



851 SW 6th Ave., Suite 1600  
Portland, OR 97204  
1.503.294.7025  
fax: 1.503.294.7130

Webtrends Sales  
1.888.932.8736  
sales@Webtrends.com

**Europe, Middle East, Africa**  
+44 (0) 1784 415 700  
emea@Webtrends.com

For offices worldwide, visit:  
[www.Webtrends.com](http://www.Webtrends.com)

WHITEPAPER / **USING WEBTRENDS ANALYTICS WITH MICROSOFT SILVERLIGHT.**

# Using Web Analytics with Microsoft Silverlight

Michael S. Scherotter, Developer Evangelist, Microsoft

## EXECUTIVE SUMMARY

Webtrends and Microsoft Silverlight™ make a valuable combination for measuring the effectiveness of Web 2.0 technologies. When you analyze the performance of Rich Internet Applications (RIAs) and media players created with Microsoft Silverlight with the powerful behavioral data and online usage insight you gain through Webtrends, it helps you measure how web sites and marketing campaigns perform against your business goals. Because Webtrends uses JavaScript page tags to measure online behavior, it's easy to integrate with Silverlight's JavaScript programming model.

## USING MICROSOFT SILVERLIGHT TO CREATE INTERACTIVE ONLINE EXPERIENCES

Silverlight is the new cross-platform RIA deployment vehicle from Microsoft. It allows the creation of interactive web applications that employ high quality streaming media, vector graphics, images, and animation. Web developers can create interactive applications knowing that they'll present identical user experiences on the majority of web browsers in use today. Deployed as a browser plug-in control for FireFox, Mac Safari, and Internet Explorer, Silverlight can be programmed using familiar JavaScript and XAML, a declarative XML-based language. This architecture enables using AJAX-style programming patterns for building responsive, interactive, applications. An event model within the Silverlight control allows for web developers to easily add JavaScript-based event handling for mouse, keyboard, and media streaming events that are triggered by invisible markers embedded in the media.

## Configuring Silverlight RIA with Webtrends Analytics

### GETTING STARTED: DCSMULTITRACK

Webtrends tracks traditional web pages via a JavaScript tag that executes upon page load. With RIAs and other implementations that rely on post-page-load events, Webtrends utilizes a helper function to collect information about visitor activities. This helper function is called `dcsMultiTrack`.

The purpose of `dcsMultiTrack` is to handle the packaging of information and transmission of that data to the data collector when a post-page-load event occurs. `dcsMultiTrack` may be triggered automatically via standard event handlers, or can be called manually.

### HOW DCSMULTITRACK WORKS

Activating the `dcsMultiTrack` function requires associating the activity you want to track with a piece of JavaScript code called an event handler. An event handler specifies the action to take when a condition such as a mouse click or a form element change occurs.

When a visitor performs an action that calls the `dcsMultiTrack` function, `dcsMultiTrack` activates the page's JavaScript tag and sends the same information gathered during a

normal view of that page. Normally this would merely duplicate a page view that has already been recorded. However, dcsMultiTrack can overwrite or supplement any data you choose from the hosting page, so you can specify exactly what information is recorded for that specific collection event.

The following sections show how to construct these virtual hits to represent events such as clicks to download PDF documents, off site links, and Flash movie events.

### **IMPLEMENTING DCSMULTITRACK**

Before coding the pages that you want to track, you must add the dcsMultiTrack function to the Webtrends JavaScript tag.

To implement the dcsMultiTrack function, follow these steps. Note: If you use Webtrends On Demand and you created your data source using Webtrends Analytics v8.0 or higher, this function is included in your JavaScript tag. Otherwise, contact Webtrends Support to get a copy of this function.

**Step 1:** Locate the dcsMultiTrack function. If you use the Webtrends SmartSource Data Collector with Webtrends software, this function is stored in the multitrack.js file in this directory: SDC Installation Directory\util\javascript\

**Step 2:** Open the multitrack.js file, select all the code in the file, and copy it.

**Step 3:** Open your Webtrends JavaScript tag file, paste the code, and save the file.

### **DCSMULTITRACK COMMAND SYNTAX**

Web analysts are a subset of business users. They provide deep scrutiny and examination of analytics data to derive insight into online performance.

