

Advanced Implementation

Managing Data from Multiple Websites

Abstract

The basic Google Analytics installation is one tracking code for each website. This method neatly stores and reports data from each website in its own separate silo, but makes it difficult for libraries to understand how website users traverse and interact with the various websites libraries maintain. Chapter 3 of Library Technology Reports (vol. 49, no. 4) “Maximizing Google Analytics: Six High-Impact Practices” introduces the high-impact practice of tracking separate subdirectories, subdomains, and domains all in one Google Analytics profile. Learn the benefits and drawbacks to tracking multiple websites and how to customize your Google Analytics tracking code to effectively monitor and report your entire library’s website usage data.

Library websites are strange beasts. They are often a complex blend of one or more websites created in-house, vendor catalogs, discovery layer services, blogs, social media presences—the list could go on. While you may own or have been given access to your own domain (yourlibrary.org), you may have other web presences that are hosted in different subdomains (catalog.yourlibrary.org or blog.yourlibrary.org) or on a completely separate domain. A website that is hosted on a vendor’s web server, such as the popular LibGuides, will typically be on a domain separate from your library’s website. You may think that you could just set up your main website, your online catalog, your discovery layer tool, your LibGuides instance, and all of the other sites that comprise your complete library web presence with their own tracking numbers in their own profiles, and you’d be absolutely right. In fact, the Google Analytics default is using one web property ID (i.e., UA-XXXXXX-X) per domain. Many libraries configure their multiple web tools in this way, which

makes sense from an internal staff perspective; the web team is evaluating the parts of the web presence that they build, the ILS administrator may track and assess use of the catalog, and reference and instruction staff want to know how their LibGuides are being used.

However, this doesn’t reflect the reality of our customers’ experiences; even though the back end of all of these things reside in separate systems, the end-user experience served up by this mishmash of library products in a web browser ought to be as seamless as possible. Assuming that your institution has adopted a user-centric data collection model—you have, right?—you may want to reconsider simply assigning separate profiles to separate tools. It’s important to think about how all of these web presences work together and whether they’re all part of the same user experience or not. Think about your audiences and how the sites are intended to be used. If they are intended for different purposes or audiences, it might not make sense to track them together. In this chapter, we’ll talk about the rationale for deciding when and how to track multiple sites together (or not!) and then take a look at how to accomplish this with some creative adjustments to your JavaScript tracking code and your Google Analytics account settings.

Concepts to Know

Since this is a slightly more advanced topic, please make sure you understand the following terms before diving in:

- **Domain.** The base or smallest part of a URL, e.g., yourlibrary.org or yourcollege.edu. (Technically speaking, a domain name is just a language-based,

memorable way to get to a web server, whose true address is one or more IP numbers.)

- Subdomain. A “child” of a domain, e.g., `kids.yourlibrary.org` or `library.yourcollege.edu`. Subdomains are commonly separate entities or sites. As in the second example here, your library website may itself be a subdomain of a parent institution or entity.
- Subdirectory. A directory inside your domain, e.g., `yourlibrary.org/kids` or `yourcollege.edu/library`. As with subdomains, it’s common for library websites to be contained in a subdirectory of a parent institution, even if it’s separately hosted, developed, and maintained.
- Google Analytics tracking cookie. Cookies are small text files that are placed in a visitor’s web browser when the visitor accesses a website. These cookies are designed to collect and send data about the users’ interactions on that website to a web server. The Google Analytics tracking script uses a cookie to determine unique visitors, new versus returning visitors, how often visitors return to the site, and various other essential data points. There are different types of cookies, but Google Analytics uses a first-party cookie, which collects data only from the single domain that it was set to track. Essentially, this means the cookie cannot combine data from multiple domains even if you own or control them—at least, not without some customizations to the tracking script that alter how the cookie collects data.¹ What does this mean for libraries? Any library wanting to track multiple domains or subdomains—such as a library’s website, catalog, and blog—in one profile must first implement cross-domain tracking. Cross-domain tracking is the process of tracking different websites (domains and subdomains) by altering the default cookie settings to combine and report data from each of those identified sites in one profile.

