



WEBTRENDS  
Marketing Lab™ 2

# WebTrends Analytics Software Implementation and Maintenance Guide

July 2009 Edition | © 2009 WebTrends Inc.



## Disclaimer

This document and the software described in this document are furnished under and are subject to the terms of a license agreement or a non-disclosure agreement.

EXCEPT AS EXPRESSLY SET FORTH IN A LICENSE AGREEMENT, WEBTRENDS INC. PROVIDES THIS DOCUMENT AND THE SOFTWARE DESCRIBED IN THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

You agree that you shall not loan, sell, or otherwise transfer this document or the software described in this document.

Except as expressly set forth in a license agreement, you agree that you shall not reproduce, store in a retrieval system, or transmit in any form or by any means, electronic, mechanical, or otherwise, all or any part of this document or the software described in this document. Some companies, names, and data in this document are used for illustration purposes and do not represent real companies, individuals, or data.

This document may include technical inaccuracies or typographical errors. WebTrends Inc. may make improvements in or changes to the software described in this document at any time.

© 1996-2009 WebTrends Inc. All rights reserved.

U.S. Government Restricted Rights: The software is "commercial software." If the software and documentation are being acquired by or on behalf of the U.S. Government or by a U.S. Government prime contractor or subcontractor (at any tier), in accordance with 48 C.F.R. 227.7202-4 (for Department of Defense (DOD) acquisitions) and 48 C.F.R. 2.101 and 12.212 (for non-DOD acquisitions), the government's rights in the software and documentation, including its rights to use, modify, reproduce, release, perform, display or disclose the software or documentation, will be subject in all respects to the commercial license rights and restrictions provided in the license agreement.

## Trademarks

WebTrends Visitor Intelligence™, WebTrends Visitor Intelligence logo™, WebTrends Score™, WebTrends Score logo™, WebTrends Marketing Lab™, WebTrends ml2™, WebTrends ml2 logo™, WebTrends Analytics™, WebTrends Analytics logo™, WebTrends Dynamic Search™, WebTrends Dynamic Search logo™, WebTrends Marketing Warehouse™, WebTrends Explore™, WebPosition®, True Optimization. Real Results.™, True Optimization™, WebTrends Open Exchange™, WebTrends® and the WebTrends logo™ are trademarks or registered trademarks of WebTrends Inc. or its subsidiaries in the United States and other jurisdictions. All other company and product names may be trademarks or registered trademarks of their respective companies.

© 1996-2009 WebTrends Inc. All rights reserved.



## Contact Information

Sales and General	Support and Services	Online Resources
WebTrends, Inc. 851 SW 6th Ave. Suite 1600 Portland OR 97204 Phone: 1-503-294-7025 Fax: 1-503-294-7130 US Toll Free: 1-877-WEBTRENDS (1-877-932-8736) <b>Email:</b> <a href="mailto:feedback@webtrends.com">feedback@webtrends.com</a>	<b>Direct Technical Support:</b> North America and Caribbean: 1-503-223-3023 Central and South America: 1-503-223-3023 Asia Pacific, Australia, New Zealand: 1-503-223-3023 Europe, Middle East, Africa: +44 (0) 1784-463-555	<b>WebTrends Home Page:</b> <a href="http://www.webtrends.com">http://www.webtrends.com</a> <b>WebTrends Support Center:</b> <a href="http://www.webtrends.com/support">http://www.webtrends.com/support</a> <b>WebTrends Professional Services (Consulting and Training):</b> <a href="http://www.webtrends.com/services">http://www.webtrends.com/services</a> <b>Online Customer Center:</b> <a href="http://www.webtrends.com/resources">http://www.webtrends.com/resources</a>



---

## About This Book and the Library

This guide is intended to help the WebTrends administrator install and configure WebTrends Analytics, GeoTrends, and SmartSource Data Collector. WebTrends Analytics software and WebTrends Analytics On Demand both offer a variety of powerful analysis and reporting options. The features and reports available to you are dependent on how your company is licensed to use WebTrends. If you have questions about the WebTrends features you are licensed to use, please see your WebTrends administrator.

The Implementation and Maintenance Guide provides in-depth configuration and reference information for WebTrends administrators. It includes:

- Detailed insight into WebTrends Analytics features such as Custom Reporting and translation files
- Extensive coverage of the SmartSource Data Collector
- A complete reference to WebTrends query parameters
- Information about security practices including a discussion of cookie tracking methods and instructions for using WebTrends with Secure Sockets Layer (SSL)
- Installation and configuration instructions for WebTrends conduits and plug-ins
- Instructions for backing up a WebTrends installation and restoring it in case of a failure

### Intended Audience

This book provides information for administrators who are responsible for installing and configuring WebTrends Analytics software or setting up WebTrends Analytics On Demand.

### Other Information in the Library

The library provides the following information resources:

#### Help

Provides context-sensitive information and step-by-step guidance for common tasks, as well as definitions for each field on each window.

#### WebTrends Administration User's Guide

This guide provides complete information for using WebTrends Administration to set up and customize core Marketing Lab operations such as data collection, analysis, report content and style, and visitor session tracking. It includes conceptual and procedural information about features such as custom reports, data filtering, scenario analysis, and Express Analysis; assistance with common administrative concerns such as job scheduling and table limiting; and reference information such as the WebTrends Query Parameter Reference.



## WebTrends Analytics Reports User's Guide

This guide provides users who primarily use WebTrends Analytics Reports with the information they need to navigate, customize, save, and export reports and report data.



### Note

Users who only have View Reports permissions automatically use WebTrends Analytics Reports instead of WebTrends Administration. While they can view the reports, they may not have access to any of the other controls. We recommend distributing the *WebTrends Analytics Reports User's Guide* to these users as a introduction to navigating WebTrends reports and report data.

---

## WebTrends Marketing Warehouse Software User's Guide

This guide includes information about using WebTrends Marketing Warehouse for ad hoc data analysis, using WebTrends Explore to analyze web business events by segment, and using WebTrends Score to identify qualified users based on their web site actions. WebTrends administrators can also find information about installing, implementing and using WebTrends Marketing Warehouse.

## WebTrends Marketing Warehouse On Demand User's Guide

This guide includes information about using WebTrends Visitor Intelligence for ad hoc data analysis, using WebTrends Explore to analyze web business events by segment, and using WebTrends Score to identify qualified users based on their web site actions. It also provides a detailed reference to Visitor Intelligence report data. WebTrends administrators can also find information about implementing and using WebTrends Marketing Warehouse.

## WebTrends Analytics Software Implementation and Maintenance Guide

A step-by-step guide for administrators who are responsible for installing, setting up and maintaining WebTrends Analytics Software. It includes information about licensing, JavaScript tagging, profile setup, security, cookie implementation, performance tuning, and system backups. It also includes the WebTrends Query Parameter Reference.

## WebTrends Analytics On Demand Implementation Guide

A step-by-step guide for administrators who are responsible for implementing WebTrends Analytics On Demand. It includes information about licensing, JavaScript tagging, and profile setup. It also includes the WebTrends Query Parameter Reference.

## WebTrends SmartSource Data Collector User's Guide

This guide provides instructions for installing, configuring, and maintaining WebTrends SmartSource Data Collector, including information about client- and server-side JavaScript tags and cookie tracking.

## WebTrends SmartView User's Guide

A guide to installing and using SmartView and configuring WebTrends to work effectively with SmartView reporting.

## WebTrends SmartReports User's Guide

A guide to using WebTrends SmartReports with WebTrends Analytics reporting for powerful data integration and analysis in the Microsoft Excel environment.



### WebTrends Marketing Warehouse Schema Reference

Provides an overview of the Marketing Warehouse databases for experienced database administrators. This guide helps you understand the data in the Marketing Warehouse, giving you the foundation you need to use the data productively. It provides instructions for populating the Marketing Warehouse databases using WebTrends Administration and for viewing the data once it is available. It also describes how the databases are constructed and how that affects the function of the different types of data.

### WebTrends Marketing Lab Programmer's Reference

This guide provides conceptual, procedural, and referential information that allows experienced programmers to customize WebTrends Marketing Lab data collection and reporting. It provides instructions for using the WebTrends ODBC Driver to query both the Marketing Warehouse and the WebTrends Analytics Report databases. It also includes documentation for the Active X, C, and Post Plug-Ins that can communicate with WebTrends Analytics.

### WebTrends Visitor 360 Web Services Developer's Guide

This guide for WebTrends On Demand customers provides development information for using Visitor 360 web services to populate a Marketing Warehouse with off-site data, query and retrieve Analytics Reports data, and execute named queries against Marketing Warehouse data.

### WebTrends Guide to Web Analytics

This guide provides an introductory conceptual overview of web analytics, supplemented with examples, graphics, and practical worksheets to help you understand WebTrends architecture and create a strategy for customizing WebTrends Analytics for your key business metrics. Topics covered in this guide include collecting web activity data, understanding visitor behavior, filtering and analyzing your data, measuring acquisition, conversion, and retention, and integrating web analytics data with other business data.

## Providing Feedback

Your comments are very important to us. Please take the time to let us know about your WebTrends experience by doing one of the following:

- Click **Customer Center** in the upper right corner of the WebTrends Marketing Lab banner. Then click **Contact Us** and click **Submit Product Feedback** in the right pane.
- From WebTrends Analytics Reports, click **Help > Feedback** from the upper right corner of the report.

The Feedback page of the WebTrends web site opens in a new browser window. You can use it to report a bug, request a feature, or give general feedback about your user experience.

## The Customer Center

The WebTrends Customer Center brings together a wide variety of materials to help you learn to use WebTrends Analytics more effectively, including white papers, interactive training modules, *How Do I?* Guides, and business case studies. To access the Customer Center, click **Customer Center** in the upper right corner of the WebTrends Marketing Lab banner.



## Conventions

The library uses consistent conventions to help you identify items throughout the documentation. The following table summarizes these conventions.

Convention	Use
<b>Bold</b>	<ul style="list-style-type: none"><li>• Window and menu items</li><li>• Technical terms, when introduced</li></ul>
<i>Italics</i>	<ul style="list-style-type: none"><li>• Book and CD-ROM titles</li><li>• Variable names and values</li><li>• Emphasized words</li></ul>
Luci da Consol e	<ul style="list-style-type: none"><li>• File and folder names</li><li>• Commands and code examples</li><li>• Text you must type</li><li>• Text (output) displayed in the command-line interface</li></ul>







---

# Table of Contents

## Chapter 1

<b>Licensing</b>	<b>1</b>
How WebTrends Is Licensed .....	1
Understanding Server Call Licensing .....	1
Understanding Profile Licensing .....	2
Checking the License Status .....	2
Checking Your Server Call and Event Use .....	2
Document Revision History .....	3

## Chapter 2

<b>Preparing to Install WebTrends Analytics</b>	<b>5</b>
Pre-Installation Checklist .....	5
Deciding Which Installation Method to Use .....	6
Planning for Log File Storage .....	6
Setting Up Shares for WebTrends Data .....	6
Configuring the Domain User Security Settings .....	7
Deciding Which Database to Use for the WebTrends System Database .....	8
Preparing for a Microsoft SQL Server Express Installation .....	9
Preparing for a Microsoft SQL Server Standard or Enterprise Edition Installation .....	10
Enabling TCP/IP Protocol for Microsoft SQL Server Standard or Enterprise Edition .....	11
Configuring WebTrends Services User Accounts .....	12
Creating Domain Users .....	13
Configuring the Domain User Security Settings .....	13
Configuring the Installer User Account .....	14



Optimizing Your Database for a WebTrends Installation .....	14
Design Recommendations for SQL Server Configuration .....	15
Optimizing Your Disk Layout to Improve Performance .....	17
Using a Local Storage Disks Disk Layout .....	17
Using a SAN Disk Layout .....	18
Deciding Which Ports to Use .....	19
Document Revision History .....	20

## Chapter 3

<b>Installing WebTrends Analytics on One Computer</b> .....	<b>21</b>
Understanding WebTrends Components .....	21
Upgrade Considerations .....	22
Installing on One Computer .....	22
Logging In to WebTrends Analytics .....	25
What's Next? .....	25
Document Revision History .....	26

## Chapter 4

<b>Distributed Installation of WebTrends Analytics Software</b> .....	<b>27</b>
Who Should Use a Distributed Installation? .....	27
Understanding WebTrends Components .....	28
Upgrade Considerations .....	28
Updating Storage Locations .....	29
Preparing to Upgrade an Existing Installation .....	29
Performing a Distributed Installation .....	30
Logging In to WebTrends Analytics .....	33
What's Next? .....	34
Managing Your Distributed Installation .....	34
Viewing Available Resources .....	34
Using Groups to Distribute Work .....	34
Monitoring Hosts in Your Installation .....	34
Document Revision History .....	35



## Chapter 5

### Unattended Installation of WebTrends Software

37

Installation Guidelines .....	37
Preparing for an Unattended Installation .....	37
Prerequisites for WebTrends Components .....	38
MSI Files to Install for WebTrends Components .....	40
Installing the Prerequisites .....	40
Performing an Unattended Installation of WebTrends .....	41
Using Command Line Installation Syntax .....	42
Using MSI Public Property Parameters .....	43
Which Parameters Are Required When You Install MSI Files? .....	43
Identifying MSI Properties for Visitor Intelligence .....	46
Identifying MSI Properties and Their Command Line Parameters .....	47
Installing Individual Subcomponents .....	47
Applying ADDLOCAL Rules .....	48
Specifying UI Server Subcomponents (WebTrends Main Application.msi) .....	48
Specifying Analysis Engine Subcomponents (WebTrends Analysis.msi) .....	49
Specifying Marketing Warehouse Subcomponents (WebTrends Marketing Warehouse.msi)	
49	
Installing Subcomponents After You Install the Component .....	50
Examples for Installing WebTrends Components .....	50
System Database Installation Example .....	50
Common/Scheduler Agent Installation Example .....	50
WebTrends System Repository Installation Example .....	50
Analysis Engine Installation Example .....	51
Marketing Warehouse Installation Example .....	51
GeoTrends Installation Example .....	51
UI Server/Main Application Installation Example .....	51
Management Application Installation Example .....	51
Visitor Intelligence Installation Example .....	52
UI Server Modules Installation Example .....	52
Performing an Unattended Uninstallation .....	52



## Chapter 6

<b>Optimizing Your Database Environment</b>	<b>55</b>
Understanding Database Partitions and File Groups .....	55
Database Maintenance .....	57
Recommendations for Bulk Extracts .....	57
Direct Access .....	58
Extract, Transform, Load (ETL) .....	58
Load Process Tuning .....	58
Tuning the Marketing Warehouse Loader .....	58
Disabling Sampling .....	59
Instance Replication and Cloning .....	59
Optimizing Marketing Warehouse with SQL Server Configuration .....	59
Recommended Settings for SQL Server Configuration .....	60
Processor Performance Settings .....	60
Memory Utilization Settings .....	60

## Chapter 7

<b>Client-Side JavaScript Integration</b>	<b>61</b>
Interactions Between the Client Browser and SDC .....	61
Customizing Your META Tags .....	63
General META Tag Information .....	63
META Tag Descriptions .....	64
Tracking Content Groups .....	64
Tracking Servers .....	65
Tracking Marketing Campaigns .....	65
Tracking Profile/Subprofile Generation .....	66
Tracking Revenue .....	67
Tracking Shopping Cart Activity .....	68
Tracking On-Site Advertising .....	68
Tracking Advertising Clicks .....	69
Tracking Customized URLs .....	69
Tracking Page Titles .....	70
Inserting the Tags .....	71
Copying the Tag to Each Page .....	71
Using Server-Side Include Files .....	71
Using Footer Templates .....	71
Tagging Best Practices .....	72
Document Revision History .....	72



## Chapter 8

### Creating WebTrends Analytics Profiles **73**

How Profiles Work .....	73
Profile Creation .....	73
Using the Profile Wizard .....	74
Advanced Profile Settings .....	74
Document Revision History .....	76

## Chapter 9

### Setting Up WebTrends Users and Roles **77**

Understanding WebTrends User Rights .....	77
Understanding Action Rights .....	77
Understanding Profile and Template Rights .....	77
Understanding WebTrends User Roles .....	78
Configuring User Roles .....	78
Applying Preconfigured Roles to Existing Users .....	79
Adding Users .....	79
About View Only Permissions .....	80
Predefined Role Settings .....	81
Document Revision History .....	82

## Chapter 10

### Tracking Visitor Sessions **83**

What is a Cookie? .....	83
Why Web Browsers Reject Cookies .....	83
Negative Impact of Third-Party Cookie Rejection .....	84
Solving Rejection with First-Party Cookies .....	85
How WebTrends Marketing Lab Uses Cookies .....	85
Methods for Generating First-Party Cookies .....	85
Using the WebTrends JavaScript Tag .....	86
Using Your Web Server to Generate Cookies .....	86
Using the WebTrends Cookie Plug-in .....	87
Configuring WebTrends for First-Party Cookie Tracking .....	87
Using the JavaScript Tag to Track Cookies .....	87
Implementing the JavaScript Tag .....	87
Specifying the First-Party Cookie Data Source .....	88
Specifying Session Tracking for First-Party Cookies .....	88
Configuring Domains .....	89



Tracking Visitors Across Domains .....	89
Converting Third-Party Cookies to First-Party Cookies .....	91
Customizing Tag-Generated First-Party Cookies .....	91
Using First-Party Cookies Without SDC .....	92
Implementing the Opt-Out Cookie .....	93
How the Opt-Out Cookie Works .....	94
Writing an Opt-Out Policy .....	94
Creating an Opt-Out Mechanism .....	95
Implementing the JavaScript Tag .....	95
Disabling Cookies .....	95
Disabling First-Party Cookies .....	95
Disabling Third-Party Cookies .....	96
Document Revision History .....	97

## Chapter 11

### Securing Your Implementation 99

WebTrends and Secure Sockets Layer (SSL) .....	99
Enabling SSL for WebTrends Analytics .....	99
Document Revision History .....	100

## Chapter 12

### Optimizing Your WebTrends Environment 101

Physical Data Storage .....	101
Log File Rotation .....	101
Archiving .....	103
Table Limiting .....	104
Scheduling Reports and Storing Reports .....	105
Maintaining and Storing Reports .....	106
Performance Issues .....	107
FTP Caching .....	107
Internet Resolution .....	107
HTML Page Title Lookups .....	108
Simultaneous Analysis .....	108
Performance Tuning for Your WebTrends Database .....	109
Improving Common Performance Problem Areas .....	109
Determining Bottlenecks .....	109
Document Revision History .....	110



## Chapter 13

### Customizing JavaScript Tags 111

Understanding the Basic JavaScript Tag .....	111
Inline HTML (webtrends.html) .....	111
External Script File .....	111
Embedded JavaScript .....	112
CDATA .....	112
Initialization .....	112
Collection .....	112
Script Disabled .....	112
External JavaScript File (webtrends.js) .....	113
Construction .....	113
Properties .....	113
Methods .....	116
Global Functions .....	121
URL Encoding .....	121
Image Array .....	121
Query Parameter Storage Objects .....	121
Adding Customized Information .....	123
SDC-Specific Query Parameters .....	123
WebTrends Query Parameters .....	124
User-Defined Query Parameters .....	124
Debugging Your Customizations .....	125
Using dcsDebug .....	125
Displaying the URL .....	125
Displaying the Query Parameter Storage Objects .....	126
Document Revision History .....	127

## Chapter 14

### Backing Up and Restoring WebTrends Data 129

How WebTrends Stores Data .....	129
Built-In Redundancies .....	130
WebTrends Data Repositories .....	131
Backing Up WebTrends Analytics Analysis Data .....	131
Viewing Available Backups .....	132
Backing Up a Microsoft SQL Server 2005 Express Edition System Database .....	132
Backing Up a MySQL System Database .....	133
Backing Up WebTrends Marketing Warehouse Data .....	134
Restoring WebTrends Analytics Data .....	134



Restoring WebTrends Marketing Warehouse Data .....	135
Recovering from Data Center Failure .....	135
Document Revision History .....	137

## Chapter 15

### **Moving Your WebTrends Analytics Installation to a New Location** **139**

Limitations .....	139
Preparing to Move Your WebTrends Analytics Installation .....	139
Moving Your WebTrends Analytics Installation .....	141
Updating the SQL Server Databases .....	141
Starting the New Installation .....	142
Document Revision History .....	143

## Chapter 16

### **WebTrends Query Parameter Reference** **145**

How WebTrends Query Parameters Work .....	145
Using WebTrends Query Parameters .....	145
Products Using WebTrends Query Parameters .....	146
Query Parameter Syntax .....	146
Name Syntax .....	146
Value Syntax .....	147
Complete Syntax .....	148
Types of Query Parameters .....	149
Auto-Configuration Parameters .....	149
Content Group Parameters .....	150
Marketing Campaign Parameter .....	150
Advertising View Parameter .....	150
Advertising Click Parameter .....	151
Server Parameter .....	151
Scenario Analysis Parameters .....	151
Title Parameter .....	155
Split Parameter .....	155
Custom Report Parameters .....	155
Search Engine Type Parameter .....	155
Web Client Parameters .....	156
Products Parameters .....	159
Transaction Parameters .....	160
Invoice Parameters .....	161



Campaign Parameter .....	162
Campaign Event Parameter .....	163
Segment Parameter .....	163
Page of Interest Parameter .....	163
On-Site Search Parameters .....	164
Registered Visitor Parameter .....	164
Content Parameters .....	164
SmartView Parameters .....	166
Stored Visitor Parameter .....	167
Visitor History Parameters .....	167
Most Recent Campaign Parameters .....	167
Most Recent Campaign Visitors .....	169
Visitor “Initial” Parameters .....	169
Elapsed Time Parameters .....	170
Historical Counts Parameter .....	171
Historical Transactions/Purchases Parameters .....	172
Search Engine Parameters .....	176
Visitor Tracking Parameters .....	178
Visitor Segmentation Parameters .....	180
SDC-Generated Visitor Parameters .....	180
Visitor Tracking Parameters .....	180
Cookie Detection Parameters .....	183
URL Truncation Parameter .....	184
HTTP Headers .....	184
JavaScript Tag Version .....	184
Site ID .....	184
Event Tracking .....	185
Parent DIV/Table ID .....	186
Click-Based Tracking .....	186
DCSID .....	186
SDC-Parameter Override Parameters .....	187
Conversion Plug-In Parameters .....	189
Document Revision History .....	189
<b>Index.....</b>	<b>191</b>





---

## Chapter 1

# Licensing

This chapter provides information about how WebTrends Marketing Lab is licensed and tells you how to check the status of your license.

## How WebTrends Is Licensed

Your license determines:

- How long you can use WebTrends.
- Which core Webtrends products you can install and use. Core products include WebTrends Analytics, WebTrends Marketing Warehouse, Score, and Ad Director.
- Which *report packs* you can use. Report packs are collections of reports that are grouped according to purpose. For example, the Commerce Report Pack includes reports about products and purchase-related visitor segments. Report packs allow you to report on the type of information you need and give you the flexibility to add report packs as your needs grow.
- Which product add-ons, such as Advanced SmartView and Custom Reporting, you can use.
- The number of server calls that WebTrends On Demand collects during your licensing period. If you reach the server call limit, you can contact your Account Manager to purchase additional server calls. For more information, see [“Understanding Server Call Licensing” on page 1](#).
- How many WebTrends Analytics profiles you can create. For more information about licensing WebTrends Analytics parent-child profiles, see [“Understanding Profile Licensing” on page 2](#). Your license does not limit the number of Marketing Warehouse profiles.
- The number of WebTrends Analytics reports you can export. If you reach the limit for report exports, contact your Account Manager to purchase additional exports.
- How many events you can collect and store in the Marketing Warehouse. If you reach the limit for events collected or stored, contact your Account Manager to purchase additional events.

## Understanding Server Call Licensing

WebTrends Analytics On Demand is licensed according to server calls analyzed per licensing period. For example, during a 12 month period, your WebTrends Analytics activation may be licensed for up to 20 million server calls.

In a licensing context, a *server call* is defined as a file with a web page extension (such htm, html, asp) that is requested by a web site visitor to one of the domains being analyzed. However, for WebTrends On Demand, all hits (or files) regardless of the extension are considered server calls.



If WebTrends Analytics analyzes data that is processed at a later date by a different profile or by the same profile, the server call is not counted again. Server calls are counted before applying any data filters. Although multiple frames on one web page are counted as separate (multiple) server calls, you can avoid this problem by excluding the WebTrends JavaScript tag from the secondary frames.

## Understanding Profile Licensing

Your license determines the number of profiles you can create. You can distribute the total number of profiles among your *child accounts*. Child accounts are sub-accounts within your main license.

Child accounts are counted differently toward your license total depending on whether you use Basic Analysis or Full-Featured Analysis to create Parent-Child profiles.

A Basic Analysis Parent-Child profile with up to 10 child profiles counts as one profile. Each group of 10 additional child profiles counts as one additional licensed profile. For example, a Parent-Child profile using Basic Analysis with between 11 and 20 children counts as 2 profiles. A Parent-Child profile using Basic Analysis with between 21 and 30 children counts as 3 licensed profiles, and so on.

For a Full-Featured Analysis Parent-Child profile, each child profile counts as one profile. For example, a Parent-Child profile with 3 children counts as 3 licensed profiles.

## Checking the License Status

The **Edit Account** dialog box in WebTrends Accounts allows you to view the status of your profiles, custom reports, and report export limits.

**To check your license status:**

1. From WebTrends Analytics Accounts, select **Account > Edit Account** in the left pane.
2. Click **Settings**.

## Checking Your Server Call and Event Use

You can view the number of server calls and events collected in WebTrends Analytics Accounts.

**To view your server calls balance:**

In the left pane of WebTrends Accounts, select **Account Usage > Summary**. Use this dialog to view:

- Server calls collected for the current month to date
- Server calls collected for the calendar year to date
- Server calls collected during the life of your account



## Document Revision History

Table 1: Document Revision History contains a summary of changes made to this document beginning with the release of WebTrends Analytics, version 8.7.

**Table 1: Document Revision History**

Software Version	Date of Last Update	Summary of Changes
v8.7	March, 2009	Added Document Revision History section
v8.7d	July, 2009	Added footer link to Documentation Center.





---

## Chapter 2

# Preparing to Install WebTrends Analytics

This chapter discusses how to prepare for a WebTrends Analytics installation. Review this chapter before installing WebTrends Analytics. In this chapter:

- “Deciding Which Installation Method to Use” on page 6
- “Planning for Log File Storage” on page 6
- “Setting Up Shares for WebTrends Data” on page 6
- “Configuring the Domain User Security Settings” on page 7
- “Deciding Which Database to Use for the WebTrends System Database” on page 8
- “Configuring WebTrends Services User Accounts” on page 12
- “Deciding Which Ports to Use ” on page 19

For information about preparing to install WebTrends Marketing Warehouse, see *WebTrends Marketing Warehouse User’s Guide*.

## Pre-Installation Checklist

- Collect all valid license keys for the product, add-on licenses, or subscriptions.
- Review the hardware, operating system, and browser requirements to ensure optimum performance of WebTrends Analytics. For more information, see the [Analytics Software System Requirements](#).
- If anti-virus software is installed on your WebTrends-dedicated computers, decide whether you want to disable it. If the anti-virus software is activated when you install WebTrends, the installation program warns you that it may conflict with analysis. For more information, see the [Analytics Software System Requirements](#).
- Determine which installation method is right for you. For more information, see “Deciding Which Installation Method to Use” on page 6.
- Consider how much storage you will need for WebTrends system data. For more information, see “Setting Up Shares for WebTrends Data” on page 6.
- Create shares for WebTrends data. For more information, see “Setting Up Shares for WebTrends Data” on page 6.
- If you plan to use a custom installation and want to be able to schedule reports in Microsoft Word or Excel format, configure a domain user account for WebTrends. For more information, see “Configuring the Domain User Security Settings” on page 7.



## Deciding Which Installation Method to Use

WebTrends supports two types of installations, an installation on one computer and a distributed installation with multiple computers each running the WebTrends components that your organization needs to maximize processing and report delivery. Each method has benefits and special considerations.

### Single Computer

Allows you to choose which optional WebTrends components you want to install on a computer dedicated to WebTrends. For example, you have the option to use your own pre-installed Microsoft SQL Server database as your system database for storing WebTrends data. You can also run WebTrends services as a domain user giving you the ability to schedule reports in Microsoft Word or Excel format, and install the Express Analysis database engine. For more information, see [“Installing WebTrends Analytics on One Computer” on page 21](#).

### Distributed (Multiple Computers)

Allows you to choose which optional components you want to install and where you want to install them. Distributed configurations are used by large organizations and those with high-traffic sites or a large number of report users. For more information, see [“Distributed Installation of WebTrends Analytics Software” on page 27](#).

## Planning for Log File Storage

As a best practice, you should store your log files on a network location that the computers in your WebTrends implementation can access. Typically, this is accomplished by storing log files on a shared drive.

How much storage you need depends largely on your site traffic and how long you plan to store your processed log files. For more information about optimizing log file storage, see [“Optimizing Your WebTrends Environment” on page 101](#).

If you use WebTrends Marketing Warehouse, there are additional storage requirements. For more information, see the *WebTrends Marketing Warehouse User's Guide*.

## Setting Up Shares for WebTrends Data

Before you install WebTrends Analytics components, you need to create directories for WebTrends working directories. Because each path must be accessible by each service and computer in the installation, be sure you share each folder and give the domain users used to run each WebTrends service in your installation full control permission before beginning the installation. Consider dedicating machines for storing your system and configuration data. As a general rule, plan on 5 GB of data per profile each year. You can store each repository on a separate system if necessary.

During installation, you specify the storage locations for the WebTrends data repositories discussed in the previous section. For example, the path to the Analysis Data Repository share for the Main Application computer that has a name of “wtmain” could be \\wtmain\storage\analysis.

You need to set up the following shared directories for your WebTrends Analytics implementation:

### Analysis Data Repository

Stores the data that is processed to generate reports; accessed primarily by the Analysis engines. How much data is stored depends on a number of variables including the amount of web site traffic, table limit settings, the quantity and complexity of your custom reports, and if you use WebTrends Visitor History feature, the amount of visitor history data that you store.



### Report Data Repository

Stores the data used to display reports; is accessed by the UI Server and the Analysis engines. The size of the Report Data repository depends on the number of custom reports, your table limit settings, and the report period and storage profile settings. Sample path: \\wtmain\storage\analysis

### Configuration Repository

Stores the settings WebTrends needs in order to run. These flat files require relatively little storage space for an Analytics installation, but can grow large in the case of a Marketing Warehouse installation.

### Backup Repository

Contains WebTrends configuration data that can be used to restore your WebTrends Analytics installation if necessary. By default, WebTrends maintains seven daily backup archives and four weekly archives for each profile.

### Express Analysis Log File Storage

Contains copies of the SmartSource data files that Express Analysis analyzes for constantly updated reporting.

### Express Analysis Report Data Repository

Stores the data used to display Express Analysis reports; is accessed by the UI Server and the Express Analysis engines.

### Marketing Warehouse Import Directory

Contains any lookup tables and lookup table configuration files you want your Marketing Warehouse to import.



#### Note

WebTrends does not support administrative shares, such as \\systemname\c\$

---

## Configuring the Domain User Security Settings

If you want to be able to schedule reports in Microsoft Word or Excel format, the domain user account must have certain security settings. During installation, you will specify the domain user account for WebTrends services to use. If you choose to split your WebTrends services between multiple domain accounts, this information applies only to the account associated with your WebTrends UI services.

#### To give the domain user account the required security settings:

1. From the Windows Local Security Settings dialog, select **Local Policies > User Rights Assignment**.
2. Select **Act as part of the operating system**, right-click, and select **Properties**.
3. Click **Add User or Group**, type a domain user account, and click **Check Names**. You must specify a specific domain user and not a group. The domain user must be a member of the local administrators group on the computer where WebTrends Analytics is installed.
4. Repeat the previous step for the **Log on as a service policy** and the **Log on locally/Allow log on locally** policy.



## Deciding Which Database to Use for the WebTrends System Database

WebTrends Analytics uses a small database for system and configuration data. The database coordinates the work of all other components. During installation, you can choose to install Microsoft SQL Server 2008 Express Edition or use your own installation of Microsoft SQL Server Express, Standard or Enterprise Edition.

If you choose to use your own Microsoft SQL Server Standard or Enterprise Edition database, or if you choose to install your own copy of Microsoft SQL Server 2008 Express Edition, you must install Microsoft SQL Server before you install the WebTrends System Database component. Depending on how you plan to use WebTrends and the available resources on the System Database server, you may want to dedicate a separate server to the System Database. However, to assure proper performance and reliability, you should always install the System Database to a dedicated SQL instance. If you plan to support additional non-WebTrends applications on separate instances of SQL server on the same server, check the [Analytics Software System Requirements](#) to ensure that your server's available resources comfortably exceed the system requirements for WebTrends.



Refer to the table below for supported database and server configurations. For more information about installation configurations, see the [System Requirements](#)



### Note

The databases shown in the **Analytics and Marketing Warehouse** row of this table are only required if you plan to install the Marketing Warehouse XMDB on the same computer as WebTrends Analytics System Database. If you plan to install Marketing Warehouse databases on a different computer, use the requirements for **Analytics only**.

:

Products to Install	Database	Required Operating System
Analytics only	Microsoft SQL Server 2005 Express, Standard or Enterprise Edition	Windows Server 2003 x86 or Windows Server 2003 x64
	Microsoft SQL Server 2008 Express, Standard or Enterprise Edition	Windows Server 2008 x64
Analytics and Marketing Warehouse	Microsoft SQL Server 2008 Express Edition (included with the installer)	Windows Server 2008 x64
	Microsoft SQL Server 2008 Express, Standard or Enterprise Edition you install yourself	Windows Server 2008 x64

## Preparing for a Microsoft SQL Server Express Installation

Only install SQL Server Express Edition if you plan to install Analytics on one computer. For Marketing Warehouse installations and distributed installations of Analytics, install the appropriate versions of SQL Server Standard or Enterprise Edition as specified in the table, above, on all computers in the installation. See the [System Requirements](#) for more information

## Using the WebTrends Installer to Install SQL Server 2008 Express

The simplest way to install Analytics on one computer is to allow the WebTrends installer to install and configure SQL Server 2008 Express for you. No special preparation is required. The installer automatically loads SQL Server 2008 Express Edition onto your server, the installs the system database, and provides the host name, instance name, and port number for you.



## Using Your Own Installation of SQL Server Express

If you plan to install Analytics on one computer only, you can install SQL Server 2005 Express Edition or SQL Server 2008 Express Edition for the system database. Use the guidelines for installing and configuring SQL Server Standard or Enterprise Edition, below.

## Preparing for a Microsoft SQL Server Standard or Enterprise Edition Installation

This section contains instructions that must be followed before installing WebTrends Analytics software.

### Installing Microsoft SQL Server for Your WebTrends System Database

The Microsoft SQL Server (SQL Server 2005 or SQL Server 2008) installation must be installed on a system dedicated to storing and supporting only WebTrends-related databases. While SQL Server can maintain multiple databases, the system resource and configuration requirements of the WebTrends database server environment demand a dedicated server to assure proper performance and reliability.

If you plan to use SQL Server for your WebTrends system database, you must follow the installation protocol.

#### To follow the installation protocol:

1. Select English as the default language when you install SQL Server.
2. Install the SQL Server software before beginning any WebTrends product installation. For more information, see System Requirements.
3. Install the WebTrends system database on the same computer where the SQL Server was installed. For a distributed installation, this should happen before you install any other WebTrends Analytics software component. For a basic installation, you can install more components at the same time as the WebTrends system database.

For a distributed installation, the SQL Server must be installed on a separate system from the WebTrends Analysis server. The WebTrends system database server should be a stand-alone database server.

For a distributed installation, you must run the installation of the WebTrends system database component from the computer where your SQL Server is installed in order for the upgrade process to work properly.

## Setting up Microsoft SQL Server

WebTrends Analytics requires certain setup steps for Microsoft SQL Server.



#### Note

When you install the WebTrends System Database component, the installer prompts you for information about how your Microsoft SQL Server is set up. It is helpful to write down the details of the setup configuration so they are available when you install WebTrends.



**To set up Microsoft SQL Server:**

1. Create two empty SQL Server databases. Name one of them `wtMaster` and the other `wt_Sched`. These are the “system databases”.
2. Specify a port for WebTrends services to connect to the system databases..

**Note**

If you install Microsoft SQL Server using the default instance, you must use a fixed port.

3. Configure a login account using Windows Authentication. This login is used by the installer. The login account must:
  - Be a member of the `securi tyadmi n` and `dbcreator` server roles.
  - Have `db_owner` or `db_ddl admi n`, `db_datareader`, and `db_datawri ter` permissions for the WebTrends databases.

The installation requires the user name and password for this user. For more information, see the Microsoft SQL Server documentation.

4. Enable the TCP/IP Protocol for Microsoft SQL Server. Enabling this protocol prevents a TCP/IP connection error. For more information, see “Enabling TCP/IP Protocol for Microsoft SQL Server.”
5. Make a note of the host name of your SQL Server Standard or Enterprise Edition and the password for your login account. You will need this information if you are installing on multiple computers.

**Note**

Because of issues with mixing languages, WebTrends only supports SQL Servers that have the `SQL_Latin1_General_CP1_CI_AS` (case-insensitive) collation installed at the server level. The databases should then be configured with that same `SQL_Latin1_General_CP1_CI_AS` collation.

## Enabling TCP/IP Protocol for Microsoft SQL Server Standard or Enterprise Edition

If you plan to install the Marketing Warehouse components and have TCP/IP Protocol disabled, you might experience database connection problems when you set up your database from WebTrends Administration.

**To enable the TCP/IP Protocol:**

1. Open the SQL Server Configuration Manager.
2. Select **SQL Server Network Configuration** in the left pane.
3. In the right pane, double-click **Protocols for MSSQLSERVER**.
4. In the right pane, note the status of TCP/IP.
5. *If the status denotes Disabled*, right-click **TCP/IP** and select **Enable**.
6. In the left pane, select **SQL Server Services**. This populates the right pane with protocols.
7. In the right pane, right-click **SQL Server Agent (MSSQLSERVER)** and select **Restart**.



- In the right pane, right-click **SQL Server (MSSQLSERVER)** and select **Restart**.

## Configuring WebTrends Services User Accounts

During installation of the WebTrends Analytics system database, you have the option of specifying a single user account (**Typical** option) to control the Loader Service group, the UI Service group, and the System Database service group, or one account for each of the service groups (**Advanced** option). Before you install WebTrends, you need to create Active directory users for WebTrends services to run under. *If you are installing Analytics only on one computer*, you can use local users instead of Active Directory users.

If your organization does not have strict security requirements for applications that access Microsoft SQL Server, you can select **Typical** to use one user for all three service groups. This simplifies installation and maintenance.

You can also create two or three users for the service groups to run under, using the **Advanced** option. See Table 2 for a list of database permissions assigned to each group. This information may help you decide how many users to create.

### Note



Each Active Directory user must be a member of the local administration group and have permissions on all shared directories used by WebTrends on every computer in the distributed installation environment.

When you install Marketing Warehouse, the WebTrends installer creates SQL Server users associated with each of the domain user accounts you create before installation. Each SQL Server user is given only the minimum permissions to allow the associated service group to perform its tasks. For information about the SQL Server permissions assigned to each service group, see Table 2..

**Table 2: Database Permissions**

Service Group	Application(s)	System Database	Configuration Database	Event Database
Loader Service	Event Database Loader Services	<b>Server Roles:</b> public	<b>Server Roles:</b> public, processadmin	<b>Server Roles:</b> public, processadmin, dbcreator
		<b>Database Permissions</b> data reader, data writer, exec sproc	<b>Database Permissions</b> data reader, data writer, exec sproc	<b>Database Permissions</b> db_owner or db_ddladmin, db_securityadmin, db_accessadmin, data reader, data writer, execsproc NOTE: [msdb] requires the sqlAgentUserRole



**Table 2: Database Permissions**

Service Group	Application(s)	System Database	Configuration Database	Event Database
UI Service	Tomcat service (for administration UI), User Interface Service	<b>Server Roles</b> public	<b>Server Roles</b> public	<b>Server Roles</b> public
		<b>Database Permissions</b> data reader, data writer, exec sproc	<b>Database Permissions</b> data reader, data writer, exec sproc	<b>Database Permissions</b> data reader, data writer, exec sproc
System Service	Scheduler Agent, System Monitor, Email Notification, Event Database Splitter, Log File Pre-processor	<b>Server roles:</b> public	<b>Server roles:</b> public	<b>Server Roles:</b> none
		<b>Database Permissions</b> data reader, data writer, exec sproc	<b>Database Permissions:</b> none	<b>Database Permissions:</b> db_ddladmin

## Creating Domain Users

To create users for WebTrends services using Active Directory accounts:

1. On the computer that acts as your network's domain controller, click **Start > Administrative Tools > Active Directory Users and Computers**.
2. Right-click **Users** and select **New > User**.
3. Enter a First name for the user. We recommend giving the user a name that will help you identify its purpose. For example, `WebtrendsLoaderService`.
4. Enter the same name in the **User logon name** field and click **Next**.
5. Enter a password for the account in the **Password** and **Confirm password** fields.
6. Clear the **User must change password at next logon** check box.
7. Select **User cannot change password** and **Password never expires**.
8. Click **Next**. Review your settings and click **Finish**.

*If you are creating one user for each service group (**Advanced** option), repeat steps 2 - 8 for the two remaining WebTrends service groups.*

## Configuring the Domain User Security Settings

WebTrends requires that the domain users you specify during the installation have certain security settings.

To give the domain user account the required security settings:

1. From the Windows Local Security Settings dialog, select **Local Policies > User Rights Assignment**.
2. Select **Act as part of the operating system**, right-click, and select **Properties**.



3. Click **Add User or Group**, type a domain user account, and click **Check Names**. You must specify a specific domain user and not a group. The domain user must be a member of the local administrators group on the computer where WebTrends Analytics is installed.
4. Repeat the previous steps for the **Log on as a service** policy and the **Log on locally/Allow log on locally** policy.

**To grant a user access to the computer where the UI Server installation resides:**

1. From the Windows Local Security Settings dialog, select **Local Policies > User Rights Assignment**.
2. Select **Access this computer from the network**, right-click, and select **Properties**.
3. Click **Add User or Group**, type a domain user account, and click **Check Names**. You must specify a specific domain user and not a group. The domain user must be a member of the local administrators group on the computer where WebTrends Analytics is installed.

## Configuring the Installer User Account

The user logged in during installation must be configured properly for installation to complete.

The installer user must:

- Be an administrator on each server where the installer is run
- Have access to all WebTrends shares

You can remove SQL Server and WebTrends shares permissions from the installer user after installation if necessary.

## Optimizing Your Database for a WebTrends Installation

Before you install WebTrends or SQL Server, you should set up your database to optimize performance. You can begin performance tuning by looking at your database environment and evaluating the infrastructure and capacity of your database environment. This document addresses using best practices and modifying disk layout to improve database performance by changing either local storage disks or SAN, depending on the database infrastructure that has been implemented.



## Design Recommendations for SQL Server Configuration

This section provides best practices that you can use during the installation of your database. Although you can improve performance using hardware, database configuration is crucial to improving system performance. You can use the following settings to adjust the design and setup of a SQL Server configuration and improve database performance.

Configuration Option	Definition	Best Practice
Affinity Mask	Indicates which processor (core) SQL Server spawns threads.	Set to 0 if it is a dedicated instance. If the server will host multiple instances, this setting can be used to dictate which instance uses a particular processor (core)
Affinity I/O Mask	Indicates which processor (core) SQL Server spawns threads.	Set only if needed when running multiple instances.
AWE	<p>Extends the SQL Server buffer pool beyond the 3GB limit set for virtual memory when the /3GB switch is enabled in the boot . ini . SQL Server uses a buffer pool to cache data to avoid using hard disk space. A 32-bit SQL Server does not usually go beyond 3 GB without enabling AWE to use the rest of the memory.</p> <p><b>Note</b></p> <p>The boot . ini file is located in the root folder on a Windows system.</p> <p><b>Note</b></p> <p>AWE is not needed and cannot be configured on 64-bit operating systems.</p>	Set to 1 to enable on a 32-bit operating system. Enabling AWE increases the startup time for SQL Server because it pre-allocates memory.
Lightweight Pooling	Provides a means of reducing the system overhead associated with the excessive context switching.	Set this parameter to default value of 0. Enabling this parameter only produces minor performance improvements.
Max Degree of Parallelism	Limits the number of processors (a maximum of 32) to use in parallel plan execution.	Set this parameter to default value of 0 so the engine determines which processors (core) are busy and decides the degree of parallelism.



