OMNITURE™

WHITE PAPER

IMPLEMENTING WITH AJAX

Tracking Rich Media Applications

June 6, 2006 Version 1.0

1 Overview

AJAX is an emerging concept in web design that uses multiple technologies to create and manage dynamic content on web pages. The need to track user interaction with AJAX and other rich media applications is paramount to analytics success and realization of the return on investment in the web design. The focus of this white paper is to address the question above and provide recommendations for tracking rich Internet applications, specifically those that use AJAX.

1.1 Rich Internet Applications (RIA)

Rich Internet Applications (RIAs) are changing the face of the web. They bridge the gap between the promise of on-demand technologies for the masses and realistic user experiences. RIAs have been maturing for years. The biggest leaps forward have occurred with wide-scale adoption of browsers which support the underlying technologies that power these applications. While RIA has many forms and supporting technologies, the most common, and perhaps most widely adopted, are AJAX and Flash. It is important to understand that the technology is not what defines an RIA, but instead, its usability and application.

RIA looks cool, and is changing the web, but, should you use it on your site? This is an area where caution is important. Rich Internet Applications are expensive to develop. Don't jump in without using SiteCatalyst to test the water.

1.1.1 What to Track

One of the most commonly asked questions with RIA is how to track micro-level activity separate from macro-level activity, and when it is appropriate to do either. For example, say you have an application that allows customers to uniquely configure a product. The application may have significant steps to which users are exposed. Are these steps considered page views? In addition, there are micro-level activities within each step. Should these activities be tracked as page views? What if you want to understand the flow between activities, or which features get the most activity? The trouble with RIA is that each application is unique. They are designed to mimic realistic actions, and since actions are relative to the situation, the possibilities are endless. However, most applications have a few major components: milestone steps, features, and micro-level actions within features. So, while each application requires some consideration as to what specifically should be measured, there are some generalizations that can be applied to RIA tracking.

Macro-Level Activity

Macro-level activity usually constitutes the loading of the application, which provides information on visits, visitors, instances, value to future actions, etc., but it can and should also represent major steps in the process. A good rule of thumb is that if an RIA action changes the application more than 50% (or whatever is considered significantly changing the user experience or content), then it is macro-level, and should be tracked as a page view.

Micro-Level Activity

Micro-level activity includes any changes less than 50% (or not considered as significantly changing the user experience or content). Toggling between color selections, for instance, would be considered micro-level activity. Omniture recommends that micro-level tracking be related to features. For example, in the case of toggling between colors, is it really important to understand which colors were considered? Or is it more important to know that the color selection feature was used? Perhaps both are important, and if so, capture both; but, when measuring the effectiveness of RIA, consider the feature level activity as being more valuable. All micro-level activity should be tracked as "custom links" with specifics measured through associated "traffic" variables (props and even eVars if the use needs to be measured against success events). This will ensure that page views are not inflated by micro-level activity, and allows for path analysis through the traffic variable.

1.1.2 What to Analyze

It is important to understand how effectively your RIA is driving success. Success is most commonly measured through conversions. A macro-level analysis will provide insight into RIA effectiveness as a whole. Micro-level analysis may provide insight into which features help drive conversion.

Second, you should measure efficiency of your RIA. This is an analysis of micro-level activity relative to the RIA macro metrics. Do users go through more steps than necessary to arrive at the same goal? Analysis metrics might include visits/features activity; page views/feature activity, visitors/feature activity, etc.

Finally, conduct analysis on path flow and fall out. Are users avoiding the RIA and finding another path to the goal? Run SiteCatalyst fallout reports built around the site and RIA flow. Run path analysis from landing pages to gauge the true traffic patterns. Look at barriers and incentives to guide users toward the goal.

