OMNITURE™

WHITE PAPER

REDIRECTS AND ALIASES

Properly Recording Data when Using Redirects

February 23, 2006 Version 1.0

1 Redirects and Aliases

Redirects point the browser to a new location without user interaction. They are executed at either the web browser (client-side redirect) or at the web server (server-side redirect). Because redirects do not require any user interaction, redirects are often executed without the user ever noticing. The only thing that indicates a redirect has occurred is the browser's address bar. The address bar displays a URL that is different from the link the browser initially requested.

Although there are only two types of redirects, they can be implemented in numerous ways. For example, client-side redirects can occur because the web page to which a user has pointed his or her browser contains scripting or special HTML code that redirects the browser to another URL. Server-side redirects can occur because the page contains server-side scripting or because the web server has been configured to point the user to another URL.

1.1 SiteCatalyst and Redirects

SiteCatalyst gathers some of its data from the browser, and relies upon certain browser properties. Two of those properties, the "Referring URL" (or "referrer") and the "Current URL" can be changed by a server-side redirect. Since the browser is aware that one URL has been requested, but a different URL has been returned, it clears the Referring URL. The result is the referring URL is blank, and SiteCatalyst may report that no referrer existed for the page.

1.1.1 Example: Browsing Without Redirects

Consider the following hypothetical scenario in which the user does not encounter a redirect:

1. User points her browser to www.google.com, and types, “discount airline tickets” into the search field, and then clicks the Search button.
2. The browser displays the search results including a link to your site, http://www.flywithus.com/. After displaying the search results, the browser's address bar displays the search terms that the user entered in the search field (http://www.google.com/search?hl=en&ie=UTF-8&q=discount+airline+tickets). Notice that the search terms are included in the URL query string parameters that follow http://www.google.com/search?.
3. The user clicks on the link to your hypothetical site http://www.flywithus.com/. When the user clicks on this link and lands on the flywithus.com web site, SiteCatalyst uses JavaScript to collect the referring URL (http://www.google.com/search?hl=en&ie=UTF-8&q=discount+airline+tickets) as well as the current URL (http://www.flywithus.com/).
4. SiteCatalyst reports the information collected during this interaction in various reports, such as Referring Domains, Search Engines, and Search Keywords.

1.1.2 Example: Browsing With Redirects

Redirects can cause the browser to blank out the true referring URL. Consider the following scenario:

1. User points her browser to www.google.com, and types, “discount airline tickets” into the search field, and then clicks the Search button.
2. The browser window’s address bar displays the search terms that the user typed into the search field (http://www.google.com/search?hl=en&ie=UTF-8&q=discount+airline+tickets). Notice that the search terms are included in the URL query string parameters that follow "http://www.google.com/search?". The browser also displays a page that contains the search results including a link to one of your domain names: http://www.flytohawaiiforfree.com/. This *vanity* domain is configured to redirect the user to http://www.flywithus.com/.
3. The user clicks on the link http://www.flytohawaiiforfree.com/ and is redirected by the server to your main site, http://www.flywithus.com. When the redirection occurs, the data that is important to SiteCatalyst data collection is lost because the browser clears the referring URL. Thus, the original search information used in the SiteCatalyst reports (i.e., Referring Domains, Search Engines, Search Keywords) is lost.

Implementing Redirects for SiteCatalyst in this document discusses how to leverage SiteCatalyst variables to capture the data lost in the re-direct. Specifically, the section discusses how to fix the "discount airline tickets" situation described above.

1.2 Implementing Redirects for SiteCatalyst

In order to capture SiteCatalyst data from redirects, four minor alterations need to be made to the code that creates the redirect and the SiteCatalyst .JS file. The following steps will retain the information that the original referrer (e.g. http://www.google.com/search?hl=en&ie=UTF-8&q=discount+airline+tickets in the scenario above) passes to your site.

1.2.1 Configure "Referrer Override" JavaScript Code

Omniture Live Support can provide you with a JavaScript file that contains code that will override the default functionality SiteCatalyst uses to capture the referrer from the HTTP header. The code snippet below shows two JavaScript variables, s_referrer and s_pageURL. This code is placed on the ultimate landing page of the redirect.

G Code

```
<!-- SiteCatalyst code version: H.0.
Copyright 1997-2005 Omniture, Inc. More info available at
http://www.omniture.com -->
<script language="JavaScript" src="//INSERT-DOMAIN-AND-PATH-TO-CODE-
HERE/s_code.js"></script>
<script language="JavaScript"><!--
/* You may give each page an identifying name, server, and channel on
the next lines. */var s_pageName=""
var s_server=""
var s_campaign=""
var s_referrer=""
var s_pageURL=""
```

H Code

```
<!-- SiteCatalyst code version: H.0.
Copyright 1997-2005 Omniture, Inc. More info available at
http://www.omniture.com -->
<script language="JavaScript" src="//INSERT-DOMAIN-AND-PATH-TO-CODE-
HERE/s_code.js"></script>
<script language="JavaScript"><!--
/* You may give each page an identifying name, server, and channel on
the next lines. */
s.pageName=""
s.server=""
s.campaign=""
s.referrer=""
s.pageURL=""
```

1.2.2 Modify the Redirect Mechanism

Because the browser strips referring URL, you must configure the mechanism that handles the redirect (e.g. the web server, server-side code, client-side code) to pass along the original referrer information. If you would also like to record the alias link URL, this must also be passed along to the ultimate landing page. Use the s_pageURL variable to override the current URL.