Configuration Option	Definition	Best Practice
Max Server Memory / Min Server Memory	Reconfigures the amount of memory (in megabytes) in the buffer pool used by an instance of Microsoft SQL Server.	If the host is a dedicated host, the value should be set to an 80/20 or 90/10 rule depending on the amount of physical memory. The default physical memory size is 2GB and 0GB is allocated for the buffer pool. The more memory that can be allocated, the larger the buffer pool. With a larger buffer pool, SQL Server can avoid using hard disk space and performs much faster.
Max Worker Threads	Configures the number of worker threads available to Microsoft SQL Server processes.	Set this parameter to default value. The default setting enables Scheduler to determine the number of threads available, which depends on whether the host is 32 bit or 64 bit.
Network Packet Size	Sets the packet size (in bytes) used across the entire network.	Set this parameter to default value. A different value might be recommended depending on the makeup of the network, such as the backbone, fibre, 10/100/1000, or ATM.
Priority Boost	Specifies whether Microsoft SQL Server should run at a higher Microsoft Windows NT 4.0 or Windows 2000 scheduling priority than other processes on the same computer.	Set this parameter to default value of 0, which is a priority base of 7.
Recovery Interval	Sets the maximum number of minutes per database that Microsoft SQL Server needs to recover databases.	Set this parameter to default value of 0.
Default Connection Options (ANSI NULLS and etc.)	Sets default query-processing options for user connections.	Set these parameters to their default values. If you change these values, WebTrends may produce incorrect data.



## Optimizing Your Disk Layout to Improve Performance

In addition to database configuration, you can also improve performance by adjusting the disk layout of your database. Disk layout poses the greatest bottleneck for any Relational Database Management System (RDBMS). You can improve performance by separating log files and data (Index) files because log files require sequential access and data files require random disk access. Because your data files move the drive head randomly and the logs are all sequential reads and writes, it is less efficient to store data files and logs on the same drive. For best performance, divide the logs, tempDB, data files, and the other database components onto separate drives or spindles.

Disk layout can be divided into local storage disks or storage area network (SAN). For information about using local storage disks for your disk layout, see [“Using a Local Storage Disks Disk Layout” on page 17](#). For information about using SAN for your disk layout, see [“Using a SAN Disk Layout” on page 18](#).

### Formatting for Improved Performance

Depending on your SAN or local storage disks storage system, you may need to format the drives in 64K blocks. This format usually yields anywhere from 10 to 20% improvement. SQL Server writes pages on local storage in 64K chunks by default. If the pages and the block size are not aligned, then each page will take up two blocks. With this scenario, the engine must read and write two 64K blocks for each 64K page.

- Although SQL server writes data and indexes, the smallest unit of data that SQL Server can allocate is 64 KB, which is called an extent. The formula is  $8k/page * 8 pages = 1 \text{ extent (64K)}$ . Format the NTFS in 64K block size, which will line up with the two 64K blocks and allow for faster reads of data. Matching up the NTFS format with the SQL Server I/O size optimizes the data path thus reducing the amount of I/O thrashing when querying the data. For more information, see Books Online in Microsoft's SQL Server Developer Center.
- The ideal size for a transaction log is 4K.
- Use the following command to determine the current setting.

```
fsutil fsinfo ntfsinfo c:
```

### Storing TempDB and Log Files on Separate Disks

The location of the tempDB is important if the database is a transactional based because all version control occurs here. Prior to an update, the original row is saved in case a rollback needs to be performed. To optimize performance, make sure tempDB and log files are stored to separate LUNS (logical unit numbers) consisting of a dedicated disk.

## Using a Local Storage Disks Disk Layout

You can use the following methods to tune local storage disks for better performance.

### Increase the Number of Disks

Distributing data to multiple disks always improves performance. Increasing the number of disks and spindles enables you to spread data access across more disk drives, resulting in several spindles performing the work at the same time instead of one spindle performing all the work to access the data. For example, it is better to use four 9 GB disk drives than it is to have one 36 GB disk.



## Using RAID to Distribute Data

Another way to distribute data across multiple disk spindles is to use a form of RAID. You can use either RAID 5 or RAID 0+1 each of which has benefits depending on whether you focus on budget or performance needs.

RAID 5 does not have the same performance and durability as RAID 0+1. The read operations between RAID 5 and RAID 0+1 are similar when comparing a similar number of spindles, but RAID 5 performance is less fault-tolerant and degrades more quickly under certain circumstances. A RAID 5 write operation requires a read-before-write operation to be able to recalculate the parity stripe, as well as an additional write operation for the parity stripe. In contrast, a write operation in RAID 0+1 only requires a write to each member of the mirror pair.

During a rebuild operation of a failed member, RAID 0+1 generally rebuilds much more quickly, because it is a mirror-copy operation. When RAID 5 rebuilds, it requires recalculating parity from data and back-calculating data from parity. In the unlikely circumstance that a second drive using the RAID 0+1 set fails during the rebuild, as long as the second failed drive is not the mirror pair of the first failed drive, the RAID 0+1 set survives. However, a second failed drive in a RAID 5 set, even during a rebuild, guarantees a total RAID 5 set failure.

The following table illustrates an ideal setup for local storage disks.

Database	Data	Log
Master, Model, MSDB	RAID 5 F:	RAID 5 J:
TempDB	RAID 0+1 G:	RAID 0+1 I:
WebTrends	RAID 5 H:	RAID 5 J:

## Using a SAN Disk Layout

If you use SAN, you should consult with your vendor for specific recommendations. Because SAN disks are usually shared by other intensive applications in an organization, you should have dedicated storage for your database environment.

### Formatting

You can improve performance by using the formatting recommendations provided for both local storage disks and SAN disks, but for SAN you also should remember to use the Windows System Utility DiskPart.exe to align the sectors. For more information about formatting recommendations, see [“Formatting for Improved Performance” on page 17](#). Most SAN vendors use 64K blocks when formatting, but be sure to check with your vendor.

### Number of Data Files

Instead of using one large data file, you can use parallelism to improve performance by making sure the number of data files is equal to or greater than the number of processors (or cores). Parallelism is when two or more processors (or cores) each perform part of the work at the same time.



## Deciding Which Ports to Use

WebTrends uses network ports to communicate with all the components in your installation. Review this section to make sure that other services that you plan to run on your WebTrends computers do not conflict with these ports.

During installation, you can configure the following ports:

- User Interface Server

After installation, you can modify the following ports:

- GeoTrends Server
- Report Cache Server

---

### Note



After you install the User Interface Server component, it is only possible to install the Report Cache Server using the command line method. For more information about installing the Report Cache Server, see [“Installing Subcomponents After You Install the Component” on page 50](#).

---

The following table shows the default port for each WebTrends component.

Component	Default Port	Configurable After Installation
Event Database Loader	3307	No
GeoTrends Server	7199	Yes
Report Cache Server	7299	Yes
System Database	Microsoft SQL Server Standard or Enterprise Edition: 1433 <b>Note:</b> This should be installed and configured prior to the installation of WebTrends. Microsoft SQL Server 2005 Express Edition: 1433	No

In addition, WebTrends needs to be able to access the WebTrends data stored on your network. This data is stored in the repositories discussed earlier in this chapter. For more information, see [“Setting Up Shares for WebTrends Data” on page 6](#). WebTrends runs on Windows which uses the Server Message Block (SMB) protocol to access network data. Your network administrator can tell you which port your version of Windows uses.



## Document Revision History

Table 3 contains a summary of changes made to this document beginning with the release of WebTrends Analytics, version 8.7.

**Table 3: Document Revision History**

Software Version	Date of Last Update	Summary of Changes
v8.7d	July 2009	<ul style="list-style-type: none"><li>• Updated pre-installation checklist to provide information about new features</li><li>• Removed references to Express Installation (no longer supported)</li><li>• Updated recommendations for SQL configuration</li><li>• Added footer link to Documentation Center.</li></ul>



---

## Chapter 3

# Installing WebTrends Analytics on One Computer

This chapter provides the information you need to install WebTrends Analytics software on one computer. Before you install, make sure you review the planning and preparation guidelines. For more information, see [“Preparing to Install WebTrends Analytics” on page 5](#).

For information about installing WebTrends Analytics on multiple computers, see [Chapter 4, “Distributed Installation of WebTrends Analytics Software” on page 27](#). For information on installing Marketing Warehouse, see “Installing the Marketing Warehouse” in the *WebTrends Marketing Warehouse User’s Guide*.

## Understanding WebTrends Components

The WebTrends Analytics installation program includes several components. Some of these components are required and some are optional. With a custom installation, you can decide which optional components you install on your computer.

### System Database

Stores the data that WebTrends Analytics needs to coordinate work for all components. You can use Microsoft SQL Server Standard or Enterprise Edition or Microsoft SQL Server 2005 Express Edition for the system database. The WebTrends installation program includes Microsoft SQL Server 2008 Express Edition.

### UI Server

Provides the application interface and serves reports.

### Analysis Engine

Analyzes your web data.

#### Standard Analysis Engine

A required component that performs scheduled analysis of your web traffic data.

#### Express Analysis Engine

An optional component that provides on-going analysis of your web traffic data for limited, current day only reporting. If you plan to use Express Analysis, you will need to install the Express Analysis engine component.

### GeoTrends Server

An optional component that adds geographical data about your visitors to reports.



## Upgrade Considerations

---

### Important



Upgrading WebTrends Analytics requires additional preparation if your current installation includes, or has ever included, Marketing Warehouse. Please contact technical support before upgrading.

---

### Important



You should back up your WebTrends computers before you install the upgrade. For more information, see [“Backing Up and Restoring WebTrends Data” on page 129](#).

---

If you are completing these steps to upgrade your installation, you must stop the required services before installing the upgrade.

#### To stop WebTrends services before an upgrade:

1. From the Windows Services panel, select and stop all WebTrends services except for the database services. Make sure you stop the non-database WebTrends services on all WebTrends computers.
2. Make sure your WebTrends system database is still running by ensuring that all relevant Microsoft SQL services are started.

## Installing on One Computer

This section provides instructions for installing WebTrends Analytics on a single computer. Before installing, make sure you review the planning and preparation guidelines. For more information, see [“Preparing to Install WebTrends Analytics” on page 5](#)

#### To install WebTrends Analytics on one computer:

1. If you downloaded your software from the WebTrends web site:
  - a. On the computer where you want to install WebTrends Analytics, open the downloaded installation file.
  - b. Unzip the files either to the default location or to the location of your choice. When the files are unzipped, the installation program starts automatically.
2. Otherwise, on the computer where you want to install WebTrends Analytics, insert the installation DVD-ROM into your DVD-ROM drive. If your computer is set up to use auto-run, the installation program is automatically launched. If not, choose **Run** from the File or Start menus and type `X:\setup` where *X* is the drive letter of your DVD-ROM drive. The installation program starts. Click **Next** to continue.
3. Before the WebTrends program files are copied to your computer, you must accept the End-User License Agreement. If you agree to the stated terms, click **I accept the terms in the license agreement** and click **Next**. Otherwise, click **Cancel** to exit the installation program without installing the software.



4. In the License Keys dialog, type a valid license key in the text box and click **Add License**. Repeat this step for each license key, add-on license, and subscription license that you have. Contact WebTrends Support if you do not have a valid license key.

**Note**

The computer where you are installing WebTrends must have an Internet connection in order to automatically activate a license key. If the computer does not have an Internet connection, you can manually activate the license key.

---

5. Select the directory where you will install WebTrends.

**Note**

If you are installing WebTrends on a 64-bit machine, you cannot install to C: \Program Files.

---

6. In the Select Components dialog, select the components to install. For more information about WebTrends components, see [“Understanding WebTrends Components” on page 21](#). Click **Next**.
7. *If your computer does not meet the minimum system requirements*, a message indicates the requirements that are not met. Click **No** to cancel the installation program. Click **Yes** to continue. Only install WebTrends Analytics on a dedicated computer that meets the minimum system requirements. For more information, see the [Analytics Software System Requirements](#).
8. In the System Database Connection dialog, complete the following steps:
  - a. Select **Use an existing Microsoft SQL Server 2005 or 2008 installation**.
  - b. In the **Host** text box, specify the name of the server where you installed the WebTrends system database during the WebTrends Analytics distributed installation.
  - c. In the **Instance** text box, specify the instance name of the SQL Server instance where the system database is located. *If you are using the default SQL Server instance on the Host*, leave this text box empty.
  - d. If the database instance is configured to use static ports, specify the connection port of the system database instance in the **Port** text box. Click **Next**.

**Note**

Do not specify a port unless you plan to use a custom port. The standard port for the default instance is 1433.

---

9. In the WebTrends Service Account dialog, specify whether to run all WebTrends services under a single user account, or whether to run three different types of WebTrends services under separate user accounts. Using Advanced mode provides a higher level of security by making it possible to run the UI services, the Event database services, and the remaining “system” services under three different accounts.
  - Click **Typical** to run all WebTrends services under the same user account. Then click **Next**.
  - Click **Advanced** to run WebTrends Analytics services under three different user accounts. Then click **Next**.



10. If you selected **Typical** in the previous dialog, provide user name, password, and domain information for the user account that WebTrends services will use. Click **Next**.

---

### Notes



- Make sure the security settings are properly configured for the user account selected in this step. For more information, see [“Preparing to Install WebTrends Analytics” on page 5](#).

11. If you selected **Advanced** in the previous dialog, provide user name, password, and domain information for the three user accounts that each group of WebTrends services will use. Click **Next**.
12. In the Data Storage Locations dialog, specify the storage locations for each data repository. Each path must be accessible by the WebTrends computer. As a best practice, you should specify the paths to shared directories that you create before beginning the installation. Specifying these paths using Universal Naming Convention (UNC) simplifies adding computers to your installation. For more information, see [“Setting Up Shares for WebTrends Data” on page 6](#).

Specify the following storage locations:

- Location for storing analysis data in the **Analysis Data Repository** text box.
- Location for storing report data in the **Report Data Repository** text box.
- Location for WebTrends configuration data in the **Configuration Repository** text box.
- Location for configuration data that you can use to restore your installation in the **Backup Repository** text box.
- Location for data generated by Data Scheduler data captures in the **Analysis Data Capture Repository** text box.

Click **Next**.

13. If you selected **Express Analysis Engine** in [Step 6](#), in the Express Analysis Data Storage Locations dialog complete the following steps.
- a. In the **Express Analysis Log Source** text box, specify the location where WebTrends should store Express Analysis log files.
  - b. In the **Express Analysis Report Data Repository** text box, specify the location where WebTrends should store Express Analysis reports.
  - c. Click **Next**.
14. In the WebTrends User Interface dialog, configure Microsoft IIS for the WebTrends UI server.
- a. Specify the IP address that you want to be associated with the WebTrends UI server.
  - b. Specify the port that will be used to access WebTrends Analytics. The default port is 7099.
  - c. Specify the URL that you want to be used to access WebTrends.
  - d. If you want to identify this WebTrends installation, provide a description.
  - e. Click **Next**.
15. If you selected **GeoTrends Server** in [Step 6](#), in the GeoTrends Data File Location dialog, specify the location of your GeoTrends data file.



- If you want WebTrends to use the latest version of the GeoTrends data file, click **Download File**. Save the file to a location accessible by your WebTrends computer, then click **Browse** and locate the `geotrends.dat` file.
- If you want WebTrends to use the GeoTrends data file provided on your installation DVD-ROM, click **Browse** and locate the `geotrends.dat` file on the DVD-ROM.

Click **Next** to continue.

16. The Start Installation dialog opens. Click **Install**. The WebTrends program files are installed to your computer.

17. The InstallShield Wizard complete dialog indicates that the installation is complete.

- If you want to read important information about WebTrends including new features, identified issues, and summary of your installation, leave the **View Release Notes** check box selected.*
- If you want to download the WebTrends sample files that show how to set up features and demonstrate available reports, in the Download Sample Data dialog, select the **Download Sample** check box. Click **Save**, and specify where you want to save the file. By default, the file is saved to your computer desktop. After the installation program finishes, you can install the samples that you downloaded.*
- Make a note of your WebTrends log in information.

18. Click **Finish** to close the Installation program.



#### Note

If you installed WebTrends from a downloaded installation program, be sure to preserve the downloaded directory that contains the installation files. If you want to modify WebTrends components after installation, InstallShield needs the downloaded directory to change your installation.

## Logging In to WebTrends Analytics

After you complete the installation, you can log in to WebTrends Analytics.

#### To log in:

1. Click the WebTrends Analytics desktop icon. The login dialog opens.
2. Type your network user name and password and click **Log in**. If you did not provide licensing information during the installation, you need to activate your license. For more information, see [“Logging In to WebTrends Analytics” on page 25](#).

## What’s Next?

If you are installing WebTrends for the first time, you will need to complete a few basic configuration steps to start using WebTrends Analytics.

#### To complete your implementation:

1. Create a data source. For more information, see “Understanding Data Sources” in the *WebTrends Administration User’s Guide*.
2. Create a profile. For more information, see [“Creating WebTrends Analytics Profiles” on page 73](#).



- 3. Configure session tracking. For more information, see [“Tracking Visitor Sessions”](#) on page 83.
- 4. Review your security options. For more information, see [“Securing Your Implementation”](#) on page 99.

## Document Revision History

Table 4 contains a summary of changes made to this document beginning with the release of WebTrends Analytics, version 8.7.

**Table 4: Document Revision History**

Software Version	Date of Last Update	Summary of Changes
v8.7c	July, 2009	<ul style="list-style-type: none"><li>• Updated installation workflow to include latest configuration</li><li>• Removed references to Express Installation (no longer supported)</li></ul>
v8.7d	July, 2009	Added footer link to Documentation Center.



---

## Chapter 4

# Distributed Installation of WebTrends Analytics Software

This chapter provides an overview of WebTrends distributed configuration and provide instructions for installing WebTrends Analytics software in a distributed environment. Issues related to upgrading are discussed at the end of the chapter. Before you install, make sure you review the planning and preparation guidelines. For more information, see [“Preparing to Install WebTrends Analytics” on page 5](#).

## Who Should Use a Distributed Installation?

Large organizations that support a large number of report users or have a high-traffic web sites might need to install WebTrends components across multiple computers that work together as one implementation. You might use a distributed installation to manage many complex analysis processes, distribute the impact of a large number of concurrent users viewing reports, or analyze more web site traffic than a single-computer installation can process in one day. If you plan to use the WebTrends Marketing Warehouse, you must install the Marketing Warehouse on a dedicated computer, separate from the computer(s) running other WebTrends components.

---

### Note



A distributed installation of WebTrends Marketing Warehouse requires a minimum of three computers; one computer for WebTrends Analytics, one for the Event Database Loader, and one for the Marketing Warehouse databases.

---

Your web site traffic and your reporting needs determine the number of computers in your WebTrends installation. A distributed configuration provides the flexibility to run components in the configuration that works best for your organization, whether that means installing multiple components on one computer or installing a single component, such as Analysis, on multiple computers. WebTrends components are discussed in detail in the section, [“Performing a Distributed Installation” on page 30](#).

For some suggested configurations to support a distributed installation, see the [Analytics Software System Requirements](#).



## Understanding WebTrends Components

The WebTrends Analytics installation program includes several components. Some of these components are required and some are optional. You can decide which optional components you install on your computer.

### Analysis Engine

Analyzes your web data. There are two types of analysis engines you can use:

#### Standard Analysis Engine

A required component that performs scheduled analysis of your web traffic data.

#### Express Analysis Engine

An optional component that provides on-going analysis of your web traffic data for limited, current day only reporting. If you plan to use Express Analysis, you will need to install the Express Analysis component.

### GeoTrends Server

An optional component that adds geographical data about your visitors to reports. If you want this geographic information, and your web site receives a lot of visitors, then you may want to install GeoTrends on a separate computer. Doing so allows GeoTrends to run without taking memory away from other important components. Otherwise, install GeoTrends on the machine running the Analysis Engine.



#### Note

If you plan to install WebTrends Marketing Warehouse, GeoTrends Server is required.

---

### Marketing Warehouse

Imports data from SmartSource Data Collector data files and stores the results in an SQL Server database to allow customized data exploration. For more information about these components, see the *WebTrends Marketing Warehouse User's Guide*.

### System Database

Stores the data that WebTrends Analytics needs to coordinate for all components. For more information, see ["Performing a Distributed Installation" on page 30](#). You can use Microsoft SQL Server Standard or Enterprise Edition for the system database.

### UI Server

Provides the application interface and serves reports. The number of users who view reports concurrently determines the number of computers that you need running this component. You may want to install the UI Server separately so that report viewing does not slow down when other components such as Analysis are running.

## Upgrade Considerations

This section addresses considerations for upgrading WebTrends.



## Updating Storage Locations

Before upgrading to Analytics v8.7.4, make sure the storage location paths specified in WebTrends Administration are UNC paths.

For example: \\server\volume\directory\file

### To check and update storage locations:

1. Log in to Webtrends Administration.
2. Click **System Management > Storage Locations**.
3. Change all storage location paths to UNC paths.

## Preparing to Upgrade an Existing Installation

---

 **Important**  
Upgrading WebTrends Analytics requires additional preparation if your current installation includes, or has ever included, Marketing Warehouse. Please contact technical support before upgrading.

---

 **Important**  
You should back up your WebTrends computers before you install the upgrade. For more information, see [“Backing Up and Restoring WebTrends Data” on page 129](#).

---

If you are completing these steps to upgrade your installation, you must stop the required services before you install the upgrade.

### To upgrade WebTrends:

1. From the Windows Services panel on the computer where the UI Server is installed, select and stop all WebTrends services except for the database services. Make sure you stop the non-database WebTrends services on all WebTrends computers.
2. Make sure your WebTrends system database is still running by ensuring that all Microsoft SQL services are started.
3. Run the upgrade installation program on the computer that contains the WebTrends Analytics System Database.
4. Uninstall all the components on other computers.
5. Use the installation program to install the new version of WebTrends for all other components. In the System Database Connection dialog, specify the host where the System database is installed. As a best practice, we suggest that you upgrade the Analysis Engine computers last.
6. You can use the Topology dialog to make sure that all computers in the installation are recognized. In the left pane, click **Administration > Application Settings > Monitoring > Topology**.  
If you suspect that one of the WebTrends component computers was not upgraded, check the status logs on each computer in your distributed installation for a “version incompatibility error.” This message is logged when one of the WebTrends components does not match the version of the database. For example, if you cannot view the WebTrends user interface, or if there is no new report data since the upgrade, one of the computers may not have been upgraded. Status logs are located in the *WebTrends Installation Directory\logs* directory.



## Performing a Distributed Installation

After all the computers designated for WebTrends Analytics are configured, you can proceed with the installation. Before you install, make sure you review the planning and preparation guidelines.



### Note

For detailed information on installing Marketing Warehouse, see “Installing the Marketing Warehouse” in the *WebTrends Marketing Warehouse User’s Guide*.

If you want to upgrade your installation to a newer version, see “Upgrade Considerations” on page 28.

When you install WebTrends Analytics components in a distributed configuration, you begin by installing the system database component.

### To perform a distributed installation:

1. If you downloaded your software from the WebTrends web site:
  - a. On the computer where you want to install the system database component, open the downloaded installation file.
  - b. Unzip the files either to the default location or to the location of your choice. When the files are unzipped, the installation program starts automatically.
2. Otherwise, on the computer where you want to install the system database component, insert the installation DVD-ROM into your DVD-ROM drive. If your computer is set up to use auto-run, the installation program is automatically launched. If not, choose **Run** from the File or Start menus and type `X:\setup.exe` where *X* is the drive letter of your DVD-ROM drive. The installation program starts. Click **Next** to continue.
3. Before the WebTrends program files are copied to your computer, you must accept the End-User License Agreement. If you agree to the stated terms, click **I accept the terms in the license agreement** and click **Next**. Otherwise, click **Cancel** to exit the installation program without installing the software.
4. In the License Keys dialog, type a valid license key in the text box and click **Add License**. Repeat this step for each license key, add-on license, and subscription license that you have. Contact WebTrends Support if you do not have a valid license key.



### Note

The computer where you are installing WebTrends must have an Internet connection to automatically activate a license key. If the computer does not have an Internet connection, you can manually activate the license key.

5. Select the directory where you will install WebTrends components.



### Note

If you are installing WebTrends on a 64-bit machine, you cannot install to `C:\Program Files` or to the *drive*: \Program Files path on any other drive.



6. In the Select Components dialog, select the components to install on this computer. The required components are selected by default. For more information about WebTrends components, see [“Understanding WebTrends Components” on page 28](#).

You can install the System Database with or without the UI Server. Decide whether you want the components on separate computers or on the same computer:

- *If you want to install the WebTrends System Database on a dedicated computer, perform the entire installation of the System Database on one computer, then install the UI Server on another computer, and finally install the Marketing Warehouse Databases on a third computer. Clear the check boxes for the components that you do not want to install.*
7. *If you want to install the System Database and the UI Server on the same computer, perform the entire installation of the System Database, Marketing Warehouse Databases, UI Server, and other components on one computer, then install any other components, such as Analysis engines or the Marketing Warehouse, on another computer(s). Clear the check boxes for the components that you do not want to install.*

**Note**

You must install Marketing Warehouse components on a separate server from the WebTrends System Database.

---

8. Click **Next**.
9. *If your computer does not meet the minimum system requirements, a message indicates the requirements that are not met. Click **No** to cancel the installation program. Click **Yes** to continue. Only install WebTrends Analytics on a dedicated computer that meets the minimum system requirements. For more information, see the [Analytics Software System Requirements](#).*
10. In the System Database Connection dialog, complete the following steps:
  - a. Select **Use an existing Microsoft SQL Server 2005 or 2008 installation**.
  - b. In the **Host** text box, specify the name of the server where you installed the WebTrends system database during the WebTrends Analytics distributed installation.
  - c. In the **Instance** text box, specify the instance name of the SQL Server instance where the system database is located. *If you are using the default SQL Server instance on the Host, leave this text box empty.*
  - d. If the database instance is configured to use static ports, specify the connection port of the system database instance in the **Port** text box. Click **Next**.

**Note**

Do not specify a port unless you plan to use a custom port. The standard port for the default instance is 1433.

---



11. In the WebTrends Service Account dialog, specify whether to run all WebTrends services under a single user account, or whether to run three different types of WebTrends services under separate user accounts. Using Advanced mode provides a higher level of security by making it possible to run the UI services, the Event database services, and the remaining “system” services under three different accounts. The best practice is to install WebTrends using domain accounts with read-write access to the WebTrends shared directories for that service type.
  - Click **Typical** to run all WebTrends services under the same user account. Then click **Next**.
  - Click **Advanced** to run WebTrends Analytics services under three different user accounts. Then click **Next**.
12. If you selected **Typical** in the previous dialog, provide user name, password, and domain information for the user account that WebTrends services will use. Click **Next**.

---

### Notes



- Make sure the security settings are properly configured for the user account selected in this step. For more information, see [“Preparing to Install WebTrends Analytics” on page 5](#).

13. If you selected **Typical** in the previous dialog, provide user name, password, and domain information for the three user accounts that each group of WebTrends services will use. Click **Next**.
14. In the Data Storage Locations dialog, specify the storage locations for each data repository. Each path must be accessible by the WebTrends computer. As a best practice, you should specify the paths to shared directories that you create before beginning the installation. Specifying these paths using Universal Naming Convention (UNC) simplifies adding computers to your installation. For more information, see [“Setting Up Shares for WebTrends Data” on page 6](#).

Specify the following storage locations:

- Location for storing analysis data in the **Analysis Data Repository** text box.
- Location for storing report data in the **Report Data Repository** text box.
- Location for WebTrends configuration data in the **Configuration Repository** text box.
- Location for configuration data that you can use to restore your installation in the **Backup Repository** text box.
- Location for data generated by Data Scheduler data captures in the **Analysis Data Capture Repository** text box.

Click **Next**.

15. If you selected *Express Analysis Engine* in [Step 6](#), in the Express Analysis Data Storage Locations dialog complete the following steps.
  - a. In the **Express Analysis Log Source** text box, specify the location where WebTrends should store Express Analysis log files.
  - b. In the **Express Analysis Report Data Repository** text box, specify the location where WebTrends should store Express Analysis reports.
  - c. Click **Next**.
16. In the WebTrends User Interface dialog, configure Microsoft IIS for the WebTrends UI server.



- a. Specify the IP address that you want to be associated with the WebTrends UI server.
  - b. Specify the port that will be used to access WebTrends Analytics. The default port is 7099.
  - c. Specify the URL that you want to be used to access WebTrends. If you choose to customize this URL, you need to configure your web server accordingly.
  - d. If you want to identify this WebTrends installation, provide a description.
  - e. Click **Next**.
17. If you selected *GeoTrends Server* in [Step 6](#), in the GeoTrends Data File Location dialog, specify the location of your GeoTrends data file.
- If you want WebTrends to use the latest version of the GeoTrends data file, click **Download File**. Save the file to a location accessible by your WebTrends computer, then click **Browse** and locate the `geotrends.dat` file.
  - If you want WebTrends to use the GeoTrends data file provided on your installation DVD-ROM, click **Browse** and locate the `geotrends.dat` file on the DVD-ROM.
- Click **Next** to continue.
18. In the Start Installation dialog, click **Next**. The WebTrends program files are installed to your computer.
19. The InstallShield Wizard complete dialog indicates that the installation is complete.
- a. If you want to read important information about WebTrends including new features, identified issues, and summary of your installation, leave the **View Release Notes** check box selected.
  - b. If you want to download the WebTrends sample files that show how to set up features and demonstrate available reports, in the Download Sample Data dialog, select the **Download Sample** check box. Click **Save**, and specify where you want to save the file. By default, the file is saved to your computer desktop. After the installation program finishes, you can install the samples that you downloaded.
  - c. Make a note of your WebTrends log in information.
  - d. Click **Finish** to close the Installation program.
20. Install the remaining components on your other WebTrends-designated computers in the order that you choose.

---

**Note**

If you installed WebTrends from a downloaded installation program, be sure to preserve the downloaded directory that contains the installation files. If you want to modify WebTrends components after installation, InstallShield needs the downloaded directory to change your installation.

---

## Logging In to WebTrends Analytics

After you complete the installation, you can log in to WebTrends Analytics.

### To log in:

1. Click the WebTrends Analytics desktop icon. The login dialog opens.
2. Type your network user name and password and click **Log in**. If you did not provide licensing information during the installation, you are prompted to activate your license.



## What's Next?

If you are installing WebTrends for the first time, you will need to complete a few basic configuration steps to start using WebTrends Analytics.

### To complete your implementation:

1. Create a data source. For more information, see [Choosing a Data Collection and Storage Method](#).
2. Create a profile. For more information, see [“Creating WebTrends Analytics Profiles” on page 73](#).
3. Configure session tracking. For more information, see [“Tracking Visitor Sessions” on page 83](#).
4. Review your security options. For more information, see [“Securing Your Implementation” on page 99](#).

## Managing Your Distributed Installation

WebTrends provides several features for managing the computers in your installation which are discussed in this section.

### Viewing Available Resources

The Topology dialog shows all the hosts, or computers, in your installation, the roles assigned to each host, and the available resources for each computer. A *role* is a function that the host performs according to the modules you installed on that computer. A *group* is a designation that you create primarily for the purpose of distributing work.

### To view the computers in your installation:

In the left pane, click **Administration > Application Settings > Monitoring > Topology**.

### Using Groups to Distribute Work

WebTrends allows you to create groups and assign the computers in your installation to those groups. For example, if you have multiple hosts that have the Standard Analysis Engine Role, you can use Host Groups to make sure that the computers that have sufficient system resources analyze your largest web activity data files. You could then assign profiles that process large data sets to this group.

### To work with host groups:

1. In the left pane, click **Administration > Application Settings > Host Groups**.
2. Click **New** to begin creating a new host group or edit an existing host group.

## Monitoring Hosts in Your Installation

For troubleshooting purposes, you may need to check the status of the WebTrends services. The Service Health dialog allows you manage all the WebTrends services in your installation through WebTrends Administration.

To manage WebTrends services, in the left pane, click **Administration > Application Settings > Monitoring > Service Health**.



## Document Revision History

Table 5 contains a summary of changes made to this document beginning with the release of WebTrends Analytics, version 8.7.

**Table 5: Document Revision History**

Software Version	Date of Last Update	Summary of Changes
8.7d	July, 2009	<ul style="list-style-type: none"><li>• Updated installation workflow to include latest configuration</li><li>• Removed references to Express Installation (no longer supported)</li><li>• Added footer link to Documentation Center</li></ul>





---

## Chapter 5

# Unattended Installation of WebTrends Software

An unattended installation allows you to install WebTrends Analytics without requiring your constant attention by removing the need to answer the typical questions required during the installation process. This chapter describes how to prepare for and perform an unattended installation. It also provides syntax and code examples that show how to install various WebTrends Analytics components.

## Installation Guidelines

The following list outlines guidelines you must follow for installation:

- Use this method only if you are familiar with WebTrends Analytics and your local network.
- Obtain or have administrator rights for the computer(s) on which you are installing WebTrends.
- Perform the unattended installation locally from the computer where you want WebTrends installed.
- Install WebTrends on a remote computer with third-party software that allows you to execute programs on remote systems.

## Preparing for an Unattended Installation

This section describes what you need to do before you perform an unattended installation of WebTrends Analytics. To ensure optimum performance, WebTrends Analytics should be installed on a computer dedicated solely to this software. However, you can install WebTrends components on different computers, and you don't have to install all of the components or subcomponents (but some are required).

### To prepare for your unattended installation:

- Before performing the installation, make sure your computer(s) meets the system requirements. Review the minimum system requirements provided in the Release Notes. Also, see the *WebTrends v8\_5 Release Notes.htm* file located in the /Docs directory of the WebTrends Analytics Installation DVD.
- Before performing the installation, review documentation that will help you prepare for your installation. For more information, see [“Preparing to Install WebTrends Analytics” on page 5](#) and the *Marketing Warehouse Software User's Guide*.
- Make sure you have local administrator rights on the installation computer(s).
- After you install your prerequisites, make sure you install SQL Server 2005 Express, Standard, or Enterprise Edition. For more information, see [“Deciding Which Database to Use for the WebTrends System Database” on page 8](#)



- If you have WebTrends v8.1 or later installed, you need to follow the uninstall instructions. For more information, see [“Performing an Unattended Uninstallation” on page 52](#). If you have an earlier version of WebTrends Analytics installed, uninstall that version using Windows Add or Remove Programs utility before proceeding.
- Currently, unattended upgrades are not supported.
- Review the information available in [“Installing the Prerequisites” on page 40](#) to find out which prerequisites and Microsoft Installer (MSI) files that you need to install, as well as the order that the components should be installed. Use this information to install the following WebTrends components.
  - System Database
  - UI Server \*
  - Analysis Engine \*
  - Marketing Warehouse \*
  - GeoTrends

---

**Note**

The components denoted with an asterisk (\*) contain subcomponents that you can install individually. For more information about installing subcomponents, see [“Installing Individual Subcomponents” on page 47](#).

---

Once you know the WebTrends components that you want to install and where you want to install them, you can use the tables in [“Prerequisites for WebTrends Components”](#) and [“MSI Files to Install for WebTrends Components”](#) to determine which prerequisites and MSI files should be installed for each WebTrends component.

- Plan which MSI files you need to install for each of the Marketing Warehouse products and add-ons included in your license. For more information, see [“MSI Files to Install for WebTrends Components” on page 40](#).

## Prerequisites for WebTrends Components

The following table indicates the prerequisites that need to be installed for each WebTrends component.

WebTrends Component	Required Prerequisites	Required Installation Steps
System Database	<ul style="list-style-type: none"> <li>• Windows Installer 3.1</li> <li>• JET 4.0 Microsoft Core XML Services 6.0</li> <li>• Microsoft SQL Server Native Client</li> <li>• Microsoft SQL Server 2005 Analysis Services 9.0 OLE DB Provider</li> <li>• Microsoft SQL Server 2005 Management Objects Collection</li> </ul>	Follow Steps 1 to 6 in <a href="#">“Installing the Prerequisites” on page 40</a> .



WebTrends Component	Required Prerequisites	Required Installation Steps
UI Server	<ul style="list-style-type: none"> <li>• Windows Installer 3.1</li> <li>• JET 4.0</li> <li>• Microsoft Core XML Services 6.0</li> <li>• Microsoft SQL Server Native Client</li> <li>• Microsoft SQL Server 2005 Analysis Services 9.0 OLE DB Provider</li> <li>• Microsoft SQL Server 2005 Management Objects Collection</li> </ul>	Follow Steps 1 to 6 in <a href="#">“Performing an Unattended Installation of WebTrends”</a> on page 41.
Analysis Engine	<ul style="list-style-type: none"> <li>• Windows Installer 3.1</li> <li>• JET 4.0</li> </ul>	Follow Steps 1 to 2 in <a href="#">“Performing an Unattended Installation of WebTrends”</a> on page 41.
Marketing Warehouse	<ul style="list-style-type: none"> <li>• Windows Installer 3.1</li> <li>• JET 4.0</li> <li>• Microsoft Core XML Services 6.0</li> <li>• Microsoft SQL Server Native Client</li> <li>• Microsoft SQL Server 2005 Analysis Services 9.0 OLE DB Provider</li> <li>• Microsoft SQL Server 2005 Management Objects Collection</li> </ul>	Follow Steps 1 to 6 in <a href="#">“Performing an Unattended Installation of WebTrends”</a> on page 41.
GeoTrends	<ul style="list-style-type: none"> <li>• Windows Installer 3.1</li> <li>• JET 4.0</li> </ul>	Follow Steps 1 to 2 in <a href="#">“Performing an Unattended Installation of WebTrends”</a> on page 41.



## MSI Files to Install for WebTrends Components

The following table indicates which MSI files need to be installed for each WebTrends component.

WebTrends Components	WebTrends System Database.msi	WebTrends Scheduler Agent.msi	WebTrends System Repository.msi	WebTrends Analysis.msi	WebTrends Marketing Warehouse.msi	WebTrends GeoTrends.msi	WebTrends Main Application.msi	WebTrends Management Application.msi	WebTrends Visitor Intelligence.msi	WebTrends UI Server Modules.msi
System Database	Y	Y	Y	--	--	--	--	--	--	--
UI Server	--	Y	--	--	--	--	Y	Y	Y	Y
Analysis Engine	--	Y	--	Y	--	--	--	--	--	--
Marketing Warehouse	--	Y	--	--	Y	--	--	--	--	--
GeoTrends	--	Y	--	--	--	Y	--	--	--	--

## Installing the Prerequisites

Prerequisites are third party software programs that need to be installed before you install WebTrends Analytics. You need to install prerequisites before you install the WebTrends Analytics components that you want on your computer. Only install the prerequisites that you need. If you do not have one or more prerequisite programs, you can find them on the WebTrends software installation DVD in the MSI directory. For example, you can go to `E:\MSI\prerequisite name.msi` to find a prerequisite.

Some prerequisites are not required for all WebTrends components. Use the WebTrends Components table to determine which steps that you need to perform for each WebTrends component. For more information, see [“Installing the Prerequisites” on page 40](#)

### To install prerequisites:

1. Install the appropriate Windows Installer 3.1. For example, `WindowsInstaller-KB893803-v2-x86.exe`.
2. Install the appropriate JET 4.0:



- For Windows 2003 32 bit, install `windowsserver2003-kb829558-x86-enu.exe`.
  - For Windows 2003 64 bit, install `WindowsServer2003-KB829558-i a64-ENU.exe`.
  - For XP, install `WindowsXP-KB829558-x86-ENU.exe`.
3. Install the Microsoft Core XML Services 6.0 for your processor.
    - For a 32-bit processor, install `msxml 6.msi`.
    - For an Intel 64 bit processor, install `msxml 6_i a64.msi`.
    - For a non-Intel 64 bit processor, install `msxml 6_x64.msi`.
  4. Install the Microsoft SQL Server Native Client for your processor.
    - For a 32-bit processor, install `sql ncl i.msi`.
    - For an Intel 64 bit processor, install `sql ncl i_i a64.msi`.
    - For a non-Intel 64 bit processor, install `sql ncl i_x64.msi`.
  5. Install the Microsoft SQL Server 2005 Analysis Services 9.0 OLE DB Provider for your processor.
    - For a 32-bit processor, install `SQLServer2005_ASOLEDB9.msi`.
    - For an Intel 64 bit processor, install `SQLServer2005_ASOLEDB9_i a64.msi`.
    - For a non-Intel 64 bit processor, install `SQLServer2005_ASOLEDB9_x64.msi`.
  6. Install the Microsoft SQL Server 2005 Management Objects Collection for your processor.
    - For a 32-bit processor, install `SQLServer2005_XM0.msi`.
    - For an Intel 64 bit processor, install `SQLServer2005_XM0_i a64.msi`.
    - For a non-Intel 64 bit processor, install `SQLServer2005_XM0_x64.msi`.
  7. After you install all prerequisites, you can install Microsoft SQL Server 2005 Express, Standard, or Enterprise Edition. For more information, see [“Preparing for a Microsoft SQL Server Standard or Enterprise Edition Installation”](#) on page 10

## Performing an Unattended Installation of WebTrends

After you install all of the necessary prerequisites and you know which components you want to install, you are ready to perform an unattended installation of WebTrends Analytics. Use the syntax and guidelines provided in this section to perform an unattended installation.

### To perform an unattended installation:

1. Review the Syntax guidelines to help you with your command line syntax. For more information, see [“Using Command Line Installation Syntax”](#) on page 42.
2. Because each WebTrends MSI file requires certain parameters to be set, identify which parameters need to be set for each installation. For more information, see the table in [“Which Parameters Are Required When You Install MSI Files?”](#) on page 43.
3. Refer to the table of MSI Parameters if you want to identify parameters for certain MSI installation files. This table describes available parameters. For more information, see [“Identifying MSI Properties and Their Command Line Parameters”](#) on page 47.
4. Review the information about installing subcomponents. This information shows you how to set the ADDLOCAL properties so you can install one or more subcomponents. For more information, see [“Installing Individual Subcomponents”](#) on page 47.



5. Install WebTrends MSI files in the following order. The .msi files are provided in the WebTrends Installation Package and are located in the *WebTrends\_Installation\_Directory\MSI* folder. For more information and syntax examples, see [“Examples for Installing WebTrends Components” on page 50](#).
  - a. Install WebTrends System Database.msi .
  - b. Install WebTrends Scheduler Agent.msi .
  - c. Install WebTrends System Repository.msi .
  - d. Install WebTrends Analysis.msi .
  - e. Install WebTrends Marketing Warehouse.msi (required by Marketing Warehouse).
  - f. Install WebTrends GeoTrends.msi .
  - g. Install WebTrends Main Application.msi .
  - h. Install WebTrends Management Application.msi (required by Marketing Warehouse).
  - i. Install WebTrends Visitor Intelligence.msi (required by Marketing Warehouse).
  - j. Install WebTrends UI Server Modules.msi (required by Marketing Warehouse).
6. If you installed the UI Server component, you need to stop and start the wtui service. You can stop and start the wtui service by typing the following at a command-line prompt:

```
NET STOP wtui
```

```
NET START wtui
```

## Using Command Line Installation Syntax

To perform an unattended installation of WebTrends Scheduler Agent, use the following syntax:

```
msiexec.exe /i "E:\MSI\WebTrends Scheduler Agent.msi" /q [OPTIONS]
```

where E:\ represents the disk drive that contains the WebTrends DVD-ROM and [OPTIONS] are the MSI Public Properties for the components you want to install. For more information about what to use for the [OPTIONS], see [“Using MSI Public Property Parameters” on page 43](#).

---

### Note



WebTrends currently supports three Microsoft Installer (MSI) user interface options: /q, /qb and /qn. These options set the User Interface level. The /q and /qn options turn off the user interface. A /qb option uses a basic user interface.

---

Optionally, you can include a command that logs installation information for troubleshooting. For example:

```
msiexec.exe /i "E:\MSI\WebTrends Scheduler Agent.msi" /I *v "C:\WebTrends Scheduler Agent.log".
```



## Using MSI Public Property Parameters

Use the examples in “[Examples for Installing WebTrends Components](#)” on page 50 to identify how you should incorporate MSI Properties parameters into your unattended installation syntax. For more information about the parameters you need to set for proper installation, see the table in “[Which Parameters Are Required When You Install MSI Files?](#)” on page 43.

## Which Parameters Are Required When You Install MSI Files?

You can use this section to identify which parameters you need to set for proper installation of each WebTrends component. For information about how these parameters are used, see “[Examples for Installing WebTrends Components](#)” on page 50.

