Adobe

ADOBE® ONLINE MARKETING SUITE
Powered by Omniture®

Measurement Library (PHP)

IMPLEMENTATION GUIDE

1.1 07152010

Table of Contents

PREFACE iv

CHAPTER 1 - MEASUREMENT LIBRARY (PHP) 1

OVERVIEW 1
REQUIREMENTS 2
IMPLEMENT MEASUREMENT LIBRARY (PHP) 2
Application Config Variables 2
Track Config Variables 5
Methods 6
Plug-In Architecture (Optional) 8
Implementation Examples 8
SUPPORTED SITECATALYST REPORTS 10

Preface

The Adobe® Measurement Library (PHP) Integration Guide describes using the Measurement Library (PHP) to measure the usage of your Web site. It lets you capture certain types of activity on your Web site, and forward that data to Adobe data collection servers where it is available for use in the Adobe Online Marketing Suite, including SiteCatalyst® reports.

This guide is intended for Web developers, and assumes that you are familiar with both implementing SiteCatalyst data collection code, and PHP Web development.

Release Notes

Measurement Library for PHP 1.1 includes the following updates from version 1.0:

- Better GeoSegmentation accuracy (when `sendFromServer` is enabled).
- Fixed bug so user can now append to userAgent variable.
- Image beacon is now more compliant with more mobile browsers.
- The `imageDimensions` variable now defaults to `5x5` when `mobile` is enabled.
- Refined bot detection list.
- Added debug information (HTTP headers, response, errors, etc.) when `debugTracking` and `sendFromServer` are enabled.
- Added the `debugFilename` variable (when `sendFromServer` is enabled).
- The pagename variable defaults to `$_SERVER['SCRIPT_NAME']` when neither `pagename` nor `pageURL` are set.
- Full support for CGI implementations of PHP.
- Performance enhancements.

Terms and Conditions of Use

This document and the related software described in this document is proprietary to Adobe Systems, Inc. and is supplied only under license or nondisclosure agreement. The document and software can be used or copied only in accordance with the terms of the agreement (Enterprise Terms of Use - https://sc.omniture.com/p/l10n/1.0/terms.html).

The information in this document is subject to change without notice and does not represent a commitment on the part of Adobe.

Account Support

ClientCare is available to:

- Answer specific product questions.
- Help you utilize SiteCatalyst reports to the greatest effect.
- Resolve any technical difficulties you might have.
- Help you configure SiteCatalyst variables.

Service and Billing Information

Depending on the service level you have purchased, some of the options described in this guide might not be available to you. Additionally, each account has unique billing needs. Please refer to your contract for pricing, due dates, terms and conditions. If you would like to add to or otherwise change your service level, or if you have questions regarding your current service, please contact your Account Manager.

Adobe welcomes any suggestions or feedback you might have regarding Measurement Library (PHP) or the contents of this guide. Send comments to your Account Support.

Contact Information

[CORPORATE ADDRESS]	Adobe Systems, Inc. 550 East Timpanogos Circle Orem, UT 84097
[PHONE]	1.801.722.7000
[FAX]	1.801.722.7001
[TOLL FREE]	1.877.722.7088 (support, billing and sales)
[SUPPORT E-MAIL]	clientcare@omniture.com
[SALES E-MAIL]	sales@omniture.com
[INFORMATION E-MAIL]	info@omniture.com
[CORPORATE URL]	http://www.omniture.com
[LOG-IN URL]	http://my.omniture.com

Measurement Library (PHP)

Adobe® Measurement Library (PHP) lets you use PHP to monitor and evaluate vistor activity on your Web site. Once captured, data is available for analysis by the Adobe Online Marketing Suite.