#### **WEB ANALYSTS TYPICALLY:**

Use the following syntax to invoke the dcsMultiTrack function:

```
mouseEvent="dcsMultiTrack('parameter1', 'value1', 'parameter2', 'value2');"
```

The parameter and value pairs represent the information you want Webtrends On Demand to collect when the event takes place, and mouseEvent represents the event attribute you want to use to activate dcsMultiTrack.

The values that you specify replace the values that would normally be gathered by the Webtrends JavaScript tag. You should specify different values for the event than the values specified for the page on which you called dcsMultiTrack.

In order to use dcsMultiTrack effectively, you will also need to use some Webtrends query parameters. For more information about Webtrends query parameters, see Chapter 9, "Webtrends Query Parameter Reference" in the Webtrends Analytics On Demand Implementation Guide. (You can find it in the in-product Customer Center resources.)

## RIA SCENARIOS

There are two main RIA scenarios that will be addressed here: collecting data about user interaction and collecting data about media viewer drop-off. Both of these scenarios are easy to measure using Silverlight with minimal user impact. In both cases, the granularity of data collection affects application performance, which means adding more instrumentation to a RIA may slow its responsiveness. For any web application, you may want to tune the instrumentation to balance the specific data collection needs with responsiveness requirements of the RIA.

### COLLECTING USER INTERACTION DATA

Web developers collect user interaction data for RIA in order to tune it to accomplish the specific business goals. For instance, if the user is presented with a number of option buttons, the web developer may want to enable both the business logic for each option as well as analytics logging for each button press. In the example below, the user is presented with three red option buttons.

#### Silverlight Web Analytics Demonstration

How do you measure the success of your rich internet application? With Web Analytics. Because most web analytics packages today rely on JavaScript, they are very easy to integrate with Silverlight.

This Silverlight application is collecting usage data with WebTrends On Demand.



Tell Me How

Show Me How

Try WebTrends On Demand for Silverlight

<http://blogs.msdn.com/synergist>

Once the Silverlight application is created by the interaction designer, the web developer can hook up the JavaScript code to the mouse button event handlers for each button. In this case, once the application is loaded into the Silverlight control (but before it's displayed), an event callback is triggered. In that event callback, the web developer adds the event handlers to the call JavaScript functions when specific user interactions occur. The following code adds event handlers for mouse over, mouse leave, and mouse click events for each button:

```
Analytics.Scene.prototype =  
{  
//This is called once the control has been loaded  
handleLoad: function(control, userContext, rootElement)
```

```

        {
            this.control = control;
            AddButtonHandlers(rootElement, "TellMe");
            AddButtonHandlers(rootElement, "ShowMe");
            AddButtonHandlers(rootElement, "SendInfo");
        }
    }
    //Add event handlers to each button
    function AddButtonHandlers(rootElement, name)
    {
        var element = rootElement.findName(name);
        // These are the event handlers for user feedback and business logic
        element.addEventListener("MouseEnter", "MouseOverButton");
        element.addEventListener("MouseLeave", "MouseLeaveButton");
        element.addEventListener("MouseLeftButtonUp", name);
        // This is the event handler for analytics
        element.addEventListener("MouseLeftButtonUp", "Collect");
    }
    // Log an event based on the element's name
    function Collect(sender, eventArgs)
    {
        LogEvent(sender.Name);
    }
    // Generic call to log an analytics event
    function LogEvent(text)
    {
        // This is the Webtrends function to track a RIA event
        dcsMultiTrack("DCS.dcssip", "blogs.msdn.com/synergist",
            "DCS.dcsuri", text);
    }
    // Webtrends tracking function
    function dcsMultiTrack(source, url, type, text)
    {
        //...
    }

```

Now, when the user starts interacting with the RIA, in addition to the interaction and business logic associated with each button, the Collect() function is called, which logs an event associated with the name of the visual element that triggered the event. Silverlight exposes an extensive event model that includes keyboard, mouse, focus, resizing, downloading, media playback, and error handling. Any of these events can trigger analytics collection.