Required Parameter	WebTrends System Database.msi	WebTrends Scheduler Agent.msi	WebTrends System Repository.msi	WebTrends Analysis.msi	WebTrends Marketing Warehouse.msi	WebTrends GeoTrends.msi	WebTrends Main Application.msi	WebTrends Management Application.msi	WebTrends Visitor Intelligence.msi	WebTrends UI Server Modules.msi
ADDLOCAL	n/a	Y	Y	Y	Y	Y	Y	Y	Y	Y
BACKUP_DIR	n/a	n/a	Y	n/a	n/a	n/a	n/a	n/a	n/a	n/a
ADMINUSER_DOMAIN	n/a	n/a	n/a	n/a	n/a	n/a	n/a	Y	n/a	Y
ADMINUSER_USERNAME	n/a	n/a	n/a	n/a	n/a	n/a	n/a	Y	n/a	Y
ADMINUSER_PASSWORD	n/a	n/a	n/a	n/a	n/a	n/a	n/a	Y	n/a	Y
CONFIG_DIR	n/a	n/a	Y	n/a	n/a	n/a	n/a	n/a	n/a	n/a
DB_TYPE	Y	Y	Y	Y	Y	Y	Y	n/a	n/a	n/a



Required Parameter	WebTrends System Database.msi	WebTrends Scheduler Agent.msi	WebTrends System Repository.msi	WebTrends Analysis.msi	WebTrends Marketing Warehouse.msi	WebTrends GeoTrends.msi	WebTrends Main Application.msi	WebTrends Management Application.msi	WebTrends Visitor Intelligence.msi	WebTrends UI Server Modules.msi
EX_ATTRIB_DATADIR	n/a	n/a	n/a	n/a	Y	n/a	n/a	n/a	n/a	n/a
EX_ATTRIB_HOST	n/a	n/a	n/a	n/a	Y	n/a	n/a	n/a	n/a	n/a
EX_ATTRIB_LOGDIR	n/a	n/a	n/a	n/a	Y	n/a	n/a	n/a	n/a	n/a
EX_ATTRIB_PASS	n/a	n/a	n/a	n/a	Y	n/a	n/a	n/a	n/a	n/a
EX_ATTRIB_USER	n/a	n/a	n/a	n/a	Y	n/a	n/a	n/a	n/a	n/a
FASTTRENDS_DIR	n/a	n/a	Y	n/a	n/a	n/a	n/a	n/a	n/a	n/a
GEOTRENDS_DAT	n/a	n/a	n/a	n/a	n/a	Y	n/a	n/a	n/a	n/a
INSTALLDIR	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
IPADDRESS	n/a	n/a	n/a	n/a	n/a	n/a	Y	n/a	n/a	n/a
MASTER_HOST	Y	Y	Y	Y	Y	Y	Y	n/a	n/a	n/a
MASTER_PASS	Y	Y	Y	Y	Y	Y	Y	n/a	n/a	n/a
MASTER_PORT (Microsoft SQL Server 2005 any Edition)	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a



Required Parameter	WebTrends System Database.msi	WebTrends Scheduler Agent.msi	WebTrends System Repository.msi	WebTrends Analysis.msi	WebTrends Marketing Warehouse.msi	WebTrends GeoTrends.msi	WebTrends Main Application.msi	WebTrends Management Application.msi	WebTrends Visitor Intelligence.msi	WebTrends UI Server Modules.msi
MASTER_USER	Y	Y	Y	Y	Y	Y	Y	n/a	n/a	n/a
REPORTS_DIR	n/a	n/a	Y	n/a	n/a	n/a	n/a	n/a	n/a	n/a
RT_ANALSRC	n/a	n/a	n/a	Y	n/a	n/a	n/a	n/a	n/a	n/a
<b>Note:</b> This property is only applicable when Express Analysis is installed.										
RT_REPORTS	n/a	n/a	n/a	Y	n/a	n/a	n/a	n/a	n/a	n/a
<b>Note:</b> This property is only applicable when Express Analysis is installed.										
SCHED_HOST	Y	n/a	n/a	Y	Y	Y	n/a	n/a	n/a	n/a
SCHED_PASS	Y	n/a	n/a	Y	Y	Y	n/a	n/a	n/a	n/a
SCHED_USER	Y	n/a	n/a	Y	Y	Y	n/a	n/a	n/a	n/a
SERVICEACCT	Y	Y	n/a	Y	Y	n/a	Y	n/a	n/a	n/a
SERVICEPASS	Y	Y	n/a	Y	Y	n/a	Y	n/a	n/a	n/a
UI_PORT	n/a	n/a	n/a	n/a	n/a	n/a	Y	n/a	n/a	n/a

Required Parameter	WebTrends System Database.msi	WebTrends Scheduler Agent.msi	WebTrends System Repository.msi	WebTrends Analysis.msi	WebTrends Marketing Warehouse.msi	WebTrends GeoTrends.msi	WebTrends Main Application.msi	WebTrends Management Application.msi	WebTrends Visitor Intelligence.msi	WebTrends UI Server Modules.msi
UI_DESC	n/a	n/a	n/a	n/a	n/a	n/a	Y	n/a	n/a	n/a
UI_URL	n/a	n/a	n/a	n/a	n/a	n/a	Y	n/a	n/a	n/a

## Identifying MSI Properties for Visitor Intelligence

You can use this section to identify which parameters you need to set for proper installation of WebTrends Visitor Intelligence. For information about how these parameters are used, see [“Examples for Installing WebTrends Components” on page 50](#). Visitor Intelligence has different MSI properties than other WebTrends MSI files.

The following list describes the MSI Properties for WebTrends Visitor Intelligence.msi.

### ARPSYSTEMCOMPONENT

Allows you to hide Analyzer inside Add/Remove Program control panel. Set to 1 for this functionality.

### APPOOL

Allows you to specify the AppPool Name you wish to use

### INSTALLDIR

Specifies the Visitor Intelligence directory of the installation. For example, C:\Program Files\WebTrends\common\ui server\WHReporting\

### IIS\_VRDIR

Specifies the virtual directory for Visitor Intelligence. Use WHReporting for this value.

### IIS\_PORT\_NUMBER

Specifies the HTTP port established during database installation. For example, 80. This is the same value that you use for the UI Server Port MSI Property (UI\_PORT).



## USEBOSTORE

Specifies whether or not there is an existing wtWarehouseReporting database. This parameter can be set to 0 or 1. Set this value to 0 the first time you install the UI server component, which includes Visitor Intelligence. WebTrends then creates the wtWarehouseReporting database for Visitor Intelligence. Any subsequent installations of the UI Server in the same distributed installation require that this parameter be set to 1, which tells WebTrends to use the existing wtWarehouseReporting database.

## IS\_SQLSERVER\_SERVER

Specifies the same server that is used for the XMDB. This is the same value that you use for the EX\_ATTRIB\_HOST MSI Property.

## IS\_SQLSERVER\_DATABASE

Specifies the Microsoft SQL Server database. For example, **wtWHReporti ng**.

## IS\_SQLSERVER\_USERNAME

Specifies user name you established during Microsoft SQL Server database installation. For example, **sa**. This is the same value that you use for the EX\_ATTRIB\_USER MSI Property.

## IS\_SQLSERVER\_PASSWORD

Specifies the password established during Microsoft SQL Server database installation. This is the same value that you use for the EX\_ATTRIB\_PASS MSI Property

## TRANSFORMSSECURE

Informs the installer that transforms are to be cached locally on the user's computer in a location where the user does not have write access. This parameter should be set to 1 if you want to provide secure transform storage with traveling users of Windows 2000.

## WT\_OWNERNAME

Specifies the WebTrends owner. Use **AdminDomain\AdminUsername** for this value. You also specify these values when you install the Management Application and the UI Server Mods. These values need to match the values specified in ADMI NUSER\_DOMAI N and ADMI NUSER\_USERNAME. For more information, see [“Identifying MSI Properties and Their Command Line Parameters” on page 47](#).

## Identifying MSI Properties and Their Command Line Parameters

You can use MSI properties to identify the parameters needed to install MSI files from the command line. For more information, see [“Examples for Installing WebTrends Components” on page 50](#) .



### Note

Be sure to replace the italicized items in the following table with your custom information.

## Installing Individual Subcomponents

You can install individual subcomponents for the UI Server, Analysis Engine, and Marketing Warehouse by setting the ADDLOCAL property. The ADDLOCAL property lists subcomponents separated by commas in a command-line installation, and are to be installed locally. If you want to install all of the subcomponents, you need to set the ADDLOCAL property to ALL by specifying

ADDLOCAL=ALL



in the installation command text shown in the examples at the end of this chapter. For more information, see “UI Server/Main Application Installation Example” on page 51.

## Applying ADDLOCAL Rules

The following rules apply to the ADDLOCAL Property.

- Subcomponents are case sensitive.
- Specify multiple features using a comma-delimited list with no spaces between the subcomponents.
- *If an MSI file requires an ADDLOCAL property when you install the file, you need to specify ADDLOCAL=ALL, unless the MSI file installs subcomponents.*

## Specifying UI Server Subcomponents (WebTrends Main Application.msi)

You can specify one or more of the following subcomponents for the UI Server:

- MC
- ODRC
- RDE
- Tools
- Mail

The following code sample shows all five of these options:

```
ADDLOCAL=MC, ODRC, RDE, Tool s, Mai l
```

The following table shows more information about the UI Server Subcomponents.

Property Value	Display Name	Description
MC	UI Server	Installs the UI Server component that serves reports to WebTrends users.
ODRC	Report Cache Server	Installs the UI Server component that stores reports requested by WebTrends users.
RDE	Report Exporter	Installs the UI Server component that converts HTML reports to other formats.
Tool s	Tools	Installs Administrative tools including utilities for backing up WebTrends and cleaning up deleted profiles.
Mai l	Email Notification Service	Installs the Email Notification Service.



## Specifying Analysis Engine Subcomponents (WebTrends Analysis.msi)

You can specify one or more of the following subcomponents for the Analysis Engine:

- ASU
- Realtime

The following code sample shows both of these options:

```
ADDLOCAL=ASU, Real time
```

The following table shows more information about the Analysis Engine Subcomponents.

Property Value	Display Name	Description
ASU	Standard Analysis Engine	Installs the Standard Analysis Engine.
Real time	Express Analysis Engine	Installs the Express Analysis Engine.

## Specifying Marketing Warehouse Subcomponents (WebTrends Marketing Warehouse.msi)

You can specify one or more of the following subcomponents for Marketing Warehouse:

- EventDBServerLoader
- EventDBSplitter
- ExtendedAttribDB

The following code sample shows all three of these options:

```
ADDLOCAL=EventDBServerLoader, EventDBSplitter, ExtendedAttribDB
```

The following table shows more information about the Marketing Warehouse Subcomponents.

Property Value	Display Name	Description
EventDBServerLoader	Event Database Server/Loader	Analyzes SmartSource Data Collector log files and moves processed data into the Marketing Warehouse.
EventDBSplitter	Log File Management	Installs the Event Database Splitter, which is required for the WebTrends Marketing Warehouse.
ExtendedAttribDB	Configuration Databases	Installs the Extended Attributes database, which is required for the WebTrends Marketing Warehouse. Make sure you only install one instance of the Extended Attributes Database per distributed architecture.



## Installing Subcomponents After You Install the Component

You can install subcomponents after the component has been installed by installing the component and specifying the new subcomponent as a parameter for ADDLOCAL. For example, if you want to install the Report Cache Server after you have installed the UI Server, you can type

```
ADDLOCAL=ODRC
```

in the installation command text shown in the examples at the end of this chapter. For more information, see “UI Server/Main Application Installation Example” on page 51.

## Examples for Installing WebTrends Components

Use the examples in this section to install WebTrends Analytics components in a distributed configuration. You need to install the System Database component first, followed by the Marketing Warehouse Configuration database if you are licensed to install Marketing Warehouse.



### Note

In the following examples, E: \ represents the disk drive that contains your WebTrends Analytics installation disk and C: \ represents the drive on your computer where you are installing WebTrends Analytics.

### System Database Installation Example

The following example shows the full command text for a Database installation:

```
msiexec /i "E:\MSI\WebTrends System Database.msi" /qb! /I *v "C:\Program Files\WebTrends\logs\install\WebTrends System Database.log" INSTALLDIR="C:\Program Files\WebTrends" MASTER_USER=sa SCHED_USER=sa MASTER_PASS=abcd SCHED_PASS=abcd MASTER_HOST=myComputer SCHED_HOST=myComputer DB_TYPE=SQLServer SERVICEACCOUNT=myDomain\myUser SERVICEPASSWORD=myPassword
```

### Common/Scheduler Agent Installation Example

The following example shows the full command text for a Common/Scheduler Agent installation:

```
msiexec /i "E:\MSI\WebTrends Scheduler Agent.msi" /qb! /I *v "C:\Program Files\WebTrends\logs\install\WebTrends Scheduler Agent.log" INSTALLDIR="C:\Program Files\WebTrends" MASTER_USER=dbUsername MASTER_PASS=dbPassword MASTER_HOST=myComputer DB_TYPE=SQLServer SERVICEACCOUNT=myDomain\myUser SERVICEPASSWORD=myPassword ADDLOCAL=ALL
```

### WebTrends System Repository Installation Example

The following example shows the full command text for a WebTrends System Repository installation:

```
msiexec /i "E:\MSI\WebTrends System Repository.msi" /qb! /I *v "C:\Program Files\WebTrends\logs\install\WebTrends System Repository.log" INSTALLDIR="C:\Program Files\WebTrends" MASTER_USER=dbUsername MASTER_PASS=dbPassword MASTER_HOST=myComputer DB_TYPE=SQLServer REPORTS_DIR="\\myComputer\myShare\reports" FASTTRENDS_DIR="\\myComputer\myShare\analysis" CONFIG_DIR="\\myComputer\myShare\config" BACKUP_DIR="\\myComputer\myShare\backup" ADDLOCAL=ALL
```



## Analysis Engine Installation Example

The following example shows the full command text for an Analysis Engine installation:

```
msiexec /i "E:\MSI\WebTrends Analysis.msi" /qb! /I *v "C:\Program
Files\WebTrends\logs\install\WebTrends Analysis.log" INSTALLDIR="C:\Program
Files\WebTrends" MASTER_USER=dbUsername SCHED_USER=dbUsername MASTER_PASS=dbPassword
SCHED_PASS=dbPassword MASTER_HOST=myComputer SCHED_HOST=myComputer DB_TYPE=SQLServer
SERVICEACCT=myDomain\myUser SERVICEPASS=myPassword
RT_ANALSRC="\myComputer\myShare\expressanalysis\src"
RT_REPORTS="\myComputer\myShare\expressanalysis\reports" ADDLOCAL=ASU RealTime
```

## Marketing Warehouse Installation Example

The following example shows the full command text for a Marketing Warehouse installation:

```
msiexec /i "E:\MSI\WebTrends Marketing Warehouse.msi" /qb! /I *v "C:\Program
Files\WebTrends\logs\install\WebTrends Marketing Warehouse.log" INSTALLDIR="C:\Program
Files\WebTrends" MASTER_USER=dbUser SCHED_USER=dbUser MASTER_PASS=dbPassword
SCHED_PASS=dbPassword MASTER_HOST=myComputer SCHED_HOST=myComputer DB_TYPE=SQLServer
SERVICEACCT=myDomain\myUser SERVICEPASS=myPassword EX_ATTRIBUTES_HOST=myComputer
EX_ATTRIBUTES_USER=myUser EX_ATTRIBUTES_PASS=myPassword EX_ATTRIBUTES_DATADIR="d:\my MDF data
path" EX_ATTRIBUTES_LOGDIR="d:\my LDF log path" ADDLOCAL=ALL
```

## GeoTrends Installation Example

The following example shows the full command text for a GeoTrends installation:

```
msiexec /i "E:\MSI\WebTrends GeoTrends.msi" /qb! /I *v "C:\Program
Files\WebTrends\logs\install\WebTrends GeoTrends.log" INSTALLDIR="C:\Program
Files\WebTrends" MASTER_USER=dbUsername SCHED_USER=dbUsername MASTER_PASS=dbPassword
SCHED_PASS=dbPassword MASTER_HOST=myComputer SCHED_HOST=myComputer DB_TYPE=SQLServer
GEOTRENDS_DAT="d:\Path\GeoTrends.dat" ADDLOCAL=ALL
```

## UI Server/Main Application Installation Example

The following example shows the full command text for a UI Server installation:

```
msiexec /i "E:\MSI\WebTrends Main Application.msi" /qb! /I *v "C:\Program
Files\WebTrends\logs\install\WebTrends Main Application.log" INSTALLDIR="C:\Program
Files\WebTrends" MASTER_USER=dbUsername MASTER_PASS=dbPassword MASTER_HOST=myComputer
DB_TYPE=SQLServer SERVICEACCT=myDomain\myUser SERVICEPASS=myPassword UI_PORT=80
UI_DESC="My Description" UI_URL="https://myComputer:80" IPADDRESS=10.10.10.10
ADDLOCAL=ALL
```

## Management Application Installation Example

The following example shows the full command text for a Management Application installation:

```
msiexec /i "E:\MSI\WebTrends Management Application.msi" /qb! /I *v "C:\Program
Files\WebTrends\logs\install\Management Application.log" INSTALLDIR="C:\Program
Files\WebTrends\ " ADMINUSER_DOMAIN=MyDomain ADMINUSER_USERNAME=AdminUsername
ADMINUSER_PASSWORD=AdminPassword ADDLOCAL=ALL
```



## Visitor Intelligence Installation Example

The following example shows the full command text for a Visitor Intelligence installation:

```
msiexec /i "E:\MSI\WebTrends\Visitor Intelligence.msi" TRANSFORMS="E:\MSI\WebTrends\Visitor Intelligence.mst" /qb! /I *v "C:\Program Files\WebTrends\logs\install\WebTrends Visitor Intelligence.log" INSTALLDIR="C:\Program Files\WebTrends\common\ui server\WHReporting\ " WT_OWNERNAM=AdminDomain\AdminUsername IIS_PORT_NUMBER=7099 IIS_SQLSERVER_USERNAME=sql Username IIS_SQLSERVER_PASSWORD=Sql Password ARPSYSTEMCOMPONENT=1 APPPOOL="VisitorIntelligenceAppPool" IIS_SQLSERVER_SERVER=sql Servername IIS_SQLSERVER_DATABASE=wtWHReporting IIS_VRDIR=WHReporting USEBOSTORE=0 TRANSFORMSSECURE=1 ADDLOCAL=ALL
```

### Note



The Visitor Intelligence MSI properties are not included in the MSI properties section in this chapter. For more information about these properties, see [“Identifying MSI Properties for Visitor Intelligence”](#) on page 46.

## UI Server Modules Installation Example

The following example shows the full command text for a UI Server Modules installation:

```
msiexec /i "E:\MSI\WebTrends\UI Server Modules.msi" /qb! /I *v "C:\Program Files\WebTrends\logs\install\UI Server Modules.log" INSTALLDIR="C:\Program Files\WebTrends\" ADMINUSER_DOMAIN=MyDomain ADMINUSER_USERNAME=AdminUsername ADMINUSER_PASSWORD=AdminPassword ADDLOCAL=ALL
```

## Performing an Unattended Uninstallation

When you perform an unattended uninstallation of WebTrends Analytics 8.1 or later, you must uninstall the components in a certain order. If you no longer have the WebTrends Analytics installation DVD-ROM, please contact support for the MSI files or product codes that you can use when uninstalling.

### Note



You must always uninstall the Common/Scheduler Agent component last.

### To uninstall WebTrends Analytics:

1. Uninstall UI Server Modules.

For example, type the following at a command prompt:

```
msiexec /x "E:\MSI\WebTrends\UI Server Modules.msi" /qb! /I *v "C:\Program Files\WebTrends\logs\uninstall\WebTrends UI Server Modules Uninstall.log"
```

2. Uninstall Visitor Intelligence

For example, type the following at a command prompt:



```
msi exec /x "E:\MSI\WebTrends Visitor Intelligence.msi" /qb! /I *v "C:\Program Files\WebTrends\logs\uninstall\WebTrends Visitor Intelligence Uninstall.log"
```

### 3. Uninstall Management Application

For example, type the following at a command prompt:

```
msi exec /x "E:\MSI\WebTrends Management Application.msi" /qb! /I *v "C:\Program Files\WebTrends\logs\uninstall\WebTrends Management Application Uninstall.log"
```

### 4. Uninstall UI Server/Main Application

For example, type the following at a command prompt:

```
msi exec /x "E:\MSI\WebTrends Main Application.msi" /qb! /I *v "C:\Program Files\WebTrends\logs\uninstall\WebTrends Main Application Uninstall.log"
```

### 5. Uninstall GeoTrends

For example, type the following at a command prompt:

```
msi exec /x "E:\MSI\WebTrends GeoTrends.msi" /qb! /I *v "C:\Program Files\WebTrends\logs\uninstall\WebTrends GeoTrends Uninstall.log"
```

### 6. Uninstall Marketing Warehouse

For example, type the following at a command prompt:

```
msi exec /x "E:\MSI\WebTrends Marketing Warehouse.msi" /qb! /I *v "C:\Program Files\WebTrends\logs\uninstall\WebTrends Marketing Warehouse.log"
```

### 7. Uninstall Analysis Engine

For example, type the following at a command prompt:

```
msi exec /x "E:\MSI\WebTrends Analysis.msi" /qb! /I *v "C:\Program Files\WebTrends\logs\uninstall\WebTrends Analysis.log"
```

### 8. Uninstall WebTrends System Repository

For example, type the following at a command prompt:

```
msi exec /x "E:\MSI\WebTrends System Repository.msi" /qb! /I *v "C:\Program Files\WebTrends\logs\uninstall\WebTrends System Repository.log"
```

### 9. Uninstall System Database

For example, type the following at a command prompt:

```
msi exec /x "E:\MSI\WebTrends System Database.msi" /qb! /I *v "C:\Program Files\WebTrends\logs\uninstall\WebTrends System Database Uninstall.log"  
DROP_MASTER_DB=TRUE
```



#### Note

If you set DROP\_MASTER\_DB to True, all WebTrends system databases will be dropped.

### 10. Uninstall Scheduler Agent

For example, type the following at a command prompt:

```
msi exec /x "E:\MSI\WebTrends Scheduler Agent.msi" /qb! /I *v "C:\Program Files\WebTrends\logs\WebTrends Scheduler Agent.log"
```





---

## Chapter 6

# Optimizing Your Database Environment

This chapter discusses ways to increase efficiency in the Marketing Warehouse database environment, and recommends settings for the SQL Server configuration that may improve performance, and tuning information for the Marketing Warehouse Loader.

## Understanding Database Partitions and File Groups

WebTrends leverages table partitioning to enhance performance. In large databases and data warehouses, managing subsets of data and indexes increases efficiency, because operating on subsets of data is faster than operating on entire tables.

In SQL Server 2008, the benefits of partitioning are increased because all available threads are allocated to each partition relevant to a query, in a round robin fashion. In SQL 2005, if only one partition happened to be involved in a query, all threads were allocated to that query. However, if multiple partitions were involved in a query, only one thread per partition was allocated. This meant that, for a comparison of two types of data that happened to be in different partitions, performance would degrade (with the cause not apparent to the user).

Figure 1, Figure 2, and Figure 3 show the organization of file groups and data files in the Marketing Warehouse databases. All partitioned tables are aligned, that is, the data and indexes are located within the same file group or partition scheme.

WebTrends leverages SQL Server 2008 table partitioning in the Event Database. All events are grouped into monthly filegroups, with a default retention of 13 months. Events (visit, search, etc.) are partitioned on daily, weekly, or other intervals. You can modify this setting per event. Regardless of partition size, all events for a given month are stored in one monthly filegroup, in a single file. Additional files can be added if there is a need to minimize disk I/O. The number and location of files varies depending on the environment, as shown in the figures.

For more information about the partitioning strategy and how it can improve data access, see the Marketing Warehouse Schema Reference.



Figure 1: Event Database File Group and Data File Layout

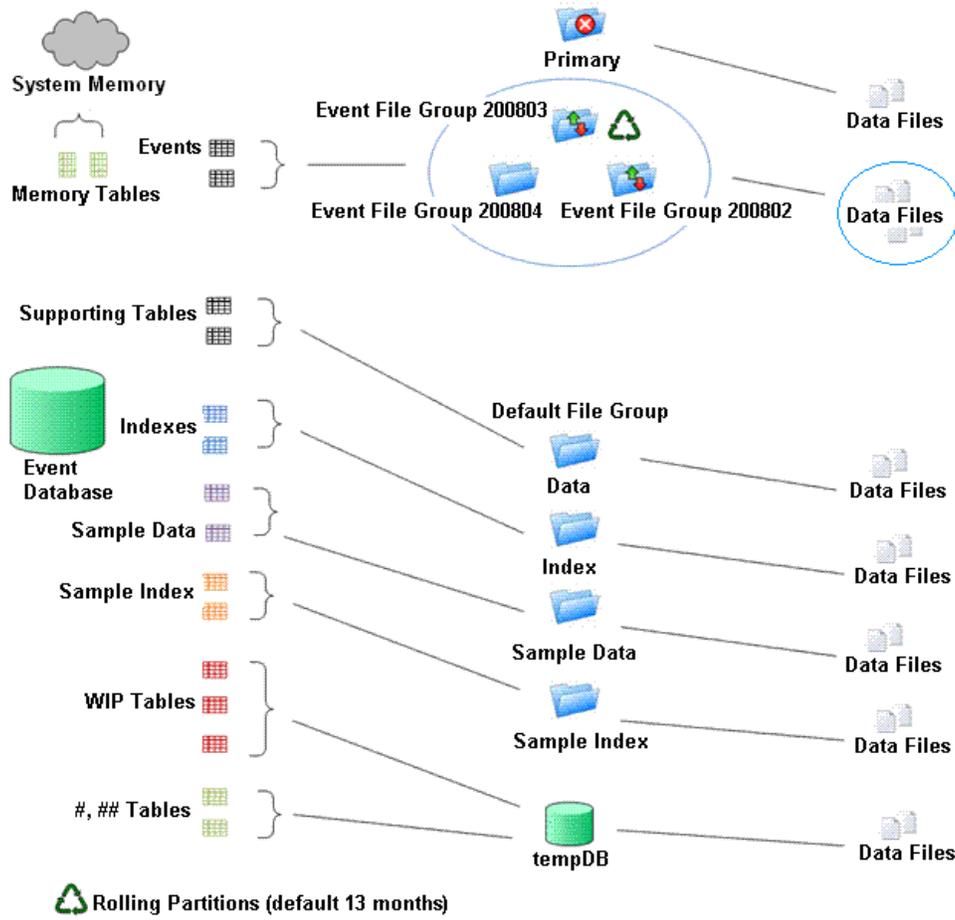
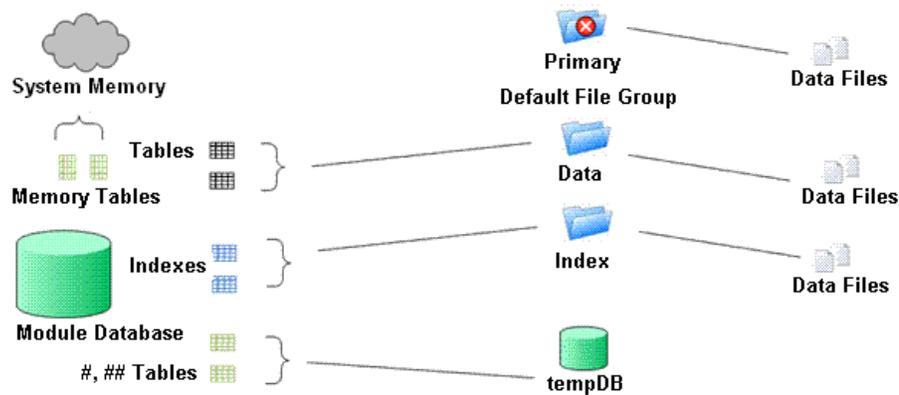
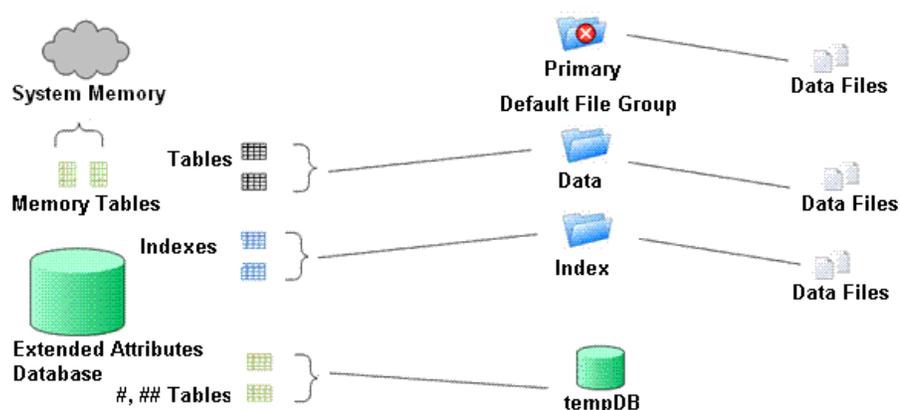


Figure 2: Module Database File Group and Data File Layout



**Figure 3: Extended Attributes Database File Group and Data File Layout**

## Database Maintenance

Due to the size and loading requirements of the Marketing Warehouse data warehouse, regular database maintenance is required to ensure optimal performance. In addition to regular backups, you should:

- Re-index all tables weekly: Because there are frequent deletes, inserts and updates (depending on whether the index is cluster or non-cluster), indexes are not optimized. Over time, indexes become fragmented, which degrades disk I/O performance. Regular index rebuilds ensure continuity and reduce the number of reads.
- Update statistics twice weekly: Updating statistics helps the engine generate efficient execution plans. Based on the number of rows in the statistics, the engine can determine whether to do a full table scan, or use indexes. If statistics are stale, the engine may decide to do a full table scan, based on information that there are only a thousand rows when there are a million.

## Recommendations for Bulk Extracts

When there is a need to share data between Marketing Warehouse and other systems, you have several options, depending on the intended use of the extracted data: direct access, Extract, Transform, Load (ETL), and instance replication.

To determine the appropriate approach, consider the following:

- Size and volume: how much data needs to be extracted?
- Timing: how long does it take to extract the data? How frequently is it refresh? Does it need to be real-time?
- Usage: will the extracted data need to be transformed, or joined to external tables? What events are required?

You should use dirty reads with direct access and ETL to prevent contention with the loader.



## Direct Access

For live data and small amounts of data, use Data Manipulation Language (DML) statements against the databases. With direct access, you can query the Event Database directly using straight SQL commands to retrieve data of interest, but you cannot create additional objects in the Event Database, so Data Definition Language (DDL) is not supported. If you need to create temporary working tables or customized data structures, the Custom schema is available for that purpose, but note that changes made to the Custom schema may not be preserved in successive releases. A number of providers (OLE DB, ODBC, JDBC, ADO DB) can be used to access WebTrends data. An entity-relationship diagram is available in the Marketing Warehouse Schema Reference.

## Extract, Transform, Load (ETL)

For large volumes of data, use a commercially available or in-house ETL tool, and schedule batch jobs during off-peak hours to minimize resource contention and locking.

## Load Process Tuning

You can modify load processes to improve performance by changing parameters in the Marketing Warehouse Loader, or by eliminating unnecessary processes.

## Tuning the Marketing Warehouse Loader

The default parameter settings are intended to be suitable for a wide variety of customer environments, but in you can alter these settings to achieve performance gains according to specific load characteristics.

To increase the throughput of the loader, you can modify the `ThrottledMaximumDataMinutes` parameter in the `eventdb.ini` file as follows, according to the load:

```
ThrottledMaximumDataMinutes=180
```

The density of the log files determines the correct value. For smaller profiles (that is, a lower-density log file), this value will be higher than for large profiles (higher-density logs). To determine the maximum data minutes to load, use a trial and error approach, slowly increasing the value until you see an "out of memory" error in the user interface. After inducing this error, decrease the `ThrottledMaximumDataMinutes` value until the error disappears.

If the traffic varies from day to day, then this setting should account for the variance and be set to accommodate the higher traffic days. For example, if a site has more hits or events on weekends than on weekdays, then set this value accordingly.

The loader is a 32-bit application. We recommend that it run on a 64-bit operating system due to memory allocation. On a 64 bit operating system, you can observe memory usage to find the optimal number of minutes to process for each batch. As loading occurs, the memory will grow until it reaches 4GB. Ideally, the memory consumption will not exceed 4GB. When it reaches this limit, the application will display an "out of memory" error and attempt to restart loading (no data loss occurs).

For more information on the memory limitation, see:

[http://msdn.microsoft.com/en-gb/library/aa366778.aspx#physical\\_memory\\_limits\\_windows\\_server\\_2008](http://msdn.microsoft.com/en-gb/library/aa366778.aspx#physical_memory_limits_windows_server_2008)



## Disabling Sampling

The sampling engine runs as a post-process event from the loader. If you are not using Explore, then the extra storage and processing time spent collecting sample data is unnecessary overhead. Sampling is enabled by default (`IsServiceEnabled=1`), but you can disable it by updating the `wt_Sched` database directly with these queries:

To disable sampling for all profiles:

```
update wt_profileaddonmodule set IsServiceEnabled=0
```

To disable sampling for a single profile:

```
update wt_profileaddonmodule set IsServiceEnabled=0 where profileID={profile ID}
```

The initial row in the `wt_profileaddonmodule` is created when a new profile is established. After profile creation, the sampling row is never removed or updated when a profile is edited. If the row is manually updated, the update remains in place unless the profile is deleted.

## Instance Replication and Cloning

If a replica of an instance is required for sharing, use SAN snapshots, cloning, or BCV. You should limit any down time to one hour, and schedule it in off-peak hours.

# Optimizing Marketing Warehouse with SQL Server Configuration

You can use these SQL Server 2008 features to improve performance and increase efficiency in managing Marketing Warehouse environments.

**Implement these according to your needs, following Microsoft best practices:**

- Backup compression - Backup speed is increased, thereby reducing device I/O.
- Policy-based management - Policies are defined and enforced across the enterprise (for example, `AutoShrink=False`).
- Central management servers - Enables central storage of server groups and registered servers, providing easier access to the registration properties of a group of servers.
- Data collector - Provides a single point for data collection across database servers and applications, and supports dynamic tuning for data collection.
- Resource Governor - Manages workload and resource usage, providing the ability to limit CPU and memory usage per application request.

**These features are best implemented with WebTrends professional services engagement and validation:**

- Database mirroring - Compresses the outgoing log stream from the principal database to the mirror database, thereby minimizing the network bandwidth used by database mirroring.
- Data compression - All fixed length data types are stored in variable length format, but the semantics of the data type remains unchanged; that is, from the perspective of the application it is still the fixed length data type. Data redundancy is minimized in columns in one or more rows on a given page, because redundant data is stored once on the page and then referenced from the multiple columns.
- Log shipping - Enables shipping of transaction logs to other servers.



- Failover clustering - Provides high-availability support for an entire SQL Server instance. Prerequisite: Windows Server failover cluster.
- Peer-to-peer replication - Every database is a member of a publish-subscribe model, which improves performance and provides failover for all databases.
- Scalable shared databases (SSD) - Employs a read-only database that can be scaled; useful for offloading reporting, or ad-hoc, read-only queries by operations staff. Requires Enterprise Edition.

## Recommended Settings for SQL Server Configuration

You can improve the performance of Marketing Warehouse by tuning these SQL Server parameters in response to processor performance degradation, insufficient memory, or disk I/O or network issues.

You should not change Default Connection Options. If you change these settings, WebTrends may produce incorrect data. Although WebTrends overwrites incorrect settings at the session level for functions, procedures, views, and so on, changing this setting may still affect the results.

### Processor Performance Settings

**Affinity Mask:** The affinity mask determines on which processor (core) SQL Server spawns threads. In a dedicated instance, the setting should be 0. If the server hosts multiple instances, use this setting to indicate which instance uses a particular processor (core).

**Affinity I/O Mask:** The affinity I/O mask is similar to the affinity mask. Set it only if needed for servers hosting multiple instances.

**Max Degree of Parallelism:** WebTrends recommends setting this to 0, which indicates that the engine will detect which processors are busy and specify the degree of parallelism.

### Memory Utilization Settings

**AWE Enabled:** (from existing internal doc--needs rewrite) WebTrends recommends MS SQL Server 2005 Enterprise (x64) whenever possible. AWE does not apply to the 64-bit version and it should not be checked. AWE is used to extend the SQL Server buffer pool beyond the virtual address space of 3GB when 3GB is enabled in the boot.ini file (32-bit limitation). SQL Server uses a buffer pool to cache data to avoid going to disk. A 32-bit SQL Server does not go beyond 3 GB. AWE enables it to use the rest of the memory. Keep in mind that SQL Server takes longer to start because it pre-allocated this memory when AWE is enabled.

**Max Server Memory/Min Server Memory:** For a dedicated host, this value should be set to 80 /20 or 90/10, depending on the amount of physical memory (the default is 2GB and 0). The more memory allocated, the larger the buffer pool. With a larger buffer pool, SQL Server does not have to access the disk for data and performs much faster.



---

## Chapter 7

# Client-Side JavaScript Integration

WebTrends On Demand and SmartSource Data Collector (SDC) use a special JavaScript tag that you place on your web pages to collect activity. When a visitor accesses a page from your web site, this JavaScript tag initiates interactions between the visitor's browser and either WebTrends On Demand (for service only) or SDC (for software). The tag collects data about the visitor's browser and activity and transmits this information to WebTrends On Demand or SDC. For WebTrends Analytics reports, an analysis engine aggregates this data with that of all other visitors to your site, stores it, and makes it available to you in reports that you can view and download. For WebTrends Marketing Warehouse, an analysis engine analyzes this data, which is stored in the Marketing Warehouse and made available to you in WebTrends Explore and WebTrends Visitor Intelligence.

This chapter explains how client-side data collection works, and describes how to customize your WebTrends JavaScript tag using META tags to collect the data that interests you.

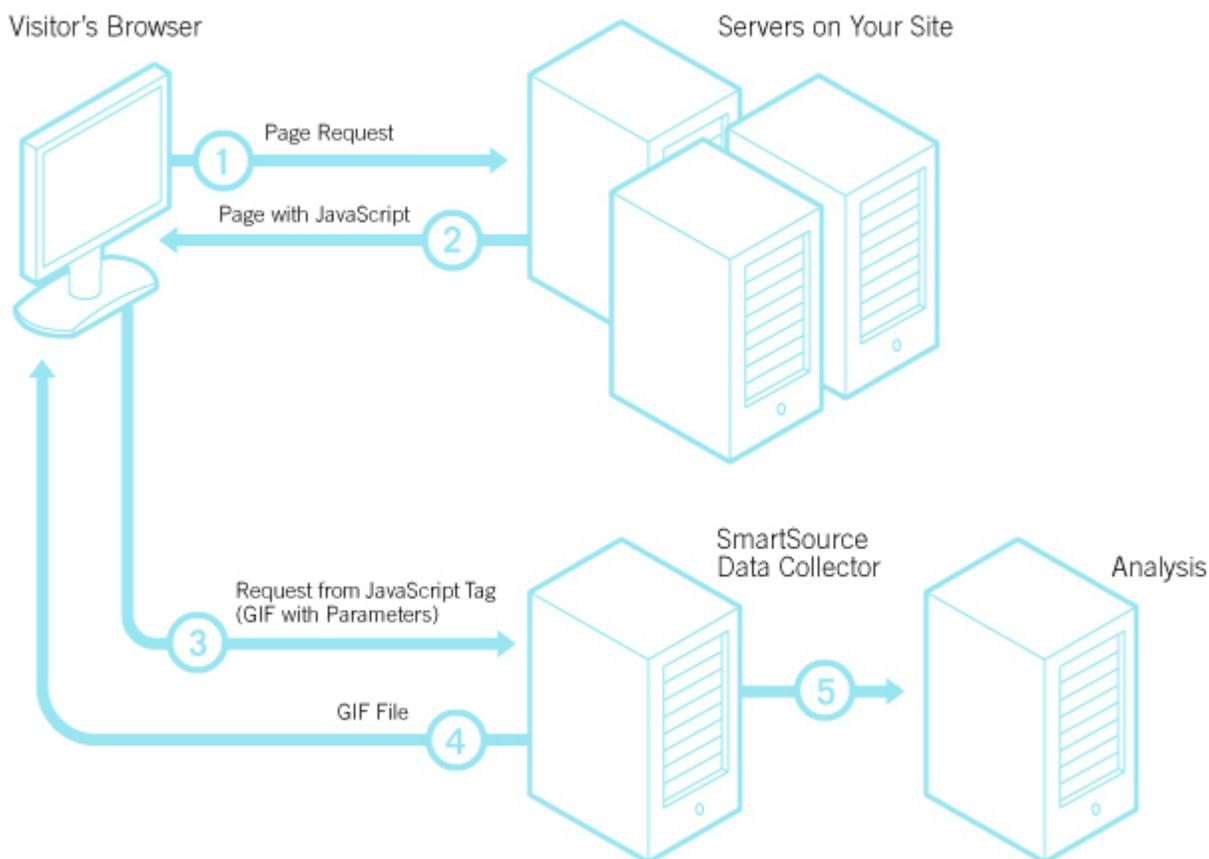
## Interactions Between the Client Browser and SDC

After deploying the WebTrends JavaScript tag on your web pages, you are set up for a series of interactions between your visitors, your web site, and SmartSource Data Collector. The first interaction is between your visitors and your web site.

When a visitor accesses a page from your web site, the JavaScript tag initiates an interaction between the visitor's browser and sends that data to the SmartSource Data Collector (SDC), which WebTrends On Demand uses to collect web activity data. The data is analyzed, stored, and made available to you in your WebTrends Analytics reports, WebTrends Explore, or WebTrends Visitor Intelligence.



The following illustration shows an overview of the interaction process.



**The following interactions take place:**

1. A visitor wants to view a page on your site. This initiates a page request to your web server.
2. Your web server sends the page to the visitor that contains your WebTrends JavaScript tag.
3. The JavaScript tag triggers a request for a GIF file with additional tagging parameters and cookie attached. This image request is sent to the SmartSource Data Collector or WebTrends On Demand.
4. The GIF file and cookie is sent to the visitor.
5. The image request with the parameters is collected and analyzed.

All interactions between your visitors and WebTrends take place at the browser on the client side (your visitor's side). With client-side integration, there is no interaction between WebTrends On Demand and your web servers.



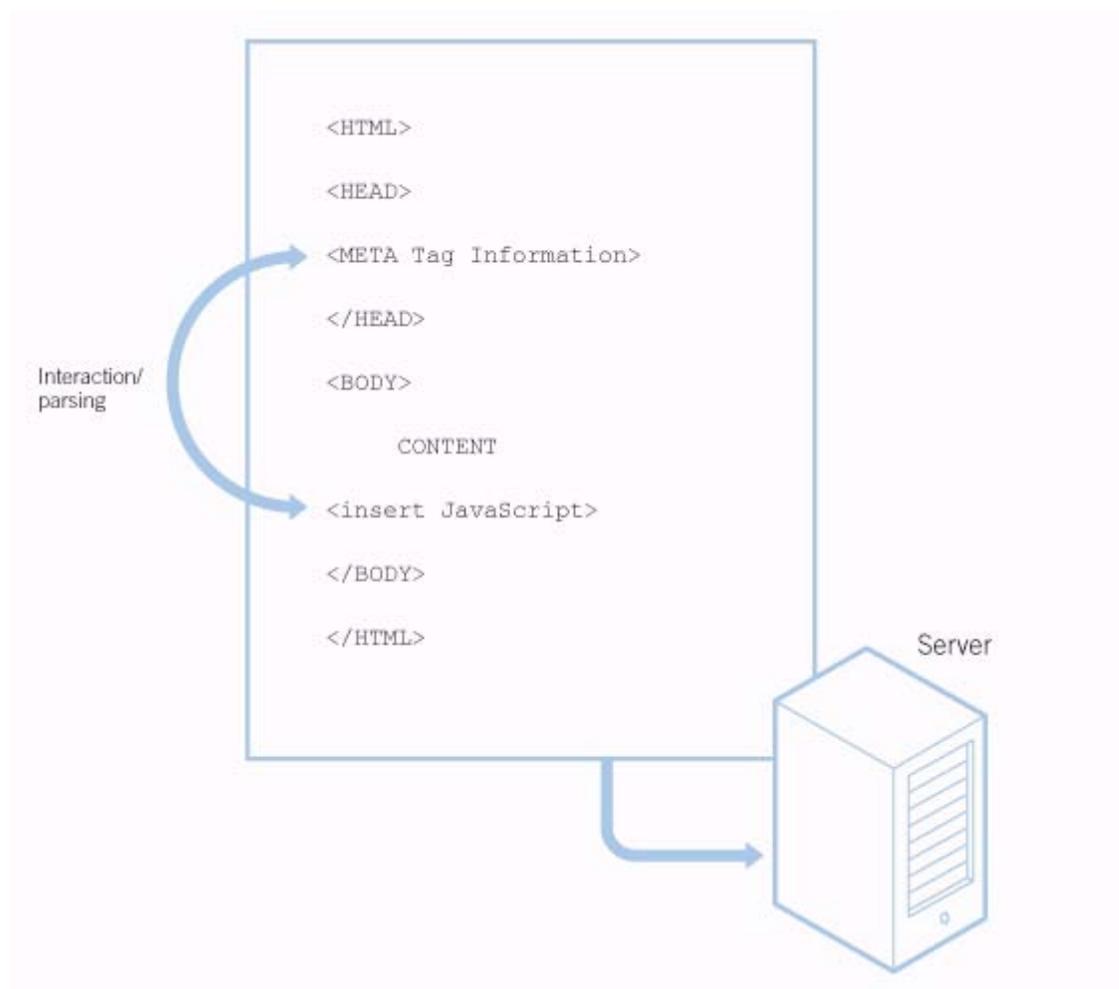
## Customizing Your META Tags

After you put the WebTrends JavaScript tag on your web page, you may also modify the META tags on your web pages because the tag interacts with the META tags and stores the resulting information in SmartSource Data Collector log files.