This section includes the following topics:

- Overview
- Requirements
- Implement Measurement Library (PHP)
- Supported SiteCatalyst Reports

1.1 Overview

The Measurement Library provides the following key benefits:

- **Streamlined Implementation for Non-JavaScript Tagging:** Management Library provides an alternative to hard-coding JavaScript Web beacons on your Web pages, using familiar syntax and tools (for example, `track` and `tracklink`).
- **Integrated Data Tracking:** Measurement Library lets you incorporate external data into your data collection (external applications, off-line data, etc.)
- **Tagless Implementation:** Measurement Library supports server-to-server (Web server to Adobe data collection server) data transfers that can greatly simplify implementation and maintenance of application and Web site tracking. Server-side measurement also provides a global implementation option independent of the front-end application (mobile application, Web site, off-line application, etc.)
- **Visitor ID Management:** Measurement Library provides additional visitor identification options that improve visitor tracking, including limiting third-party cookie usage, using the carrier's subscriber ID, and using server-side tracking of visitor IDs.

Before implementing Measurement Library (PHP), you should understand the following concepts and limitations:

- Measurement Library (PHP) consists of the following software components. You can download Measurement Library software from SiteCatalyst's Code Manager. For information about using Code Manager, see "Code Manager" in the Admin Console User Guide, which is available in the Online Marketing Suite help system.

  OmnitureMeasurement.class.php: A PHP class library. Copy the class library to your Web server where it is accessible to PHP scripts.

- To view debugging information related to you Web site, you can use either the JavaScript debugger or standard Web browser output. Debug information includes request, request headers, response, and errors, when `sendFromServer` and `debugTracking` are enabled. For more information, see "debugTracking" on page 4.

  If you do not have the JavaScript debugger, contact your Account Manager.
- When using Measurement Library outside of a Web context, any data specific to an HTTP request will not be available (for example, pageURL, userAgent, etc.)

1.2 Requirements

- Measurement Library supports PHP 5 or later. For more information, visit www.php.net.
- Measurement Library requires the cURL library installed to use direct server-to-server requests. For more information, visit www.php.net/curl.
- Measurement Library assumes that all strings are in UTF-8 format. You must convert any non-UTF-8 strings before using them in Measurement Library object (for example, using `iconv()`).

1.3 Implement Measurement Library (PHP)

Once installed, Measurement Library (PHP) provides the following variables and methods for configuring event tracking on your Web site.

- Application Config Variables
- Track Config Variables
- Methods
- Plug-In Architecture (Optional)
- Implementation Examples

NOTE: When you set a variable directly on the object, the value acts as the default value until you change it on the object. However, you can use the variable overrides parameter to set a one-time-use value that persists only for the current track call, then resets to the default value. You cannot unset variables with variable overrides. For information about using variable overrides, see “Implementation Examples” on page 8.

Application Config Variables

The following variables are set when the Web page loads, and typically do not need to change. However, you can change them as needed.

VARIABLE	REQUIRED	DESCRIPTION
`account`	Yes	The report suite or report suites (multi-suite tagging) that you wish to track. Separate multiple report suites with commas. You cannot override this variable using variable overrides. For example: `$s->account = 'myrsid';`
`dc`	No	The Adobe data collection servers that receive the event tracking data. Consult your Account Manager to find out how you should set this value. 112 specifies the San Jose, CA collection servers, while 122 specifies the Dallas, TX collection servers. By default, this variable is 112 (San Jose). For example: `$s->dc = '112';`

VARIABLE	REQUIRED	DESCRIPTION
`linkTrackEvents`	No	A comma-separated list of events that restricts the current set of events for link tracking. If `$s->linkTrackEvents` is not defined, Measurement Library (PHP) sends all events. For example: `$s->linkTrackEvents = 'purchase,event1';`
`linkTrackVars`	No	A comma-separated list of variable names that restricts the current set of variables for link tracking. If `$s->linkTrackVars` is not defined, Measurement Library (PHP) sends all variables with values. Even though pageName is sent, its Page View count is not incremented. For example: `$s->linkTrackVars = 'events,prop1,eVar49';`