Can’t I Just Use the Same Tracking Code on Two or More Sites?

Applying the same web property ID to all of your sites in which you have implemented Google Analytics would indeed result in data being collected from all of them, but without some modifications to the tracking code itself, your data will be badly skewed. For example, a visitor entering the main library site (`yourlibrary.org`) and then accessing a subdomain of that site (`blog.yourlibrary.org`) that uses the same uncustomized tracking code will be counted as two separate visitors. Once the visitor navigates to the subdomain, the session on the main site is ended and a new session on the subdomain is begun because a cookie is

automatically generated for the main domain and another for the subdomain. This inflates the reports Visits and Visitor Count, which are critical to almost all the reports in Google Analytics. Additionally, the individual sites—in this case, the main site and the subdomain—are shown in reports as referrers to one another. This can be particularly frustrating because it cannot show the actual navigation summary, from the user entering the main site to navigating to the blog on the subdomain: all you can infer is that the visitor was referred to the blog from the library’s main site.

With cross-domain tracking, you will use the same tracking code with a few additional customizations to the script to track all those separate sites as one proper site. We’ll take a look at some examples of when and how you would want to combine or separate to track across multiple domains or subdomains and then review the tweaks required to your Google Analytics tracking setup to implement your ideal tracking scenario.

Traditional Tracking: Using One Web Property ID per Domain

Using one web property ID per domain is the default method of tracking sites in Google Analytics, where each web presence is tracked in its own profile—so the library website, catalog, blog, and any other web presence you manage are each given its own profile and unique tracking code. A site that has several related subdomains, like a blog site that contains multiple blogs (`blog1.yourlibrary.org`, `blog2.yourlibrary.org`, etc.) could be tracked using one tracking code, but this approach is not recommended because it generates the same problems as using the same tracking code on separate domains discussed earlier. We’ll discuss the best practices for subdomain tracking later in the chapter.

The primary benefit of tracking one profile per domain is that the data is in a single profile, allowing you to easily see the total use of the individual websites without having to create additional segments or filters. Each site can have its own set of unique goals, custom variables, and profile settings. However, there are several drawbacks to this approach. First, all that data is in separate reports, so if you ever needed to report on multiple sites, you would have extra work to combine the necessary data. Additional steps are required to track how those sites interact with each other. By default, Google Analytics will track interactions between the sites as referrals found in the Traffic Sources reports. Libraries must set up a method to track exits to these other sites to accurately distinguish proper exits from bounces—it will show where users are navigating off a site in order to go to another library-managed site

rather than just leaving a site. Finally, you cannot track goals across the separate domains.

If you opt to track all sites separately, you still can separate some of the site's data into different filtered profiles. For example, a large library that contains many branches or departments with each branch's website in a subdirectory (e.g., www.yourlibrary.edu/englishlibrary/ or www.yourlibrary.edu/engineeringlibrary/) of the main library site can track the entire site's presence (www.yourlibrary.edu) with one tracking code to create a master profile and then create filtered profiles for each department library. Filters can be predefined (using IP addresses or even subdirectories of the site), but you can also create completely custom filters using regex (regular expressions) to define exactly which set of pages you want tracked in the filter. (See chapter 4 for more information about regex and filtering.) These filtered profiles allow those individual department libraries to see the usage of their own websites without having an impact on the master profile that contains data from all the library's websites. But be aware that this filtered profile approach can also skew your visitor data because the site is still sharing a single cookie! Data reported on a filtered profile for a department library will be influenced by visitor activities from a different filtered profile on the site if the visitor crosses multiple subdirectories. A better practice is to implement individual subdirectory tracking.

Tracking Subdirectories Separately

Rather than using filters to track separate subdirectories, you can customize your tracking code to track solely a subdirectory on your website. The ideal scenario for implementing this method is when you have a large site where different subdirectories are essentially their own stand-alone websites, such as the example earlier of the large library website that contains the different branch websites as subdirectories on a single domain. However, it could also include smaller cases, like a blogging platform where individual blogs are identified as separate subdirectories (e.g., libraryblog.org/blog1 or libraryblog.org/blog2).