The following sections discuss all the META tags that interact with the JavaScript tag. Note that the META tags do not have to appear in a particular order in the web page. However, they are presented here in an order that helps you more readily understand the structure of your web page.

WebTrends offers one of a few data tagging implementations that segregate page-specific information from the main script, maximizing code modularity and reuse.

The following illustration provides an overview of the JavaScript tag and META tag interaction.



## General META Tag Information

The general syntax of the supported META tag is as follows:

```
<META NAME="name" CONTENT="content">
```



Include the META tag between the <HEAD> and </HEAD> tags.

The *name* represents the parameter name.

The *content* represents the parameter value.

Many META tags support more than one parameter. Separate multiple parameters by a semicolon “;” as shown in the following example:

```
<META NAME="name" CONTENT="content1; content2; content3. . .">
```

## META Tag Descriptions

The following subsections discuss the META tags that may be needed to add or modify on your web pages.

Note that all of the META tags that you use begin with WT., for example, WT. cg\_n, which you can use to track content groups.

## Tracking Content Groups

WebTrends can report on visitors according to the content group and content subgroup of the pages they visit. To do so, your site must capture the content group and subgroup in a parameter when the page is visited, and then pass the parameter values to WebTrends. Content subgroups are optional.

The following META tags track content groups:

```
<META NAME="WT. cg_n" CONTENT="Name">
```

Defines the name of the content group.

```
<META NAME="WT. cg_s" CONTENT="subName">
```

Defines the name of the content subgroup. This tag is optional.

### Sample: Single Content Group and Subgroup

A university web site reports the number of visitors to its pages. The web site denotes the content group DegreeReq for each page that describes the requirements. The web site then assigns a subgroup designation for each page about a particular field of study, for example, Lit for Literature.

Using the WT. cg\_n and WT. cg\_s tags, your META tag would look like this:

```
<META NAME="WT. cg_n" CONTENT="DegreeReq">
```

```
<META NAME="WT. cg_s" CONTENT="Li t">
```

### Example: Multiple Content Groups

Building on the single content group example, the Admissions Department is also interested in visitors to these pages. An additional group can be added so that the pages are reported for both content groups. Note that Math has been added as a subgroup. Multiple content groups and subgroups are separated by semi-colons.

```
<META NAME="WT. cg_n" CONTENT="DegreeReq; Admi ssi ons">
```

```
<META NAME="WT. cg_s" CONTENT="Li t; Math">
```



## Sample Log File

After the JavaScript interacts with this META tag information, it generates logs that look something like this:

```
2008-10-24 00:08:18 proxy7.hotmail.com - W3SVC3 web1 192.168.1.1 GET /ads/default.asp
redir=products&ad=http%3A//
www.bi.guiversity.edu&WT.cg_n=DegreeReq;Admissions&WT.cg_s=Lit;Math...
```

The italicized text represents the META-related order information captured by the JavaScript and placed in the log file.

## Tracking Servers

If your site is hosted on multiple servers, a server cluster, or a server farm, and you want to evaluate the performance of your load balancer, WebTrends can track page views for each server. To do so, populate the following META tag on all pages on each server:

```
<META NAME="WT.sv" CONTENT="name">
```

Defines the name of the machine that serves the web page.

If you have two servers (Server1 and Server2), you would make two copies of the META tag and designate CONTENT="Server1" for deployment to pages on the first server and CONTENT="Server2" for deployment to the same pages on the second server.

For a server farm, you can extract the value of the built-in server name and dynamically assign it to the META tag using server-side scripting.

## Sample Log File

After the JavaScript interacts with this META tag information, it generates log files that look something like this:

```
2007-03-04 00:08:18 proxy7.hotmail.com - W3SVC3 web1 192.168.1.1 GET /ads/default.asp
redir=products&ad=http%3A//www.phonedea.com&WT.sv=Server1...
```

The italicized text represents the META-related order information captured by the JavaScript and placed in the log file.

## Tracking Marketing Campaigns

WebTrends can report visitor activity that relates to a marketing campaign. You can place the following META tags on the landing page to identify the name of the campaign and the type of campaign.

```
<META NAME="WT.mc_id" CONTENT="Campaign ID">
```

Identifies the ID of the marketing campaign.

## Landing Page

The landing page is the first page that visitors see when they visit your site. Normally, this is your home page, but for effective marketing campaign tracking, you can bring visitors to a page exclusively used for your marketing campaign.



## Example Marketing Campaign

To attract new students, a university launches a marketing campaign by sending recruitment email to all graduating high school seniors in a metropolitan area. The email links to a special landing page in the university's web site, containing the following META tag to track marketing campaigns.

```
<META NAME="WT. mc_id" CONTENT="1X2GG34">
```

The Campaign ID 1X2GG34 represents recruits to be contacted by email.

## Sample Log File

After the JavaScript interacts with this META tag information, SDC generates log files that look something like this:

```
2007-03-04 00:08:18 proxy7.hotmail.com - W3SVC3 web1 192.168.1.1 GET /ads/default.asp
referrer=products&ad=http%3A//www.biuniversity.edu&WT.mc_id=1X2GG34 . . . .
```

The italicized text represents the META-related order information captured by the JavaScript and placed in the log file.

## Tracking Profile/Subprofile Generation

If you are using parent/child profiles, the split profile META tag allows you to identify the pages that are associated with each child profile. In addition to putting this META tag on your web pages, you also create a parent profile in the UI that specifies the WT.sp parameter as well as the values that identify your child profiles. The child profile values that you specify should match the values of the CONTENT= string. Use the following META tag to track child profiles.

```
<META NAME="WT.sp" CONTENT="profile name">
```

Defines the identification string for creating the child profile, and is used to track child profile activity. For more information about parent/child profiles, see the Administration Help. For more information about parent/child profiles, see the Administration Help.

## Example Tracking Profile/Subprofile Generation

Suppose you are an Internet Service Provider and you collect web traffic data for three different customers whose account numbers are: 11111, 22222, 33333. You can use Parent/Child profiles to split-out traffic based on a WebTrends Query Parameter WT.sp.

```
<META NAME="WT.sp" CONTENT="11111">
```

The child profile name 11111 will be created automatically by the WebTrends Analytics. Traffic from pages containing this META tag will be directed into this profile.

## Sample Log File

After the JavaScript interacts with this META tag information, it generates log files that look something like this:

```
2007-03-04 00:08:18 proxy7.hotmail.com - W3SVC3 web1 192.168.1.1 GET /ads/default.asp
referrer=products&ad=http%3A//www.phonedeeper.com&WT.sp=Wireless%20Phones . . .
```



The italicized text represents the META-related order information captured by the JavaScript and placed in the log file.

## Tracking Revenue

To track Commerce revenue, configure your web site to populate the META tags on your confirmation pages. Your web site captures transaction information, typically using an order form. Configure your site to pass the values from the form to a META tag so that WebTrends can track the transactions, aggregate them, and include them in your reports.

To track revenue, include the following META tags:

**<META NAME="WT. pn\_sku" CONTENT="ProductSKU">**

Identifies the SKU of the product. Use semicolons to pass multiple SKUs for the order.



### Note

WT. pn has been replaced with WT. pn\_sku. WT. pn can still be used, but it does not work with product SKUs.

**<META NAME="WT. pc" CONTENT="ProductCategory">**

Defines the category of the product. Use semicolons to pass multiple categories.

**<META NAME="WT. tx\_u" CONTENT="units">**

Defines the quantity purchased. If the order contains multiple products, pass a semicolon-delimited list of units.

**<META NAME="WT. tx\_s" CONTENT="subtotal ">**

Defines the total cost for each WT. pn\_sku value passed. If the order contains multiple SKUs, pass a semicolon-delimited list of values for this parameter. However, do not pass a dollar sign (\$) or comma(,) in the subtotal variable.

## Example of Multiple Usage

You can pass multiple orders to the variables in the META tags by using a semicolon-delimited list. For example, the following tag represents two products:

**<META NAME="WT. pn\_sku" CONTENT="1X11GG34; 2YR5R53">**

**<META NAME="WT. tx\_u" CONTENT="2; 7">**

**<META NAME="WT. tx\_s" CONTENT="130.00; 150.00">**

## Sample Log File

After the JavaScript tag interacts with these META tags, it generates log files that may look something like this:

```
2007-03-04 00:08:18 proxy7.hotmail.com - W3SVC3 web1 192.168.1.1 GET /ads/default.asp
redir=products&ad=http%3A//www.phonedea.com&WT.pn_sku=1X11GG34; 2YR5R53
&WT.tx_u=2; 7&WT.tx_s=130.00; 150.00 . . . .
```



The italicized text represents the META-related order information captured by the tag and placed in the log file.

**Note**

In the log file %20 is an ASCII representation for a blank space.

---

## Tracking Shopping Cart Activity

You can use META tags that track shopping cart activity. At analysis, these tags are interpreted as steps along the path that leads to a successful completion of the shopping activity. WebTrends preconfigured Purchase Conversion Funnel includes four steps. For more information, see [“Scenario Analysis Parameters for Shopping Cart Analysis” on page 153](#).

## Tracking On-Site Advertising

Visitors often view advertisements that they do not necessarily click on. You can use On-Site Advertising to determine the number of visitors to your web site who view particular ads. With this feature you can produce advertising reports for each of your clients.

If you are selling advertising space on your web site, for example, you can collect traffic statistics to help determine pricing schedules.

The following META tag tracks advertising views:

```
<META NAME="WT. ad" CONTENT="name">
```

This metatag defines the name of the advertisement viewed on this page. You can designate multiple ad views using semicolons.

### Example On-Site Advertising

To attract business to your online hotel reservation system, you place a promotional advertisement on a page. You want to find out how many visitors viewed this advertisement. You can use the WebTrends Query Parameter WT.ad for this purpose.

```
<META NAME="WT. ad" CONTENT="Weekend Special Rate">
```

The advertising name “Weekend Special Rate” represents an advertisement that was viewed on the page.

### Sample Log File

After the JavaScript tag interacts with this META tag information, SDC generates log files that look something like this:

```
2007-03-04 00:08:18 proxy7.hotmail.com - W3SVC3 web1 192.168.1.1 GET /ads/default.asp  
referrer=products&ad=http%3A/www.phonedeeper.com&WT.ad=Weekend%20Special%20Rate . . .
```

The italicized text represents the META-related order information captured by the JavaScript tag and placed in the log file.



## Tracking Advertising Clicks

When a visitor to your site clicks on an ad, that action is referred to as an *Ad Click*. The following META tag tracks advertising clicks:

```
<META NAME="WT. ac" CONTENT="name">
```

Defines the name of the advertisement clicked to reach a particular web page. The Ad Click must contain an external redirect back to the client. The redirect needs to include the necessary code to generate a hit to the SDC server. You can designate multiple Advertising Clicks using semicolons.

### Example Tracking Advertising Clicks

You have a banner ad that links to another site with weekend getaway ideas and you receive payment for each referral to the site. You can use the WebTrends Query Parameter *WT. ac* to track clicks on the banner ad.

```
<META NAME="WT. ac" CONTENT="Weekend Pointer">
```

The advertising name "Weekend Pointer" represents an advertisement that was clicked on your page.

### Sample Log File

After the JavaScript tag interacts with this META tag information, SDC generates log files that look something like this:

```
2007-03-04 00:08:18 proxy7.hotmail.com - W3SVC3 web1 192.168.1.1 GET /ads/default.asp  
redirect=products&ad=http%3A//www.phone.dealer.com&WT.ac=Weekend%20Pointer . . .
```

The italicized text represents the META-related order information captured by the JavaScript tag and placed in the log file.

## Tracking Customized URLs

WebTrends reports only the base URL when it compiles reports on pages identified by URLs. "Top Pages" and "Page Views Trend" are reports that use page URLs. These reports could become overwhelming and meaningless because URL parameters are used to carry many different kinds of information in addition to dynamic page identification. There can be many variations to a single page URL. The parameters make it seem as if there are many unique URLs when there is only one, the base URL.

Dynamic pages are an exception. With dynamic pages, WebTrends counts each URL with different parameters. By customizing a URL, you can track dynamic pages by changing the URL before it is passed to WebTrends.

### META Tags That Customize URLs

The following META tags allow you to customize URLs:

```
<META NAME="DCS.csuri" CONTENT="uri-stem">
```

Assigns the information to the cs-uri-stem field of the log file.

### Sample Log File

If the URL looks like this without any customization:



`http://www.asiate.com/browse.asp?UID=14&Cat=Rock&Artist=Your_Band&Album=Jar_of_Files`

the Top Pages report only shows `http://www.asiate.com/browse.asp`.

If you write a server-side script to dynamically convert the parameters to page names, the URL might look like this:

`http://www.asiate.com/Rock/Your_Band/Jar_of_Files.asp`.

Place that URL in a META tag as follows:

```
<META NAME="DCS.dcsuri" CONTENT="http://www.asiate.com/Rock/Your_Band/
Jar_of_Files.asp">
```

After the JavaScript interacts with this META tag information, it generates log files that look something like this:

```
2007-08-10 00:06:06 192.168.100.40 - web1 GET /Rock/Your_Band/Jar_of_Files.asp...
```

The italicized text represents the META-related URL information captured by the JavaScript and saved in the log file.

## Tracking Page Titles

You may want to modify a page title before sending it to WebTrends in the following cases:

- You are dealing with dynamic content pages identified by URL parameters, and the page title represents the title of the base URL page rather than the dynamic content page.

Unless you modify the page titles, all pages have the same title in the reports.

- All pages have been assigned the same title, for reasons of style or company policy.

Even though URLs are displayed in addition to page title, the entire URL cannot be depended upon to distinguish one page from another.

Use server-side scripts to change the title to something that reflects the content of the pages so that you can identify them in reports. Next, pass the customized page titles to WebTrends, using the following META tag:

```
<META NAME="WT.title" CONTENT="title">
```

Defines the name of the title for this page.

## Example Page Title

You would like to specify the name of the page that is displayed on your reports. You are using a hosting service that does not allow you access to the <TITLE> tag on your page. You can assign a title for the page using the WebTrends Query Parameter `WT.title`.

```
<META NAME="WT.title" CONTENT="Advertising">
```

The page title "Advertising" represents the name of the page that is displayed on your reports.

## Sample Log File

After the JavaScript tag interacts with this META tag information, SDC generates log files that look something like this:



2007-03-04 00:08:18 proxy7.hotmail.com - W3SVC3 web1 192.168.1.1 GET /ads/default.asp?referrer=products&ad=http%3A//www.phonedea.com&WT.ti=Advertising . . .

The italicized text represents the META-related order information captured by the JavaScript tag and placed in the log file.

## Inserting the Tags

You can insert the JavaScript tag in several ways: by copying appropriate tag versions to each of your web site pages, using server-side includes on appropriate web servers, or by inserting the tag in the footer template. The method depends on your needs, your maintenance practices, and the programming resources available to you.

Once you have deployed the tag to your site, you can begin viewing reports of your visitor activity and of the revenue your site generates. Reports are typically available 24 hours after the tags are deployed.

## Copying the Tag to Each Page

Copy the same tag or individually modified copies of the code to your web site pages. To minimize the impact on your web site, place the tag as close as possible to the `</BODY>` tag.

## Using Server-Side Include Files

Server-side includes (SSI) are enabled by default on Internet Information Server and Apache web servers. You can either configure the server to run SSI on all files with the extensions you use for your web pages (.htm, .html), or you may need to change your page extensions (to .stm, .shtm, and .shtml, for example).

### To set up the include file and the include statements:

1. Place the include file containing the JavaScript tag in it in a location accessible to every page of your site.
2. Place an include statement on all of your web site pages. Be sure to use the correct file extensions. For example, if your include file is named `code_include.inc` and located in the `mysite` directory, you place the following include statement on your web pages:

```
<!--#include virtual="/mysite/code_include.inc"-->
```

## Using Footer Templates

If your web site uses header and footer templates, you can place the JavaScript tag in the footer template. Place it as close as possible to the `</BODY>` tag.

Keep in mind that a commerce confirmation page must include the revenue tracking code. You need to configure the confirmation page to include both the JavaScript tag and the revenue tracking code. Keep in mind that other pages cannot include the revenue tracking code.



## Tagging Best Practices

### Web Page Editor Issues

A number of HTML editors actually modify your HTML code and can break JavaScript. Make sure that your HTML editor does not modify the tag in any way.

### Specify Character Sets On Tagged Pages

As a best practice, you should include a character set META tag on the pages that have your WebTrends JavaScript tag. For example, `<META HTTP-EQUIV="Content-Type" CONTENT="text/html ; charset=iso-8859-1">`.

If you don't specify the character set, a browser may use the character set defined on the previously viewed web site. If that character set does not match the one you intend to be used on your own, your tags may not be consistently encoded. The resulting reports would show a single web page as several different web pages, and the text for the pages that were not encoded correctly may not be properly displayed.

## Document Revision History

Table 6 contains a summary of changes made to this document beginning with the release of WebTrends Analytics, version 8.7.

**Table 6: Document Revision History**

Software Version	Date of Last Update	Summary of Changes
v8.7	March, 2009	Added Document Revision History section.
v8.7d	July, 2009	Added footer link to Documentation Center.



---

## Chapter 8

# Creating WebTrends Analytics Profiles

This chapter describes the kind of information stored in report profiles and provides instructions for creating a profile using the Basic and Advanced profile wizards. For more information about using each dialog in the profile wizards, see the Administration Help.

## How Profiles Work

Profiles specify all of the information needed to generate reports from a web data file. They define the location of your web server data and how it should be analyzed. For example, profiles can specify information such as:

- The type of web data WebTrends analyzes and where to find it
- Whether your web site resides on a single server or on multiple servers
- The location of your home page
- Whether to apply data filters
- Which users can access the profile
- Which reports to create
- When to update reports

Each profile is associated with a set of log files from which it draws data and one or more templates, which determine the set of reports that can be rendered from the analyzed data. When you analyze a profile, WebTrends creates a set of Report databases. You can use WebTrends Analytics Reports to view reports for a specified profile and template based on the data in the Report databases.

## Profile Creation

**To begin creating a profile:**

1. In the left pane, click **Administration > Web Analysis > Reports & Profiles**.
2. Click **New**.



## Using the Profile Wizard

WebTrends allows you to create profiles in two modes. In the Basic mode of profile creation, WebTrends uses the most common settings to produce reports quickly with minimum configuration. The Basic mode creates a Standard Full-Featured Analysis profile. You can also use the Advanced profile settings to customize your profile by providing more information about your web data, your site configuration, and your reporting preferences.

### To create a profile using the Basic wizard:

1. In the Profile Name dialog, provide a name for your profile and specify the web site domain name.
2. If you are a WebTrends On Demand user, specify the time zone you want to use when displaying your reports.
3. In the Data Sources dialog, specify a data source (if you have already configured one) or create a new data source. A data source identifies the location of the web data that you want to track with this profile.
4. In the Report Packs dialog, specify the licensed report packs you want to use with the current profile. Report Packs determine what kinds of reports WebTrends creates, and thus the type of data included in reports for this profile.
5. In the Session Tracking dialog, specify how to identify user sessions for the profile.
6. In the Summary dialog, review your settings.

## Advanced Profile Settings

Use the Advanced Profile settings if you want to configure special WebTrends features such as:

- Advanced SmartView reports
- Profile analysis scheduling (WebTrends software users only)
- Advanced reporting features including:
  - Campaign and Scenario Analysis
  - Content Group Analysis
  - Hit and Visit Filters
  - URL Parameter Analysis
  - URL Search and Replace

Click **Additional Settings** in the last wizard dialog to access these advanced settings.

You should also use Advanced Profile settings if you want to create special types of profiles such as Parent-Child profiles. For more information about configuring these specialized profile types, see the Help and the *WebTrends Administration User's Guide*.

### To create a profile using the Advanced Profile settings:

1. In the Profile Name dialog, provide a name for your profile and specify the web site domain name.
2. If you are a WebTrends On Demand user, specify the time zone you want to use when displaying your reports.
3. Select the **Advanced profile options** check box. You see the Advanced Profile Options dialogs in the wizard listed in the left column.



4. Click **Next** to complete the dialogs. For detailed information about each setting, click the **Help** icon. To configure more profile options than the wizard provides, click **Additional Settings** in the Summary dialog.

## Advanced Profile Dialogs

The following dialogs are included in the Advanced Profile wizard. For detailed information about how to complete each dialog, see the Help. Dialogs marked with an asterisk (\*) may not be displayed depending on the wizard settings you choose. However, choosing Advanced Settings will typically display all the dialogs for the Advanced Profile wizard.

### Data Sources

Specifies the location of the web data file.

### General

Specifies the portion of the log to analyze, whether to retrieve HTML page titles, whether to enable Express Analysis, and time zone behavior. (If you are a WebTrends On Demand user, specifies the time zone to be displayed when displaying reports).

### Home\*

Specifies the location of the site home page.

### Host Binding\*

Specifies whether analysis should run only on certain computers or groups of computers.

### Page File Types\*

Specifies which file types WebTrends counts as page views.

### Parent Child\*

Specifies the settings used to create a Parent-Child profile.

### Post-Analysis\*

Specifies whether any programs should run immediately after analysis.

### Pre-Analysis\*

Specifies whether any programs should run immediately before analysis.

### Profile Class

Specifies the type of web data to use and whether to create a standard, Advanced SmartView, Event Database, or Parent-Child report.

### Profile Name

Specifies the name of the profile, the site domain name (used to identify the SmartView domain and the web site URL) and (for WebTrends On Demand users) the time zone.

### Schedule

Specifies when analysis should occur.

### Session Tracking\*

Specifies whether to apply profile-specific settings for tracking user sessions.

### Site Configuration\*

Specifies whether your web data is stored in logs or a WebTrends Warehouse, and whether the web server resides on one server or multiple servers.



**SmartView\***

Specifies whether to enable SmartView reporting. For advanced SmartView profiles, specifies whether to analyze the home page domain or another domain.

**Summary**

Specifies the settings established for the profile.

**URL Rebuilding\***

Specifies whether to apply settings that modify URLs before analysis to provide more accurate reporting.

## Document Revision History

Table 7 contains a summary of changes made to this document beginning with the release of WebTrends Analytics, version 8.7.

**Table 7: Document Revision History**

Software Version	Date of Last Update	Summary of Changes
v8.7	March, 2009	Added Document Revision History section.
v8.7d	July, 2009	Added footer link to Documentation Center.



---

## Chapter 9

# Setting Up WebTrends Users and Roles

WebTrends Marketing Lab provides a highly configurable array of user rights and roles to help you set up user access that reflects your organizational security model. Depending on your organizational structure, you may find that you want to provide varying levels of access to different features and objects within WebTrends Marketing Lab.

This chapter describes how to create users and assign user rights to various features of WebTrends Marketing Lab. At the most basic level, creating a user grants access to a WebTrends login. However, by selecting specific rights for each user, you can define very specific and detailed user rights according to each user's job function and data access requirements. If you have more than a handful of WebTrends users, you can reduce configuration time using groups of rights called *roles*. Instead of repeatedly defining many individual rights for each user, you can streamline rights assignment by assigning each new or existing user a role.

## Understanding WebTrends User Rights

When you configure rights for a user or role, WebTrends provides three different types of rights: action rights, profile rights, and template rights. Selecting these rights defines the permissions granted to a user or role. Action rights are designed to control access to functionality. Profile and template rights provide access to the data and reports contained in specific profiles and templates.

WebTrends user rights are cumulative, and greater rights grant implied access to lesser ones. For example, granting Create rights also grants View, Edit, and Delete rights. Similarly, if you have only View rights to all profiles through the Action Rights dialog, you can still be granted Edit rights to any specific profile in the Profile Rights dialog.

## Understanding Action Rights

Action rights are Create, View, Edit, and Delete rights to specific Marketing Lab features and functionality. (These rights were known as User Rights in previous versions.)

Action rights can provide different levels of functionality to users or groups of users. For example, users in your organization who primarily use WebTrends Marketing Lab to view reports or schedule report exports probably do not need rights to manage users, set system options, and perform administrative functions. By providing users only with the rights they need, you can ensure that your configuration is secure and your report users do not become confused by too many configuration options.

## Understanding Profile and Template Rights

Profile and Template rights are Create, View, Edit, and Delete rights to individual profiles and templates within WebTrends Marketing Lab.



Profile and Template rights can provide better security and more targeted report content. In cases where different users and roles should not see the same data, you can use these rights to determine which profile data users can access and the number and arrangements of the reports they see. For example, Marketing Lab users who provide third-party reporting services often use profile rights to make sure each company only has access to its own data.

Similarly, you can limit template rights to make only certain groups of reports visible to each user or role. For example, if you only want to provide Search reports to your Search Marketing group, you can create a template that contains only those reports and grant template rights to each user in the group. Alternately, you can create a Search Marketing role that has those rights and assign it to all users in the group.

**Note**

Granting access to all profiles or all templates listed in the Profiles Rights and Template Rights dialogs does not also grant access to newly created profiles and templates. To grant rights to all current and future templates and profiles, use the Profiles and Templates rights in the Action Rights dialog.

## Understanding WebTrends User Roles

User roles are predefined sets of user rights that you can apply to more than one user. If you have multiple users who need the same permissions, you can avoid recreating the same rights for each user by creating a role. Then assign the same role to each user by selecting the role in the General dialog of the User settings.

## Configuring User Roles

**To create a user role:**

1. In the left pane, click **Administration > Application Settings > User Management > Roles**. A list of the current users roles opens.
2. Click **New**.
3. Specify a name for the role. For example, you could name a role based on a job function such as Accounting, a third-party company name, or a role within WebTrends such as "Report Scheduler." Click **Next**.
4. In the Action Rights dialog, set rights to WebTrends Marketing Lab functionality for this user. For more information about Action rights, see ["Understanding Action Rights" on page 77](#). For more information about individual Action rights, see Help. Click **Next**.
5. In the Profile Rights dialog, set rights to specific report profiles for this user. For more information, see ["Understanding Profile and Template Rights" on page 77](#).
6. In the Template Rights dialog, set rights to specific report templates for this user. For more information, see ["Understanding Profile and Template Rights" on page 77](#).



## Applying Preconfigured Roles to Existing Users

User roles overwrite any existing rights for each user. If you have existing users and want to migrate them to use roles, you can edit each user and select a role in the General dialog.

---

### Caution



After you assign a role to an existing user and click Save, you cannot recover any previous user rights settings associated with that user. Because WebTrends On Demand used a different method to assign user rights in previous releases, WebTrends On Demand administrators should plan carefully before mapping and applying roles.

---

## Adding Users

The following procedure shows how to add users in WebTrends Administration. For more information about managing users in WebTrends On Demand, see the *WebTrends On Demand Accounts User Guide* or the WebTrends On Demand Accounts Help.

---

### Tip



If you plan to create customized user roles and assign them to the users you create, configuring roles first will save you configuration time. For more information about roles, see [“Understanding WebTrends User Roles” on page 78](#).

---

### To add a user:

1. In the left pane, click **Administration > Application Settings > User Management > Users**. A list of the current user accounts opens.
2. Click **New**.
3. If you want to assign a user role to this user rather than specifying individual user rights, select a role from the list.
4. In the **Login Name** text box, type the name for the user to use to log in to WebTrends.
5. In the **First Name** and **Last Name** text boxes, type the user's names.
6. In the **Email** text box, specify the user's email address.
7. If you want to be able to sort users by an internal group name that is meaningful to your organization, type the name in the **Group Name** text box. For example, type Campaign Managers if you want to be able to identify a set of people in that group and adjust their rights or roles more efficiently. Clicking on the Group column in the Users dialog box lets you
8. In the **User Authentication** section, choose one of the following options:
  - If you want the user to log in using a network user name and password, select **Use OS Authentication for this user's password**.



- If you want the user to log in using a password you specify, select **Use WebTrends authentication for this user's password**.

**Note**

If your web server is configured for operating system authentication, make sure you provide users permissions on the local machine, or have a domain account. (If you choose not to set up a local user account, use WebTrends authentication.) For full details on setting up a local user, please refer to your operating system's documentation.

9. If you selected **Use WebTrends Authentication...**, specify the user's password in the **New Password** and **Verify Password** text boxes.
10. If at any time you need to disable this user's WebTrends access, select the **Disable User** check box here. Disabling access prevents the user from logging in to WebTrends Marketing Lab. It does not delete or remove the user.
11. Click **Next**.
12. In the Action Rights dialog, set rights to WebTrends functionality for this user. For more information about individual rights, see Help. Click **Next**.
13. In the Profile Rights dialog, set rights to specific report profiles for this user. For more information, see ["Understanding Profile and Template Rights" on page 77](#).
14. In the Template Rights dialog, set rights to specific report templates for this user. For more information, see ["Understanding Profile and Template Rights" on page 77](#).
15. In the Preferences dialog, specify this user's time zone, so that all WebTrends events are displayed in the user's local time.
16. Specify the session timeout for this user. This setting determines how long this user can remain inactive before WebTrends automatically ends the session.
17. The **Summary** dialog lists the current settings for the user you are adding. To save these settings, click **OK**. The user is added to the **Users** list.

## About View Only Permissions

For all users who have no other permissions than View Reports, make sure the user has view rights assigned in the template. Otherwise, the user with View Reports only rights will not be able to view any reports if they are not listed in the template. Users with View Reports rights have access to WebTrends Analytics Reports only.

**Note**

Users who have View Only permissions are also automatically granted access to view profiles. However, because of the extreme limitation of View Only permissions, viewing profiles is not enabled for these users.

The *WebTrends Analytics Reports User's Guide* is available for those using WebTrends Analytics Reports. You can download this guide from the WebTrends Customer Center.



## Predefined Role Settings

The following graphic shows the settings for each of the predefined user roles included in WebTrends. You can use this comparison to determine whether the predefined roles are appropriate for your users. Settings marked N/A are not available rights for a specific feature or functionality.

Permission	Create			Delete			Edit			View		
	Administrator	Report Manager	Report User									
Accounts (WTOD only)	1	0	0	1	0	0	1	0	0	1	1	1
Analysis Options	1	1	0	1	1	0	1	1	0	1	1	0
Application Settings (Software Only)	1	0	0	1	0	0	1	0	0	1	0	0
Custom Reports	1	1	0	1	1	0	1	1	0	1	1	1
Custom Reports in Advanced Mode	N/A	N/A	N/A	N/A	N/A	N/A	1	1	0	N/A	N/A	N/A
Data Sources	1	0	0	1	0	0	1	1	0	1	1	0
Event Database Free Form Query Access (Software Only)	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	1	0	0
Event Database Full Named Query Access	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	1	0	0
Export to Database	1	1	1	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Export Word , CSV, PDF Reports	1	1	1	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Goals	1	1	0	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Log File Import (WTOD only)	1	0	0	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Log File Download (WTOD only)	1	0	0	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Marketing Warehouse Configuration	N/A	N/A	N/A	N/A	N/A	N/A	1	0	0	N/A	N/A	N/A
Marketing Warehouse Profiles	1	0	0	1	0	0	1	1	0	1	1	0
ODBC Driver	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	1	1	0
Profiles	1	0	0	1	0	0	1	1	0	1	1	1
Report Configuration	1	1	0	1	1	0	1	1	0	1	1	0
Report Designer	1	1	0	1	1	0	1	1	0	1	1	0
Reports	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	1	1	1
Scheduled Export To Database	1	1	1	1	1	0	1	1	0	1	1	1
Scheduled Word, CSV, PDF, Reports	1	1	1	1	1	0	1	1	0	1	1	1
Scheduler (WT Software only)	1	0	0	1	0	0	1	0	0	1	0	0
Score	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	1	1	1
Score Rule Set Configuration	N/A	N/A	N/A	N/A	N/A	N/A	1	1	0	N/A	N/A	N/A
Score Rule Set Enablement	N/A	N/A	N/A	N/A	N/A	N/A	1	1	0	N/A	N/A	N/A
Segment	1	1	0	N/A	N/A	N/A	N/A	N/A	N/A	1	1	1
Shared Bookmarks	1	1	1	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
SmartView	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	1	1	1
Templates	1	1	0	1	1	0	1	1	0	1	1	0
Translation File Import (WTOD only)	1	1	0	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Users	1	0	0	1	0	0	1	0	0	1	1	1
Visitor History Export	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	1	1	0
Visitor Intelligence (Warehouse Reports)	1	1	0	N/A	N/A	N/A	N/A	N/A	N/A	1	1	1
Web Services	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	1	1	0
WebTrends Explore	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	1	1	1



## Document Revision History

Table 8 contains a summary of changes made to this document beginning with the release of WebTrends Analytics, version 8.7.

**Table 8: Document Revision History**

Software Version	Date of Last Update	Summary of Changes
v8.7	March, 2009	Added Document Revision History section.
v8.7d	July, 2009	Added footer link to Documentation Center.



---

## Chapter 10

# Tracking Visitor Sessions

In order to provide the most insightful data, WebTrends requires that you use a strong method of identifying visitors. For WebTrends Analytics, strong identification methods are cookies and authenticated user IDs. For WebTrends Marketing Warehouse, cookies are required to track visitor-related data. Because cookies are the recommended best practice, this chapter focuses on cookies. It explains the differences between first-party cookies and third-party cookies and describes how you can use cookies to track visitor sessions. For information about alternative methods of identifying visitors for session tracking, see “Visitor Identification” in the *WebTrends Guide to Web Analytics*.

This chapter assumes that you use WebTrends Analytics On Demand or have installed WebTrends SmartSource Data Collector (SDC) and have configured an SDC site map. For more information about installing SDC and configuring an SDC site map, see the “SmartSource Data Collector Installation” in the *WebTrends SmartSource Data Collector User’s Guide*. If you do not intend to use first-party cookies with SDC, see [“Using First-Party Cookies Without SDC” on page 92](#).



### Note

This chapter does not apply to WebTrends Analytics On Demand Small Business.

---

## What is a Cookie?

A cookie is a piece of identifying data, typically created by a web server. A web site sends a cookie to a visitor’s browser and stores it on a visitor’s computer either temporarily (for that visit session only) or permanently on the hard disk (or until the visitor deletes them). Temporarily stored cookies are called session cookies. Cookies stored on the hard disk are called persistent cookies.

Persistent cookies can identify a visitor as a new or returning visitor by storing a value that uniquely identifies each visitor. If a visitor has been to the site before, a cookie is sent to the web server with the request for a particular page. The web server checks for the presence of a cookie in the request and if no cookie is detected, the web server generates the cookie and sends it with the response to the visitor’s browser. When the visitor returns, the cookie is included in the request, the web server detects the cookie and recognizes the visitor as a returning visitor. The web server then writes the cookie to the log file in the cs(Cooki e) field.

## Why Web Browsers Reject Cookies

Whether web browsers are likely to accept a cookie strongly depends on whether the cookie is a first-party cookie or a third-party cookie.



A cookie served from a domain other than the domain that your visitor requests from your web site is considered a *third-party cookie*. WebTrends On Demand and SmartSource Data Collector (SDC) have historically used cookies as the primary method to obtain visitor information.

Tracking visitors accurately is paramount for confidence in your web analytics results. Studies by leading analyst research firms such as Jupiter Research and Forrester have indicated that increasingly high-rates of cookie rejection and deletion by Internet users makes third-party cookies an unreliable method for collecting and reporting on web marketing results. In fact, Jupiter currently believes third-party cookie rejection rates are as high as 28%.

In response, WebTrends conducted its own research, analyzing third-party cookie rejection rates for 5 billion visitor sessions between January 2004 and April 2005. WebTrends research found cookie rejection rates to be somewhat lower but still significant, revealing that on average 12% of Internet user traffic is blocking or preventing third-party cookies from being set on computers, and that this trend can be as high as 17% for some vertical industries, such as retail. In addition, when analyzing the third-party cookie rejection trend since the beginning of 2004, WebTrends findings show that third-party cookie rejection has increased 4x to its current rate.

Internet users commonly reject third-party cookies as part of their security measures. Some of the most common reasons for the increase in the rejection of third-party cookies are:

- Anti-spyware programs are designed to remove cookies that surreptitiously monitors visitors' web activities. These programs often consider hosted web analytics services to be spyware and thus target their cookies for removal from your visitors' computers.
- Current browser technologies such as Microsoft's Internet Explorer and Mozilla's Firefox make it easier for visitors to reject third-party cookies.

## Negative Impact of Third-Party Cookie Rejection

Cookie-dependent analytics solutions rely on the cookie as the method to identify one unique browsing session from another. There are a number of business issues that arise from third-party cookies being rejected or deleted on a regular basis:

- **Inaccurate Visitor Metrics:** At its most fundamental level, if an Internet user has configured the browser security settings to automatically reject third-party cookies, that visitor will not be properly counted in your web analytics results. As mentioned earlier, market estimates project this to be anywhere from 12% to 28% of Internet users on average.
- **Deceiving Retention Based Metrics:** Taking this one step further, if "John Doe" visits your web site on May 15 and accepts the third-party cookie, he will be recognized as a new visitor. If John then deletes all of his third-party cookies with his anti-spyware application on May 16th and returns to the site on May 17th, the analytics solution will identify John as a new visitor, since he no longer has the cookie on his computer. This would have an impact of under representing your retention based metrics such as your repeat visitor rate.
- **Inaccurate Conversion Metrics:** Cookie deletion also has an impact on your conversion rate for new visitors versus repeat visitors.

Conversion rate = (conversion actions taken/number of visitors) X 100

where a conversion action is an action indicating visitor conversion, such as an order, and the visitors may be new or repeat visitors.



As pointed out in example #2 (directly above), if the cookie is being systematically deleted, repeat visitor rates are going to be under-counted and new visitor rates are going to be over-counted, skewing your conversion rate metric by which you analyze your site's overall effectiveness.

- **Unreliable Campaign, Search and Merchandising Reports:** In addition to tracking the behavior of a visitor to the site in general, many analytics providers correlate visitor response and site interaction to a specific campaign, search engine or product in an attempt to understand precisely which campaign or merchandising offer inspired the Internet user to take an action; much of this information can rely on information stored in the cookie. If the cookie is rejected or deleted from the Internet user's browser, reports designed to identify latent or deferred conversion to a campaign or merchandising offer will be misrepresented. It is also important to note that the longer that you track conversion to an individual marketing activity, the more likely it is that your metrics are inaccurate, as the likelihood the user deletes the third-party cookie increases.

## Solving Rejection with First-Party Cookies

For most business models, *first-party cookies* are regarded as the most reliable method to measure visitor activity. A cookie served directly to your visitors by your own web server is registered by the browser as a first-party cookie. Whereas a third-party cookie is set by the analytics vendor, an entity with which the web site visitor does not have a relationship, the first-party cookie is set by the business or organization with which the Internet user has specifically chosen to do business. Because of this relationship, the first-party cookie is deemed a more secure cookie by the user.

First-party cookies are considered less of a security risk than third-party cookies and are more likely to be accepted by the browser. By issuing first-party cookies, your benefits include:

- Most accurate visitor metrics
- Compatibility with data collected from existing WebTrends data sources that used WebTrends third-party cookies

If you use WebTrends Analytics software, you may already have a method of setting first-party cookies. If you use WebTrends Analytics On Demand or WebTrends Analytics software with SDC, your WebTrends JavaScript tag is configured to use first-party cookies by default. For more information, see ["Using the JavaScript Tag to Track Cookies" on page 87](#).

## How WebTrends Marketing Lab Uses Cookies

As long as the visitor's browser accepts cookies and the visitor does not delete the cookie from the computer, WebTrends can use the cookie to determine whether the visitor is a returning visitor or a first-time visitor. WebTrends can also use cookies to strongly identify visitors and use this information to develop a rich repository of visitor history which you can use for reporting.

## Methods for Generating First-Party Cookies

You can generate first-party cookies that WebTrends can use to track visitor sessions using one of the following methods:

- Allow the WebTrends JavaScript tag to serve cookies
- Configure your web server to serve cookies
- Use the WebTrends Cookie Plug-in to serve cookies



This section discusses each of these methods.

## Using the WebTrends JavaScript Tag

If you use WebTrends On Demand or WebTrends software with SDC, your best choice for generating first-party cookies is by allowing your JavaScript tag to generate them. By default, the JavaScript tag generates the first-party cookie and passes it in the query string as the `WT.co_f` query parameter. With this method, you do not need to configure your web server to generate cookies.

## Using Your Web Server to Generate Cookies

Most modern web servers contain functionality for serving cookies. If your web server is already configured to serve cookies, you should use this cookie to identify your visitors. This method is suitable whether you use WebTrends software or WebTrends On Demand.

This section describes how some commonly used web servers deliver cookies.

### Apache Web Server

Apache provides the `mod_usertrack` module for click stream logging of visitor activity on a site. `Mod_usertrack` sets a cookie with a unique identifier. Enable `mod_usertrack` so by adding this dynamically shared object to the `LoadModule` list, and setting the “CookieTracking on” directive in the `httpd.conf` file. The following directives provide additional control: `CookieDomain`, `CookieExpires`, `CookieName`, `CookieStyle`.

For more information, see [www.apache.org](http://www.apache.org).

### Microsoft Internet Information Server (IIS)

#### Active Server Pages

Microsoft ASP supports the notion of a session management through the `Session` object. Session keys are stored in the `ASPSESSIONID` cookie.

As an alternative, you can manage your own tracking cookie using the `Response.Cookies` Collection of the `Response` and `Request` objects.

For more information, see [www.microsoft.com](http://www.microsoft.com).

#### Site Server

Microsoft Site Server includes a User identification Filter (ISAPI filter) called `mss_log.dll`. This filter generates a 32-byte GUID that is stored in the `SITESERVER` cookie.

For more information, see [www.microsoft.com](http://www.microsoft.com).

### iPlanet/SunOne

iPlanet/SunOne’s servlet engine supports Java Server Pages. You can manage your own tracking cookie by using the `Cookie` class and the `HttpServletRequest.addCookie` method, and the `HttpServletRequest.getCookies` method.

For more information, see [www.java.sun.com](http://www.java.sun.com).



- Modifying the WebTrends JavaScript tag to serve cookies
- Modifying the WebTrends JavaScript tag to serve cookies

## Using the WebTrends Cookie Plug-in

The WebTrends Cookie Plug-in is software that you can install on your web server to generate first-party cookies. If your web server cannot be configured to serve cookies, and you use WebTrends software without SDC, this is your best choice for generating first-party cookies.

The Cookie Plug-in supports Apache, Microsoft IIS, and iPlanet/SunOne web servers. For more information about installing and using the Cookie Plug-in, see the *Cookie Plug-in User's Guide*.

## Configuring WebTrends for First-Party Cookie Tracking

If you use *WebTrends On Demand* or *WebTrends* software with *SmartSource Data Collector*, you can configure your JavaScript tag to recognize the first-party cookie method that you use. By default, the JavaScript tag generates the first-party cookie for you. For more information, see [“Using the JavaScript Tag to Track Cookies” on page 87](#).

If you use *WebTrends* software without *SmartSource Data Collector*, you simply need to create a Session Tracking definition configured to use your cookie.

## Using the JavaScript Tag to Track Cookies

If you use *WebTrends On Demand* or *WebTrends* software with *SmartSource Data Collector*, the JavaScript tag is configured to track first-party cookies by default. Use the Tag Builder at [tagbuilder.webtrends.com](http://tagbuilder.webtrends.com) to create your tag and specify any additional cookie settings. The following procedure has four main steps.

Use the following steps in sequence to configure first-party cookie tracking.

18. Implement the JavaScript tag you created using the Tag Builder. See [“Implementing the JavaScript Tag” on page 87](#).
19. Edit a profile and specify the first-party cookie data source. See [“Specifying the First-Party Cookie Data Source” on page 88](#).
20. For the same profile, specify session tracking for first-party cookies. See [“Specifying Session Tracking for First-Party Cookies” on page 88](#)

## Implementing the JavaScript Tag

Implement the WebTrends JavaScript tag on all the pages that you want to track. You can place the JavaScript tag anywhere between the <body> and </body> tags on a web page. Placing the tag at the top of the page directly after the <body> tag allows the tag to execute even if the page does not fully load. However, we recommend you place it at the bottom of the page just before the </body> tag to ensure the JavaScript tag is only activated after the page fully loads and all the information that the tag needs is available.