`trackingServer`
`trackingServerSecure`
	No	Identifies the collection domain.
		If you are using a first-party cookie in other SiteCatalyst implementations, set these variables to the same value as your other implementations.
		If you want to use these variables, you must set them both. You cannot set only one tracking server variable.
		For example:
		`$s->trackingServer = 'metrics.corp1.com';`
		`$s->trackingServerSecure = 'smetrics.corp1.com';`

`userAgent`	No	The HTTP `User-Agent` field that is sent with each track call to Adobe data collection servers. By default, Measurement Library sets a default value for this variable that includes the `User-Agent` field from the requesting Web page. If necessary, you can append to this default value. For example: `$s->userAgent .= 'more data';`
`visitorNamespace`	No	The domain used to set cookies. For more information about namespace, see the **SiteCatalyst Implementation Guide**. For example: `$s->visitorNamespace = 'mycompany';`
`currencyCode`	No	The Currency Code used for purchases or currency events that are tracked in your Web site. For example: `$s->currencyCode = 'USD';`
`visitorID`	No	Unique identifier for each visitor. For example: `$s->visitorID = '12345';`
`linkLeaveQueryString`	No	Enables (`true`) or disables (`false`) preserving the query-string on the URL for links to Web pages from your Web site. By default, this variable is `false`. You cannot override this variable using variable overrides. For example: `$s->linkLeaveQueryString = true;`

VARIABLE	REQUIRED	DESCRIPTION
`ssl`	No	Enables (`true`) or disables (`false`) sending tracking data via SSL (HTTPS). By default, this variable uses the same setting as the Web page where the PHP script executes. You cannot override this variable using variable overrides. For example: `$s->ssl = true;`
`debugTracking`	No	This flag applies only when the `sendFromServer` flag is enabled. Enables (`true`) or disables (`false`) viewing debug information in the Web browser. By default, this variable is `false`. Debug information includes the request, request headers, response, and errors, when `sendFromServer` and `debugTracking` are enabled. You cannot override this variable using variable overrides. For example: `$s->debugTracking = true;` **NOTE:**If `sendFromServer` is disabled, debug data is still accessible through the JavaScript debugger.
`debugFilename`	Yes	Specifies a file to receive debug information. If not specified, Measurement Library sends debug information to the Web browser. For example: `$s->debugFilename = ‘/home/debug.log’;`
`usePlugins`	No	Enables (`true`) or disables (`false`) using plug-ins as part of a delegate implementation. By default, this variable is `false`. You cannot override this variable using variable overrides. For example: `$s->usePlugins = true;`
`cookieDomainPeriods`	No	Specifies the number of periods in the cookie domain. For example, www.mysite.com has two periods. www.mysite.com.jp has three periods. Measurement Library uses this variable for visitor ID management. By default, this variable is 2. For example: `$s->cookieDomainPeriods = 3;`
`cookieLifetime`	No	Specifies the cookie’s lifetime, in seconds. If this variable is not set, SiteCatalyst uses default cookie expiration settings. Measurement Library uses this variable for visitor ID management. By default, this variable is 5 years. For example: `$s->cookieLifetime = 86400;` (one day in seconds) or `$s->cookieLifetime = 'SESSION';`
`mobile`	No	Specifies the order in which cookies and Subscriber IDs are used to identify visitors. When `mobile = false`, Adobe uses cookies first. By default, this variable is `false`. When enabled, Measurement Library automatically shortens the Adobe image request. You cannot override this variable using variable overrides. For example: `$s->mobile = false;`