For this tracking method, you need to add one line to your tracking code to tell the cookie to track only pages from that subdirectory. If you wanted to track `blog1` separately from `blog2`, the script would look like this:

```
<script type="text/javascript">
var _gaq = _gaq || [];
_gaq.push(['_setAccount',
'UA-XXXXXX-XX']);
_gaq.push(['_setCookiePath', '/
blog1']);
_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);
})();
</script>
```

The bolded portion highlights the addition to the tracking code that sets the cookie to track only web pages in the `blog1` subdirectory. This option assumes you are not interested in tracking data from the rest of the site—all other site data must be collected using a different tracking code.

The above method works well when tracking just one subdirectory on the entire site, but what if you want to track multiple subdirectories on a site, such as tracking both `blog1` and `blog2` in one profile? That takes a little more work, but it is possible. Again, this method assumes you are interested in tracking selected subdirectories, and not the entire site. Essentially, you add another line to the tracking code (bolded in the following script excerpt):

```
<script type="text/javascript">
var _gaq = _gaq || [];
_gaq.push(['_setAccount',
'UA-XXXXXX-XX']);
_gaq.push(['_setCookiePath', '/
blog1']);
_gaq.push(['_trackPageview']);
_gaq.push(['_cookiePathCopy', '/
blog2']);
(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);
  }) ();

</script>

```

This script tracks the blog1 data and tells the cookie to also be used for tracking blog2. Make sure the script is embedded on every single page in the sub-directories that need to be tracked.

Tracking Multiple Subdomains in One Profile

Another scenario common in libraries is the use of subdomains in URLs for identifying and branding multiple sites as part of the same organization. For example, in a public library setting, there may be a main library web page (e.g., `yourlibrary.org`), and a children's site (e.g., `kids.yourlibrary.org`). It might be that these resources are considered two separate user experiences, and it makes sense to track them separately. This scenario also applies even if one of these sites is hosted remotely; for example, academic libraries may use a subdomain to host LibGuides. You could opt to track this subdomain separately from the library's main site. In this case, simply create two profiles and track these web presences separately. Yet, if you wanted to track multiple subdomains in one profile, you could implement subdomain tracking. Subdomain tracking allows you to monitor data from the entire domain in one profile, making it easier to see how visitors navigate and interact with the complete site.

Setting up subdomain tracking is fairly simple—just turn on the Subdomains option as you automatically generate the code when creating the profile. An additional line of script (bolded in the following example) will be in your tracking code that sets the cookie to be shared across the multiple subdomains. For example, if you wanted to track `yourlibrary.org` and a children's site at `kids.yourlibrary.org` in one profile, you embed the following script on both sites:

```

<script type="text/javascript">

var _gaq = _gaq || [];
_gaq.push(['_setAccount',
  'UA-XXXXXX-XX']);
_gaq.push(['_setDomainName', '
  yourlibrary.org ']);
_gaq.push(['_trackPageview']);

(function() {
var ga = document.
  createElement('script');
  ga.type = 'text/javascript';
  ga.async = true;

```

Figure 3.1
Creating a filter for a subdomain

```

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);
}) ();

</script>

```

Once the tracking script is in place, all the data from both subdomains is collected and reported in a master profile; the script tricks Google Analytics into thinking that the content on both of the sites belongs to one domain. You can still create filtered profiles to track both of these subdomains separately. Create a predefined filter to include the traffic to just that domain, as shown in figure 3.1. This allows you to still analyze visitor data at the single subdomain level while keeping the master profile intact.

Interpreting Content Reports in Combined Profiles

By implementing subdomain or cross-domain tracking, you may notice it is difficult to distinguish between different pages in Content reports because Google Analytics does not provide the full URL of the page—only the information after the hostname or subdomain. Hence, you could have listed in your reports multiple `index.html` pages that refer to very different pages! To fix this issue, you will need to apply a custom filter to the master profile that tells Google Analytics to always provide the full hostname/subdomain of the

page data. Figure 3.2 demonstrates the configuration for that specific filter. Note that (.*) is regex, which pulls the full hostname (extract A) and full request URI (extract B) from the data collected in Google Analytics, and the constructor combines the data from extract A and extract B to create a full URL in all of the reports in the profile to which the filter is applied. This advanced filter will help make any combined profile less confusing.