In addition to placing the tag directly in your web pages, there are other methods for tagging your pages which can make it easier to tag many pages quickly. Alternatively, you can place the tag in a client-side include file or place the tag in a footer template. For more information on these options and tagging best practices, see “Client-Side JavaScript Integration” in the *WebTrends Analytics Implementation and Maintenance Guide*.

## Specifying the First-Party Cookie Data Source

In this step, you edit a profile and specify the first-party cookie data source for that profile.



### Note

If your web site has multiple domains and you want to tracking visitors across them, you must create a separate data source for each domain. For more information, see “Configuring Domains” on page 89 and “Tracking Visitors Across Domains” on page 89.

### To specify the first-party cookie data source:

1. In the left pane, click **Administration > Web Analysis > Reports & Profiles**.
2. Mouse over a profile and click **Edit** on the Action menu.
3. Click **Analysis > Data Sources**.
4. Click **New**. If you use WebTrends On Demand, you cannot specify a new data source. You simply need to specify the first-party cookie data source for this profile. Click **Save**, and go to “[Specifying Session Tracking for First-Party Cookies](#)” on page 88.
5. In the Data Sources dialog specify the name of the server and select the data source.
6. Click **Save**, and your new data source for first-party cookies appears in the list of data sources for that profile.

## Specifying Session Tracking for First-Party Cookies

Session Tracking definitions determine how WebTrends identifies visits and counts unique visitors.

### To specify session tracking:

1. In the left pane, click **Administration > Web Analysis > Reports & Profiles**.
2. Mouse over a profile and click **Edit** on the Action menu.
  1. Click **Analysis > Session Tracking**.
  2. Clear the **Always Use Default Definition** check box.
  3. Click **Track User Sessions Using First Party Cookie**.
  4. Click **Save**.



## Configuring Domains

By default, the domain for the first-party cookie is populated with the actual domain that served the page. If you need to track visitors across different domains, make sure that you configure your JavaScript tag to set the domain that should be associated with your first-party cookie.

Consider the following domains:

- www.newstuff.webtrends.com
- www.standardstuff.webtrends.com
- www.ultra.cool.things.webtrends.com

All of these domains are subdomains of www.webtrends.com, which is a root domain. Therefore, in the WebTrends On Demand user interface, you would specify .webtrends.com (note the leading period) to track cookies across these domains.

Also, note that another domain such as www.webtrends.store.com can be a separate root domain that belongs to the same WebTrends account. Using WebTrends cross-domain tracking, the same visitor ID can be moved from one domain to another. This is because WebTrends On Demand can recognize that all four domains are members of the same account (WebTrends).

You can specify the domain for your cookie when you add or edit a data source.

### To specify your domain:

1. In the left pane, click **Administration > Application Settings > Data Sources**.
2. Mouse over a data source and click **Edit** on the Action menu.
3. Click **SmartSource Data Collector**.
4. Click **Tracking**.
5. Select the **Set the First-Party Cookie domain** check box.
6. Type the name of the domain you want to use. Be sure to precede the domain name with a period. Doing so insures that all sub-domains are rolled up into the domain. If you do not add the period, the cookie is set to the actual domain which serves the page. For example, type .webtrends.com

## Tracking Visitors Across Domains

If you use WebTrends Analytics On Demand or WebTrends Analytics software with SmartSource Data Collector and you have multiple domains, your visitors will have a different first-party cookie set on each domain as well as a third-party cookie for your account. They will be reported as unique visitors to each domain when you use first-party cookie session tracking methods. However, WebTrends can track your first-party cookies across domains by using the third-party cookie that identifies your domain.



You can create a separate profile to track your visitors across your domains using the Account Rollup data source, keeping in mind that this data source uses the WebTrends third-party cookie for tracking visitors. The visit and visitor counts will be different when using this Account Rollup data source, compared to your more reliable first-party cookie profiles. However, it can provide you with meaningful insight into your account traffic if needed.

**Note**

Cross-domain tracking applies only to SDC data files. You cannot analyze both web server data and SDC data file and then perform cross-domain cookie tracking using the SDC account rollup data source.

As a best practice, WebTrends recommends that you use first-party cookies to identify enterprise-wide, cross-domain behavior and trends. This method leverages a WebTrends third-party cookie to establish the first-party cookie, which tracks visitors across the specified domains in your data sources. If the visitor rejects third-party cookies, the first-party cookies continue to identify the visitor; however, that particular visitor appears as a different visitor for each domain.

**To track visitors across domains in WebTrends Analytics software:**

1. Create a separate data source for each domain. For more information, see [“Configuring Domains” on page 89](#).
2. In the left pane, click **Administration > Web Analysis > Reports & Profiles**.
3. Edit a profile.
4. Select **Analysis > Session Tracking**.
5. Select **Track User Sessions Using First-Party Cookie (Account Rollup)**. You might need to clear the **Always Use Default Definition** check box.
6. Click **Save**.

**Note**

You can make further modifications to your profile setting if necessary by editing the profile after you have saved it.

**To track visitors across domains in WebTrends Analytics On Demand:**

1. Create a separate data source for each domain. For more information, see [“Configuring Domains” on page 89](#).
2. Create your tag using the Tag Builder at [tagbuilder.webtrends.com](http://tagbuilder.webtrends.com).
3. Select **Web Analysis > Data Sources** and then select all of the data sources you want to analyze for this profile. Be sure that for each data source checked here, you generate the tag using **Tag Builder** and that you update the tag on every page of the domains affected. Missed pages will not be counted. The tag generated on the Data Sources tab does not support cross-domain tracking for Marketing Warehouse profiles.
4. Select **Web Analysis > Reports & Profiles**.
5. Edit the specified profile.
6. Select **Analysis > Session Tracking**.



7. Select session tracking options:
  - For Marketing Warehouse profiles, select the option that matches your selection under Warehouse Specific Visitor Tracking on the Advanced Settings tab in **Tag Builder**. The default selection in Tag Builder and in Session Tracking is **Use standard WebTrends Visitor ID (WT.vt\_sid)**.
  - For WebTrends Analytics profiles, select **Track User Sessions Using First-Party Cookie (Account Rollup)**.
8. Click **Next**.
9. Click **Save**.
10. The reports for this profile show aggregated data for the domains that belong to that account.

**Note**

WebTrends On Demand Business Edition does not support cross-domain tracking.

## Converting Third-Party Cookies to First-Party Cookies

If you have an existing WebTrends On Demand account that uses third-party cookies, you can add the first-party cookie tracking to your JavaScript tag. After the tag is implemented on your site, visitors immediately begin to receive the first-party cookies. WebTrends On Demand and SmartSource Data Collector (SDC) have built-in logic to stitch visitor records together to ensure a smooth transition from the third-party cookie methodology to the new first-party cookie methodology.

### To convert third-party cookies to first-party cookies:

1. Modify the JavaScript tag as described in “[Configuring WebTrends for First-Party Cookie Tracking](#)” on [page 87](#).
2. Update all your web pages to use the new JavaScript tag.
3. Wait 24 hours until analysis of the visitors who were tracked using the third-party cookie session tracking has completed.
4. Change the profile so it uses the first-party cookie session tracking definition.

## Customizing Tag-Generated First-Party Cookies

If you use WebTrends On Demand or WebTrends software with SDC, you can customize the persistence and expiration date of your first-party cookie.

### Creating Session Cookies

If you want to generate session cookies rather than persistent cookies, you remove the expiration date parameter from the cookie. However, this is not recommended because WebTrends cannot use session cookies to accurately track unique visitors to your site. Also, visit counts are inaccurate if the visitor closes the browser, reopens it and immediately returns to your site. In this case, the visitor is identified as a new unique visitor, and the visit is considered a new visit to your site.



**To create session-based first-party cookies:**

1. In the left pane, click **Administration > Application Settings > Data Sources**.
2. Edit the data source.
3. Click **SmartSource Data Collector**.
4. Modify the following line from the JavaScript tag text box:  

```
var expiry=""; expires="+dExp.toGMTString();
```

so that it looks like this:  

```
var expiry="";
```
5. Click **Download this tag** to save your new tag.
6. Click **Save**.
7. Implement the tag on your web site, replacing any existing tags, and redeploy the updated pages to your web site.

Since session cookies are only valid for the current visit, they cannot be used to accurately report on many aspects of visitor data, including unique visitors, campaign tracking, commerce tracking, search engine history, and other visitor history based analysis reports. Because the use of session cookies may alter the statistics in reports from what you are used to viewing, you should try session cookies on one profile as a test model and look at the numbers in the resulting report to see if that is what you were expecting. After you accept the results, you can apply session cookies to other profiles.

## Configuring Cookie Expiration

With WebTrends v7.5 and higher, the first-party cookie set by the JavaScript tag is configured to expire in 10 years. You can change the expiration by modifying the time value parameter in the statement.

**To configure the first-party cookie expiration:**

1. In the left pane, click **Administration > Application Settings > Data Sources**.
2. Edit the data source.
3. Click **SmartSource Data Collector**.
4. Edit the following line from the JavaScript tag text box:  

```
var dExp=new Date(dCur.getTime()+315360000000);
```
5. 315360000000 represents the total number of milliseconds in 10 years. Change this value to the number of milliseconds from the current time until the time that you want the cookie to expire. For example, if you want the cookie to expire in 60 days, then change this value to  $5183940000 = 60 \text{ (days)} * 24 \text{ (hrs per day)} * 60 \text{ (minutes per hour)} * 60 \text{ (seconds per minute)} * 1000 \text{ (milliseconds per second)}$ .
6. Implement the modified tag, replacing the existing tags on your pages, and redeploy the updated pages to your web site.

## Using First-Party Cookies Without SDC

If you use WebTrends Analytics software without SDC, you can use first-party cookies to identify visitors simply by creating a Session Tracking definition that specifies your cookie and applying it to your profiles.



**To create a Session Tracking definition:**

1. In the left pane, click **Administration > Web Analysis > Options > Session Tracking**.
2. Click **New**.
3. Type a name for the definition in the **Description** field.
4. Select **Use the following alternate method(s)**.
5. Select the **Cookie** check box.
6. In the Track Sessions by Cookie dialog, select **Use this cookie**.
7. Specify the name of the cookie that your web server uses to identify visitors. For example, WEBTRENDS\_ID is the name of the cookie that the WebTrends Cookie Plug-in uses by default.

**To use cookie session tracking in profiles:**

1. In the left pane, click **Administration > Web Analysis > Reports & Profiles**.
2. Add or edit a profile.
3. Click **Analysis > Session Tracking**.
4. Clear the **Always use default definition** check box.
5. Select the cookie tracking definition that you created.
6. Click **Save**.

## Implementing the Opt-Out Cookie

Because of misconceptions about the nature of cookies, some of your visitors might have concerns about how cookies that your web site generates might be used to track their Internet behavior. Using the WebTrends JavaScript, you can implement a method that allows visitors to your site to opt out of being tracked by your first-party cookie. The JavaScript supports *full opt-out*, in which the JavaScript does not collect data about the visit and does not set a cookie.

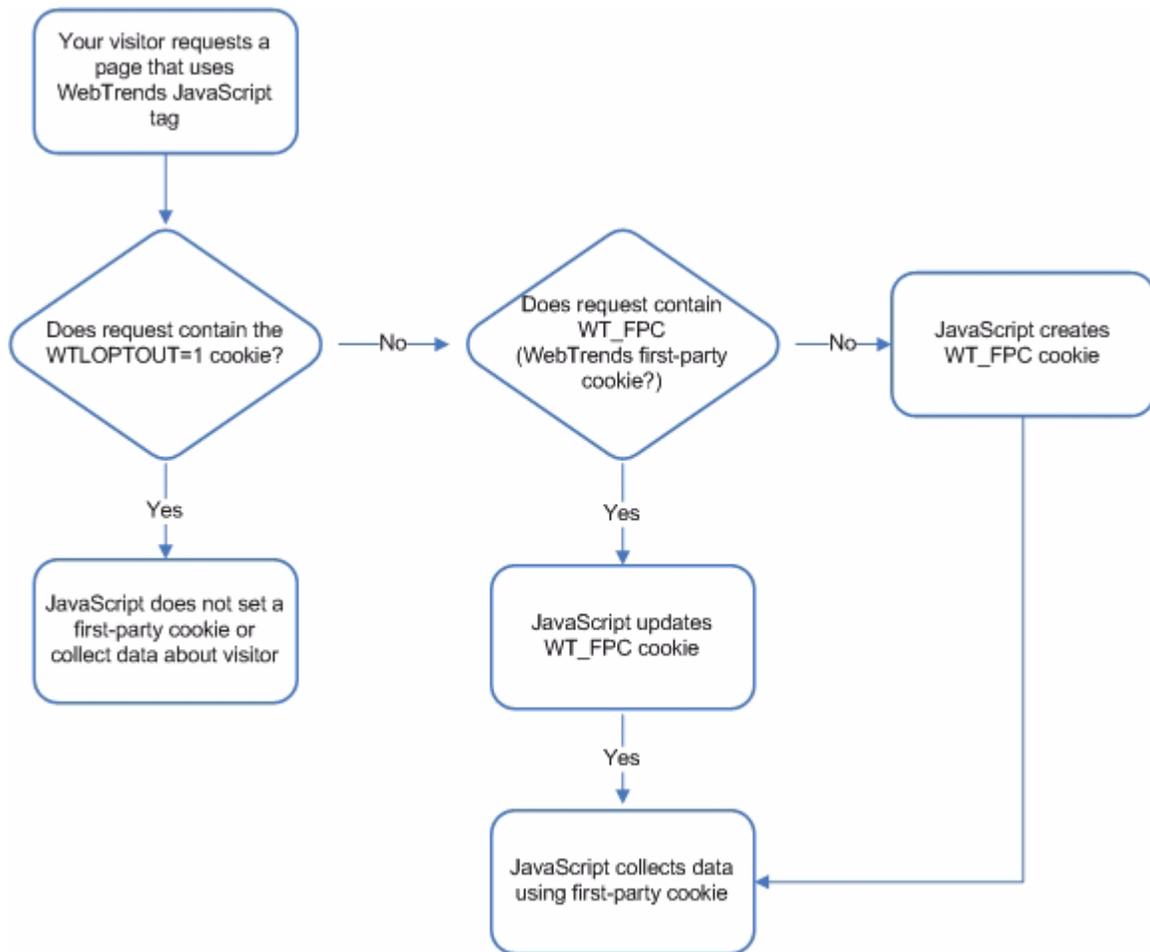
Implementing the full opt-out cookie involves the following high-level steps:

1. Write a policy about how your organization uses first-party cookies and post that policy on your site. For more information, see [“Writing an Opt-Out Policy” on page 94](#).
2. Create a web page that allows your visitors to specify their tracking preference, and implement a method on your web site to set the full opt-out cookie, WTLOPTOUT=1. For more information, see [“Creating an Opt-Out Mechanism” on page 95](#).
3. Deploy a WebTrends v8 or higher JavaScript tag on your site. For more information, see [“Implementing the JavaScript Tag” on page 95](#).



## How the Opt-Out Cookie Works

After you implement the opt-out cookie on your web site, the WebTrends JavaScript tag makes sure that no data is collected about visitors who have requested full opt-out. The following graphic shows the logic involved:



## Writing an Opt-Out Policy

As a best practice, you should create a policy that describes how your organization uses first-party cookies and the differences between anonymous opt-out and full opt-out. Make it clear to visitors that accepting the opt-out cookie from your web server is the mechanism from preventing your first-party tracking cookie from being set and prevents data about their activity your on your site from being collected. Cookie-wary visitors should be aware that if they delete the opt-out cookie, your web server identifies them as a new visitor and sets a first-party cookie that can be used to collect data.

You can view the opt-out language that WebTrends Inc. uses at the following URL:

<http://ondemand.webtrends.com/support/optout.asp>



## Creating an Opt-Out Mechanism

Your web site developer will need to implement a way to set the opt-out cookie for visitors who specify that they do not want tracked on your site. The JavaScript tag looks for a cookie named WTLOPTOUT set to a value of 1.

Because cookies are domain-specific, if you have multiple domains, you need to set the opt-out cookie for each domain. If you have multiple sub-domains, use the domain attribute in the cookie to specify the main domain. For example, `ondemand.webtrends.com` is a sub-domain of `webtrends.com`. In order to set a cookie that is identified for both the domain and all sub-domains, the web site developer passes `domain=.webtrends.com` for the domain attribute as shown in the following example:

```
WTLOPTOUT=yes expires= Thu, 14 Jan 2016 18:40:50 UTC; path=/: domain=.webtrends.com
```

The following example shows JavaScript that sets a full opt-out cookie:

```
<html >
<body>

<script type="text/javascript"><!--

// test for existence of WTLOPTOUT cookie
if (document.cookie.indexOf("WTLOPTOUT=")==-1){

// compute cookie expiration
var dCur=new Date();
var dExp=new Date(dCur.getTime()+315360000000);

// initialize cookie attributes
var expiry=""; expires=""+dExp.toGMTString();
var path=""; path="/";
var domain=""; domain=.webtrends.corp";

// WTLOPTOUT cookie does not exist so set it
document.cookie="WTLOPTOUT=1"+expiry+path+domain;
}

//-->
</script>

</body>
</html >
```

## Implementing the JavaScript Tag

If you used earlier versions of WebTrends Analytics software or WebTrends Analytics On Demand, you need to generate a new JavaScript tag for each data source and update the tag on your site. For more information, see “Implementing the JavaScript Tag On Your Web Pages” in Administration Help.

## Disabling Cookies

You can configure WebTrends Analytics On Demand and SmartSource Data Collector to not set cookies if your organization’s policy does not allow visitor data to be tracked using cookies.

## Disabling First-Party Cookies

If your organization decides not to set a first-party cookie, you can edit your SmartSource data sources and re-generate your JavaScript tag so that first-party cookie tracking is disabled.



**To disable first-party cookie tracking:**

1. In the left pane, click **Administration > Application Settings > Data Sources**.
2. Mouse over a SmartSource data source and click **Edit** on the Action menu.
3. Click **SmartSource Data Collector**.
4. Click **Tracking**.
5. Clear the **Enable First-Party Cookie Tracking** check box.
6. Click **Generate Tag**.
7. Implement the new JavaScript tag on your web site.

## Disabling Third-Party Cookies

If you use first-party cookies, you may want to disable the third-party cookies that are set by WebTrends Analytics On Demand or SmartSource Data Collector. Although setting both first-party and third-party cookies may seem unnecessary, there are several reasons that you should consider using both. WebTrends uses third-party cookies for the following purposes:

- To set a new first-party cookie for returning visitors who previously were only identifiable by the third-party cookie. If you disable third-party cookie tracking, all visitors are considered new visitors until the first-party cookie is set. This only a consideration if you are upgrading from an earlier version.
- To track visitors across multiple domains.
- To identify visitors for session tracking when third-party cookie is available, but a first-party is not available.

### Disabling Third-Party Cookies for WebTrends On Demand

You can prevent WebTrends On Demand from setting third-party cookies for all hits by editing your data source and adding the following line to the JavaScript tag:

```
DCS. dcscfg=1;
```

### Disabling Third-Party Cookies for SmartSource Data Collector

You can prevent SmartSource Data Collector (SDC) from setting third-party cookies.

- To prevent SDC from setting a third-party under any circumstance, edit the `dcs. cfg` file and set the `enabled` setting in the `[cookieserver]` section to `false`.
- To disable third-party cookies for all hits using the JavaScript tag, edit the `dcs. cfg` file and add or edit the following line: `cfgbyhit=true`. Also, edit your data source and add the following line to the JavaScript tag: `DCS. dcscfg=1;`



## Document Revision History

Table 9 contains a summary of changes made to this document beginning with the release of WebTrends Analytics, version 8.7.

**Table 9: Document Revision History**

Software Version	Date of Last Update	Summary of Changes
v8.7	March, 2009	Added Document Revision History section.
v8.7d	July, 2009	Added footer link to Documentation Center.





---

## Chapter 11

# Securing Your Implementation

This chapter discusses methods for securing your WebTrends Marketing Lab implementation.

## WebTrends and Secure Sockets Layer (SSL)

Sensitive information is often transmitted between web clients and web servers. Typically, this information is protected by sending data in an encrypted form that is decrypted on the receiving side. The Secure Sockets Layer (SSL) protocol provides several features that enable secure transmission of web data. These features include data encryption, server authentication, and message integrity. WebTrends Analytics uses Microsoft Internet Information Services (IIS) to provide the user interface. IIS supports the SSL protocol.

If you use WebTrends SmartSource Data Collector in your WebTrends Analytics software installation, see “Securing SmartSource Data Collector” in the *WebTrends SmartSource Data Collector User’s Guide* for more information.

## Enabling SSL for WebTrends Analytics

When you install WebTrends Analytics, a web site is created in Microsoft IIS that provides the user interface. To enable SSL for WebTrends Analytics, you need to configure this web server in IIS to use SSL. See the Microsoft IIS documentation for details on setting up SSL and creating valid certificates.

### To access IIS settings and online Help:

1. From the Windows control panel on the computer running your WebTrends UI Server, click **Administrative Tools**.
2. Click **Internet Information Services**.



## Document Revision History

Table 10 contains a summary of changes made to this document beginning with the release of WebTrends Analytics, version 8.7.

**Table 10: Document Revision History**

Software Version	Date of Last Update	Summary of Changes
v8.7	March, 2009	Added Document Revision History section.
v8.7d	July, 2009	Added footer link to Documentation Center.



---

## Chapter 12

# Optimizing Your WebTrends Environment

WebTrends Analytics software can be a very resource-intensive application. In addition to the resources that analysis requires, WebTrends needs to store log files, summary tables, report tables, perhaps a web data warehouse, system and external databases, IP addresses, and page titles. This chapter discusses how you can improve performance and use fewer resources. It also discusses the costs associated with limiting resource usage.

Where relevant, each section includes recommendations for handling environmental variables. These recommendations are based on the average web site requirements. However, each web site has its own unique characteristics, and you should use your own judgment and experience to adjust these recommendations to the requirements of your web site.

## Physical Data Storage

How well WebTrends performs may depend on the decisions you make about storing and backing up data. The following sections discuss how to balance log, report, and backup storage with application performance.

### Log File Rotation

With web traffic analysis that relies on web server logs, the first consideration you must make is how long to hold onto the raw, unaggregated log files. You may need to access old log files to reanalyze them. For example, you might want to reanalyze raw data based on new configuration settings. Or you might need to reanalyze the log file from a server belonging to a cluster that was not available at the original time of analysis and then add that reanalysis into an entire day's worth of logs.

In a log file, a typical hit might range roughly from 250 to 750 bytes. Given that number, consider what happens if your site experiences an average of 10,000 hits per day. This means that your log file can be anywhere from 2.5 MB to 7.5 MB. If your site experiences up to 5,000,000 hits per day (not unusual for enterprise-level organizations) your log file size can easily be several gigabytes. For large organizations with extremely active web sites, generating terabytes of data per year is common.



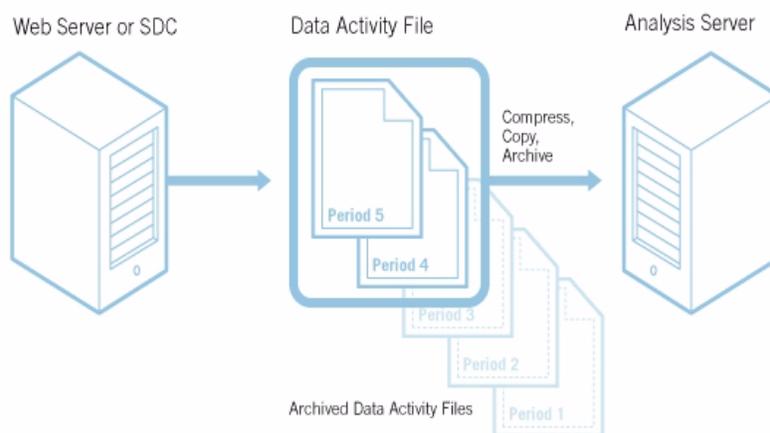
Because data activity file sizes for even a daily web data activity file can require gigabytes of storage space, most organizations implement a log file rotation scheme that keeps computing resources available for processing tasks. Depending on the volume of traffic that your site experiences, you may wish to rotate (roll over) log files daily, weekly, or monthly.



**Note**

When an IIS server rolls over daily, it closes one log file and starts a new file at 12:00 A.M. Greenwich Mean Time, not at midnight local time.

The following illustration shows a basic overview of log file rotation, rollover, and archiving.



Rotation schedules can also depend on how you access your log files, and how often you intend to report on those log files. WebTrends should always be configured to analyze log files that have been closed; that is, log files that have been rotated and will no longer be written to with new traffic. If the log files will be analyzed using FTP, the entire log file needs to be transferred to the analysis engine before analysis.



**Note**

Creating a network share on the folder where the logs are located on the remote machine will improve performance as the log files can then be analyzed directly from the remote machine without the need to first be copied to the analysis machine.

Typically, organizations rotate their log files daily, however, you can rotate log files more frequently if needed. After you rotate the log files and analyze them, determine how long to archive them. How long you archive log files depends on your reasons for keeping the data. Some organizations never intend to re-analyze their data, so they discard data shortly after analysis. Other organizations keep their data forever. Most organizations archive data for a period between one quarter and one year.

**Recommendations**

- Rotate log files daily. Consider rotating log files hourly if you access your log files using FTP, and if your site experiences a large amount of traffic.
- Archive analyzed log files for one year.



## Archiving

Archiving is a critical part of maintaining a fail-safe environment and designing an archiving strategy is particularly important in a web analytics environment. The WebTrends Archiving feature allows administrators to automate backups of WebTrends databases after incremental analysis cycles.

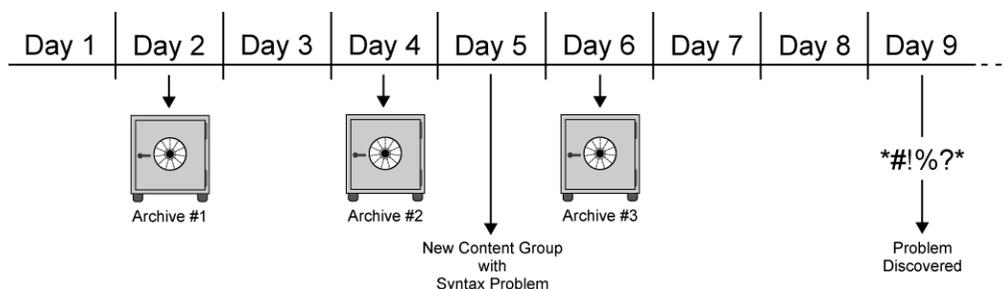
Occasionally, after you analyze data, you may need to return your analysis to a point when you knew the analysis results were based on a correct configuration. For example, suppose you add a new content group to your WebTrends installation. This content group contains a group of new pages that relate to a new product. A week later, when you review your weekly report, you discover that the content group is not included in your reports. Investigation shows that improper syntax was used to define the pages in the content group. As a result, WebTrends did not analyze hits to those pages.

If you created periodic backup copies of your summary tables database along the way, WebTrends Analytics software offers the ability to take a snapshot of the database. Depending on what the analysis software is configured to create, the snapshot may include a copy of the daily, weekly, monthly, quarterly, and/or yearly summary tables at a point in time. You can restore that copy in the event that you run into problems with your analysis later on. After you restore the data to the last known good copy, you will need to fill in the data that was not contained in that backup. This requires you to restore and re-analyze the raw log files for the data from the time of the backup to the most current log file. For more information about how to back up and restore your WebTrends Analytics installation, see [“Backing Up and Restoring WebTrends Data” on page 129](#).

Let's go back to the earlier example in which the content group was incorrectly set up. If your web site experiences a significant amount of traffic, and each daily log file analysis requires around 10 minutes to run, you might determine that you can afford the time it would take to re-analyze up to twenty-eight days of data at any given time. You also decide that 28 days is enough time to discover any issues, given that you review reports once a week. You can afford to store four backups of the data. This means that when you create a fifth backup, it replaces the oldest backup.

In this situation, a sensible solution is to back the data up every seven days, and maintain four backups. This solution allows you to maximize the amount of storage space you have and assures that you will catch any problems with the data long before your oldest archive is overwritten.

The following diagram shows the archiving scenario:



In this situation, you have two options:

1. Correct the syntax for the new content group and re-analyze the data, and then go back and import all the raw log files from day one (assuming you still have those log files).
2. Go back to the last known good set of summary tables and then re-analyze the data from that day up to the current day. In this case, you would restore Archive 2, the last archive that contained data without the syntax problem, correct the syntax for the new content group, and then you would re-analyze the raw log file data up to the current day.



As you can imagine, creating and maintaining multiple backup copies of an entire database can require substantial storage space on your computer. It's important to balance the storage space you have available with the number of backup copies you can afford to store at any given time. This calculation also depends on how long it would take to restore lost data, which in turn depends on how much traffic your site experiences, which summary tables you choose to create, and how powerful your system is.

How often you may need to back up data also depends on how closely you monitor the results of your data. If you only review results once a day, creating backups every day or every two days may be sufficient, because you will probably find any issues within a few days.

## Recommendations

- Check how much disk storage space you have to save the backups versus the average size of a backup.
- Determine how long it takes to restore data by analyzing it from the raw log file. This is affected by how much traffic your site generates, which summary tables you choose to create (daily, weekly, monthly, etc.), and how fast your system can process the data.
- Figure out how soon you are likely to find issues that may necessitate restoring a backup by how closely and frequently you monitor your analysis results.
- Make sure you store your backups in a location that allows you to restore them in the event of a disk failure. For more information about how to back up and restore your WebTrends Analytics installation, see .

## Table Limiting

Your system only has so much physical memory (called random access memory or RAM) in which to store the results of analysis. When data requirements exceed that memory, it has to use virtual memory, exchanging data as needed from RAM to the hard disk and back to RAM. This can create a low performance situation known as *thrashing*, in which a large amount of activity (swapping pages of data in and out of RAM) accomplishes very little.

Unfortunately, there is no perfect solution to the issue of overwhelming your memory with data. However, there are measures you can take to reduce how often your system has to swap data out to the disk. You can add more RAM, which up to a point will increase performance.

Most computers with 32-bit processors can address only 4 GB of memory (that is, virtual address space, regardless of how much physical RAM you might have), and they usually allocate half to user processes and half to the operating system. This limitation creates a 2 GB per-process limit. If you have 4 GB (or more) in a machine, two simultaneous user processes can each use 2GB of physical RAM. You can configure some versions of Windows versions, such as Windows 2003 Advanced Server, to provide 3 GB of memory for user processes and 1 GB for the operating system. WebTrends can use 3 GB of available memory.



Another approach is to limit the amount of data that you store in your summary database tables. The trade-off with this approach is that by limiting the amount of entries in a summary table, you only collect records up to the point when you reach that limit. For example, if you limit the Top Pages table to 10,000 pages, WebTrends only aggregates data for the first 10,000 pages entered in the table. Any new pages encountered in the Web data activity file after that will not be entered in the table, but is recorded in the “Other” section. This means that if your site experiences a great deal of traffic and has 200,000 or 300,000 pages, then limiting it to the top 10,000 will significantly impact the accuracy of your reports. However, if you were to perhaps limit it to the top 50,000, or use Virtual Top Pages, you might expect to get a reasonably accurate representation of the top pages in your reports.

In addition to requiring less storage space in RAM, limiting tables also reduces the time spent inserting data into the database. This time savings is fairly minimal in comparison to the time savings achieved by avoiding swapping data out to the hard disk.

Whether you have to limit table sizes depends on three factors:

- System processing speed
- Amount of RAM
- Tables being created (daily, weekly, monthly, quarterly, and/or yearly)

System processing speed affects how long the instructions and data must stay in main memory, while the amount of RAM affects how much data can be kept in main memory at any given time. And finally, the periods for which you have chosen to generate reports determine which tables exist and have data aggregated in them. If you have selected to aggregate data in yearly tables, toward the end of a year, you would be maintaining almost an entire year’s worth of data. Because the summary tables have to be loaded in RAM to aggregate the data, the larger the amount of data, the more likely that you may have to swap out to hard disk.

For more detailed information about limiting tables, see “Optimizing Reports Using Table Limiting” in the *WebTrends Administration User’s Guide*.

## Scheduling Reports and Storing Reports

There are several decisions you have to make about reports.

- Which reports to generate – daily, weekly, monthly, or yearly?
- How frequently to run an analysis – every five minutes, every ten minutes, or once a day?
- How long to keep a given report – do you store each daily report for one month, two months, or longer?
- How many elements to store in a report – 100, 2,000, or 20,000?

Reporting is one of the key elements to consider when deciding how to allocate resources, because the report rendering process itself uses significant system resources.

Rendering reports is a processing-intensive task. The report engine must first look up all the information requested by the report templates. It must then create tables and graphs that are populated with all the requested information. Depending on the report periods you request (such as daily, monthly, and yearly), your report engine may have to render one or more different reports for each report type.

After you create reports, each report requires a fair amount of storage space. Each stored report can occupy a fair amount of memory—up to 1 MB of memory, for example, for a basic report that comes packaged with WebTrends Analytics software.



Therefore, you should always consider the amount of time and resources involved in generating reports. For example, if it takes an hour to generate a complete day's report, and you generate a report every hour, it will take more than an hour to generate the report because of the overhead involved in shutting down and starting up processes. Your system may also experience thrashing if you generate reports too frequently.

## Recommendations

Many IT departments prune reports to contain only the tables/charts that may be of interest to the particular audience. Limiting report templates to only the reports you need makes them less daunting and more accessible, and reduces processing time and storage needs. You should track which reports are viewed by business users and remove those that are never accessed from your report templates.

## Maintaining and Storing Reports

By default, WebTrends Analytics copies the top-most elements from the analysis tables to the report database (also called the On Demand Database). You can increase the number of elements WebTrends Analytics copies to the report database, but as you increase this number the performance of the On Demand Database decreases. In general, we recommend limiting the On Demand Database for best performance.

WebTrends also allows you to control the number of reports kept over a period of time. You could, for example:

- Delete all daily reports that are more than 90 days old.
- Keep weekly reports only over the last 52 weeks.
- Keep only eight quarterly reports and two yearly reports.

By limiting the number of reports to keep in the On Demand Database, you can reduce the storage space required.

There is a trade-off between keeping large amounts of data and maintaining a robust database that generates reports efficiently. Some organizations may find great value in keeping a large volume of historical data, no matter what the cost in performance. Other organizations may find little value in maintaining daily reports from the previous year. This decision depends on your organization's needs and resources.

### To limit the data stored in the On Demand database:

In the left pane, click **Administration > Application Setting > System Management > Retention > Report**.



## Performance Issues

### FTP Caching

If you are analyzing a log file that you must access using FTP, you will need to physically transfer that log file to a local drive. You can either use your WebTrends Analytics software to take care of the log file transfer, or you may set up your own procedure to bring the log files over before running the analysis. After the log file is stored locally, you can use the WebTrends Analytics software to unzip the compressed file, or you can set up your own process. Either way, after you move the log file to your local drive, you must decide how long to store it. Just as with uncompressed web data, this decision depends on how often you expect to re-analyze the log file, how much data the log file contains (which affects how long it takes to transfer the log file using FTP), and how much local storage space you can afford to allocate.

Consider the following choices for handling your FTP cache:

- Delete the file from the cache after analysis completes.
- Keep the file in the cache for a specified number of days.
- Keep the file in the cache until the cache reaches a maximum size, at which point the oldest files in the cache will be replaced by new, incoming files.
- Keep the file in the cache, but delete the file if it is not accessed within a specified period of days.

### Internet Resolution

When your web server generates a log file, it can either be configured to look up the client machine's IP address as it creates the log file in a process known as reverse DNS, or it can leave the IP address unresolved. The more efficient approach is to look up the IP address during log file creation; however, because this process (known as Internet resolution) takes some of the server's resources to perform this lookup, web site content delivery may be negatively affected. For this reason, many web servers are not configured to perform a lookup.

The reality is that when reviewing reports about your visitors, just receiving the IP address of your visitor does not give you much insight. An IP address can't let you easily see that many of your visitors come from the competition, or that many of your visitors come from a company with whom you are trying to establish more business.

IP addresses also affect visitor counts, because multiple IP addresses can resolve to the same domain name.

WebTrends Analytics software gives you the option to look up IP addresses from DNS servers. After a lookup, WebTrends stores IP addresses in a cache so future analyses can use the information locally, rather than having to go through DNS servers to locate the information. You need to determine the value of having IP addresses translated into meaningful names versus the loss of disk space that the cache of resolved addresses occupies. Typically, the address cache has a size limit. When the cache reaches the limit the oldest entries are deleted to make room for more recent ones. Because DNS lookups use memory, you should also weigh the impact of looking up IP addresses on analysis performance.

### Recommendations

- Determine how important it is to have the looked up values of IP addresses in your reports. The space required by these looked up values can be fairly minimal, but the performance slowdown can be noticeable. Most people tend to have the lookup performed if the Web server did not already do this.



- Note that a company may use many IP addresses that are assigned to them but only register a few of these addresses as domains. For example, a company may have many proxy servers with addresses that connect to the Internet, yet since the company doesn't expect anyone to connect to the proxy, it hasn't assigned a domain to the proxy. Consider using WebTrends GeoTrends, which will resolve IP addresses more accurately than DNS. That is, GeoTrends identifies the companies that registered the IP addresses. GeoTrends also provides pertinent geographic and demographic information for your web analysis.

**To specify Internet resolution settings:**

1. In the left pane, click **Administration > Web Analysis > Options > Analysis**.
2. Click **Internet Resolution**.

## HTML Page Title Lookups

In the log file, requested content is recorded as the URL for that item. The URL could be for a gif or a jpeg image, it could be for a downloaded file, or it could be for a page. WebTrends Analytics software can look up the actual page titles that are recorded in the Title tags in each HTML page. However, if you choose this option, you will have to dedicate some space on your hard drive for the results of the page title lookups. Just like the resolved IP address cache, the cache for HTML page title lookups is managed by setting either the maximum number of entries allowed in the cache at any given time, and/or the maximum number of days that a page title can remain in the cache. Again, you have to balance the usefulness of the looked up titles against the cache space they require and the performance hit your system takes during the initial lookup.

### Recommendation

Determine how important it is to have the HTML page titles of the URLs in your reports. The space required by these looked-up values can be fairly minimal, but the performance slowdown can be noticeable. Most people perform the lookup to make reports more meaningful.

**Note**

Web site security can impede or prevent HTML title lookups. You may need to configure a user name and password to get the data.

**To specify whether to look up HTML page titles:**

1. In the left pane, click **Administration > Web Analysis > Options > Analysis**.
2. Click **General**.

## Simultaneous Analysis

WebTrends Analytics software is a multi-threaded application, meaning that it can run multiple processes simultaneously. Depending on the number and speed of the processors and memory in your Analysis Engine system, you may increase performance by running more than one analysis at a time.

### Recommendations

- Have no more than one simultaneous analysis for each processor in the analysis system.



- Each processor should have at least 2 GB of RAM.

## Performance Tuning for Your WebTrends Database

This section explains how to improve database performance after you install your SQL Server database and WebTrends software. Database tuning can speed up report viewing and streamline the interaction between the database and the EDB loader engine.

### Improving Common Performance Problem Areas

After installing your database, you can improve common performance problem areas by maintaining indexes and statistics and monitoring performance counters as described in the following sections.

#### Maintaining Indexes

An index is a structure that speeds retrieval of rows from a table. An index contains keys built from one or more columns in the table. As you modify data in a table, the information in the index becomes scattered, or fragmented. Because heavily fragmented indexes can degrade query performance and reduce the response time of WebTrends, you should maintain your index by periodically rebuilding or defragmenting them.

The following Microsoft knowledge base articles explain the differences between rebuilding and defragmenting as well as describe how to implement these processes.

##### SQL Server 2005

<http://msdn2.microsoft.com/en-us/library/ms189858.aspx>

##### SQL Server 2000

<http://msdn2.microsoft.com/en-us/library/aa933127.aspx>

#### Monitoring Performance Counters

You can eliminate performance counters by using the Waits and Queues methodology, which is an effective way to quickly identify and resolve application performance problems because it enables you to discover new and potentially unexpected problem areas with limited guesswork. For more information about this methodology, see the following URL:

[http://www.microsoft.com/technet/prodtechnol/sql/bestpractice/performance\\_tuning\\_waits\\_queues.mspx](http://www.microsoft.com/technet/prodtechnol/sql/bestpractice/performance_tuning_waits_queues.mspx)

You should increase memory or improve processing based on the recommendations provided by Microsoft.

### Determining Bottlenecks

Microsoft TechNet provides a useful article to help you determine bottlenecks and narrow down the cause of a performance issue. For more information, see "[Troubleshooting Performance Problems in SQL Server 2005](#)."



## Document Revision History

Table 11 contains a summary of changes made to this document beginning with the release of WebTrends Analytics, version 8.7.

**Table 11: Document Revision History**

Software Version	Date of Last Update	Summary of Changes
v8.7	March, 2009	Added Document Revision History section.
v8.7d	July, 2009	Added footer link to Documentation Center.



---

## Chapter 13

# Customizing JavaScript Tags

The WebTrends Tag Builder produces code known as the JavaScript tag. The WebTrends JavaScript tag contains code that enables you to collect visitor data after the tag is implemented on your Web site. This chapter discusses the contents of the tag before you modify it, and covers more advanced ways you can modify it to get additional information.

If you use WebTrends On Demand or SmartSource Data Collector and want to learn to tag your Web site and create META tags that automatically collect visitor information, see [Chapter 7, "Client-Side JavaScript Integration"](#) on page 61.

## Understanding the Basic JavaScript Tag

This section breaks the tag into functional parts and describes how the tag performs data collection.

The JavaScript tag is divided into the following two files:

- **webtrends.html**, which contains the markup you place in the HTML body of your Web pages.
- **webtrends.js**, which contains the JavaScript file that you host on your Web server.

## Inline HTML (webtrends.html)

The markup that goes in the HTML body of your Web pages is enclosed inside HTML comments similar to the following:

```
<!-- START OF SmartSource Data Collector TAG -->  
<!-- Copyright (c) 1996-2008 WebTrends Inc. All rights reserved. -->  
<!-- Version: 8.5.0 -->  
<!-- Tag Builder Version: 2.1.0 -->  
<!-- Created: 9/23/2008 10:32:55 AM -->  
  
...  
<!-- END OF SmartSource Data Collector TAG -->
```

The markup is comprised of three HTML `<script>` blocks, and one HTML `<noscript>` block.

## External Script File

The first HTML `<script>` block contains a `src` attribute referencing an external script file named **webtrends.js**:

```
<script src="webtrends.js" type="text/javascript"></script>
```

Note that the file location specified in the `src` attribute can be a full or relative URI. This depends on where it resides on your Web server.



The `webtrends.js` file contains the object constructor function named `WebTrends`. This function is used to store, manipulate, and send data to WebTrends data collection servers.

## Embedded JavaScript

The second and third HTML `<script>` blocks contain embedded JavaScript. These blocks must remain separate for the tag to function properly (that is, they cannot be combined into one block).

## CDATA

The embedded JavaScript in both blocks is wrapped inside CDATA sections:

```
//<![CDATA[  
...  
//]]>>
```

CDATA sections are used to make the tag XHTML compliant. Certain characters found in embedded JavaScript will cause XML parsers to fail ("`>`" and "`;`"). XML parsers ignore everything inside a CDATA section, so the tag will not cause a parsing failure.

The CDATA sections are commented out using JavaScript single-line comments ("`//`"). This prevents the JavaScript interpreter from throwing an error.

If XHTML compliance is not a requirement, then the CDATA sections may be removed.

## Initialization

The second HTML `<script>` block performs tag initialization.

```
var _tag=new WebTrends();  
_tag.dcsGetId();
```

The first line instantiates an object named `_tag` using the `WebTrends` object constructor function (defined in `webtrends.js`). The `_tag` object serves as a namespace that we can use to access properties and invoke methods.

The second line invokes a method of the `_tag` object named `dcsGetId()`. This method performs an inline JavaScript request for the file `wtid.js`. This request is used to obtain a unique visitor identifier that is persisted in the WebTrends first party cookie.

## Collection

The third HTML `<script>` block performs data collection.

```
_tag.dcsCollect();
```

This line invokes a method of the `_tag` object named `dcsCollect()`. This method performs an image request for the file `dcs.gif`. The image request is the mechanism used to pass information to WebTrends data collection servers.