VARIABLE	REQUIRED	DESCRIPTION
botDetection	No	When enabled, Measurement Library detects if the HTTP User-Agent field is a bot, and does not send bot-generated data to Adobe. By default, this variable is false. You cannot override this variable using variable overrides. However, you can override the getBots() method in the OmnitureMeasurement class to supply your own bot detection list. For example: $s->botDetection = true;
imageDimensions	No	Specifies the width and height of the image request, in pixels. By default, this variable is 5x5 when mobile = true, and 1x1 otherwise. You cannot override this variable using variable overrides. For example: $s->imageDimensions = '5x5';
sendFromServer	No	When enabled, Measurement Library makes a direct HTTP request from the server instead of rendering an image tag to the Web browser. For more information, see "track()" on page 6. By default, this variable is false. You cannot override this variable using variable overrides. For example: $s->sendFromServer = true;

Track Config Variables

The following variables are typically set before calling one of the track methods. For more information about these tracking variables (except where noted) see the SiteCatalyst Implementation Guide., which is available in the Online Marketing Suite help system.

pageName
pageURL
referrer
purchaseID
transactionID
channel
timestamp*

server
pageType
dynamicVariablePrefix
variableProvider
campaign
state

zip
events
products
prop (1-50)
eVar (1-50)
hier (1-5)
list (1-3)

* The following variable is specific to Measurement Library:

VARIABLE	DESCRIPTION

timestamp	Specifies a historical timestamp to associate with a hit when recorded by Adobe data collection servers.
	The timestamp variable uses UNIX timestamp format.
	For example: $s->timestamp = time()-86400; //yesterday
	NOTE: The timestamp variable is required for, and should be used only with, timestamp-enabled report suites in order to properly record hits.

Methods

The following tracking methods provide functionality to configure tracking data and send it to Adobe data collection servers.

NOTE: When using variable overrides, use the variable name as the key in a key-value pair. Also, use only strings as values in the key-value pairs.

METHOD	DESCRIPTION
`track()`	When `sendFromServer = false`, `track` renders an image tag to the Web browser. When `sendFromServer = true`, `track` makes a direct HTTP request to Adobe data collection servers. In either case, `track` sends any Track Config Variables that have values.
`track($variableOverrides)`	Same as track, except you can pass in a list of key-value pairs that indicates temporary variable overrides for that track call.
`trackLink($linkURL, $linkType, $linkName)`	When `sendFromServer = false`, `trackLink` renders an image tag with custom, download or exit link data to the Web browser. When `sendFromServer = true`, `trackLink` makes a direct HTTP request to Adobe data collection servers. Use `trackLink` to track all activity that should not capture a page view. `trackLink` has the following parameters: **url**: Identifies the clicked URL. If no URL is specified, the name is used. Use this only when linking to a Web page from your Web site. Otherwise, pass in `null` for this parameter. **type**: Identifies the link report that will display the URL or name. Supported values include: • "o" (Custom Links) • "d" (File Downloads) • "e" (Exit Links) **name**: The name that appears in the link report. If no name is specified, the report uses the URL. **NOTE:** To collect data, you must specify either the `name` or `url` parameter. When not using one of these parameters, pass in null as the value.
`trackLink($linkURL, $linkType, $linkName, $variableOverrides)`	Same as trackLink, except you can pass in a list of key-value pairs that indicates temporary variable overrides for that trackLink call.

`clearVars()`	Clears the following track config variables on the object:
	`channel`
	`events`
	`purchaseID`
	`transactionID`
	`products`
	`state`
	`zip`
	`campaign`
	`prop (1-50)`
	`eVar (1-50)`
	`hier (1-5)`
	`list (1-3)`