Tracking Multiple Domains in One Profile: Cross-Domain Tracking

There are times when you may want to combine all data from different websites into one seamless profile. Let's say you want to track all your library's web presences in one profile in order to track the complete usage of your library's sites and how those sites direct visitors to each other to create one giant library web-o-sphere. To successfully do this, you will need to identify all web presences the library manages and use cross-domain tracking to monitor those different domains in one profile. This process is similar to tracking multiple subdomains—you need to alter the tracking code to set the cookie to track the different sites. In this example, we want to combine the tracking for a main library's website (yourlibrary.org) and a hosted discovery tool (discoverytool.com). Since these sites are in different domains, each will need its own customized tracking code that uses the same web property ID. The code for the library's website would be:

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

var _gaq = _gaq || [];
_gaq.push(['_setAccount',
'UA-XXXXXX-XX']);
_gaq.push(['_setDomainName', '
yourlibrary.org']);
_gaq.push(['_setAllowLinker',
true]);
_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.
getElementsByName('script')
[0]; s.parentNode.
```

Figure 3.2
Creating a filter to display the full URL in reports

```
insertBefore(ga, s);
})();

</script>
```

The `_setDomainName` method creates the domain name for the cookie, while the `_setAllowLinker` enables cross-domain tracking. The code for the discovery tool shares the same web property ID to ensure the data is tracked in the same profile. However, the `_setDomainName` needs to be slightly different in order to track the data from the actual domain the code is on. Each different domain to be tracked will need its own tracking script with the correct domain name set. Here is the example code for the discovery tool:

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

var _gaq = _gaq || [];
_gaq.push(['_setAccount',
'UA-XXXXXX-XX']);
_gaq.push(['_setDomainName', '
discoverytool.com']);
_gaq.push(['_setAllowLinker',
true]);
_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);
})();
</script>

```

Customizing the tracking code for each site is just the first step. The next step involves identifying where a site refers visitors to the different domains—this could be links, or possibly forms. Once they are identified, additional JavaScript needs to be added to each link or form to successfully pass the same cookie settings between the two domains.² To put this into context, say that the library’s website has a link to the discovery tool on its home page so an additional script was added to that link:

```

<a href="http://discoverytool.
com/"
onclick="_gaq.push(['_link',
'discoverytool.com']); return
false;">Discovery Tool</a>

```

However, the library’s website also has an embedded search widget that allows visitors to use the discovery tool from its home page. Since that search widget uses a form to send the visitor’s search to the discovery tool, the form script is slightly altered to pass the cookie information:

```

<form name="searchwidget"
action="http://discoverytool.
com" method="post" onsubmit="_
gaq.push(['_linkByPost',
this]);">

```

If you miss this step, the domains will not share the same cookie, which means you are not truly tracking cross-domain. Again, it will inflate your visitor data and treat interactions between sites as referrals. A great best practice is to identify all referring elements (links or forms) to any of the domains being tracked in one profile before starting this process. This will help identify how much setup time will be required during implementation. Remember, by default, tracking different domains as one creates a master profile that contains all the sites’ data, but filtered profiles can be created to analyze the data from each site.

You might be wondering if it’s worth the effort to implement cross-domain tracking, and rightfully so. The work required for altering the tracking script in the ways required across all of the desired platforms may not be possible or worth the return on the investment of your time. Consider figure 3.3, from DePaul University