## Script Disabled

The `<noscript>` block contains an inline image request similar to the following:



```
<div></div>
```

The request is hard-coded to contain a special URI-stem and a minimal subset of WebTrends Query Parameters. You may add parameters if necessary.

The `dcsuri` parameter is logged as the URI-stem with the value of `/noj avascript`.

WebTrends Analytics can be configured to report on this filename. `WT.js=No` tells WebTrends Analytics that the page view was from a non-JavaScript enabled browser.

The `<img>` tag is wrapped inside a `<div>` tag to make the tag XHTML compliant. If XHTML compliance is not important to you, then the `<div>` tag may be removed.

## External JavaScript File (webtrends.js)

The **webtrends.js** file contains an object constructor function named `WebTrends`. This function can be thought of as a class containing properties and methods. All properties and methods are publicly available, and an object of this class can be instantiated using the new operator. Once you have instantiated an object, you can modify properties, and invoke methods.

### Construction

The inline HTML portion of the tag uses the JavaScript new operator to call the object constructor function as follows:

```
var _tag=new WebTrends();
```

In this case, the resultant object is named `_tag`. All properties and methods are accessed using the `_tag` qualifier.

During construction, properties are initialized using settings specified during tag creation process. For example, suppose you specified a `dcsID` value of `dcs5w0txb10000wocrvqy1nqm_6n1p` when you created the tag. At construction time, the `dcsID` property is initialized as follows:

```
this.dcsid="dcs5w0txb10000wocrvqy1nqm_6n1p";
```

This property can be accessed using the `_tag` qualifier as follows:

```
alert("The DCSID is: " + _tag.dcsid);
```

### Properties

The `WebTrends` function contains several properties. All properties are public in scope. Some properties are designed to be modified, others are read-only. Some properties are only present when applicable options were selected during tag creation. As the tag evolves over time, new properties are likely to be added.

To access a property from within a `WebTrends` constructor function method, prefix the property with `this`. For example, the `dcsTag` method accesses the `WT.ad` property as follows:

```
this.WT.ad="";
```

To access a property globally, prefix the call with `_tag`. For example, a custom parameter could be set from the `<script>` block in the HTML inline as follows:

```
_tag.DCSext.team="Mariners";
```



## User Modifiable Properties

Property	Type	Present when...	Description
dcsid	String	always	SmartSource site id (DCSID)
domain	String	always	Data collection server domain name or IP address
timezone	Number	always	Timezone of customer web site (integers between -12 and 12)
fpcdom	String	always	WebTrends First Party Cookie domain attribute
onsitedoms	String	track clicks to download, offsite, or anchor links are enabled	Onsite domain names (comma separated list or regular expression)
downloadtypes	String	track clicks to download links is enabled	Download file types (comma separated list)
rightclicktypes	String	track right clicks to download links is enabled	Right click file types (comma separated list)
navigatontag	String	any click event tracking enabled and navigation area selected is DIV or TABLE	Navigation area that encloses a link or button (None, DIV, TABLE)
customerfpc	String	use customer provided cookie is enabled	Name of cookie used to seed visitor id value (WT.co_f)
adclickparam	String	create ad view from links containing this parameter is enabled	Name of parameter used to designate an ad click (default is WT.ac)
metanames	String	capture these custom meta tags is enabled	Names of custom meta tags (comma separated list)
trackevents	Boolean	any click event tracking enabled	Programmatically enable/disable click event tracking on a page



evi	Object	set query parameter using this cookie is enabled	Object that contains properties for cookie name (cookie), query parameter to set (qp), cookie crumb (crumb), cookie crumb separator (sep)
enabled	Boolean	always	Programmatically enable/disable data collection by tag
i18n	Boolean	always	Programmatically enable/disable extended internationalization support
fpc	String	always	WebTrends First Party Cookie name



## Read Only Properties

Property	Type	Present when...	Description
that	Object	always	Private variable used to store "this" object for closure
DCS	Object	always	Properties mapped to SDC-specific query parameters
WT	Object	always	Properties mapped to WebTrends Query Parameters
DCSext	Object	always	Properties mapped to custom query parameters
images	Array	always	Image array that holds 1x1 pixel request (dcs.gif)
index	Number	always	Index into image array
qp	Array	assign custom query parameters to WebTrends parameters enabled	Temporary storage of custom query parameters assigned using the dcsQP() function
exre	Object	always	Anonymous function containing regular expression
re	Object	always	Anonymous function containing regular expression list used to URL encode query parameter values
vtid	String	use alternate WebTrends Visitor Id is enabled	Alternate visitor id for Warehouse seeded by query parameter or cookie

## Methods

The WebTrends constructor function contains several methods. All methods are public in scope. Some methods are only present when applicable options were selected during tag creation. As the tag evolves over time, new methods are likely to be added.

Generally, the method names are prefixed with `dcs`. To access a method from within another method, prefix the call with `this..` For example, the `dcsTag` method calls the `dcsCreateImage` method as follows:

```
this.dcsCreateImage(P);
```

To access a method globally, prefix the call with `_tag..` For example, the HTML inline `<script>` tag calls the `dcsCollect` method as follows:



```
_tag.dcsCollect();
```

The following table contains the high level methods of consequence contained in the tag.

Method	Return type	Parameter list	Description
dcsGetId	none	none	Performs an inline JavaScript request to data collection servers. If no WebTrends First Party Cookie is present on the client, a request is for the file wtid.js occurs. This file contains an unique visitor identifier that is persisted in the WebTrends First Party Cookie and passed via a WebTrends Query Parameter (WT.co_f).
dcsGetCookie	String containing cookie value or null	name - String containing cookie name	Reads a cookie name and returns its value.
dcsGetCrumb	String containing cookie crumb value	cval - String containing cookie value crumb - String containing cookie crumb name sep - String containing cookie crumb separator	Reads a cookie crumb from WebTrends First Party cookie and returns the crumb value.
dcsGetIdCrumb	String containing cookie crumb value	cval - String containing cookie value crumb - String containing cookie crumb name	Reads the "id" cookie crumb from WebTrends First Party cookie and returns the crumb value.
dcsIsFpcSet	Number indicating if cookie was set (0=success, non-0=failure)	name - String containing cookie name id - String containing id crumb value lv - String that is the lv crumb value ss - String containing ss crumb value	Tests to see that the WebTrends First Party Cookie was successfully created.
dcsFpc	none	none	Sets WebTrends First Party Cookie and associated visit-related visit query parameters.
dcsOther	none	none	Reads customer provided cookie (specified in customerfpc property) and uses its value for unique visitor identification.



dcstP	none	none	Tracks activity for SmartView using a meta tag. This function sets a cookie named WT_DC to identify SmartView pages. As a best practice, you should not modify any code in this function. For more information about using your JavaScript tag to track pages for SmartView, see the SmartView User's Guide.
dcsGetMeta	String containing meta tag content or null	name – String containing meta tag name	Reads a meta tag name and returns its content.
dcsAdSearch	none	none	Scans page for links containing ad click query parameters (specified in adclickparam property) and populates WebTrends Query Parameter for ad view (WT.ad).
dcsQP	String containing query parameter value or empty string	N – String containing query parameter name to search for	Search query string for a query parameter name and return its value.
dcsIsOnsite	Boolean indicating if domain name is onsite (true=onsite, false=not onsite)	host – String containing domain name to test	Compares domain name with list of onsite domains (specified in onsitedoms property) and determines if a match occurred.
dcsTypeMatch	Boolean indicating if a given filename type matches a list of types	pth – String containing filename to test typelist – String containing the list of file types	Extracts filename type and compares it with a list of filetypes and determines if a match occurred.
dcsEvt	Object containing click event	evt – Object containing event tag – String containing HTML tag name	Walks up the DOM hierarchy looking for a click target event with matching HTML tag name. Helper method for event handlers.
dcsNavigation	String containing element id or classname	evt – Object containing event	Walks up the DOM hierarchy looking for a matching HTML tag name (specified in navigation tag property). Helper method for event handlers.
dcsBind	none	event – String containing mouse event func – Object containing event handler function	Binds event handling function to mouse event.



dcsET	none	none	Binds event handling functions to mouse event.
dcsMultiTrack	none	String pairs containing query parameter key/values to send to data collection servers	Helper method for globally available dcsMultiTrack function.
dcsCleanup	none	String pairs containing query parameter key/values to retain	Clears out values in query parameter storage objects. Used in conjunction with dcsMultiTrack. String pairs passed as parameters are retained.
dcsSetProps	none	String pairs containing query parameter key/values to retain	Helper method for dcsMultiTrack, dcsCleanup.
dcsSplit	Array of items	list - String containing comma separated list of items	Parses comma separated list of items into array and removes leading and trailing white space from each item.
dcsDownload	none	Object containing event	Download links click event handler.
dcsRightClick	none	Object containing event	Right click download links click event handler.
dcsDynamic	none	Object containing event	Dynamic links click event handler.
dcsFormButton	none	Object containing event	Form button click event handler.
dcsOffsite	none	Object containing event	Offsite links click event handler.
dcsAnchor	none	Object containing event	Anchor links click event handler.
dcsImageMap	none	Object containing event	Image map click event handler.
dcsMetaCap	none	none	Reads meta tags looking for matching names (specified in metanames property) and assigns content values to custom query parameters.
dcsEvi	none	none	Reads cookie (specified in evi.cookie property) and assigns value to query parameter (specified in evi.qp property). Optionally reads cookie crumb (specified in evi.crumb property) using separator (specified in evi.sep property).



dcsAdv	none	none	Wrapper method that dispatches several lower-level utility methods. Each utility method corresponds with a particular feature.
dcsVar	none	none	Gathers page data and assigns it to query parameter storage objects.
dcsEscape	String containing encoded value	S – String containing value to encode REL – RegEx object containing list of characters to encode	URL-encodes query parameter values.
dcsA	String containing query parameter key/value	N – String containing query parameter key V – String containing query parameter value	Helper function used to compose data collection URL.
dcsEncode	String containing encoded query parameter value	S – String containing query parameter value	Helper function used to URL-encode query parameter values.
dcsCreateImage	none	dcsSrc – String containing data collection URL	Populates the Image array with the data collection URL. This causes the image request to occur.
dcsMeta	none	none	Reads meta tags looking for names containing WT, DCS, or DCSExt. If found, the content values are assigned to corresponding query parameter storage objects. This method is called after dcsVar so that meta tag assignments tag precedence over all other query parameter assignments.
dcsTag	none	none	Composes data collection URL from query parameter storage objects and passes URL to dcsCreateImage method.
dcsCollect	none	none	High level function called to perform data collection request.
wtbody			Extends function object to maintain “this” object using a closure. Used by click event handlers.



## Global Functions

In addition to the methods found in the WebTrends constructor function, the tag contains a few globally accessible functions. Access these functions through the global namespace using `window.` prefix rather than `_tag.` prefix.

Function	Return type	Parameter list	Description
<code>dcsMultiTrack</code>	none	String pairs that are query parameter key/values to send to data collection servers	Sends data collection request. Typically in response to an event.
<code>dcsDebug</code>	none	none	Creates popup window with data collection URL components.

## URL Encoding

Certain characters can cause problems when used in query parameter values. For example, for a WebTrends query parameter assignment of `WT.ti="The Gettysburg Address";` WebTrends Data Collection servers write the following value to the log file:

```
&WT.ti=The Gettysburg Address
```

The space characters in this value cause problems because the space character is used to separate fields within a log file. The solution is to URL encode all query parameter values. URL encoding means replacing certain characters with their hexadecimal equivalents of the form `%XX` where `%` is the escaping character and `XX` is the character's numeric ASCII value. URL encoded characters are properly rendered in WebTrends reports. Continuing with this example, the URL-encoded form is as follows:

```
&WT.ti=The%20Gettysburg%20Address
```

Note that space characters have been replaced by `%20`. The tag URL encodes the following characters: tab, space, `#`, `&`, `+`, `?`, `"`, `\`, and non-breaking spaces. These characters are defined in the regular expression list. The regular expression list contains regular expressions to search for, and the corresponding `%XX` replacement strings. Regular expression properties are used as arguments to the `string.replace` method. The tag URL encodes parameter values by passing them as arguments into the `dcsEscape` method.

## Image Array

The tag uses an array of Image objects so that many hits to SDC from a single Web page can be made in rapid succession. The data collection request URL is assigned to the Image object's `src` property. This causes the browser to load the image into the image cache. This is a fast and efficient way to load the image that lends itself to repeated invocations of the tag.

## Query Parameter Storage Objects

The following objects are properties of the WebTrends constructor function: DCS, WT, DCSext.



Each object contains query parameter mappings. Property name/value pairs correspond to the query parameter name/value pairs that are ultimately passed to WebTrends Data Collection servers. Each object serves as a container to group together like types of query parameters.

Object	Query Parameter Types Contained Therein
DCS	SDC-specific
WT	WebTrends Query Parameters
DCSext	User-defined

Adding customized information to the tag involves assigning property name/value pairs to the appropriate object. During object instantiation, empty objects are created. You can manually assign customized query parameters to the objects. For more information about adding custom query parameters, see [“Adding Customized Information” on page 123](#). The `dcsMeta` and `dcsVar` methods automatically populate the objects with property name/value pairs. The `dcsTag` method enumerates values contained in each object, and composes the query parameter string that is passed to SDC.

### SDC-Specific Query Parameters (DCS Object)

The DCS Object stores SDC-specific query parameters. The property names correspond to the following SDC-specific parameters: `dcsqry`, `dcsref`, `dcsaut`, `dcsmet`, `dcssta`, `dcSSIP`, `dcspro`, `dcsbyt`, `dcsdat`, `dcsP3p`, `dcscfg`, `dcscip`.

For example, by default the `dcsref` parameter is assigned as shown here:

```
DCS.dcsref=window.document.referrer;
```

This assignment is transformed into a query parameter as follows:

```
&dcsref=encoded contents of DCS["dcsref"]
```

For more information about using each of these query parameters, see [“SDC-Parameter Override Parameters” on page 187](#).

### WebTrends Query Parameters (WT Object)

The WT object stores WebTrends query parameters. The property names correspond to the type/attribute portion of the WebTrends query parameter.

For example, by default the `WT.ti` parameter is assigned this way:

```
WT.ti=document.title;
```

This is transformed into a query parameter as follows:

```
&WT.ti=contents of WT["ti"]
```

Note that `WT.` is added to the beginning of the object property name to create the query parameter key.

### User-Defined Query Parameters (DCSext Object)

The DCSext object is used to store user-defined or “custom” query parameters.



For example, suppose you want to insert your own query parameter, `apparel=jacket`. This is how the `DCSext.apparel` parameter is assigned:

```
DCSext. apparel ="j acket";
```

Alternatively, you can use the meta tag:

```
<meta name="DCSext. apparel " content="j acket">
```

which is transformed into the following query parameter:

```
&apparel =j acket
```

Be careful when naming user-defined query parameters. Do not use any of the parameter names used by SDC-specific parameters or by the WebTrends query parameters because those names are reserved.

## Extended Internationalization Support

The JavaScript tag has extended internationalization support primarily for handling search phrases and query parameters in Asian languages. To enable this support, set the `i18n` property `true`. This support must be used in conjunction with WebTrends Conversion Plug-in. For more information, see “Internationalization and WebTrends” in the *WebTrends Administration User’s Guide*.

## Adding Customized Information

You can add customized query parameters by assigning property name/value pairs to the query parameter storage objects: `DCS`, `WT`, `DCSext`. As a best practice, you should add customized query parameters through HTML meta tags. You can do this quickly and easily using an HTML authoring tool. For more information about this process, see “Tagging Web Pages for SDC” chapter in *SmartSource Data Collector User’s Guide*.

Alternatively, you can assign properties directly to the object in the inline HTML portion of the tag, beneath the “Add custom parameters here” comment. Note that in the case of a redundant parameter definition, an HTML meta tag assignment takes precedence over a direct object assignment. This permits override capability through HTML meta tags.

## SDC-Specific Query Parameters

The JavaScript tag assigns several of the SDC-specific query parameters in the `dcsvar` method by default. Typically, you should not change these parameter assignments because they are required for base functionality. In a few cases, you can define additional SDC-specific parameters for added functionality.

The following example uses the `dcscfg=1` parameter to disable WebTrends Third Party Cookie.

You could use the following meta tag definition:

```
<meta name="DCS. dcscfg" content="1">
```

Alternatively, you could assign a value directly:

```
_tag. DCS. dcscfg="1";
```

Either approach results in the following query parameter definition:

```
&dcscfg=1
```



## WebTrends Query Parameters

WebTrends Query Parameters are specially designed parameters that provide a powerful way to collect meaningful web data. WebTrends Query Parameters are passed in URLs through JavaScript to WebTrends Analytics for analysis. For more information, see [“WebTrends Query Parameter Reference” on page 145](#).

By default, the WebTrends JavaScript tag defines several WebTrends query parameters. These parameters are used by pre-configured custom reports in WebTrends. You can get additional insight by adding parameters recognized by the WebTrends Auto-Configuration feature.

For example, suppose you want a page to indicate that a shopper had just added a “widget” to the shopping cart. To use the WebTrends preconfigured “Shopping Cart” Scenario Analysis, you need to set the following parameters: WT. pn\_sku, WT. si\_n, WT. si\_p or WT. si\_x.

One way to pass these parameters is by using meta tags on your Web pages. In this example, the following meta tag definitions are used:

```
<meta name="WT. pn_sku" content="91x2G439bnM">
<meta name="WT. si_n" content="ShoppingCart">
<meta name="WT. si_p" content="CartAdd">
```

Alternatively, you could assign the parameters directly. In this case, the parameters are used as shown in the following example:

```
_tag.WT. pn_sku=91x2G439bnM;
_tag.WT. si_n=ShoppingCart;
_tag.WT. si_p=CartAdd;
```

Either approach results in the following query parameter definitions:

```
&WT. pn_sku=91x2G439bnM&WT. si_n=ShoppingCart&WT. si_p=CartAdd
```

## User-Defined Query Parameters

You can also use custom (user-defined) query parameters. These query parameters are typically used in WebTrends custom reports as measures or dimensions. Be careful when naming custom query parameters. Do not use any of the parameter names used by SDC-specific parameters or by WebTrends query parameters because these names are reserved.

For example, to add a custom query parameter named wtprod and assign it the value WebTrends, you can use the following meta tag definition:

```
<meta name="DCSext.wtprod" content="WebTrends">
```

Alternatively, you could directly assign the parameter value as follows:

```
_tag.DCSext.wtprod="WebTrends";
```

Either approach results in the following query parameter definition:

```
&wtprod=WebTrends
```



## Debugging Your Customizations

To debug the WebTrends JavaScript tag, there is no substitute for having a source level debugger. Microsoft Visual Studio for Internet Explorer and Venkman or Firebug for Firefox are both invaluable for setting breakpoints and inspecting variables. Browser plug-ins that permit sniffing of HTTP requests are also useful. HTTPWatch for Internet Explorer and HTTPFox or Firebug for Firefox can be handy. And finally, you may write JavaScript code to help inspect the data being sent to data collection servers. The following are a couple of ideas with regard to debugging.

### Using `dcsDebug`

The tag contains a global function named `dcsDebug` that can be used to inspect the most recent data collection request. Invoke this function from the address bar of your browser as follows:

```
javascript:dcsDebug()
```

This will create a popup window that presents the data collection URL in an easily to read form. The URL is broken up into it's constituent parts: protocol, domain, path, query parameters. Additionally, cookie information is presented.

Note that your browser may be configured to disable popup windows. If this is the case, the browser typically displays a warning, and asks if you would like to enable the popup. An affirmative response will display the debug information.

To perform simple debugging tasks, it is sometimes useful to use JavaScript itself to display the contents of variables.

The following examples contain some code snippets that are useful for customizing the tag.

### Displaying the URL

Another way to look at the data collection URL is to a `document.write` statement inside the `dcsCreateImage` method as follows:

```
WebTrends.prototype.dcsCreateImage=function(dcsSrc){
```

```
    // Add the following line
    document.write(dcsSrc);

    if (document.images){
        this.images[this.index]=new Image();
        this.images[this.index].src=dcsSrc;
        this.index++;
    }
    else{
        document.write('<IMG ALT="" BORDER="0" NAME="DCSIMG" WIDTH="1" HEIGHT="1"
        SRC="'+dcsSrc+'">');
    }
}
```



This technique displays the URL directly into the Web page. Alternatively, you can use the JavaScript `alert` function instead of `document.write` to display the URL in an alert dialog.

## Displaying the Query Parameter Storage Objects

The `dcsDebugObjects` function below shows a way to inspect the contents of the query parameter storage objects used by the tag:

```
function dcsDebugObjects(){
    if (typeof(_tag)!="undefined"){
        var DCS=_tag.DCS;
        var WT=_tag.WT;
        var DCSExt=_tag.DCSExt;
        var prop="";
        var data="";
        for (prop in DCS){
            if (DCS[prop]){
                data+="\nDCS. "+prop+" = "+DCS[prop];
            }
        }
        for (prop in WT){
            if (WT[prop]){
                data+="\nWT. "+prop+" = "+WT[prop];
            }
        }
        for (prop in DCSExt){
            if (DCSExt[prop]){
                data+="\nDCSExt. "+prop+" = "+DCSExt[prop];
            }
        }
        window.alert(data);
    }
}
```



This function displays the contents of each object in an alert dialog. It is presented here to give an idea of how to access these important data structures in the tag.

## Document Revision History

Table 12 contains a summary of changes made to this document beginning with the release of WebTrends Analytics, version 8.7.

**Table 12: Document Revision History**

Software Version	Date of Last Update	Summary of Changes
v8.7	March, 2009	Added Document Revision History section.
v8.7d	July, 2009	Added footer link to Documentation Center.





---

## Chapter 14

# Backing Up and Restoring WebTrends Data

This chapter describes how to back up your WebTrends Marketing Lab installation and how to restore it in the event that your data becomes corrupt. It includes general information about data storage, specific instructions for backing up and restoring components of WebTrends Analytics and WebTrends Marketing Warehouse data and configuration, and some discussion of how to recover from a large-scale failure of your installation.

Information about data restoration is limited to instructions for restoring your profile data from a recent backup. If you have a serious failure or you need to restore a large amount of historical data, you should contact Support for assistance.

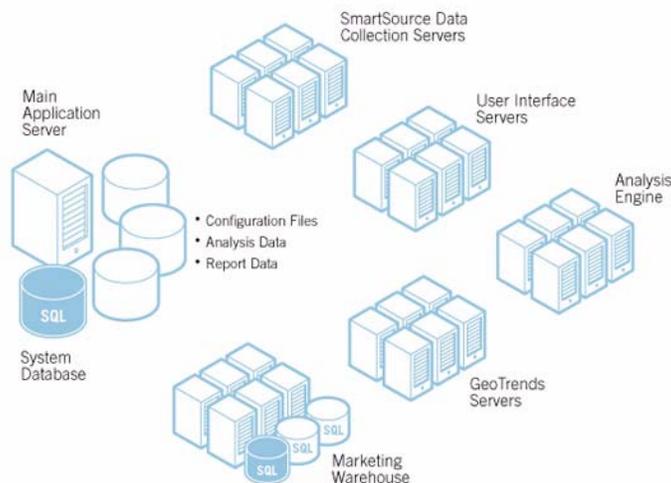
## How WebTrends Stores Data

WebTrends Analytics software stores your data both in user-defined file locations and in the WebTrends system database. To completely restore your WebTrends installation, you need to restore these files and the WebTrends system database. WebTrends recommends that you keep ongoing, coordinated backups of these components so that, in the event that you need to restore your data, you can restore it from closely aligned time periods.

WebTrends Marketing Warehouse software stores data in several different Microsoft SQL Server database locations as well as in the WebTrends system database; WebTrends Visitor Intelligence also stores data in an OLAP cube configuration. If you use Marketing Warehouse data, you should back up these locations regularly. For more information about required backups for Marketing Warehouse data, see [“Backing Up WebTrends Marketing Warehouse Data” on page 134](#).



WebTrends 7.0 and higher provides the ability to scale individual components by adding more computers to accommodate the need for greater size and capacity. The following diagram shows a typical WebTrends Distributed Architecture environment:



All data in this typical scenario is stored on the main application server. As you can see from the number of components with more than one server, this configuration provides some degree of fail-over and redundancy for many critical pieces of the operation.

The WebTrends System Database is the central repository for all system-level configuration information. The System Database can run on either Microsoft SQL Server 2005 Express Edition, or Microsoft SQL Server 2000 Standard or Enterprise Edition. The WebTrends environment cannot run without the System Database intact, so a regular backup is critical to every operation. Microsoft SQL Server users should use their preferred tools and processes for backups.

## Built-In Redundancies

As noted previously, many components of the Distributed Architecture environment can be easily built to expand for capacity or provide a higher level of availability by adding more systems. The following systems can be redundantly configured (and load balanced) or rebuilt relatively easily and quickly if no backup is available:

- The SmartSource Data Collector
- The User Interface server
- The WebTrends Analysis Engine
- The Event Database Loader
- The GeoTrends database

These systems do not hold any data permanently, and are very quick to install and configure.



## WebTrends Data Repositories

The non-database storage locations for your WebTrends data are specified during installation. However, you can change these storage locations. To determine where your data is stored, or to change your data storage locations, in the left pane, click **Administration > Application Settings > System Management > Storage Locations**.



### Note

Storing your Backup Repository on a different computer from your WebTrends installation allows you to recover data in the event of a system failure.

WebTrends creates the following data (file-based) repositories:

#### Analysis Data Repository

Contains analyzed log data in compressed form.

#### Report Data Repository

Contains report-ready data that can be accessed by on-demand reporting.

#### Configuration Data Repository

Contains configuration settings for the WebTrends Web Analysis module.

#### Backup Repository

Contains any data created by WebTrends Analytics backup jobs. For more information about backing up data, see [“Backing Up WebTrends Analytics Analysis Data”](#).

#### Visitor History Export Repository

Contains data created by Visitor History exports.

#### Event Database Source Directory

Contains the log files before the Event Database Loader imports them into the Marketing Warehouse.

## Backing Up WebTrends Analytics Analysis Data

When you create a profile backup job for a WebTrends Analytics profile, WebTrends backs up the analysis data for the profile. (After you restore, you can then regenerate your report data for the period after the backup by running an analysis based on the restored profile.) You can create backup jobs automatically for all new profiles or create them manually for individual profiles. WebTrends stores all data from backup jobs in the Backup Repository. For more information about specifying a backup location, see [“WebTrends Data Repositories”](#) on page 131.



### Note

When you first create a profile backup job, WebTrends also automatically backs up the configuration settings for the entire Web Analysis module. Configuration backups are used for reference by Support when recreating an older configuration. You cannot restore configuration data from a configuration backup. When you create a configuration backup, WebTrends does not display the backup in the Available Backups list.



**To create profile backups for all future WebTrends Analytics profiles:**

1. In the left pane, click **Administration > Application Settings > System Management > Backup/Restore > Backup Options**.
2. Select the **New profiles will have backup jobs created** check box. WebTrends creates a backup job for each subsequent profile you create. The backup event runs every 24 hours.

**To create a backup job for a single profile:**

1. In the left pane, click **Scheduler > Scheduled Jobs**.
2. Click **New**.
3. In the Job Type dialog, click **Backup Job**.
4. In the Profile dialog, select the profile you want to back up.
5. Click **Next**.
6. In the General dialog, type a description for the backup job. The description should identify the job as a backup job and link it to the correct profile.
7. Click **Next**.
8. In the Host Binding and Priority dialog, specify whether you want the backup job to run on any available host, or, if you have host groups configured in your distributed installation, select a host group. Host groups help you manage system resources in a distributed installation and are defined in **Administration > Application Settings > System Management > Host Groups**.
9. Select a job priority level. The job priority determines which jobs run first when multiple jobs are assigned to the same computer or host group. If all jobs are set to the same priority, jobs run in the order they enter the queue.
10. Click **Next**.
11. In the Schedule dialog, specify when and how often you want the backup job to run. By default, backup jobs run once per day.
12. Click **Next**.
13. Review the summary in the Summary dialog. Click **Save** to schedule the backup.

## Viewing Available Backups

The Available Backups list shows all the backups that were created by a successful backup job, including the profile name and the date when they were created. To view the list of available backups in WebTrends Administration, select **Administration > Application Settings > System Management > Backup/Restore > Restore Backup**.

## Backing Up a Microsoft SQL Server 2005 Express Edition System Database

Starting with Marketing Lab v2.1, Marketing Lab provides Microsoft SQL Server 2005 Express Edition as an alternative to Microsoft SQL Server Standard or Enterprise Edition for WebTrends Analytics users.



You have two options for backing up a Microsoft SQL Server 2005 Express Edition database. You can download Microsoft SQL Server Management Studio Express from the Microsoft web site if you prefer to back up your database using a graphical interface. Alternately, you can use the `sqlcmd` utility included with Microsoft SQL Server 2005 Express Edition.

#### To back up a SQL Express database using `sqlcmd`:

1. Open a command prompt and navigate to the `sqlcmd` subdirectory in the Microsoft SQL Server 2005 Express Edition installation directory. By default, `sqlcmd` is located in the `\Program Files\Microsoft SQL Server\90\Tools\Binn` directory.
2. Enter `sqlcmd`.
3. Enter `BACKUP DATABASE database_name` where *database\_name* is the name of the SQL server database.
4. Enter `TO DISK = 'pathname\database_name.bak'` where *database\_name* is the name of the SQL server database and *pathname* is the fully qualified path to the directory where you will create the database backup.
5. Enter `WITH FORMAT;`
6. Enter `GO`

## Backing Up a MySQL System Database

For WebTrends Marketing Lab versions earlier than v2.1, the system database is typically stored using a MySQL database. You can back up a MySQL WebTrends system database using the `mysql dump.exe` application, located in the `installation\directory\common\database\mysql\binn` directory of your WebTrends Analytics installation.

#### To back up a WebTrends system database using `mysql dump.exe`:

1. From a command prompt, navigate to the WebTrends Analytics installation directory and enter:

```
mkdir storage\backup\mysql backup
```

2. Enter:

```
cd common\database\mysql\binn
```

3. Enter:

```
mysql dump --opt --user=Database UserName --password=Database Password --result-file=..\..\..\storage\backup\mysql backup\backup.sql --all_databases
```

Make sure that *Database UserName* and *Database Password* are the user name and password specified during installation.

All databases are backed up to `\storage\backup\mysql backup\backup.sql` relative to the root of the WebTrends installation.

4. Rename the backup file to indicate the date the backup was made. For example, if the backup was made on July 1, 2007, rename the backup file to `backup_2007-07-01.sql`.



## Backing Up WebTrends Marketing Warehouse Data

WebTrends Marketing Warehouse data resides in a Microsoft SQL Server database. To determine where your Marketing Warehouse data is stored, edit the profile in WebTrends Administration and click **Analysis > Marketing Warehouse**. This dialog shows the location of the Microsoft SQL database and the names of each component database. You can access these locations using Microsoft SQL Server Management Studio or another third-party tool.

Marketing Warehouse profile data includes the following components:

- Event Database (EDB)
- Module Database (MDB)
- Extended Attributes Database (XDB)
- Analysis Services Database (containing profile-specific cube information)

You should maintain regular backups of these databases using the SQL management tools of your choice. You should also back up your Analysis Services Database each time you make changes to your cube configuration. Back up the Analysis Services Database both before and after you change the cube configuration.

As a best practice, you should also back up the following Marketing Warehouse components:

- The XMDB\_System\_Master database contains global information about Extended Attributes databases and can reside on the same Microsoft SQL database or in a remote location.
- wtWHReporting resides on the same Microsoft SQL server location as XMDB\_System\_Master.

Finally, even if you do not use WebTrends Analytics profiles, you should always back up your WebTrends System Database. For information about backing up your system database, see [“Backing Up a Microsoft SQL Server 2005 Express Edition System Database” on page 132](#) or [“Backing Up a MySQL System Database” on page 133](#).

For information about restoring a Marketing Warehouse profile, see [“Restoring WebTrends Marketing Warehouse Data” on page 135](#).

## Restoring WebTrends Analytics Data

This section describes how to restore analysis data for a profile from a backup. To recover from a serious failure, you should contact WebTrends Support.

**To restore WebTrends Analytics data from a profile backup:**

1. Determine the date of the last good backup, or the date from which you want to begin reanalysis. For more information, see [“Viewing Available Backups” on page 132](#).
2. Set your WebTrends Administration configuration settings to the state you want to use for reanalysis and future analyses. You can determine whether and when configuration changes were made using the Change History feature in WebTrends Administration. Use WebTrends Administration to make any required configuration changes.
3. In the left pane, click **Administration > Application Settings > System Management > Backup/Restore > Restore Backup**.
4. Mouse over the backup for the profile and date you want to restore and click **Restore Backup** on the Action menu.



5. In the Restore Backup dialog, click **Restore Backup** to continue.
6. Click **OK**.
7. After you finish restoring, re-run analysis for the profile. WebTrends Analytics detects the dates of analyzed data that was restored and then reanalyzes any newer data from the log files available in this profile's data source.

## Restoring WebTrends Marketing Warehouse Data

The following procedure is designed to restore data for a profile, not for your entire Marketing Warehouse installation. It includes instructions for restoring Visitor Intelligence cube data. For information about disaster recovery, you should contact WebTrends Support.

### To restore data for a Marketing Warehouse profile:

1. In the left pane, click **Administration > Web Analysis > Reports & Profiles**.
2. Edit your Marketing Warehouse profile and select **Analysis > Event Analysis**.
3. Click **Analysis Paused**. Pausing analysis allows Marketing Warehouse to continue collecting data while event data loading and other analysis processing is suspended.
4. Using the third-party SQL tools of your choice, restore the Event Database, Module Database, and Extended Attributes Database for the profile. For information about locating and backing up these components, see [“Backing Up WebTrends Marketing Warehouse Data” on page 134](#).
5. To restore from the .xml a file created when you backed up the Analysis Services Database, open a query in Microsoft SQL Server Management Studio or execute wtascmd.exe.
6. Perform a complete cube reprocess:
  - a. Open a command prompt and navigate to the wtCube directory.
  - b. Execute the following command:

```
wtcube profi l egui d -processful l
```

where profi l egui d is the ID for the profile you are restoring. You can find this ID by editing the profile in WebTrends Administration and clicking **Summary**.
7. In the left pane, click **Administration > Web Analysis > Reports & Profiles**.
8. Edit your Marketing Warehouse profile and select **Analysis > Event Analysis**.
9. Click **Analysis Enabled**.

## Recovering from Data Center Failure

Planning for a data center failure can be complex. You can use any of several methods to accomplish a quick recovery. Here are a few things to keep in mind before beginning:

- If the disaster recovery planning is limited to a situation where a data center is only temporarily unavailable, disaster recovery planning could focus specifically on providing access to reporting data, and not analysis.
- The WebTrends environment uses Web log data in the analysis process, but makes no provisions to back up or restore Web log data. The customer is responsible for ensuring that the original log data remains available.

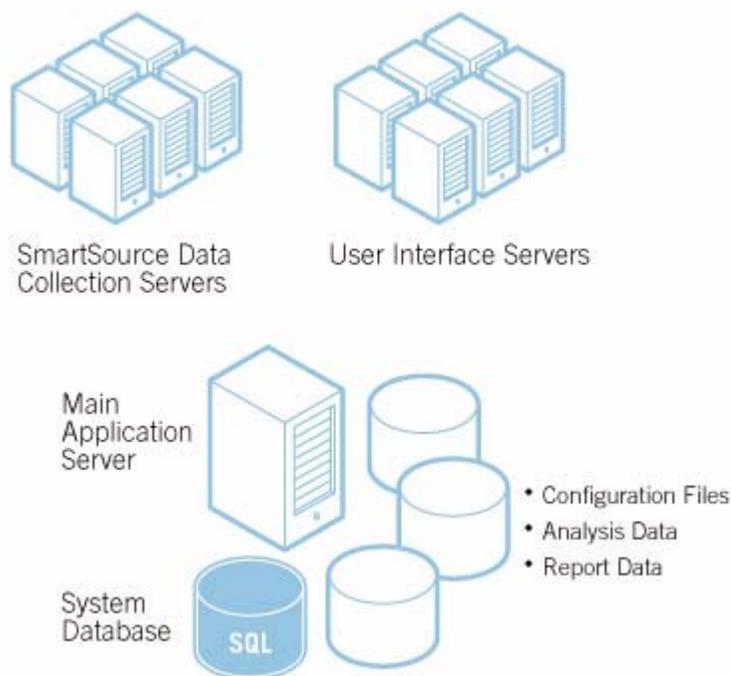


- The complete storage requirements for a WebTrends installation can be quite large. As a result, copying the data between data centers for fail-over can be time-consuming. Before you proceeding with this scenario, ensure that the bandwidth capacity between data centers is sufficient to meet the needs of the environment.

### Option 1: Data Collection and Reporting Redundancy

If your data loss is temporary, you can use a secondary environment to provide reporting. If you use this solution, no new data analysis can occur until the primary data center is recovered. Users will be able to log in to the WebTrends environment located at the secondary data center and view historical reports, but current data will not be analyzed. If you use the WebTrends SmartSource Data Collector, we recommend enabling a redundant SDC installation at the secondary data center so you can continue collecting SDC data while the primary data center is unavailable.

The following diagram shows the systems required for this option to work correctly:



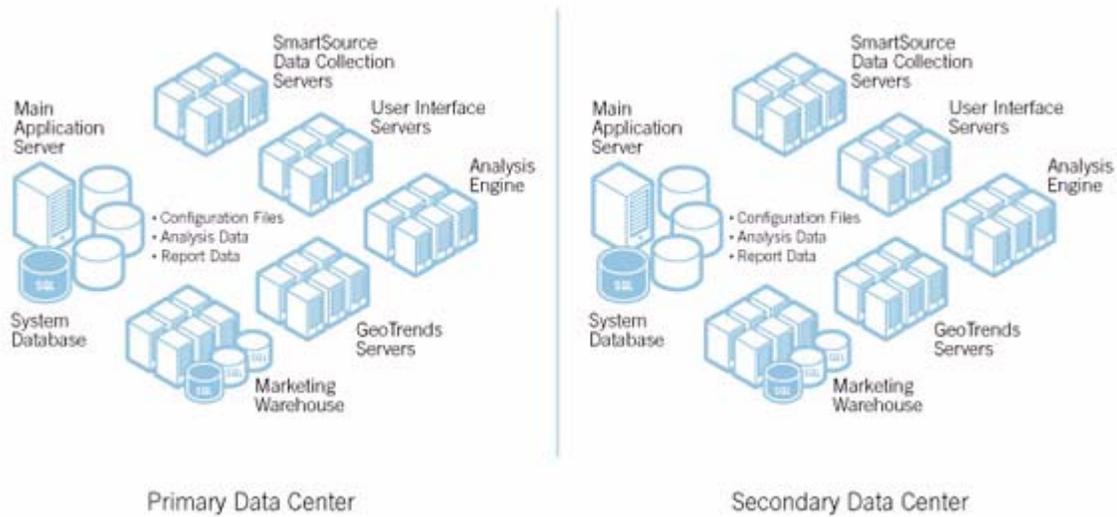
As an alternative, to reduce costs in the secondary data center, you could install the UI server directly on the main application server. Report performance and capacity will decrease slightly as a result, but this trade-off is often acceptable for a temporary situation.

### Option 2: Full Redundancy of the WebTrends Environment

To plan for recovery after a total failure, you can duplicate the complete WebTrends system at the secondary data center to provide full capacity and full fail-over in the event of a catastrophic failure of the primary data center. In this scenario, you deploy an exact copy of each system at the secondary data center. Because WebTrends does not provide an automated method of fail-over, you would have to accomplish this with the help of your local IT resources.



The following diagram shows a completely redundant deployment:



## Document Revision History

Table 13 contains a summary of changes made to this document beginning with the release of WebTrends Analytics, version 8.7.

**Table 13: Document Revision History**

Software Version	Date of Last Update	Summary of Changes
v8.7	March, 2009	Added Document Revision History section.
v8.7d	July, 2009	Added footer link to Documentation Center.





---

## Chapter 15

# Moving Your WebTrends Analytics Installation to a New Location

When WebTrends is installed on a single computer, you can move the installation to a new computer by following the procedures in this chapter. You may want to move your installation in the case of a hardware failure, a hardware upgrade/replacement, or an operating system upgrade. For requirements beyond the scope of this document, such as moving a distributed architecture installation, please contact WebTrends Support.

## Limitations

The following limitations apply to moving your WebTrends installation:

- The instructions contained in this chapter only apply to WebTrends installations on a single computer.
- The installation path on the new computer must exactly match the installation path on the original computer, including the drive letter. If you want to make any changes to this setup when you move your installation, please contact WebTrends Support.
- *If you use the WebTrends Marketing Warehouse*, you cannot move your Marketing Warehouse databases. Your WebTrends Event Database, Module Database, and Extended Attributes Database will remain on the original computer.
- If you do not want to move the WebTrends Analytics System Database, then you do not need this document.
- These instructions only apply to a WebTrends System database that uses Microsoft SQL Server 2005.

## Preparing to Move Your WebTrends Analytics Installation

Before moving your WebTrends installation, you need to perform the following steps. These steps prepare both the original computer and new computer for moving WebTrends Analytics and the System Database.



### To prepare your WebTrends Analytics installation:

1. On the original computer, cancel all active WebTrends jobs and profiles, then stop all the WebTrends services. To do this, run `SystemShutdown.exe mode=stop` from the WebTrends installation directory. For example, type the following command at a command prompt:

```
C:\WebTrends Installation Directory\modules\analysis\SystemShutdown.exe mode=stop
```



#### Note

This shutdown command waits for jobs to complete before shutting down WebTrends services.

---

2. If the WebTrends data storage folder on the original computer uses a Uniform Naming Convention (UNC) path, then rename storage to `storage.ori gi nal`. For example, rename `\\server name\storage` to `\\server name\storage.ori gi nal`.

Because WebTrends overwrites the contents in the storage folder during installation, you need to move the data so you can restore it after installation.

---



#### Note

If your storage data is saved in multiple locations, you must repeat this step for each storage data folder. For example, if you store report and analysis data in separate locations.

---

3. Follow the steps for setting up the SQL Server. For more information, see [“Preparing to Install WebTrends Analytics” on page 5](#).
4. Using the same version of WebTrends that exists on the original computer, install WebTrends Analytics, including the System Database, on the new computer.
5. In the Select Installation Type dialog, type or browse to the WebTrends installation path. This path must be the same installation path used to install WebTrends Analytics on the original computer.

Typically, the installation path is `C:\Program Files\WebTrends`. You can find the installation path on the original computer by navigating to the WebTrends folder in Windows Explorer. Copy the full path from the Windows Explorer address bar.

6. After installation completes, restart the new computer to ensure that the installation is complete and the services are running properly.
7. Stop all the WebTrends services on the new computer. To do this, run `SystemShutdown.exe mode=stop` from the WebTrends installation directory. For example, type the following command at a command prompt:

```
WebTrends Installation Directory\modules\analysis\SystemShutdown.exe mode=stop
```

8. Rename the storage folder from `storage` to `storage.temp` on the new computer. For example, rename `C:\Program Files\WebTrends\storage` to `C:\Program Files\WebTrends\storage.temp`.

Because WebTrends created a new storage folder during installation, you need to rename `storage` to `storage.temp` on the new computer so you can later restore the existing data in `storage.ori gi nal` to `storage` as shown in the [“Moving Your WebTrends Analytics Installation” on page 141](#).



## Moving Your WebTrends Analytics Installation

You can move WebTrends, the system database, and the user data by performing the steps in this section. You can move your SQL Server databases to a new location by using the Attach and Detach functions. For information about using these functions, see <http://support.microsoft.com/kb/224071>.

The following link provides other recommendations for migrating databases from Microsoft:

<http://msdn2.microsoft.com/en-us/library/ms189624.aspx>

### To move your WebTrends Analytics installation:

1. Detach the `wt_sched` and `wtmaster` databases that were created on the new computer. Using Windows Explorer, rename these databases to `wtmaster.temp` and `wt_sched.temp`.
2. Detach the `wtmaster` and `wt_sched` databases from the original computer, copy the original `wtmaster` and `wt_sched` databases to the new computer, then attach the new computer to the original databases.
3. Because WebTrends overwrites the contents in the storage folder during installation, you need to restore the original data by performing one of the following steps:
  - If the storage folder on the new computer uses a local file system path, you can move or copy the contents of the storage folder from the original computer to the storage folder on the new computer.
  - If the storage folder uses a UNC path, then you need to copy or rename the contents from `storage.original` to `storage`. For example, copy `\\server name\storage.original` to `\\server name\storage`.

## Updating the SQL Server Databases

This section provides the commands and information necessary for updating the SQL Server Database. You can run these commands from SQL Server Management Studio.