`manageVisitorID($persistVisitorIDInLinks = false, $redirectOnGeneratedVisitorID = false, $additionalRedirectVars = array())`	Allows Measurement Library to manage visitor identification instead of Adobe data collection servers. Call `manageVisitorID` before any output is rendered to the page. Measurement Library does the following to manage visitor identification: • Obtains a visitor ID by looking for the mobile subscriber ID (if `mobile = true`). Otherwise, Measurement Library generates an ID. A mobile subscriber ID can only be obtained if PHP is installed as an Apache module. • Sets the visitor ID in a cookie (using `cookielifetime` and `cookiedomainperiods`). • Persists the visitor ID in all links and forms on the page by setting `persistVisitorIDInLinks = true` (this parameter's default value is `false`). Close all form tags, and do not use '>' in href attributes. Setting `persistVisitorIDInLinks = true` prevents any content from being flushed to the page before the entire page has been rendered. • Will redirect the page to itself with the visitor ID in the query string if a visitor ID has been generated by setting `redirectOnGeneratedVisitorID = true`. This parameter's default value is `false`. Since the redirect is an HTTP GET request, do not set this parameter on an HTTP POST. • Additional query string key-value pairs may be added to the redirect URL using the `additionalRedirectVars` parameter. This parameter's default value is an empty array. **NOTE:**The manageVisitorID method utilizes both a query parameter and cookie named "`s_vid`". You should avoid using a query parameter or cookie of this name for any other purpose.

TIP: Page view tracking (track) and link tracking (trackLink) sends all variables that have values (non-null, non-empty). You should reset or empty all variables, as needed, before calling track or trackLink.

Plug-In Architecture (Optional)

Measurement Library supports a plug-in architecture that lets you build custom plug-ins that extend Measurement Library (PHP) functionality. Use this functionality to modify Measurement Library (PHP) object to manipulate data before sending it to Adobe data collection servers.

Measurement Library leverages the standard PHP callback mechanism to do this. The following example demonstrates how this is done:

```
class delegate {
   function doPlugins($other) {
      //custom code
      $other->prop6 = 'plugin';
   }
}

$s = new OmnitureMeasurement();
$s->account = 'myreportsuiteid';
$s->doPlugins = array('delegate', 'doPlugins'); //PHP callback
$s->usePlugins = true;
$s->pageName = 'Main View';
$s->track();
```

Implementation Examples

The following examples demonstrate some common use cases for Measurement Library (PHP):

- (Instance instantiation) Include Measurement Library
- (Instance instantiation) Instantiate instance and setup account configuration variables
- (Instance instantiation) Instantiate instance and setup account configuration variables
- (track) Send a Page View Only
- (track) Send a Page View with Channel Information
- (track) Send a Page View with Channel and Custom Insight (prop) Information
- (track) Send Page View data using variable overrides
- (trackLink) Send a Custom Link
- (trackLink) Send a Custom Link with Channel Information
- (trackLink) Send a Download Link with Channel and Custom Insight (prop) Information
- (trackLink) Send an Exit Link with no variables

(Instance instantiation) Include Measurement Library

```
require_once 'OmnitureMeasurement.class.php';
```

(Instance instantiation) Instantiate instance and setup account configuration variables

```
//Instantiate instance
$s = new OmnitureMeasurement();
//Setup application config variables
$s->account = 'myaccount';
$s->mobile = true;
```

The following instantiation example uses the singleton interface.

```
//Instantiate instance
$s = OmnitureMeasurement::getInstance();
//Setup application config variables
$s->account = 'myaccount';
$s->mobile = true;
```

NOTE: Do not use the singleton interface when using multiple instances of `OmnitureMeasurement` in a single request.

(track) Send a Page View Only

```
//Set Variables
$s->pageName = 'Some Page Name';
$s->channel = null;//clears any previously set value for channel
$s->prop1 = null;//clears any previously set value for prop1
$s->track();
```

(track) Send a Page View with Channel Information

```
//Set Variables
$s->pageName = 'Some Page Name';
$s->channel = 'Some Site Section';
$s->track();
```

(track) Send a Page View with Channel and Custom Insight (prop) Information

```
//Set Variables
$s->pageName = 'Some Page Name';
$s->channel = 'Some Site Section';
$s->prop1 = 'Some Value';
$s->track();
```