Since there are many ways to implement a redirect, you may need to check with your web operations group to identify the specific mechanisms that execute redirects on your web site. The example below shows how one might pass along the original referrer and alias link information to the ultimate landing page using a client side redirect mechanism.

Since there are many ways to implement a redirect, you may need to check with your web operations group or your online advertising partner to identify the specific mechanisms that execute redirects on your web site.

1.2.3 Capture the Original Referrer

Normally, SiteCatalyst will obtain the referring URL from the browser's document.referrer property, and the current URL from the document.location property. By passing values to the referrer and pageURL variables, you can override the default processing. By passing a value to the referrer variable, you are telling SiteCatalyst to ignore the referrer information in the document.referrer property and to use an alternative value that you define.

Therefore, the final version of the landing page would need to contain the following code to correct the issues introduced in the "discount airline tickets" scenario.

G Code

```
<!-- SiteCatalyst code version: G.7.
Copyright 1997-2004 Omniture, Inc. More info available at
http://www.omniture.com --><script language="JavaScript"><!--
/* You may give each page an identifying name, server, and channel on
the next lines. */
var s_pageName=""
var s_server=""
var s_campaign=""
var s_referrer="http://www.google.com/search?hl=en&ie=UTF-
8&q=discount+airline+tickets"
// Setting the s_pageURL variable is optional.
var s_pageURL=http://www.flytohawaiiforfree.com
```

H Code

```
<!-- SiteCatalyst code version: H.0.
Copyright 1997-2005 Omniture, Inc. More info available at
http://www.omniture.com -->
<script language="JavaScript" src="http://INSERT-DOMAIN-AND-PATH-TO-CODE-
HERE/s_code.js"></script>
<script language="JavaScript"><!--
/* You may give each page an identifying name, server, and channel on
the next lines. */
s.pageName=""
s.server=""
s.campaign=""
s.referrer=http://www.google.com/search?hl=en&ie=UTF-8&q=discount+airline+tickets
// Setting the s.pageURL variable is optional.
s.pageURL="http://www.flytohawaiiforfree.com"
```

1.2.4 Verify Referrer with the SiteCatalyst Debugger

Run a test to verify that the referrer, originating URL (s_server) and campaign variables are being captured. These variables will be represented as the following parameters in the SiteCatalyst Debugger.

	URL or Query String Value	Value as Shown in the SiteCatalyst Debugger
Original Referrer	`http://www.google.com/search%3Fhl%3Den%26ie%3DUTF826q%3Ddiscount%2Bairline%2Btickets`	`r=http:/ref=www.google.com/search?hl=en&ie=UTF-8&q=discount+airline+tickets`
Page URL	`http://www.flytohawaiiforfree.com`	`g=http://www.flytohawaiiforfree.com` This value will appear in the SiteCatalyst debugger if the pageURL variable is used
Ultimate Landing Page URL	`http://www.flywithus.com`	This value will NOT appear in the SiteCatalyst debugger if the pageURL variable is used

The text that the debugger displays should correspond to the following example.

Image

```
http://flywithuscom.112.2o7.net/b/ss/flywithuscom/1/H.0-Pd-R/s61944015791667?[AQB]
ndh=1
t=11/5/2004 12:4:57 5 360
pageName=Welcome to FlyWithUs.com
r=http://ref=www.google.com/search?hl=en&ie=UTF-8&q=discount+airline+tickets
cc=USD
g=http://www.flytohawaiiforfree.com
s=1280x1024
c=32
j=1.3
v=Y
k=Y
bw=1029
bh=716
ct=lan
hp=N
[AQE]
```

After verifying that the SiteCatalyst debugger displays these variables, it is always helpful to confirm that the search terms and the original referring domain (prior to the redirect) are registering traffic in the SiteCatalyst interface.

1.3 Notes

You might be tempted to use SiteCatalyst's getQueryParam plug-in to populate the referrer variable and the pageURL variables with the original referrer and the alias link. While the getQueryParam plug-in is an easy way to populate SiteCatalyst variables with query string values, it must be implemented in connection with a temporary variable so that legitimate referrers are not overwritten when the query string is empty. The best way to use getQueryParam is in connection with the getValue plug in as outlined with the following pseudo-code, which should be added to the "doPlugins()" function within the "s_code_remote.js" or "s_code.js" file as shown below.

1.3.1 G Code

```
/* Plugin Config */
s_usePlugins=true
```

```
function s_doPlugins(){
 /* Add calls to plugins here */
 s_vp_getQueryParam('tempVar','origref');
 if(s_vp_getValue('tempVar'))
  s_vp_getQueryParam('s_referrer',"origref');
```

1.3.2 H Code

```
/* Plugin Config */
s.usePlugins=true
function s_doPlugins(s) {
       /* Add calls to plugins here */
       // Copy original referrer into s.referrer if it exists
       var tempVar
       tempVar=s.getQueryParam('origref')
       if(tempVar)
              s.referrer=tempVar;
}
s.doPlugins=s_doPlugins
```

OMNITURE™

CALL 1.877.722.7088
1.801.722.0139

www.omniture.com
info@omniture.com

550 East Timpanogos Circle
Orem, Utah 84097

OMNITURE™