### To update the SQL Server databases:

1. On the new computer, run the following Transact-SQL commands:

```
Use wt_sched
del ete from wt_taskqueue;
del ete from wt_hostmetri chi story;
del ete from wt_taskstatus;
del ete from wt_hoststatus;
```
2. On the new computer, update the following fields to use the new computer name and IP address:
  - Update the **HostName** field in the `wt_host` table of the `wt_sched` database.
  - Update the **HostIP** field in the `wt_host` table of the `wt_sched` database.
  - Update the **ServerName** field in the `wt_server` table of the `wtmaster` database.
3. Also on the new computer, verify that the following fields are still correct:
  - The **Path** field in the `wt_accountdata locati on` table of the `wtmaster` database.
  - The **WRCPATH** field in the `wt_wrcversi on` table of the `wtmaster` database.



## Starting the New Installation

To run your new installation, you need to start WebTrends Analytics and license your new computer to use WebTrends Analytics.

### To start and license the new installation:

1. From your computer with the new WebTrends installation, use a command prompt to navigate to the WebTrends Installation directory.
2. Start all the WebTrends services on the new computer. To do this, run `SystemShutdown.exe mode=start` from the WebTrends installation directory. For example, type the following command at a command prompt:

```
WebTrends Installation Directory\modules\analysis\SystemShutdown.exe mode=start
```



#### Note

You must use this command to start WebTrends services.

---

3. To deactivate the original licensing information, run the following command:

```
\common\lib\wtlicman -d
```

4. To license the new installation, run the following command:

```
\common\lib\wtlicman -a -k license key
```

5. From the Windows Services panel, verify that all WebTrends services are running.
6. Access the WebTrends user interface from a web browser. If the user interface is not available, restart the WebTrends - User Interface service in the Windows Services panel.
7. Verify that you can analyze and view report data.
  - *If you cannot successfully analyze and view report data, call WebTrends Support.*
  - *If your new installation works correctly, uninstall WebTrends Analytics software from the original computer.*



#### Caution

Restarting WebTrends services on the original computer can corrupt your data if the storage folder uses a UNC path.

---



## Document Revision History

Table 14 contains a summary of changes made to this document beginning with the release of WebTrends Analytics, version 8.7.

**Table 14: Document Revision History**

Software Version	Date of Last Update	Summary of Changes
v8.7	March, 2009	Added Document Revision History section.
v8.7d	July, 2009	Added footer link to Documentation Center.





---

## Chapter 16

# WebTrends Query Parameter Reference

Web analytics is about making business sense out of web visitor behavior – the same common-sense analysis that business professionals have been applying to traditional offline business for decades. Gaining the appropriate insight to enable your organization to make smarter business decisions means understanding your site's business objectives and determining the appropriate web metrics to provide that information.

Today's business environment typically involves working at remote and off-site locations. Consequently, the people responsible for the web site and those analyzing visitor activity on the web site to measure web metrics often do not work side-by-side. That is, they are frequently in different groups and locations. This situation necessitates coordination between developing and delivering content, and reporting and analyzing the activity results.

To facilitate interaction between departments, locations and individuals, WebTrends provides a feature designed to place the power of analysis into the hands of those individuals who are most interested in it. This feature, WebTrends query parameters, uses typical web site instrumentation to facilitate the analytics end-users want to see.

If you are reading this chapter in the online Help, you can quickly find any parameters of interest by searching the Help. If you want to read the entire chapter, use the Help's Table of Contents to see all the topics in this chapter.

## How WebTrends Query Parameters Work

By implementing WebTrends query parameters when designing your web site, reporting becomes a part of the web site design process, eliminating the considerable coordination required and reducing the misunderstandings that can happen when the objectives of many groups are at play. WebTrends query parameters are passed in URLs through JavaScript, captured in SmartSource Data Collector log files, and ultimately used by WebTrends Analytics software or WebTrends Analytics On Demand. WebTrends Analytics uses these parameters to analyze your web activity and to produce reports.

## Using WebTrends Query Parameters

If one of your goals is to limit the amount of manual intervention that is required to produce valuable reports, you can implement WebTrends query parameters in your web pages to automate the configuration of many features in WebTrends Analytics, making the WebTrends solutions more insightful with reduced effort.



For example, you can use WebTrends query parameters in HTML META tags to automatically configure page titles, product names and product information, or campaign names. You could also implement a custom “Campaign Submission” page in a company intranet that sends the WebTrends query parameter to automatically configure WebTrends Analytics to report on new campaigns, allowing you to spend less time on system configuration.

## Products Using WebTrends Query Parameters

The following WebTrends products take advantage of WebTrends query parameters:

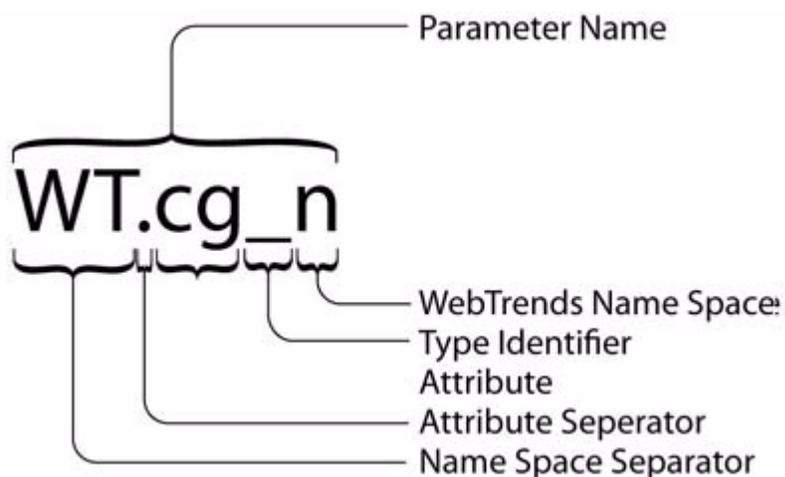
- SmartSource Data Collector
- WebTrends Analytics On Demand
- WebTrends Analytics software
- WebTrends Marketing Warehouse
- WebTrends Visitor Intelligence
- WebTrends Score

## Query Parameter Syntax

WebTrends query parameters follow specific syntax that includes name, value, and format. The following sections describe each of these syntax elements

### Name Syntax

The name of each WebTrends query parameter uses the following syntax:



Although WebTrends Analytics handles query parameter names without considering case, JavaScript object names are case-sensitive. Therefore, if you want JavaScript tagging to capture events, you must adhere to specified naming conventions. WebTrends reserved query parameters require an upper case name space, either WT or DCS, and the type identifier in lower case.



In addition to the query parameters covered in this chapter, the following name spaces are reserved for WebTrends:

**WT.i\_**

Used for product integrations with WebTrends partners.

**WT.z\_**

Used by WebTrends Professional Services for customer integrations.

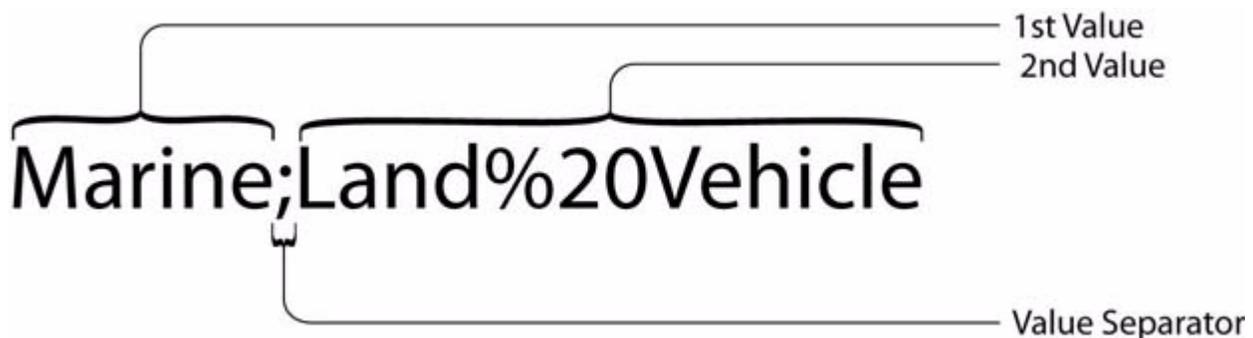
As a best practice, use the following syntax to create custom query parameters:

`DCSext.w_custom_identifier`

For example, you could create a parameter called `DCSext.w_articleid` to track an article ID.

## Value Syntax

The values associated with WebTrends query parameters use the following syntax.



Each WebTrends query parameter name may have one or more values. Some parameters can be specified in pairs or in groups of related parameters. When related parameters have multiple values, these values may be correlated and their position becomes important as shown in the following example:

`WT.si_n=name1;name2&WT.si_x=position1;position2.`

In the previous example, if correlation is specified in the report, `name1` is associated with `position1` and `name2` with `position2`.

If there are multiple values, they are typically separated using a semicolon (;). You can use other separators, but you must specify the separator in the dimension or measure setting that is based on the parameter.

**Note**

Not all WebTrends query parameters support multiple values. For example, HTML Title Page allows only one value.

If the parameter value contains a semicolon, it must be hex-encoded ("%3B") to differentiate it from the separator.

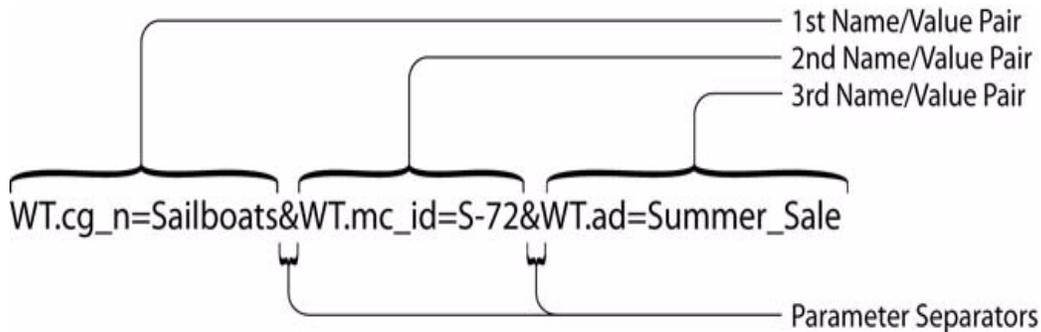


## Numerical Value Format

Unless stated otherwise, numerical values must be specified using the simple US format using the period as the decimal separator with up to 2 decimals and without thousand separators. For example, 12345.67.

## Complete Syntax

The WebTrends query parameters are represented as name/value pairs. The name/value pairs adhere to the following syntax:



The WebTrends query parameters are logged in SmartSource Data Collector log files. Each name/value pair is separated by an ampersand (&).

The WebTrends query parameters must co-exist with other web site well-known parameters. This means that WebTrends query parameters can be mixed with standard well-known parameters. The WebTrends query parameters can be separated from the other well-known parameters using the ampersand delimiter. While a WebTrends namespace is part of each query parameter, it is conceivable that parameter collisions may occur. If a query parameter string contains duplicate key values, the first instance is used and the others are discarded.

## Syntax Examples

### Single Parameter With a Single Value

The following example shows a page associated with the "Finance Offer" advertising view:

```
WT. ad=Fi nance%20ffer
```

### Single Parameter With Multiple Values

The following example shows a page associated with both the "Finance Offer" and "FishFinder Offer" advertising views:

```
WT. ad=Fi nance%20ffer; Fi shFi nder%20ffer
```

### Related Parameters With a Single Value

The following example shows a page associated with the Campaign "New Product" and the Campaign Event Type "click."

```
WT. mc_i d=New%20Product&WT. mc_ev=cl i ck
```



## Related Parameters With Multiple Values

The following example shows a page associated with the Campaign “Get Results” and “New Product” with the Campaign Event Type “Click.” Note that because the second parameter uses a single value, if correlation is used, the “Click” value of the second parameter is associated with both values of the first parameter.

```
WT.mc_id=Get%20Results;New%20Product&WT.mc_ev=Click
```

## Types of Query Parameters

WebTrends query parameters consist of the following groups:

### Auto-configuration parameters

Recognized by WebTrends and used to auto-configure certain features. For more information, see [“Auto-Configuration Parameters” on page 149](#).

### Custom report parameters

Associated with preconfigured custom reports. For more information, see [“Custom Report Parameters” on page 155](#).

### SmartView parameters

Used by WebTrends SmartView. For more information, see [“SmartView Parameters” on page 166](#).

### Stored visitor parameters

Used for identifying visitors when the Visitor History database is exported. For more information, see [“Stored Visitor Parameter” on page 167](#)

### Visitor History parameters

Related to visitor properties added by WebTrends during the analysis process. For more information, see [“Visitor History Parameters” on page 167](#).

### SDC-generated visitor parameters

Generated by SmartSource Data Collector (SDC). For more information, see [“SDC-Generated Visitor Parameters” on page 180](#).

### SDC-parameter override parameters

Override SDC parameters on the client side. For more information, see [“SDC-Parameter Override Parameters” on page 187](#).

### Conversion plug-in parameters

Used by the encoding conversion plug-in. For more information, see [“Conversion Plug-In Parameters” on page 189](#).

### Content reports parameters

Used to include web 2.0 content information in reports. For more information, see [“Content Parameters” on page 164](#).

## Auto-Configuration Parameters

WebTrends uses certain query parameters to automatically configure the appropriate advanced features and create related reports. The following types of auto-configuration parameters are available:

- Content Group (see [“Content Group Parameters” on page 150](#))
- Marketing Campaign (see [“Marketing Campaign Parameter” on page 150](#))



- Advertising View (see “Advertising View Parameter” on page 150)
- Advertising Click (see “Advertising Click Parameter” on page 151)
- Server (see “Server Parameter” on page 151)
- Scenario Analysis (see “Scenario Analysis Parameters” on page 151)
- Title (see “Title Parameter” on page 155)
- Split (see “Split Parameter” on page 155)

## Content Group Parameters

A Content Group definition uses the following parameters:

- WT. cg\_n
- WT. cg\_s

You can specify multiple content groups per page. The Sub-Content Group parameter is optional. If none of the Content Groups on a particular page contain Sub-Content Groups, WebTrends Analytics may ignore the WT. cg\_s parameter. If the WT. cg\_s parameter is included, each Content Group value must have an associated Sub-Content Group value. The Sub-Content Group value may be empty.

The Marketing Warehouse also uses this parameter. For more information about the complete set of query parameters that the Marketing Warehouse uses, see “Configuring Your Web Site to Collect Marketing Warehouse Data” in *WebTrends Marketing Warehouse User’s Guide*.

### WT. cg\_n

WT. cg\_n=*Name*[; ...]

This parameter identifies the name of a Content Group. The maximum length is for each *Name* is 64 bytes.

### WT. cg\_s

WT. cg\_s=*SubName*[; ...]

This parameter identifies the name of a Sub-Content Group. This parameter is optional. The maximum length is for each *SubName* is 64 bytes.

## Marketing Campaign Parameter

With WebTrends v7.0 and higher, WT. mc\_id replaces WT. mc\_n and WT. mc\_t. For more information, see “Campaign Parameter” on page 162.

## Advertising View Parameter

The Advertising View definition uses of the WT. ad parameter, which supports multiple Advertising Views per page.

### WT. ad

WT. ad=*Name*[; ...]

The name of the advertisement viewed on a particular web page. The maximum length for each *Name* is 64 bytes.



## Advertising Click Parameter

The Advertising Click definition uses the WT. ac parameter, which supports a single Advertising Click per page. If a page contains multiple ads, you can design the page to respond to clicks so that each click generates a hit to SDC.

### WT. ac

WT. ac=*Name*

The name of the advertisement clicked to reach a particular web page. To capture this information, the Advertising Click must contain an external redirect back to the client. The redirect needs to include the necessary code to generate a hit to SDC or WebTrends On Demand. The maximum length for each *Name* is 64 bytes.

## Server Parameter

The Server definition uses the WT. sv parameter, which supports a single server per page.

### WT. sv

WT. sv=*Name*

This parameter identifies the name of the web server that served the web content. This is used for server cluster load balanced reports. The maximum length for *Name* is 256 bytes.

---

### Note



The WT. sv parameter requires that the profile be set up as clustered with at least one server. This is the server that is used if an incoming hit does not explicitly have a server defined.

---

## Scenario Analysis Parameters

Scenario Analysis parameters specify well-known paths or processes in your web site and are typically used to measure conversion and abandonment.

In order to report on Scenario Analysis in WebTrends Analytics, you must also specify your Scenario Analysis definitions in WebTrends Administration.

The Marketing Warehouse also recognizes these parameters. For more information, see “Configuring Your Web Site to Collect Marketing Warehouse Data” in *WebTrends Marketing Warehouse User’s Guide*.

Scenario Analysis definitions use the following attributes:

WT. si\_n – Name of the Scenario Analysis

WT. si\_p – Identifies the step by name

WT. si\_x – Identifies the step by position

WT. si\_cs – Identifies the step in which conversion occurs for Marketing Warehouse profiles.



You can configure multiple Scenario Analysis per page.



**Note**

Because Scenario Analysis dimensions and measures are visit-based, they should only be used with visit-based dimensions and measures. If you create a custom report that uses Scenario Analysis dimensions with a hit-based dimension such as Page Views, the reported page view count will include all the page views of every visitor who saw one of the steps of the scenario, which inflates the actual page view count.

For example, if WT. pn\_sku, which is hit-based, is specified on the same page as the Product View scenario step, and you create a custom report using Product SKU as the dimension and four scenario steps as measures, the Product SKU will show a visit for each step of the Scenario, even though it was only specified on one of the steps

**WT. si\_n**

WT. si\_n=*Name*[; ...]

Identifies the name of the Scenario Analysis. The maximum length for each *Name* is 64 bytes. When configuring a new Scenario Analysis definition using WebTrends Administration, make sure that the Scenario Analysis name matches this parameter value.

When specifying multiple Scenario Analysis names, you must also specify multiple step names or step numbers. Each step name or step number is associated with the scenario name having the same position in the list.

**WT. si\_p**

WT. si\_p=*StepName*[; ...]



**Note**

- WebTrends Analytics supports this parameter for Analytics Reports. Use WT. si\_x to identify steps for Marketing Warehouse.
- Do not use WT. si\_p if you specify the step by position using the WT. si\_x parameter.

Identifies the step by name. When configuring a new Scenario Analysis step using WebTrends Administration, the step name needs to match this parameter value.

**WT. si\_x**

WT. si\_x=*StepPosition*[; ...]

Identifies the step by numeric position. You must specify an integer value for the value.



**Note**

Do not use if specifying the step by name using the WT. si\_p parameter. Also, for WebTrends Marketing Warehouse only, to translate the position numbers displayed in WebTrends Explore into something more meaningful, create a Scenario Analysis Definition in WebTrends Administration. Ensure that you configure a step name for each step.

**WT. si\_cs**

WT. si\_cs=*1/0*[; . . . ]



Identifies whether the page is the step in which conversion occurs when the value is 1. Most likely, this page is the last step in your Scenario.

In addition to this parameter, the web server must also pass the step query parameter for this page. WebTrends Analytics recognizes either WT. si \_p or WT. si \_x for steps. Marketing Warehouse recognizes WT. si \_x for steps.

Multiple values are supported. If multiple values are passed for this parameter and other Scenario event parameters, the Marketing Warehouse correlates the values.

The WebTrends Marketing Warehouse recognizes this parameter. However, this parameter is currently not included in any WebTrends Analytics preconfigured custom reports.

## Scenario Analysis Parameters for Shopping Cart Analysis

Use Scenario Analysis parameters to track shopping cart activity. WebTrends Analytics supports these parameters for the purchase conversion funnel report. Marketing Warehouse supports these parameters for multi-dimensional reporting through WebTrends Visitor Intelligence and for visitor segmentation in WebTrends Explore.

Use the following parameter names and values to tag your shopping cart pages. If you use WebTrends Marketing Warehouse or think that you might in the future, use the Step by Number method. If you use WebTrends Analytics, you can use either the Step by Name method or Step by Number method.

Step by Name Method (WT.si_p)	Step by Number Method (WT.si_x)	Query Parameter Purpose and Use
WT. si _n=ShoppingCart	WT. si _n=ShoppingCart	Identifies the shopping cart scenario. WebTrends Analytics uses the ShoppingCart value to create a purchase conversion funnel report. Marketing Warehouse uses the ShoppingCart value to provide top-level Scenario data.
WT. si _p=CartView	WT. si _x=1	Identifies the product view step
WT. si _p=CartAdd	WT. si _x=2	Identifies the cart start step
WT. si _p=CartCheckout	WT. si _x=3	Identifies the cart check out step
WT. si _p=CartComplete	WT. si _x=4	Identifies the cart complete step
WT. si _cs=1	WT. si _cs=1	Identifies the step in which conversion occurs

To collect shopping cart activity for reporting, you need to set up your purchases pages with shopping cart query parameters and create a Scenario Analysis definition in WebTrends Administration. For information about Scenario Analysis definitions, see Administration Help.



**To implement shopping cart scenario parameters using WT.si\_p:**

1. Configure the product detail pages where visitors can click a button to buy a product. These pages differ from the shopping cart basket page that contains all items in the cart.
  - a. Use WT. si \_n=Shoppi ngCart to identify tagged pages with your shopping cart scenario.
  - b. Use WT. si \_p=CartVi ew to identify tagged pages as the first step.
2. Configure the pages where your visitors add or remove items from the shopping cart.
  - a. Use WT. si \_n=Shoppi ngCart to identify tagged pages with your shopping cart scenario.
  - b. Use WT. si \_p=CartAdd to identify tagged pages as the second step in the scenario.
3. Configure the pages where your visitors start the checkout process.
  - a. Use WT. si \_n=Shoppi ngCart to identify tagged pages with your shopping cart scenario.
  - b. Use WT. si \_p=CartCheckout to identify tagged pages as the third step in the scenario.
4. Configure the page where your visitors have successfully completed the purchase:
  - a. Use WT. si \_n=Shoppi ngCart to identify tagged pages with your shopping cart scenario.
  - b. Use WT. si \_p=CartCompl ete to identify tagged pages as the fourth step in the scenario.

**To implement shopping cart scenario parameters using WT.si\_x:**

1. Configure the product detail pages where visitors can click a button to buy a product. These pages differ from the shopping cart basket page that contains all items in the cart.
  - a. Use WT. si \_n=Shoppi ngCart to identify tagged pages with your shopping cart scenario.
  - b. Use WT. si \_x=1. to identify tagged pages with the first step in the scenario.
2. Configure the pages where your visitors add or remove items from the shopping cart.
  - a. Use WT. si \_n=Shoppi ngCart to identify tagged pages with your shopping cart scenario.
  - b. Use WT. si \_x=2 to identify tagged pages with the second step in the scenario.
3. Configure the pages where your visitors start the checkout process.
  - a. Use WT. si \_n=Shoppi ngCart to identify tagged pages with your shopping cart scenario.
  - b. Use WT. si \_x=3 to identify tagged pages with the third step in the scenario.
4. Configure the pages where your visitors have successfully completed the purchase.
  - a. Use WT. si \_n=Shoppi ngCart to identify tagged pages with your shopping cart scenario.
  - b. Use WT. si \_x=4 to identify tagged pages with the fourth step in the scenario.
  - c. *If you want to track the purchase complete page as the conversion step for use in WebTrends Marketing Warehouse, use WT. si \_cs=1.*

**Note**

In order to provide useful names for WebTrends Explore and WebTrends Visitor Intelligence users, you also need to create a Scenario Analysis definition through WebTrends Administration. Ensure that you configure a step name for each step.



## Title Parameter

The Title parameter, WT. ti , supports a single page title per page.

### WT. ti

WT. ti = *Ti tle*

The HTML title of the associated web content. If this parameter is found in parameter list, the value is used in the reports. When present, no other page title lookups are performed. The maximum length for *Ti tle* is 1024 bytes.

WT. ti always overrides other methods of page title lookup. In a case where two different pages (both having WT. ti ) end up being rebuilt to the same URL (via URL Rebuilding), the last page seen wins.

## Split Parameter

The Split parameter, WT. sp, is the default parameter WebTrends uses to split log files for Parent Child profiles. The parameter supports multiple values per page in order for any given page to be part of multiple Child profiles.

### WT. sp

WT. sp = *ProfileName[; ...]*

The Split parameter is used by the Parent Child profile generator to build child profiles. As a best practice, keep the length of this parameter as short as possible. The format of this value is an alphanumeric string (for example, UTF-8).

## Custom Report Parameters

This section discusses parameters that are included in a series of preconfigured custom reports. These query parameters are grouped into categories as follows:

- Search Engine – see “[Search Engine Type Parameter](#)” on page 155.
- Web Client – see “[Web Client Parameters](#)” on page 156.
- Transaction – see “[Transaction Parameters](#)” on page 160.
- Invoice – see “[Invoice Parameters](#)” on page 161.
- Campaign – see “[Campaign Parameter](#)” on page 162.
- Campaign Event – see “[Campaign Event Parameter](#)” on page 163.
- Content – see “[Campaign Event Parameter](#)” on page 163.

## Search Engine Type Parameter

### WT. srch

WT. srch = *SearchEngineType*

The Search Engine Type parameter in the query string of the URL from a referring search engine identifies the link as one for a paid search engine phrase.



WebTrends Analytics uses this parameter for Custom Reporting and Visitor History to differentiate a paid search engine from a reference to an organic search engine. Only `WT. srch=1` has meaning. No other values are defined or recognized.



### Note

Search engine results using `WT. srch=1` are limited to only those search engines that are listed on the WebTrends search engine availability list. Using `WT. srch=1` will not produce search engine campaign results for search engines that are not recognized by WebTrends.

The Marketing Warehouse also uses this parameter. For more information, see “Configuring Your Web Site to Collect Marketing Warehouse Data” in *WebTrends Marketing Warehouse User’s Guide*.

## Web Client Parameters

This section describes parameters that are associated with web client (browser) properties. These parameters are typically used for creating custom dimensions that you include in custom reports.

### WT. tz

*WT. tz=Timezone*

Indicates the web client’s time zone and is the offset of the web client from UTC. The value is expressed in hours.

Positive values are given without a sign (`WT. tz=2`), but negative values require a minus sign (`WT. tz=-12`). Values can range from -12 to +14.

Example:

```
var dCurrent = new Date();
var tzQueryParam = "&WT. tz=" + escape(dCurrent.getTimezoneOffset());
```

To represent Pacific Standard Time, use the following parameter:

`WT. tz=-8`

### WT. bh

*WT. bh=BrowsingHour*

Indicates the web client’s local time of day on a 24-hour clock. Values can range from 0 to 23.

### WT. ul

*WT. ul=UserLanguage*

Indicates the web client’s user language.

Example:

```
var ulQueryParam = "&WT. ul=" + navigator.appName == "Netscape" ?
escape(navigator.language) : escape(navigator.userLanguage);
```

### WT. cd

*WT. cd=ColorDepth*

Indicates the web client’s screen color depth. It is represented as the number of color bits to which the web client computer’s video display control is set.

Example:

```
var cdQueryParam = "&WT. cd=" + escape(screen.colorDepth);
```

### WT. sr

*WT. sr=ScreenResolution*



Indicates the web client's screen resolution. It is expressed as the gross width and height of the web client's monitor. The format of the value is *wi dthXhei ght* (for example, 800X600).

Example:

```
var srQueryParam = "&WT. sr=" + escape(screen. wi dth) + "x" + escape(screen. hei ght);
```

#### WT. j o

*WT. j o=IsJavaEnabl ed*

Indicates if the web client has enabled Java. Valid values are Yes and No.

Example:

```
var joQueryParam = "&WT. j o=" + navi gator. j avaEnabl ed() ? "Yes" : "No";
```

#### WT. j s

*WT. j s=IsJavaScri ptEnabl ed*

Indicates whether the web client supports and/or has enabled JavaScript. Valid values are Yes and No.

For example:

```
<SCRI PT LANGUAGE="JavaScri pt">
... CODE ABOVE WHERE YOU WANT THI S MODI FICATION TO GO...
var jsQueryParam = "&WT. j s=Yes";
... CODE BELOW WHERE YOU WANT THI S MODI FICATION TO GO...
</SCRI PT>
```

```
<NOSCRI PT>
<IMG BORDER="0" NAME="DCSI MG" WI DTH="1" HEI GHT="1" SRC="http: //l ocal host/
nj s. gi f?dcsuri =/noj avascri pt&WT. j s=no">
</NOSCRI PT>
```



#### Note

You can use WebTrends to set up a custom reports (augmenting the Pages reports) that keys on **WT. j s** and provides a targeted count of no JavaScript hits.

#### WT. j v

*WT. j v=JavaScri ptVersi on*

Indicates the version of JavaScript supported by the web client. If JavaScript is not enabled, this parameter should not be included.

Example:

```
var jvQueryParam = "";
<SCRI PT Language"Javascri pt1. 0">
jvQueryParam = "&WT. j v=1. 0"
</SCRI PT>
```

```
<SCRI PT Language"Javascri pt1. 1">
jvQueryParam = "&WT. j v=1. 1"
</SCRI PT>
```

```
<SCRI PT Language"Javascri pt1. 2">
jvQueryParam = "&WT. j v=1. 2"
</SCRI PT>
```

#### WT. ct

*WT. ct=connecti ontype*

Identifies the visitor's connection type. You can use this parameter in custom reports to determine whether visitors can download media on your site that requires high-bandwidth connections.



The JavaScript tag generates this value using data from Microsoft Internet Explorer 5 or higher. Valid values are `lan`, `modem`, and `offline`. For all other browsers, the JavaScript tag generates a value of `unknown`.

**WT.hp**

*WT.hp=ishomepage*

Indicates whether your visitors have configured a URL as their home page. It is only available for visitors using Microsoft Internet Explorer 5 or higher. Valid values are `1` and `0`.

You can use this parameter to filter a report based on URLs or pages, and report on pages that are used as home pages by visitors.

**WT.bs**

*WT.bs=browsersize*

Identifies the actual size of the visitor's browser window. It is expressed as the width and height of the web client window in pixels. The format of the value is `widthxheight` (for example, `924x212`).

**WT.fi**

*WT.fi=iflashinstalled*

Indicates whether your visitors have installed the Macromedia Flash browser plug-in. Valid values are `Yes` and `No`.

For more information see, the `WT.fv` parameter.

**WT.fv**

*WT.fv=flashversion*

Indicates the version of the Macromedia Flash browser plug-in if installed. Requires the presence of `WT.fi=Yes`.

web. For example, `WT.fv=7.0`.

**WT.le**

*WT.le=languageencoding*

Specifies the character set used by the web client to render the current document. This parameter can be used to troubleshoot internationalization issues. The format of this value is an alphanumeric string (for example, `UTF-8`). Go to the following site for a list of valid values:

<http://www.iana.org/assignments/character-sets>

**WT.mle**

*WT.mle=metalinguageencoding*

Specifies the character encoding set by the web client to render the current document. If the page includes the `http-equiv="Content-Type" META` tag, the WebTrends JavaScript tag generates this parameter and passes it as an alphanumeric string (for example, `UTF-8`). The parameter is used for troubleshooting internationalization issues.

For example, the JavaScript tag generates a value of `WT.mle=UTF-8` for the following tag:

```
<meta http-equiv="Content-Type" content="text/html ; charset=UTF-8">
```

**Note**

In order for this parameter to be generated, you must set the `gl18n` variable in the JavaScript tag to `true`. For more information, see "Customizing the JavaScript Tag" in the *WebTrends Analytics Software Implementation and Maintenance Guide*.

Go to the following site for a list of valid values:

<http://www.iana.org/assignments/character-sets>



**WT. em**

WT. em=*esc/uri*

Specifies the encoding method supported by the web client. This parameter can be used to troubleshoot internationalization issues. Valid values are *esc* for the JavaScript `escape()` function and *uri* for the JavaScript `encodeURIComponent()` function.

**WT. sl v**

WT. sl v=*Silverlight\_version*

Specifies the version of the Microsoft Silverlight plug-in installed on the visitor's web client. If Silverlight is not enabled, this parameter is not provided.

## Products Parameters

You can use Products parameters to report on transaction activity for products on your site. Several preconfigured custom reports use these query parameters.

These parameters support multiple values except where noted. When multiple values are passed, the order of the values is important because they correlate to other parameters. For example, multiple product values for the WT. pn\_sku parameter correlate to the number of units passed in the WT. tx\_u parameter.

---

**Note**

You cannot use these query parameters as dimensions and measures in Scenario Analysis funnels. Products parameters are hit-based while scenario parameters are visit-based and should not be used together in reports.

---

**WT. pn\_sku**

WT. pn\_sku=*productSKU[; ...]*

Identifies the SKU (a unique numeric identifier) of the product. The Marketing Warehouse also uses this parameter. For more information, see "Configuring Your Web Site to Collect Marketing Warehouse Data" in WebTrends Marketing Warehouse User's Guide.

---

**Note**

WT. pn\_sku replaces the WT. pn and WT. pc parameters. WT. pn represented the name of the product. WT. pc represented the category of the product.

---

**WT. pn\_id**

WT. pn\_id=*productID[; ...]*

Optional: Identifies the product identifier of a product. If possible, product IDs should be unique values to preserve lookup data integrity. The following parameters are automatically added by the product translation process when the product SKU is found in the product translation file. For more information about translation files, see "Using Lookup Tables for Analytics Reports" in the *Administration User's Guide*.

---

**Note**

Product IDs typically map to multiple Product SKUs. For example, a sporting goods company might have an item with a specific ID and several SKUs corresponding to various colors.

---



**WT. pn\_fa**

WT. pn\_fa=*productFamily*[; ...]

Identifies the family of the product.

**WT. pn\_gr**

WT. pn\_gr=*productGroup*[; ...]

Identifies the group of the product.

**WT. pn\_sc**

WT. pn\_sc=*productSubCategory*[; ...]

Identifies the sub-category of the product.

**WT. pn\_ma**

WT. pn\_ma=*productManufacturer*[; ...]

Identifies the manufacturer of the product.

**WT. pn\_su**

WT. pn\_su=*productSupplier*[; ...]

Identifies the supplier of the product.

## Transaction Parameters

**WT. tx\_t**

WT. tx\_t=*Type*[; ...]

In earlier versions of WebTrends, this parameter was used to identify transaction types. Because WebTrends Analytics does not use this query parameter in any preconfigured custom reports, it is no longer supported.

**WT. tx\_u**

WT. tx\_u=*Units*[; ...]

Identifies the quantity in the transaction. Pass a positive integer for this value.

If an order contains multiple products, separate the numbers of units for each product using a comma or semi-colon (configurable) in the WT. tx\_u variable.

When associating this measure with the product dimension in a WebTrends Analytics custom report, make sure you select the correlation option to ensure that the first number of units is associated with the first product, the second number of units with the second product, and so on.

The Marketing Warehouse also recognizes this parameter. For more information, see “Configuring Your Web Site to Collect Marketing Warehouse Data” in *WebTrends Marketing Warehouse User’s Guide*.

**WT. tx\_s**

WT. tx\_s=*Subtotal* [; ...]

Identifies the total cost for each product in the order.

The format of this field must match the currency format that WebTrends Analytics is configured to analyze. Do not include a currency symbol and be sure pass the value in *dollars.cents* format. For example, if you globally define your currency as US-Dollars in WebTrends Analytics, the format of this parameter set to a \$4500 cost is: WT. tx\_s=4500.00. Do not include a currency symbol or a comma in the value.

If an order contains multiple products, the totals for each product should be separated by a comma or semi-colon (configurable) in the WT. tx\_s variable.

When associating this measure with the product dimension in a custom report, make sure you select the correlation option to ensure that the 1st amount is associated with the 1st product, the 2nd amount with the 2nd product, and so on.



The Marketing Warehouse also recognizes this parameter. For more information, see “Configuring Your Web Site to Collect Marketing Warehouse Data” in *WebTrends Marketing Warehouse User’s Guide*.

**WT. tx\_e**

WT. tx\_e=*event*

Identifies a product-related event. WebTrends Analytics uses this parameter as qualifier in preconfigured measure definitions in order to determine which product to count for a measure. WebTrends Marketing Warehouse uses this value to qualify certain preconfigured events. For more information, see “Configuring Your Web Site to Collect Marketing Warehouse Data” in *WebTrends Marketing Warehouse User’s Guide*. You can use the parameter to report on specific activities by creating a custom report filter that you apply to a custom report.

**Note**

Even if there are multiple values specified in the WT.pn\_sku (Products), WT.tx\_u (Units) and WT.tx\_s (Revenue) parameters, the WT.tx\_e should contain a single value (the same event applies to the entire page view).

You can pass a custom value for this parameter, or if you want to track product purchases, product views, product cart additions, and product cart removals, pass one of the following values:

**WT. tx\_e=p**

Required by WebTrends Analytics and Marketing Warehouse to identify a product purchase. Although WebTrends Analytics does not use WT. tx\_e=p to determine whether a visitor is a buyer (WT. vr\_brws), the Marketing Warehouse does. Instead WebTrends Analytics uses WT. tx\_s to determine whether the visitor is a buyer. In addition, you can use the Invoice query parameters to include invoice number, date, and time for the purchase. For more information, see “[Invoice Parameters](#)” on page 161.

**WT. tx\_e=v**

Required by WebTrends Analytics and Marketing Warehouse to identify a product view.

**WT. tx\_e=a**

Identifies a product cart addition.

**WT. tx\_e=r**

Identifies a product cart removal.

**WT. tx\_cartid**

WT. tx\_cartid=*CartIdentifier*

Pass a unique value to identify a visitor’s cart. Marketing Warehouse uses this parameter to identify events associated with a specific cart. For more information, see “Configuring Your Web Site to Collect Marketing Warehouse Data” in *WebTrends Marketing Warehouse User’s Guide*.

## Invoice Parameters

Use WT. tx\_i , WT. tx\_id , and WT. tx\_it parameters together. You must enable Visitor History in order to use the Invoice parameters.

**WT. tx\_i**

WT. tx\_i=*InvoiceNumber*

Identifies the invoice number for the purchase. WebTrends Analytics uses data stored in Visitor History to make sure that a page view with an invoice number is a new purchase and not the result of a visitor refreshing the page after making a purchase. If WebTrends Analytics sees a page view with an invoice number, that page view is compared against the last three invoices for a visitor. If the WT. tx\_i value does not match the last three invoices, WebTrends considers it a new purchase.



The Marketing Warehouse also recognizes this parameter. For more information, see “Configuring Your Web Site to Collect Marketing Warehouse Data” in *WebTrends Marketing Warehouse User’s Guide*.

**Note**

WebTrends Express Analysis cannot use the invoice data to identify duplicate purchase records because the Express Analysis Engine does not use visitor history data.

**WT. tx\_i d**

WT. tx\_i d = *InvoiceDate*

The format is mm/dd/yy. A 4-digit year is also allowed: mm/dd/yyyy.

Identifies the purchase invoice date, which is based on GMT. If the invoice date is three days older than the date of the visit, it is assumed that the hit is not an actual purchase but a view of a previously bookmarked or saved page. If the invoice date was less than three days than the date of the visit, the WT. tx\_i d parameter is used to determine if the hit is a valid purchase.

For example, a visitor makes a purchase. The purchase is accounted with an invoice date. The visitor saves a bookmark to the page. Five days later, the visitor goes to the bookmarked page. This causes another hit to be sent to SDC. However, the WT. tx\_i d parameter still contains the original purchase date. WebTrends analysis sees that the date of the hit is several days after the date found in the WT. tx\_i d parameter and determines that this is not an actual purchase.

**WT. tx\_i t**

WT. tx\_i t = *InvoiceTime*

Identifies time of the invoice. The format is hh: mm: ss where hh is in a 24-hour format (0 = midnight, 23 = 11pm).

This parameter helps determine when an invoiced purchase was made. This value is used along with WT. tx\_i d and WT. tx\_i to determine if the purchase was a valid purchase or if this was a user refreshing the web page after a purchase or returning to the page to check status.

## Campaign Parameter

**WT. mc\_i d**

WT. mc\_i d = *Campaign ID*

Identifies a specific marketing campaign. Pass this query parameter to pages that you want to associate with a specific campaign. You can specify a numeric or string value.

## WebTrends Analytics Considerations

If you plan to export WebTrends Analytics report data to a SmartReport, values that are either characters or a combination of numbers and characters work best with Microsoft Excel.

If you enable Visitor History in a profile, WebTrends Analytics reads this parameter and stores it in the Visitor History database. The most recent value of this parameter is made available by Visitor History on every hit as WT. vr. rac.



## WebTrends Marketing Warehouse Considerations

The Marketing Warehouse uses this parameter to identify Ad events. For more information about the parameter set used by Marketing Warehouse, see “Configuring Your Web Site to Collect Marketing Warehouse Data” in *WebTrends Marketing Warehouse User’s Guide*.



### Note

NWT. mc\_i d replaces the WT. mc\_n and WT. mc\_t auto-configuration parameters in earlier versions of WebTrends.

## Campaign Event Parameter

### WT. mc\_ev

WT. mc\_ev=*EventType*

This parameter identifies an ad event type.

WebTrends Analytics and Marketing Warehouse recognize the following values:

WT. mc\_ev=cl i ck identifies an ad clickthrough event.

## WebTrends Analytics Considerations

WebTrends Analytics does not currently provide preconfigured custom reports that use the cl i ck value. However, you can also specify a custom value to identify a custom event type, create a custom report filter based on this value, and apply it to a custom report.

## Segment Parameter

### WT. seg\_X

WT. seg\_X=*Segment*

Identifies a segment of interest. X can be 1, 2, 3, or 4. For example, WT. seg\_1=*Segment 1*. This parameter identifies values associated with this segment, and you can store these values in Visitor History. The Visitor History function recognizes these values and stores them in the Visitor History database. For more information, see “[Visitor Segmentation Parameters](#)” on page 180.

## Page of Interest Parameter

### WT. pi

WT. pi =*Page i denti fication*

This parameter identifies a page on your site that is critical to evaluating performance. *Page i denti fication* can be any string. When you enable **Page of Interest Unique Visitor Tracking** in the Visitor History dialog of a profile, WebTrends stores the values for the WT. pi parameter for each unique visitor in the Visitor History database. You can limit the amount of disk space used to store these values by keeping the strings as short as possible.

As a best practice, you should only identify key pages on your site with this parameter because WebTrends stores a maximum of 20 pages for each unique visitor for each profile.

Use this parameter to create a Page of Interest dimension that can be associated with a measure based on the WT. vr. pi v Visitor History parameter. For more information, see WT. vr. pi v on “[Visitor Tracking Parameters](#)” on page 178.



## On-Site Search Parameters

On-site search parameters allow you to collect activity about your on-site search tool.

### WT. oss

WT. oss=*Search phrase*

Identifies a word or a phrase that visitors submit for an on-site search.

### WT. oss\_r

WT. oss\_r=*number of results*

Identifies whether or not an on-site search is successful. This parameter should be specified on the same hit as WT.oss and should be set to the number of results whenever the on-site search is successful, or to 0 when the search fails (no result).

WebTrends Analytics uses this parameter in preconfigured custom report filters. WebTrends Marketing Warehouse uses it provide data for the Number of Results attribute.

## Registered Visitor Parameter

### WT. rv

WT. rv=1

If you use WebTrends Marketing Warehouse, your web server should pass this parameter with a value of 1 when a visitor has completed a registration process. For more information, see “Configuring Your Web Site to Collect Marketing Warehouse Data” in *WebTrends Marketing Warehouse User’s Guide*.

## Content Parameters

The parameters defined in this section can be used to populate custom reports that include web 2.0 content information.

### WT. rss\_f

WT. rss\_f=*FeedName*

This parameter identifies the RSS subscription feed. To indicate that a feed read request was made for an RSS feed, use this parameter with WT. rss\_ev=f.

WT. rss\_f=Sports&WT. rss\_ev=f indicates that the RSS feed name WT. rss\_f= “sports” was the object of an RSS feed request event.

To indicate that a subscription request was made for an RSS feed, use this parameter with WT. rss\_ev=s.

For example, WT. rss\_f=News&WT. rss\_ev=s indicates that the RSS feed name “News” was the object of an RSS subscription event.

### WT. rss\_a

WT. rss\_a=*ArticleName*

This parameter identifies the RSS article. Use this parameter with WT. rss\_ev=a.

WT. rss\_a=Global %20shortage%20of%20flu%20vaccine&WT. rss\_ev=a indicates that the RSS article name “Global shortage of flu vaccine” was the object of an article read request event.

### WT. rss\_ev

WT. rss\_ev=a or f or s

This parameter identifies the RSS related event that has occurred, such as an article request

WT. rss\_ev=a, feed read request WT. rss\_ev=f, or subscription WT. rss\_ev=s.



**WT. clip\_t**

WT. clip\_t=*MediaType*

This parameter identifies the type of media that users have accessed. Use this parameter with WT. clip\_n= and with WT. clip\_ev=.