(track) Send Page View data using variable overrides

Specifies a temporary variable value used only for track call.

```
//Create array of variable overrides, and pass into track call
$trackData = array('pageName' => 'Some page name',
               'prop1' => 'Some prop val');
$s->track($trackData);
```

(trackLink) Send a Custom Link

```
$s->trackLink(null, 'o', 'Some Action Name');
//you can pass in null for linkURL it if doesn't apply
```

(trackLink) Send a Custom Link with Channel Information

```
//Set Variables
$s->channel = 'Some Site Section';
$s->trackLink(null, 'o', 'Some Action Name');
//you can pass in null for linkURL it if doesn't apply
```

(trackLink) Send a Download Link with Channel and Custom Insight (prop) Information

```
//Set Variables
$s->channel = 'Some Site Section';
$s->prop1 = 'Snoop';
```

```
$s->trackLink('http://www.somedownloadURL.com', 'd', 'Some Action Name');
```

(trackLink) Send an Exit Link with no variables

```
//Set Variables
$s->channel = null;//clears any previously set value for channel
$s->prop1 = null;//clears any previously set value for prop1
$s->trackLink('http://www.someexitdomain.com', 'e', 'Some Action Name');
```

1.4 Supported SiteCatalyst Reports

The following SiteCatalyst reports include data submitted by Measurement Library (PHP). For more information about these reports, see the SiteCatalyst User Guide.

Site Metrics Report Support

- Page Views
- Hourly Unique Visitors
- Daily Unique Visitors
- Monthly Unique Visitors
- Yearly Unique Visitors
- Visits
- File Downloads

Finding Methods Report Support

NOTE: These reports are available only if the developer has access to the referrer information, which might not be feasible in some circumstances.

- Referring Domains
- Referrers
- Return Frequency
- Daily Return Visits
- Return Visits
- Visit Number

Visitor Profile Report Support

- Domains
- Full Domains
- Top Level Domain
- Languages (available only if PHP is installed as an Apache module)
- Visitor Detail
- Last 100 Visitors

GeoSegmentation Report Support

- Country

- State
- City

Mobile Technology Report Support

- Devices
- Manufacturer
- Screen Size
- Screen Height
- Screen Width
- Cookie Support
- Image Support
- Color Depth
- Audio Support
- Video Support

Segmentation Report Support

- Most Popular Pages
- Most Popular Channels
- Most Popular Servers
- Key Visitors
- Pages Viewed by Key Visitors
- Custom Insight
- Custom Links
- Custom Insight Properties 1-50

Pages Report Support

- Page Summary
- Most Popular Pages
- Clicks to Page
- Time Spent on Page
- Pages Not Found

Entries & Exits

- Entry Pages
- Exit Pages
- Exit Links

Complete Paths Report Support

- Full Paths
- Longest Paths

- Path Length
- Time Spent per Visit
- Single-page Visits

Advanced Analysis Report Support

- Previous Page
- Next Page
- Previous Page Flow
- Next Page Flow
- PathFinder
- Fall-out

Purchases

- Conversions & Averages
- Revenue
- Orders
- Units

Shopping Cart

- Conversions & Averages
- Carts
- Cart Views
- Cart Additions
- Cart Removals
- Checkouts

Products Report Support

- Conversions & Averages
- Products
- Cross-Sell
- Categories

Campaigns

- Conversions & Averages
- Campaigns

Customer Loyalty

- Customer Loyalty

Sales Cycle

- Days Before First Purchase

- Days Since Last Purchase
- Visit Number
- Daily Unique Customers
- Monthly Unique Customers
- Yearly Unique Customers

Finding Methods

- Referring Domains
- Original Referring Domains

Visitor Profile

- Countries
- Languages
- Time Zones
- States
- ZIP/Postal Codes
- Domains

Site Path

- Entry Pages
- Original Entry Pages
- Pages per Visit
- Time Spent on Site
- Content Groups

Custom Variables

- Custom E-Commerce Variables 1-50