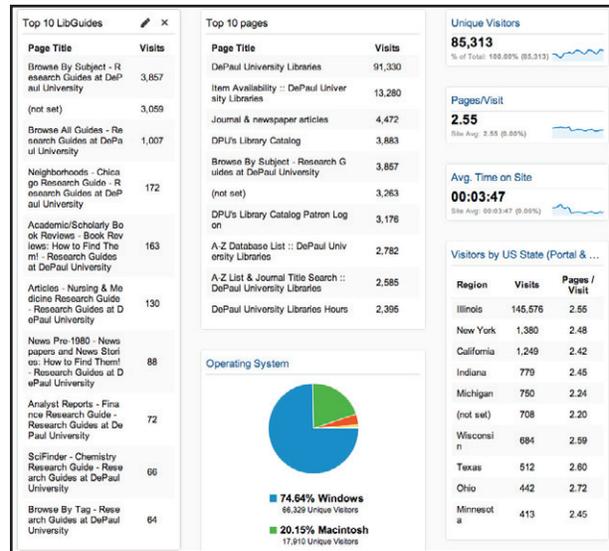


Figure 3.3 Google Analytics dashboard displaying data gathered from cross-domain tracking, DePaul University

Libraries, which displays DePaul University Libraries’ “über dashboard,” a veritable Holy Grail for library web analytics lovers. The library staff at DePaul credit cross-domain tracking with making analysis easier, allowing goal tracking across different platforms, and providing a “holistic” view of user activity.³

Considerations before Implementing an Advanced Tracking Method

Does Your Tool Allow Customized Tracking Code?

When implementing cross-domain tracking, be sure to consider any possible limitations to adding tracking code to the web products you wish to combine under one web property ID—see chapter 2 for more information about specific platforms. For example, your catalog vendor may not allow any customizations to be made to the JavaScript snippet that returns analytics data, which could have a negative impact (the escalated visitor/domain referral problem) on a cross-domain implementation. Further, for cross-domain tracking, you will also need to ensure that you can modify links between the domains to prevent them from being listed as referrers from one another.

Does Your Library Need to Use Cross-Domain Tracking?

As with any analytics decision, whether or not you want to implement advanced tracking configurations depends on what you want to find out. It may take a while to configure and test a new setup; will there be a return on the investment of your time? Again,

consider the audience and purpose of your different sites and tools: which are meant to be used together? Which stand on their own? Would you benefit from being able to track goals for common user actions spanning more than one domain, like a website/catalog interaction?

What About Sites Already Being Tracked?

If you are already tracking multiple sites with the one web property ID/one domain Google Analytics default, you will have to abandon the profiles for the domains that you will combine into one. This is okay, so long as it fits your master assessment plan; further, the data for those abandoned web property IDs will still be available to you for Google Analytics's 25-month retention period, even if you stop using them. In other words, you will still have the data that you've already collected, but you won't be able to aggregate it.

Conclusion

In a perfect world, we'd all be able to define our own web environments easily and precisely. In reality, we share virtual space with other entities, subdivide domains to host more than one site, and attempt to combine multiple domains into a single user experience. The default Google Analytics configuration of one web property ID per domain doesn't necessarily always reflect how we would like to track site usage. This chapter has demonstrated several basic methods

libraries can use to manage advanced tracking scenarios. See the additional resources listed below for cross-tracking code examples and methods. Despite the amount of time and effort required for cross-domain tracking, it can help libraries better manage their data from multiple web presences and create an optimal, user-centric tracking scenario.

Additional Resources

- “Domains & Directories,” Google Developers website, <https://developers.google.com/analytics/resources/concepts/gaConceptsDomains>.
- “Google Analytics Tracking Code,” Google Developers website, <https://developers.google.com/analytics/devguides/collection/gajs/methods>.
- “Tracking Multiple Domains—Web Tracking (ga.js),” Google Developers website, <https://developers.google.com/analytics/devguides/collection/gajs/gaTrackingSite>.

Notes

1. Brian Clifton, *Advanced Web Metrics with Google Analytics*, 3rd ed. (Indianapolis, IN: John Wiley & Sons, 2012), PDF e-book, chap. 2.
2. Ibid, chap. 7.
3. M. Ryan Hess, “Über Analytics: Customizing Google Analytics to Track Multiple Library Platforms” (presentation, Internet Librarian, Monterey, CA, Oct. 22–24, 2012), http://conferences.infoday.com/documents/158/B104_Hess.pdf.