## COLLECTING MEDIA STREAMING DATA

When content providers stream video or audio on the web, they want to know drop off rate: when people stop watching or listening to content by closing the browser or navigating away from a page. Obviously there are many factors that can contribute to this behavior (many of which are beyond the scope of the web browser model), but the event model in Silverlight can help measure some of the user actions that signify the cessation of media consumption.

Any video or audio media stream that is played in Silverlight can have markers embedded in it. Markers are invisible data elements that are part of the media stream which are embedded at specific times within the stream. These markers can be added to the stream when it is encoded or through the Silverlight Document Object Model (DOM) when played back. Windows Media Encoder and Expression Media Encoder are two Microsoft products that can be used to add Markers to a media stream as it is being encoded.

Event handlers can be added to the Silverlight RIA that so that when the media is played on the client system, JavaScript code is called and analytics information is logged. Markers can be added at regular intervals throughout the length of the content or only at specific times linked to the content; for example before/during/after commercials. Once that information is collected, it becomes clear how web users are viewing media content on a Silverlight application. In the following code, two new media event handlers have been added to the "Movie" element, in the Load event handler for the Silverlight RIA:

```
Analytics.Scene.prototype =  
{  
  
    //This is called once the control has been loaded  
    handleLoad: function(control, userContext, rootElement)  
    {  
  
        this.control = control;  
        AddButtonHandlers(rootElement, "TellMe");  
        AddButtonHandlers(rootElement, "ShowMe");  
        AddButtonHandlers(rootElement, "SendInfo");  
        var movie = rootElement.findElement("Movie");  
        movie.addEventListener("MediaOpened",  
        "AddMarkers");  
        movie.addEventListener("MarkerReached",  
        "MarkerReached");  
  
    }  
  
    }  
    // Add markers every 5 seconds  
    function AddMarkers(movie, eventArgs)  
    {  
  
        var markers = movie.markers;  
        var duration = movie.NaturalDuration.Seconds;  
        var host = movie.getHost().content;
```

```

    for (var i = 0; i <= duration; i = i + 5)
    {
        var markerXaml = "<TimelineMarker Type='Name'
        Text='Saw" + i + " Time='0:00:00'/>";
        var marker = host.CreateFromXaml(markerXaml);
        marker.time.seconds = i;
        markers.Add(marker);
    }
}
// Called whenever a marker is reached in the media playback
function MarkerReached(sender, eventArgs)
{
    LogEvent(sender.Source + "/" + eventArgs.Marker.Text);
}

```

Once the media is loaded, then markers can be added to the media stream. If the media already has markers in it, this step is unnecessary. In the code above, markers are added every five seconds in the media stream with the format of "Saw5", "Saw10", "Saw15", etc. An event is logged with the media filename and the marker text. Other properties such as whether the application is full-screen, the media download status, and the volume can also be captured. If the application developer exposes standard playback controls to the user (such as play, stop, fast-forward, rewind, and pause), then events triggered by those actions can be logged via analytics as well in a similar manner.

## CONCLUSIONS

As more web sites use RIAs to enhance their user experience, understanding the behavior of those users becomes more important. The architecture of Silverlight makes it possible to integrate web analytics such as Webtrends into the web development workflow before, during, or after Silverlight assets are created. As a result, companies that use Silverlight are more likely to understand and adapt to user behavior across their entire web site.

## ADDITIONAL RESOURCES

- Microsoft Silverlight: <http://microsoft.com/silverlight>
- Webtrends: <http://www.Webtrends.com>

## About Webtrends Inc.

Webtrends is a trusted analytics advisor in the business of collecting, analyzing, delivering and ultimately transforming data into understanding. Webtrends delivers the industry's most recognized search engine marketing, visitor intelligence, and analytics solutions to enable companies to understand their customers, drive engagement, and enhance marketing and brand awareness. Thousands of global organizations, including Microsoft, Reuters, General Mills and Ticketmaster have chosen Webtrends business solutions and client services expertise to optimize their customers' online experiences. Webtrends was the first web analytics company, founded over 15 years ago.

Webtrends is a registered trademark of Webtrends Inc. in the United States and other countries. All other trademarks and registered trademarks are the properties of their respective owners. © 2007 - 2009 Webtrends Inc. All rights reserved.