1.1.3 Suggested Metrics

- RIA Visits
- RIA Visitors
- RIA Page Views
- RIA Feature Activity (Custom Links) measure click activity by feature

1.1.4 Suggested Analysis

- RIA Feature Activity / RIA Page Views
- RIA Feature Activity / RIA Visits -
- RIA Page Views / Success Metric – Conversion Ratio: measures application effectiveness
- Total RIA Activity / Success Metric – Conversion Ratio: measures application efficiency
- Feature RIA Activity / Success Metric – Conversion Ratio: measures application feature efficiency
- Path Flow to and from RIA
- Fallout Rates through RIA conversion process

2 Implementing with AJAX

2.1 Designing the Solution

Implementing with AJAX is exactly like deploying code on a standard HTML page. The business has questions that need answers, the needs are assessed and variables assigned, and the design is applied and deployed. These concepts should be familiar if you have already been through the initial stages of implementation.

The difference when throwing AJAX into the mix is first understanding the level of detail that needs to be gathered. The potential of content changing on the page (macro-level) or tracking attributes of the application (micro-level) will determine which variables will need to be set and which method of sending data to Omniture will work best.

2.2 Deploying the Code

There are two functions in the JavaScript code that allow you to send data to SiteCatalyst. There are some distinct guidelines that should be followed to know which method should be used to send data.

2.2.1 Collecting Macro-Data (Page)

The t() function in Omniture's code will send a standard format image request, incrementing total site page views. All Omniture variables that have been assigned values will send data. The primary focus of using this function within RIAs revolves around the value of the pageName variable. You should use this function when:

- The new content is considered a page view for the site or is considered moving from one "page" to another
- The content of the page changes more than 50% (or whatever is considered significantly changing the user experience or content)
- The page path must be tracked for each user interaction with the RIA

Sending Macro-Data (Page)

If an image request was previously made on the same page, you should first clear the values of the previously-set variables. This can be accomplished by writing a simple JavaScript function to clear the Omniture variables. The next step would be to set the values appropriate for the changed content, namely the pageName variable. Finally, after the variables are set call the t() function.

Syntax

```
<clear/set variables>

void(s.t());
```

Example

```
s.pageName="New Page"

s.prop1="some value"

void(s.t());
```

2.2.2 Collecting Micro-Data (Link)

The tl() function is different in that it does not send the pageName variable and is also selective in the other variables that are sent. Because of this restrictive design, it is well suited for tracking micro-level statistics such as button clicks, link clicks, scrolling, and other similar events. You should use this function when:

- You do not want to include a new page name or inflate page view statistics for the site
- The tracking of user interactions with buttons, links, or specific features within a page/application is desired
- You want to track the interaction path of the application

NOTE: Path analysis may be enabled for traffic variables (not restricted to page pathing) which may be a useful tool in analyzing the order in which users interact with your page or application. For example, you could see that a user interacted first with "button1" and then clicked on "button4" before leaving the application. This link-level path can be traced without inflating page-level statistics by using the tl() function.

Sending Micro-Data (Link)

If an image request was previously made on the same page, you should first clear the values of the previously-set variables. This can be accomplished by writing a simple JavaScript function to clear the Omniture variables. Second, set the linkTrackVars and linkTrackEvents variables if you have not already done it in the s_code.js file. The next step would be to set the values appropriate for the changed content, namely the pageName variable. Finally, after the variables are set call the tl() function.

For additional reference and details, please refer to the Implementation Manual under the Custom Links heading.

Syntax

```
s=s_gi('RSID');

<set linkTrackVars and linkTrackEvents> (if applicable)

<set new variables>

s.tl(this,'o','Link Name');
```

Example

```
s=s_gi('myreportsuiteid');

s.linkTrackVars="prop1,eVar1,events"; s.linkTrackEvents="event1";

s.prop1="some value"; s.eVar1="another value"; s.events="event1";

s.tl(this,'o','My Link Name');
```

2.2.3 Placement of Code

There are generally two places to track data with AJAX – at the time of the request or in the reply. In most cases, macro-data (page information) should be sent at the time of reply by having the code embedded in the HTML of the new content. For micro-data tracking (links, etc.) it is more common to use a custom links approach by inserting the code in the onClick attribute of the link, button, etc.

OMNITURE™

CALL 1.877.722.7088
1.801.722.0139

www.omniture.com
info@omniture.com

550 East Timpanogos Circle
Orem, Utah 84097

OMNITURE™