasa.gov)

Agencies using earlier versions of the DAP GA code will need to change their existing DAP GA script blocks to the form specified in this documentation. The file referenced will need to overwrite the previous version in use on the site.

NOTE to developers using CMSes: Some CMSes (e.g. Drupal) combines multiple script files into single script files. However, a slip up can occur that the file is both combined and added separately. Developers running such CMSes should be careful that the code is either combined or added separately. We recommend that the code is added separately.

Appendix

1. Script Block Examples by Scenario

Scenario 1: Basic DAP GA v1.7 script block implementation

- All sites' script blocks will have these attributes:
`id="_fed_an_js_tag" type="text/javascript"`
- All script blocks require a src to be specified. If federated-analytics.js is to be hosted in the same location as previously, the src will remain unchanged.
- All script blocks should, at a minimum, specify the Agency's standard abbreviation

```
<script id="_fed_an_js_tag" type="text/javascript"  
  src="/js/federated-analytics.all.min.js?agency=DHS">  
</script>
```

This example would result in all default parameters being applied, Agency will be tracked as "DHS" and sub agency as "DHS-www.fema.gov."

Scenario 2: Specifying a sub-agency and customizing visitor cookie timeout (to override the 24 month default with a 6 month period):

```
<script id="_fed_an_js_tag" type="text/javascript"
  src="/js/federated-analytics.all.min.js?agency=DHS
  &sub-agency=FEMA&vcto=6">
</script>
```

Scenario 3: Specifying a sub-agency, tracking to a parallel Web Property ID UA-55555-55 and a custom visitor cookie timeout (to override the 24 month default with a 6 month period):

```
<script id="_fed_an_js_tag" type="text/javascript"
  src="/js/federated-analytics.all.min.js?agency=DHS
  &sub-agency=FEMA&pua=UA-55555-55&vcto=6">
</script>
```

Scenario 4: Specifying a sub-agency, tracking to a parallel Web Property ID UA-55555-55 and YouTube tracking:

```
<script id="_fed_an_js_tag" type="text/javascript"
  src="/js/federated-analytics.all.min.js?agency=DHS
  &sub-agency=FEMA&pua=UA-55555-55&yt=true">
</script>
```

2. Actual Path Examples

Specifying an agency and a sub-agency:

```
<script id="_fed_an_js_tag" type="text/javascript"
  src="http://www.fema.gov/sites/all/modules/usfedgov_google_a
  nalytics/js/federated-analytics.all.min.js?mmx1ti&agency=DHS
  &sub-agency=FEMA">
</script>
```

Note that FEMA already used a query string parameter `mmx1ti`. In such cases, simply add the DAP parameters beginning with a `&`, being standard Query String Parameter Syntax.

3. Youtube Tracking

Documentation specifically on GAS' Youtube tracking may be found here: https://github.com/CardinalPath/gas#_gastrackyoutube. The following should be noted:

- There are different methods of embedding Youtube videos. DAP GA only supports the iframe method.
- Tracking is not possible from IE6 and IE7 (technical: they don't support HTML5's postMessage)
- The following events will be tracked: 'play', 'pause', 'finish' and 'error'.
- DAP GA v.1.73 also specifies 33%, 66% and 90% viewed.
- These are selected to reduce the number of events (often 25, 50, 75 and 90 are tracked) and to cater for a common problem that many viewers stop or exit during the last few % before getting to "finish"
- The documentation recommends appending enablejsapi=1 to the video URL the parameter but this is impractical. Ver 1.73 set people d s the the force parameter to true to have GAS add the parameter if it is missing.