WT. clip\_t=Windows%20Media&WT. clip\_n=Milton%20Waddams%20Presents&WT. clip\_ev=v

In this example, the clip type WT. clip\_t= is “Windows Media,” the name of the clip WT. clip\_n= is “Milton Waddams Presents,” and the event type WT. clip\_ev=v is view.

**WT. clip\_n**

WT. clip\_n=*MediaClipName*

This parameter identifies the name of the clip WT. clip\_n= that users have accessed. Typically, use this parameter with event type, WT. clip\_ev=, and media type WT. clip\_t=.

**WT. clip\_ev**

WT. clip\_ev=*EventType*

This parameter identifies the type of media-related event that has occurred, such as a view of a media event, WT. clip\_ev=v.

**WT. ria\_a**

WT. ria\_a=*ApplicationName*

This parameter identifies the name of the Rich Internet Application (RIA) accessed. Typically, use this parameter with WT. ria\_c= to identify RIA content, WT. ria\_f= to identify the RIA feature, and WT. ria\_ev= to identify the RIA event type.

WT. ria\_a=Homepage%20Interactive%20promo%20with%20video%20and%20mp3&WT. ria\_c=Vegas%20Video%201&WT. ria\_f=Play&WT. ria\_ev=play

In this example, the parameter identifies the RIA application as “Homepage interactive promo with video and mp3,” the RIA content as “vegas,” the RIA feature used as “play,” and the RIA event type as “play.”

**WT. ria\_c**

WT. ria\_c=*RIAContent*

This parameter identifies the RIA content, WT. ria\_c=. Typically, use this parameter with WT. ria\_a=, WT. ria\_f=, and WT. ria\_ev=.

**WT. ria\_f=**

WT. ria\_f=*RIAFeature*

This parameter identifies the RIA feature accessed, WT. ria\_f=. Typically, use this parameter with WT. ria\_a=, WT. ria\_c=, and WT. ria\_ev=.

**WT. ria\_ev=**

WT. ria\_ev=*RIAEvent*

This parameter identifies the RIA event that has occurred, such as the selection of a button or feature that is part of the RIA application used. Actions taken in the RIA application such as “play,” “zoom,” or “spin” are examples of an RIA event.

**WT. cgm\_t=**

WT. cgm\_t=*CGMType*

This parameter identifies the type of consumer generated media, WT. cgm\_t=. Use this parameter with WT. cgm\_ev=.

WT. cgm\_t=Blog&WT. cgm\_ev=c



In this example, the parameter identifies the consumer generated media type as “blog,” and the event `WT.cgm_ev=c` is a comment.

**WT.cgm\_ev=**  
`WT.cgm_t=CGMEvent`

This parameter identifies the consumer generated media event that has occurred, such as a post (p) or comment (c).

**WT.test\_v=**  
`WT.test_v=Variant`

This parameter identifies the test variant, `WT.test_v=` for the web site. This parameter can be populated then used to compare activity such as the scenario step conversion rate of two or more test variants.

`WT.test_v=Product%20Layout%20B`

In this example, the web site test variant used is “Product Layout B.”

## SmartView Parameters

This section contains parameters defined specifically for use with WebTrends SmartView. For information about configuring WebTrends for SmartView, see the *SmartView User's Guide*.

**WT.svl**  
`WT.svl=any string`

For SmartView to differentiate multiple links on a web page that all lead to the same URL, use the `WT.svl` query parameter to uniquely identify the links. For example, if you have two links on your home page that both go to the store page, you should use the SmartView query parameter to identify each link.

### To use the SmartView query parameter:

1. Place the `WT.svl` parameter on every page where multiple links lead to the same page. For example, `http://www.mydomain.com/?WT.svl=link1`.
2. Assign each link a unique value. SmartView uses `WT.svl` to assign the appropriate measure values to individual links.

---

### Notes



Do not include this parameter in your URL Rebuilding definitions. WebTrends automatically recognizes this parameter and uses it only when creating SmartView custom reports. If you include `WT.svl` in a URL Rebuilding definition, non-SmartView reports are affected. If you exclude it in a URL Rebuilding definition, WebTrends won't be able to use it to differentiate links when creating SmartView reports.

---

**WT.tsp**  
`WT.tsp=1`

Identifies transition source pages for SmartView. The JavaScript creates and passes this query parameter if you enabled SmartView page transition tracking in the JavaScript tag and you tagged the source page with the SmartView page transition META tag. A source page is a page that you want to be tracked for SmartView reporting. For more information, see “Configuring SmartView Using JavaScript Tags” in the *SmartView User's Guide*.



Do not use this query parameter to tag your web site. You can use this parameter to focus the analysis on only page transition pages using custom report filters.

**WT. ttp**

WT.ttp=1

Identifies transition target pages for SmartView. The JavaScript creates and passes this query parameter if you enabled SmartView page transition tracking in the JavaScript tag and you tagged the previously viewed page with the SmartView page transition META tag. For more information, see “Configuring SmartView Using JavaScript Tags” in the *SmartView User’s Guide*.

Do not use this query parameter to tag your web site. You can use this parameter to focus the analysis on only page transition pages using custom report filters.

## Stored Visitor Parameter

The Stored Visitor parameter identifies the unique visitor ID you assign to your visitors.

**WT. dcsvi d**

WT. dcsvi d=*anystring*

### WebTrends Analytics Considerations

If you enable Visitor History in a profile, when WebTrends Analytics detects this parameter, it is stored in the Visitor History database. When the Visitor History database is exported, this parameter is exported along with each visitor for the purpose of identifying visitors. For more information about visitor history see “Visitor History Parameters” on page 167.

### WebTrends Marketing Warehouse Considerations

Marketing Warehouse uses this parameter to populate the **External Visitor ID** field of the Visitor table, which allows you to link to external visitor data in the Extended Attributes Database. For more information about the complete set of query parameters that Marketing Warehouse uses, see “Configuring Your Web Site to Collect Marketing Warehouse Data” in the *WebTrends Marketing Warehouse User’s Guide*.

## Visitor History Parameters

The WebTrends analysis process can generate and maintain parameters that support visitor profiling when you enable Visitor History in your profiles. These visitor-related parameters are stored in a Visitor History Table for each profile that has Visitor History enabled. Because Visitor History parameters are handled by WebTrends, you should not use these parameters in META tags on your web pages.

Parameters that describe elapsed time periods in days are calculated as a complete 24-hour period. Thus, if a visitor visits for the first time at 1:00 on Monday, then any visit before 1:00 on Tuesday is considered as zero days since the first visit (WT. vr. fvd), even though the actual day is different.

## Most Recent Campaign Parameters

WebTrends generates and maintains these parameters when you enable Visitor History and select the Campaign History category in your profile. For more information, see “Visitor History Parameters”.

**WT. vr. rac**

WT. vr. rac=*MostRecentCampaign*



Identifies the visitor's most recent campaign. Of all of the campaigns, the "most recent" campaign is the one that was most recently added to the visitor history table. This is a single value (no multiples allowed). This parameter is not set if the visitor has never had a campaign.

## Unique Visitors for Campaigns

You can use the following set of parameters to track unique visitors for daily, weekly, monthly, quarterly, yearly, and lifetime campaigns. The active campaign list is used to determine the values. Therefore, you can get an additional campaign unique visitor when it is referenced again. These parameters are only available for reporting if you use the default campaign translation file. For more information, see "Lookup Tables for Drilldowns" in the *Administration User's Guide*.

### WT.vr.rac\_dc

WT.vr.rac\_dc=*CampaignDemandChannel*

Identifies the campaign demand channel. It is set to the description corresponding to the ID in the translation file.

### WT.vr.rac\_de

WT.vr.rac\_de=*CampaignDescription*

This parameter is set to the description corresponding to the ID in the translation file.

### WT.vr.rac\_cr

WT.vr.rac\_cr=*CampaignCreative*

This parameter is set to the creative corresponding to the ID in the translation file. A campaign creative is an attribute of a specific offer, for example, a "Buy Now!" graphic. A specific offer may consist of many creatives.

### WT.vr.rac\_ct

WT.vr.rac\_ct=*CampaignCreativeType*

This parameter is set to the creative type corresponding to the ID in the translation file.

### WT.vr.rac\_ma

WT.vr.rac\_ma=*CampaignMarketingActivity*

This parameter is set to the marketing activity corresponding to the ID in the translation file.

### WT.vr.rac\_mp

WT.vr.rac\_mp=*CampaignMarketingProgram*

This parameter is set to the marketing program corresponding to the ID in the translation file.

### WT.vr.rac\_of

WT.vr.rac\_of=*CampaignOffer*

This parameter is set to the offer corresponding to the ID in the translation file.

### WT.vr.rac\_pa

WT.vr.rac\_pa=*CampaignPartner*

This parameter is set to the partner corresponding to the ID in the translation file.

### WT.vr.rac\_pl

WT.vr.rac\_pl=*CampaignPlacement*

This parameter is set to the placement corresponding to the ID in the translation file.



## Most Recent Campaign Visitors

These parameters are generated and maintained by the WebTrends analysis process when you enable Visitor History and select the Campaign History category in your profile. For more information, see [“Visitor History Parameters” on page 167](#).

**WT. vr. rac\_d**

WT. vr. rac\_d=1

Identifies a visitor's first visit for a day for the campaign specified in WT. mc\_i d (campaign ID) on the hit.

**WT. vr. rac\_w**

WT. vr. rac\_w=1

Identifies a visitor's first visit for a week for the campaign specified in WT. mc\_i d (campaign ID) on the hit.

**WT. vr. rac\_m**

WT. vr. rac\_m=1

Identifies a visitor's first visit for a month for the campaign specified in WT. mc\_i d (campaign ID) on the hit.

**WT. vr. rac\_q**

WT. vr. rac\_q=1

Identifies a visitor's first visit for a quarter for the campaign specified in WT. mc\_i d (campaign ID) on the hit.

**WT. vr. rac\_y**

WT. vr. rac\_y=1

Identifies a visitor's first visit for a year for the campaign specified in WT. mc\_i d (campaign ID) on the hit.

**WT. vr. rac\_f**

WT. vr. rac\_f=1

Identifies a visitor's first visit for the campaign specified in WT. mc\_i d (campaign ID) on the hit.

## Visitor “Initial” Parameters

The following list describes several visitor parameters that keep track of the “first” aspects of a visitor's history with the site. Note that although the parameters use the terminology “first,” all preconfigured objects based on these parameters use the term “initial” (for example, Initial Referrer is the dimension based on WT. vr. fr). WebTrends generates these parameters when you enable Visitor History. For more information, see [“Visitor History Parameters” on page 167](#).

**WT. vr. fr**

WT. vr. fr=*FirstReferrer*

Identifies the visitor's first recorded referrer. The format is the same as that for the Referring Domains dimension (for example, googl e. com). This is set on the first hit of the first visit and does not change afterwards.

**WT. vr. fc**

WT. vr. fc=*FirstCampaign*

Identifies the visitor's first recorded marketing campaign. The format is the same as that for the Campaign dimension. Only campaigns identified using the WT. mc\_i d parameter are counted. Campaigns defined solely through WebTrends Administration are not used.

This parameter is not provided until the visitor visits with a campaign. At that point the value and parameter are set and will never change.



**WT. vr. fe**

*WT. vr. fe=FirstEntryPage*

Identifies the visitor’s first recorded page view. The format is a page URL without query parameters. This is set on the first hit of the first visit and never changes afterwards.

## Elapsed Time Parameters

These parameters are generated and maintained by the WebTrends analysis process when you enable Visitor History. For more information, see “Visitor History Parameters” on page 167.

**WT. vr. fvd**

*WT. vr. fvd=DaysSinceFirstVisit*

Identifies the days since the visitor’s first visit. This is an integer containing the days since the visitor’s first visit. The value is truncated (for example, if 47 hours has passed since the first visit, the value is 1). This parameter is not provided on a visitor’s first visit.

When using this as a measure for the Visitor dimension, use the maximum value.

When using this as a measure for dimensions other than Visitor, you usually configure the average value.

The sum of this measure has no meaning.

**WT. vr. pvd**

*WT. vr. pvd=DaysSincePreviousVisit*

Days since the Visitor’s Previous Visit. This is an integer containing the days since the visitor’s previous visit.

This parameter makes most sense when used as a visit filter. This parameter is not provided on a visitor’s first visit.

When using this parameter as a measure for the Visitor dimension, use the maximum value. When using this parameter as a measure for dimensions other than Visitor, you usually configure the average value.

The sum of this measure has no meaning.

**WT. vr. pvdb**

*WT. vr. pvdb=High/Moderate/Some/Low descriptor*

Classifies days since the visitor’s previous visit into one of four categories. This parameter is non-numeric and is used as a dimension.

You can change the number of categories by editing the `vrbucket.ini` file. By default this file is configured as follows:

```
[PVDBValues]
High = 4
Moderate = 8
Some = 12
```

The following table describes the meaning of the default values.

[PVDBValues]	Meaning
High=4	$x \leq 4$
Moderate=8	$4 < x \leq 8$



[PVDBValues]	Meaning
Some=12	$8 < x \leq 12$

By default, the maximum number of days for Some is set to 12. Any visitor whose last visit was more than 12 days ago is assigned to the Low Recency category. *Recency* is to the number of days since a visitor's most recent visit.

Example 1: Adjust all of the ranges in `vrbucket.ini` to:

```
[PVDBValues]
High = 7
Moderate = 14
Some = 21
```

Example 2: To create two buckets (for example, *High/Low*), change `vrbucket.ini` to:

```
[PVDBValues]
High = 4
Moderate = 4
Some = 4
```

The result of Example 2 is:

```
High is ≤ 4
Low is > 4
```

#### WT. vr. ppd

WT. vr. ppd=*DaysSincePreviousPurchase*

Identifies the days since the visitor's previous purchase. This is an integer value.

This parameter is typically used as a visit filter. This parameter is not provided until the visitor makes the first purchase and does change on every hit that a purchase is made after that.

When using this parameter as a measure for the Visitor dimension, use the maximum value. When using this parameter as a measure for dimensions other than Visitor, you will usually configure the average value.

The sum of this measure has no meaning.

#### WT. vr. lat

WT. vr. lat=*VisitLatency*

Visit Latency. The visit latency is the number of days since the visitor's first visit (WT. vr. fvd) divided by the number of visit intervals (WT. vr. vc). It gives an indication of the average elapsed time between visits. This parameter is not provided on the first visit.

When using this parameter as a measure for the Visitor dimension, use the maximum value. When using this as a measure for dimensions other than Visitor, you usually configure the average value.

The sum of this measure has no meaning.

## Historical Counts Parameter

WebTrends generates and maintains these parameters by the WebTrends analysis process when you enable Visitor History. For more information, see "[Visitor History Parameters](#)" on page 167.

#### WT. vr. vc

WT. vr. vc=*VisitCount*



Identifies the total number of visits recorded for a visitor. This is an integer representing the number of visits since the visitor's first visit. When using this as a measure for the Visitor dimension, use the maximum value. When using this as a measure for dimensions other than Visitor, you usually configure the average value.

The sum of this measure has no meaning.

**WT. vr. vcb**

WT. vr. vcb=*High/Moderate/Some/Low descriptor*

Classifies the value of the WT. vr. vc parameter into one of four categories.

This parameter is non-numeric and used as a dimension. You can change the number of categories by editing the `vrbucket.ini` file. By default, the file is configured as follows:

```
[VCBValues]
High = 25
Moderate = 15
Some = 5
```

The following table shows the meaning of the default values:

[VCBValues]	Meaning
High=25	$25 < x$
Moderate=15	$15 > x \leq 25$
Some=5	$5 > x \leq 15$

By default, the minimum value for Some is 5. Any visitor whose visit count value is less than 5 is assigned to the Low value category.

Example 1: Adjust all of the ranges in `vrbucket.ini` to:

```
[VCBValues]
High = 50
Moderate = 30
Some = 10
```

Example 2: To create two buckets (for example, *High/Low*), change `vrbucket.ini` to:

```
[VCBValues]
High = 25
Moderate = 25
Some = 25
```

The result of example 2 is:

```
High is >25
Low is >0 ≤25
```

## Historical Transactions/Purchases Parameters

WebTrends generates and maintains these parameters by the WebTrends analysis process when you enable Visitor History. For more information, see ["Visitor History Parameters" on page 167](#).

The following parameters are calculated using the transaction parameters. For more information, see ["Transaction Parameters" on page 160](#).



**WT. vr. vv**

WT. vr. vv = *VisitorValue*

WebTrends generates this Visitor History parameter to track the visitor's overall value, which is the value of all purchases recorded for a visitor over time. This is a floating-point value containing the amount of money spent by this visitor back to and including the visitor's first visit.

When using this as a measure for dimensions other than Visitor, you usually configure the average value.

The sum of this measure has no meaning.

**WT. vr. ltb**

WT. vr. ltb = *High/Moderate/Some/Low* descriptor

WebTrends generates this Visitor History parameter to classify the WT. vr. vv parameter value in one of four categories.

The value is non-numeric and is used as a custom report dimension.

You can change the number of categories by editing the `vrbucket.ini` file. By default, the file is configured as follows:

```
[LTBValues]
High = 750
Moderate = 500
Some = 250
```

The following table shows the meaning of the default values:

[LTBValues]	Meaning
High=750	$x > 750$
Moderate=500	$500 > x \leq 750$
Some=5	$250 > x \leq 500$

By default, the minimum value for Some is 250. Any visitor whose value is less than 250 is assigned to the Low category.

Example 1: Adjust all of the ranges in `vrbucket.ini` to:

```
[LTBValues]
High = 1500
Moderate = 1000
Some = 500
```

Example 2: To create two buckets (for example, *High/Low*), change `vrbucket.ini` to:

```
[LTBValues]
High = 750
Moderate = 750
Some = 750
```

The result of example 2 is:

```
High is >750
Low is >0 ≤750
```

**WT. vr. ppv**

WT. vr. ppv = *PreviousPurchaseValue*



WebTrends generates this Visitor History parameter to track the visitor's previous purchase amount. This is a floating-point value containing the amount spent on the most recent purchase. This parameter is not generated until a visitor makes the first purchase.

When using this parameter as a custom report measure for the Visitor dimension, use the maximum value. When using this parameter as a measure for dimensions other than Visitor, you usually configure the average value.

The sum of this measure has no meaning.

**WT. vr. vp**

WT. vr. vp=*VisitorPurchases*

WebTrends generates this Visitor History parameter to track the total number of purchases ever made by a visitor. The value is an integer containing the number of purchase transactions (not the total number of units purchased) back to and including the visitor's first visit.

When using this parameter as a custom report measure for the Visitor dimension, use the maximum value. When using this parameter as a measure for dimensions other than Visitor, you usually configure the average value.

The sum of this measure has no meaning.

**WT. vr. fpd**

WT. vr. fpd=*DaysSinceFirstPurchase*

WebTrends generates this Visitor History parameter to track the days since the visitor's first purchase. The value is an integer containing the days since the visitor's first purchase.

This parameter is best used as a visit filter or as a custom report measure. This parameter is not generated until a visitor makes the first purchase.

When using this parameter as a measure for the Visitor dimension, use the maximum value. When using this parameter as a measure for dimensions other than Visitor, you usually configure the average value.

The sum of this measure has no meaning.

**WT. vr. bpd**

WT. vr. bpd=*DaysBeforeFirstPurchase*

WebTrends generates this Visitor History parameter to track the days before the visitor's first purchase. The value is an integer containing the number of days between the visitor's first visit and the first purchase.

This parameter is best used as a visit filter or as a measure. This parameter is not provided until a visitor makes the first purchase.

When using this as a measure for dimensions other than Visitor, you usually configure the average value.

The sum of this measure has no meaning.

WebTrends generates the following set of parameters to track unique buyers and buyer's status for daily, weekly, monthly, quarterly, yearly, and lifetime periods.

**WT. vr\_brws**

WT. vr\_brws=*Buyer/Non-Buyer*

WebTrends generates this query parameter on a visitor's first visit of the day. Its value indicates whether a visitor has purchased in the past. Buyer indicates that the visitor has purchased before. Non-buyer indicates that the visitor has not purchased.



A visitor is considered a buyer if the WT. tx\_s parameter is passed in the query string for the visit. A visitor is considered a non-buyer if WT. tx\_s is not passed in the query string. In addition, WebTrends uses the following invoice parameters to evaluate whether WT. tx\_s should be used for buyer determination:

- WT. tx\_i d and WT. tx\_i t
- WT. tx\_i

If both WT. tx\_i d and WT. tx\_i t parameters are passed during the visit and are properly formatted, WebTrends uses them to evaluate whether WT. tx\_s should be used for buyer determination. If the date and time specified by these parameters is older than the time of the hit by more than the configured invoice age limit, WT. tx\_s is not used and the visitor is considered a non-buyer. The invoice age limit is set at two days, meaning that any invoices three days or older than the first hit associated with the invoice are not used for buyer determination.

If the WT. tx\_i parameter is passed during the visit, WebTrends uses it to evaluate whether WT. tx\_s should be used for buyer determination. WebTrends looks for the invoice number passed in for this query parameter in the visitor's invoice history. If it has seen this invoice number before, WT. tx\_s is not used and the visitor is considered a non-buyer. Three invoices are kept per visitor. Invoices more than two days old are purged from the visitor's invoice history.

**WT. vr\_by**

WT. vr\_by=*New Buyer/Repeat Buyer*

WebTrends generates this Visitor History parameter when a visitor makes a purchase. It indicates whether the visitor is purchasing for the first time or has purchased before.

**WT. vr.bt\_d**

WT. vr.bt\_d=1

WebTrends generates this Visitor History parameter to identify the visitor's first daily purchase. This parameter is generated and set to 1 for the first purchase from a visitor for a day.

**WT. vr.bt\_w**

WT. vr.bt\_w=1

WebTrends generates this Visitor History parameter to identify the visitor's first weekly purchase. This parameter is generated and set to 1 for the first purchase from a visitor during a week.

**WT. vr.bt\_m**

WT. vr.bt\_m=1

WebTrends generates this Visitor History parameter to identify the visitor's first monthly purchase. This parameter is generated and set to 1 for the first purchase from a visitor during a month.

**WT. vr.bt\_q**

WT. vr.bt\_q=1

WebTrends generates this Visitor History parameter to identify the visitor's first quarterly purchase. This parameter is generated and set to 1 for the first purchase from a visitor during a quarter.

**WT. vr.bt\_y**

WT. vr.bt\_y=1

WebTrends generates this Visitor History parameter to identify the visitor's first yearly purchase. This parameter is generated and set to 1 for the first purchase from a visitor during a year.

**WT. vr.bt\_f**

WT. vr.bt\_f=1

WebTrends generates this Visitor History parameter to identify the visitor's first purchase. This parameter is generated and set to 1 for the first purchase from a visitor.



## Search Engine Parameters

The Search Engine parameters keep track of initial and most recent search engines and search engine phrases for a visitor. These parameters work in conjunction with the `WT. srch` parameter to determine whether the referring search engine was from a paid search phrase. WebTrends generates and maintains these parameters when you enable Visitor History and select the Search Engine History category in your profile. For more information, see [“Visitor History Parameters” on page 167](#).

### **WT. vr. i se**

*WT. vr. i se=Initial SearchEngine*

WebTrends generates this Visitor History parameter to track the visitor's initial search engine. This parameter contains the string identifying the initial search engine for a visitor. The parameter is generated with the hit where it is recognized as the referring site. The value of the parameter never changes and is provided with the first hit of every visit after the visit in which it is recognized. The parameter is not provided until it has been recognized and set to the initial value.

### **WT. vr. i sep**

*WT. vr. i sep=Initial SearchEnginePhrase*

WebTrends generates this Visitor History parameter to track the visitor's initial search engine phrase. This parameter contains the string identifying the initial search engine phrase for a visitor. The parameter is generated with the hit where the search engine is recognized as the referring site. The value of the parameter never changes and is provided with the first hit of every visit after the visit in which it is recognized. The parameter is not provided until it has been recognized and set to the initial value.

### **WT. vr. i pd\_se**

*WT. vr. i pd\_se=Initial PaidSearchEngine*

WebTrends generates this Visitor History parameter to track the visitor's initial paid search engine. This parameter contains the string identifying the initial paid search engine for a visitor. A paid search engine referrer is identified by a `WT. srch=1` parameter in the query field of hit query string. The `WT. vr. i pd_se` parameter is provided with the hit where it is recognized as the referring site. The value of the parameter never changes and is provided with the first hit of every visit after the visit in which it is recognized. The parameter is not provided until it has been recognized and set to the initial value.

### **WT. vr. i pd\_sep**

*WT. vr. i pd\_sep=Initial PaidSearchEnginePhrase*

WebTrends generates this Visitor History parameter to track the visitor's initial paid search engine phrase. This parameter contains the string identifying the initial paid search engine phrase for a visitor. A paid search engine referrer/phrase is identified by a `WT. srch=1` parameter in the query field of hit query string. The `WT. vr. i pd_sep` parameter is provided with the hit where it `WT. vr. i pd_se` is recognized. The value of the parameter never changes and is provided with the first hit of every visit after the visit in which it is first recognized. The parameter is not provided until it has been recognized and set to the initial value.

### **WT. vr. i og\_se**

*WT. vr. i og\_se=Initial OrganicSearchEngine*

WebTrends generates this Visitor History parameter to track the visitor's initial organic search engine. This parameter contains the string identifying the initial organic search engine for a visitor. An organic search engine referrer is identified by the lack of a `WT. srch=1` parameter in the query field of hit query string. The `WT. vr. i og_se` parameter is provided with the hit where it is recognized as the referring site. The value of the parameter never changes and is provided with the first hit of every visit after the visit in which it is recognized. The parameter is not provided until it has been recognized and set to the initial value.



**WT. vr. i og\_sep**

WT. vr. i og\_sep=*Initial OrganicSearchEnginePhrase*

WebTrends generates this Visitor History parameter to track the visitor's initial organic search engine phrase. This parameter contains the string identifying the initial paid search engine phrase for a visitor. An organic search engine referrer/phrase is identified by the lack of a WT. srch=1 parameter in the query field of hit query string. The WT. vr. i og\_sep parameter is provided with the hit where it WT. vr. i og\_se is recognized. The value of the parameter never changes and is provided with the first hit of every visit after the visit in which it is first recognized. The parameter is not provided until it has been recognized and set to the initial value.

**WT. vr. r\_se**

WT. vr. r\_se=*MostRecentSearchEngine*

WebTrends generates this Visitor History parameter to track the visitor's most recent search engine. This parameter contains the string identifying the most recent search engine for a visitor. The WT. vr. r\_se parameter is provided with the hit where it is recognized as the referring site. The value of the parameter changes whenever a new search engine is recognized. It is provided with the first hit of every visit after the visit in which it is recognized. It changes whenever a new search engine is recognized. The parameter is not provided until it has been recognized and set to a first value.

**WT. vr. r\_sep**

WT. vr. r\_sep=*MostRecentSearchEnginePhrase*

WebTrends generates this Visitor History parameter to track the visitor's most recent search engine phrase. This parameter contains the string identifying the most recent search engine phrase for a visitor. The WT. vr. r\_sep parameter is provided with the hit where WT. vr. r\_se is recognized as the referring site. The value of the parameter changes whenever a new search engine/search engine phrase is recognized. It is provided with the first hit of every visit after the visit in which it is recognized. It changes whenever a new search engine is recognized. The parameter is not provided until it has been recognized and set to a first value.

**WT. vr. rpd\_se**

WT. vr. rpd\_se=*MostRecentPaidSearchEngine*

WebTrends generates this Visitor History parameter to track the visitor's most recent paid search engine. This parameter contains the string identifying the most recent paid search engine for a visitor. A paid search engine referrer/phrase is identified by a WT. srch=1 parameter in the query field of hit query string. The WT. vr. rpd\_se parameter is provided with the hit where it is recognized as the referring site. The value of the parameter changes whenever a new search engine is recognized. It is provided with the first hit of every visit after the visit in which it is recognized. It changes whenever a new search engine is recognized. The parameter is not provided until it has been recognized and set to a first value.

**WT. vr. rpd\_sep**

WT. vr. rpd\_sep=*MostRecentPaidSearchEnginePhrase*

WebTrends generates this Visitor History parameter to track the visitor's most recent paid search engine phrase. This parameter contains the string identifying the most recent paid search engine phrase for a visitor. A paid search engine referrer/phrase is identified by a WT. srch=1 parameter in the query field of hit query string. The WT. vr. rpd\_sep parameter is provided with the hit where WT. vr. rpd\_se is recognized as the referring site. The value of the parameter changes whenever a new search engine/search engine phrase is recognized. It is provided with the first hit of every visit after the visit in which it is recognized. It changes whenever a new search engine is recognized. The parameter is not provided until it has been recognized and set to a first value.

**WT. vr. rog\_se**

WT. vr. rog\_se=*MostRecentOrganicSearchEngine*



WebTrends generates this Visitor History parameter to track the visitor's most recent organic search engine. This parameter contains the string identifying the most recent organic search engine for a visitor. An organic search engine referrer/phrase is identified by the lack of a WT. srch=1 parameter in the query field of hit query string. The WT. vr. rog\_se parameter is provided with the hit where it is recognized as the referring site. The value of the parameter changes whenever a new search engine is recognized. It is provided with the first hit of every visit after the visit in which it is recognized. It changes whenever a new search engine is recognized. The parameter is not provided until it has been recognized and set to a first value.

**WT. vr. rog\_sep**

WT. vr. rog\_sep=*MostRecentOrganicSearchEnginePhrase*

WebTrends generates this Visitor History parameter to track the visitor's most recent organic search engine phrase. This parameter contains the string identifying the most recent organic search engine phrase for a visitor. An organic search engine referrer/phrase is identified by the lack of a WT. srch=1 parameter in the query field of hit query string. The WT. vr. rog\_sep parameter is provided with the hit where WT. vr. rog\_se is recognized as the referring site. The value of the parameter changes whenever a new search engine/search engine phrase is recognized. It is provided with the first hit of every visit after the visit in which it is recognized. It changes whenever a new search engine is recognized. The parameter is not provided until it has been recognized and set to a first value.

## Visitor Tracking Parameters

Visitor Tracking parameters let you track daily, weekly, monthly, quarterly, yearly, and all-time unique visitors. WebTrends generates and maintains these parameters when you enable Visitor History and select the Visit History category in your profile. For more information, see "[Visitor History Parameters](#)" on [page 167](#).

**WT. vr. vt\_d**

WT. vr. vt\_d= 1

WebTrends generates this Visitor History parameter to track daily visitor daily activity. This parameter is present and set to 1 on a new visitor's first hit for the day.

**WT. vr. vt\_w**

WT. vr. vt\_w= 1

WebTrends generates this Visitor History parameter to track weekly visitor activity. This parameter is present and set to 1 on a new visitor's first hit for the week.

**WT. vr. vt\_m**

WT. vr. vt\_m= 1

WebTrends generates this Visitor History parameter to track monthly visitor activity. This parameter is present and set to 1 on a new visitor's first hit for the month.

**WT. vr. vt\_q**

WT. vr. vt\_q= 1

WebTrends generates this Visitor History parameter to track quarterly visitor activity. This parameter is present and set to 1 on the first hit from a new visitor during a quarter.

**WT. vr. vt\_y**

WT. vr. vt\_y= 1

WebTrends generates this Visitor History parameter to track yearly visitor activity. This parameter is present and set to 1 on the first hit from a new visitor during a year.

**WT. vr. vt\_f**

WT. vr. vt\_f= 1



WebTrends generates this Visitor History parameter to track a visitor's first hit. This parameter is present and set to 1 on the first hit from a new visitor.

**WT. vr. pi v\_d**

WT. vr. pi v\_d= 1

WebTrends generates this Visitor History parameter to track daily page of interest activity. This parameter is present and set to 1 on the first hit from a new visitor to a page of interest during a day.

**WT. vr. pi v\_w**

WT. vr. pi v\_w= 1

WebTrends generates this Visitor History parameter to track weekly page of interest activity. This parameter is present and set to 1 on the first hit from a new visitor to a page of interest during a week.

**WT. vr. pi v\_m**

WT. vr. pi v\_m= 1

WebTrends generates this Visitor History parameter to track monthly page of interest activity. This parameter is present and set to 1 on the first hit from a new visitor to a page of interest during a month.

**WT. vr. pi v\_q**

WT. vr. pi v\_q= 1

WebTrends generates this Visitor History parameter to track quarterly page of interest activity. This parameter is present and set to 1 on the first hit from a new visitor to a page of interest during a quarter.

**WT. vr. pi v\_y**

WT. vr. pi v\_y= 1

WebTrends generates this Visitor History parameter to track yearly page of interest activity. This parameter is present and set to 1 on the first hit from a new visitor to a page of interest during a year.

**WT. vr. pi v\_f**

WT. vr. pi v\_f= 1

WebTrends generates this Visitor History parameter to track a visitor's first page of interest. This parameter is present and set to 1 on the first hit from a new visitor to a page of interest.

**WT. vr. cg v\_d**

WT. vr. cg v\_d= 1; . . .

WebTrends generates this Visitor History parameter to track daily content group activity. This parameter is present and set to 1 for the first hit from a new visitor to a content group during a day. If multiple content groups are specified on the hit, this parameter contains as many values as there are content groups.

**WT. vr. cg v\_w**

WT. vr. cg v\_w= 1; . . .

WebTrends generates this Visitor History parameter to track weekly content group activity. This parameter is present and set to 1 for the first hit from a new visitor to a content group during a week. If multiple content groups are specified on the hit, this parameter contains as many values as there are content groups.

**WT. vr. cg v\_m**

WT. vr. cg v\_m= 1; . . .

WebTrends generates this Visitor History parameter to track monthly content group activity. This parameter is present and set to 1 for the first hit from a new visitor to a content group during a month. If multiple content groups are specified on the hit, this parameter contains as many values as there are content groups.

**WT. vr. cg v\_q**

WT. vr. cg v\_q= 1; . . .

WebTrends generates this Visitor History parameter to track quarterly content group activity. This parameter is present and set to 1 for the first hit from a new visitor to a content group during a quarter.



**WT. vr. cgv\_y**

WT. vr. cgv\_y=1; . . .

WebTrends generates this Visitor History parameter to track yearly content group activity. This parameter is present and set to 1 for the first hit from a new visitor to a content group during a year. If multiple content groups are specified on the hit, this parameter contains as many values as there are content groups.

**WT. vr. cgv\_f**

WT. vr. cgv\_f=1; . . .

WebTrends generates this Visitor History parameter to track a visitor's first hit to a content group. This parameter is present and set to 1 for the first hit from a new visitor to a content group. If multiple content groups are specified on the hit, this parameter contains as many values as there are content groups.

## Visitor Segmentation Parameters

Visitor Segmentation parameters allow you to store the most recent value of a segmentation query parameter for inclusion in your reports. For example, your travel web site tracks visitors using a segmentation parameter, such as WT. seg\_1, to identify the visitor's "traveler type." The result of the visitor's most recent traveler type value is stored in the WT. vhseg\_1 parameter in the Visitor History database. Custom reports you create that use the WT. vhseg\_1 parameter as a dimension show statistics for the most recent value of the key parameter.

The most recent value of the key parameter, WT. seg\_x is stored in the corresponding result parameter. To report on visitor segmentation data, create a custom report that uses your result parameter as a dimension.

WebTrends generates and maintains these parameters when you enable Visitor History and select the Custom Visitor Segmentation category in your profile. You must also implement the WT. seg parameter on your web pages. For more information about WT. seg, see ["Segment Parameter" on page 163](#).

**WT. vhseg\_1**

WT. vhseg\_1=VisitorSegment1Result

**WT. vhseg\_2**

WT. vhseg\_2=VisitorSegment2Result

**WT. vhseg\_3**

WT. vhseg\_3=VisitorSegment3Result

**WT. vhseg\_4**

WT. vhseg\_4=VisitorSegment4Result

## SDC-Generated Visitor Parameters

The following subsections discuss visitor-related parameters that are generated and maintained by SmartSource Data Collector (SDC).

### Visitor Tracking Parameters

Visitor Tracking parameters allow you to track daily, weekly, monthly, quarterly, and yearly unique visitors. SDC inserts these parameters into the cs-uri-query strings.

**WT. vt\_tlv**

WT. vt\_tlv=UNIX Time



SDC generates this parameter to identify the time of the visitor's last visit. The value is expressed as the number of seconds since 1970 (standard UNIX time), which is calculated using information stored in the third-party cookie value. SDC only sets this parameter at the start of a new visit. On a visitor's first visit, the value is set to zero. If you disable cookie tracking, this parameter is not generated or passed in the query string.

The Marketing Warehouse use this query parameter to determine whether a new visit should also be counted as a new daily, weekly, monthly, quarterly, or yearly visitor. For example, if the day of the new visit is different than the day of the previous visit, relative to the GMT offset of the JavaScript tag, the visit is counted as a new daily visit.

**WT.vt\_f\_tlv**

WT.vt\_f\_tlv=*UNIX Time*

SDC generates this parameter to identify the time of the visitor's last visit. The value is expressed as the number of seconds since 1970 (standard UNIX time), which is calculated using information stored in the first-party cookie value. SDC only sets this parameter at the start of a new visit. On a visitor's first visit, the value is set to zero. If you disable first-party cookie tracking in the JavaScript tag, this parameter is not generated or passed in the query string.

The Marketing Warehouse use this query parameter to determine whether a new visit should also be counted as a new daily, weekly, monthly, quarterly, or yearly visitor. For example, if the day of the new visit is different than the day of the previous visit, relative to the GMT offset of the JavaScript tag, the visit is counted as a new daily visit.

**WT.vt\_d**

WT.vt\_d=1

SDC generates this parameter to track daily visitors for Express Analysis. This parameter is generated and set to 1 for the first hit from a new visitor for a day.

**WT.vt\_a\_d**

WT.vt\_d=1

SDC generates this parameter to track daily visitors for Account Rollup Data Sources. This parameter is generated and set to 1 for the first hit from a new visitor for a day for a given account. This parameter is used in conjunction with Account Rollup Profiles.

**WT.vt\_f\_d**

WT.vt\_f\_d=1

SDC generates this parameter to track daily visitors for Express Analysis only. This parameter is generated and set to 1 for the first hit from a new visitor when you configure your SmartSource Data Source to use the First-Party Cookie JavaScript. The First-Party Cookie JavaScript generates this parameter and its value.

**WT.vt\_s**

WT.vt\_s=1

SDC generates this parameter to track visitor sessions for Express Analysis and the WebTrends Marketing Warehouse.

This parameter is generated and set to 1 for the first hit for a new session. Cookie tracking must be enabled to set this query parameter.

**WT.vt\_a\_s**

WT.vt\_a\_s=1

SDC generates this parameter to track visitor sessions for Account Rollup Data Sources. Applies to Real Time analysis only. This parameter is generated and set to 1 for the first hit for a new session for a given account. This parameter is used in conjunction with Account Rollup Profiles.

**WT.vt\_f\_s**

WT.vt\_f\_s=1



SDC generates this parameter to track new visitor sessions for Express Analysis and WebTrends Marketing Warehouse.

This parameter is generated and set to 1 for the first hit for a new session. The First-Party Cookie JavaScript generates this parameter.

**WT. vt\_f**

WT. vt\_f= 1

SDC generates this parameter to track new and returning visitors. This parameter is generated and set to 1 for the first hit from a new visitor. This parameter is set to 2 if the visitor's browser does not accept cookies. The First-Party Cookie JavaScript generates this parameter.

**WT. vt\_f\_a**

WT. vt\_f\_a= 1

SDC generates this parameter to track new visitors for an account. This parameter is generated and set to 1 for the first hit from a new visitor for a given account. This parameter is set to 2 if the visitor's browser does not accept cookies. The First-Party Cookie JavaScript generates this parameter. This parameter is used in conjunction with Account Rollup Profiles.

**WT. vt\_si d**

WT. vt\_si d=*i d e n t i f i e r*

SDC generates this parameter to identify visitor sessions for WebTrends Marketing Warehouse. This parameter is generated on every hit to identify the visitor session.

This identifier is formed by concatenating two pieces of data:

- *Identifier*: The value of WT. co\_f. If WT. co\_f is not present on the incoming hit, SDC generates this value. This is the unique identifier that is generated at the time of a new visit. It is 32-character hexadecimal number (0-9, a-f)
- *Timestamp*: The time when the session began. This is the number of milliseconds since January 1, 1970.

**Note**

The First Party Cookie Javascript no longer emits WT. vt\_si d. Rather, it now emits the two components (*Identifier* and *timestamp*) as separate parameters. See WT. vti d and WT. vtvs for more information.

Example:

WT. vt\_si d=10. 61. 19. 29-3899933120. 29768655. 1141069

**WT. vti d**

WT. vti d=*i d e n t i f i e r*

This parameter is used to identify visitors for WebTrends Marketing Warehouse. By default, this identifier is provided by WebTrends Data Collection Servers using an in-line JavaScript request for vti d. js at the time of a new visit. Alternatively, this parameter value can come from a custom query parameter or customer cookie.

*Identifier*: Any alphanumeric string that uniquely identifies a visitor. By default this value is the same as the WT. co\_f parameter.

**WT. vtvs**

WT. vtvs=*t i m e s t a m p*

The JavaScript tag generates this parameter to identify visitor sessions for WebTrends Marketing Warehouse. This parameter is generated on every hit to identify the visitor session.



**Timestamp:** The time when the session began. This is the number of milliseconds since January 1, 1970.

## Cookie Detection Parameters

This set of parameters allows WebTrends Analytics to tie a visitor's first hit with the rest of the visitor session. These parameters are generated and maintained by SmartSource Data Collector (SDC).

### WT.co

WT.co= *Yes/No*

SDC generates this parameter to determine whether the visitor's browser supports and is configured to accept cookies. Valid values are Yes and No.

Example:

```
var coQueryParam = "&WT.co=" + navigator.cookieEnabled() ? "Yes" : "No";
```

### WT.co\_d

WT.co\_d=*Cookie\_Data*

SDC generates this parameter the first time it attempts to set the cookie. The value is set to the value of the cookie. This parameter is generated only for "first visit" hits.

WebTrends Analytics and the Marketing Warehouse use this parameter for "session stitching." The first hit of the first visitor session (which may not have a cookie) gets the cookie value in the WT.co\_d parameter. Subsequent hits that have the same value for the WebTrends cookie can be tied together with the WT.co\_d hit to form a complete picture of the session. Also, SDC passes the WT.co\_d value; it is not passed from the visitor's web browser.

The WebTrends cookie format contains the IP address of the cookie's connection address and the creation time. The creation time is represented as the number of seconds and nanoseconds since 1970 (standard UNIX time). A checksum is appended to the cookie value. The following example shows the format of the cookie value:

```
WEBTRENDS_ID=IP Address-SSSSSSSS.NNNNNN: checksum
```

The following example shows the WT.co\_d parameter with a WebTrends cookie value:

```
WT.co_d=192.168.100.40-1045156016.29542554:A2D3FC34517CE562A9D4E33EF85D7B7F
```

### WT.co\_a

WT.co\_a=*Cookie\_Data*

SDC generates this parameter the first time it attempts to set the account cookie. Note that this parameter is generated only for hits on the first visit. This parameter is used to track visitor sessions across multiple accounts in WebTrends Analytics On Demand or multiple SmartSource data sources in WebTrends Analytics software.

This parameter is generated and the value is set to the account rollup cookie's value if SDC attempted to set a first-time cookie for a given account. The global rollup cookie is named AC00KI E. The global rollup cookie contains an encoded account rollup cookie named WT\_ACCT. It contains the IP address of the cookie's connection address and the creation time. The creation time is represented as the number of seconds and nanoseconds since 1970 (standard UNIX time). A checksum is appended to the cookie value. The following example shows the format of the cookie value:

```
WT_ACCT=IP Address-SSSSSSSS.NNNNNN: checksum
```

The following example show the WT.co\_a parameter with an account rollup cookie:

```
WT.co_a=192.168.100.40-1045156016.29542554
```

### WT.co\_f

WT.co\_f=*unique identifier*



SDC generates this parameter when you enable first-party cookie tracking in the WebTrends JavaScript tag. This parameter is passed on every hit so that WebTrends Analytics and the Marketing Warehouse can use it for visitor session tracking.

A unique identifier is passed as the value of the `WT.co_f` query parameter. The format of the unique identifier depends on the first-party cookie tracking method you specified in the SmartSource Data Source settings.

You can configure the method that you want to use for your first party cookies in the SmartSource Data Source settings. For more information about first-party cookie tracking methods, see [“Tracking Visitor Sessions” on page 83](#).

## URL Truncation Parameter

SDC uses this parameter to overcome a maximum URL length limitation imposed by Internet Explorer (Microsoft Knowledge Base Article – 208427). The URL length must be 2048 or less. If the WebTrends JavaScript tag generates a URL in excess of 2048 characters and the client browser is Microsoft Internet Explorer, the hit is truncated and is passed to SDC.

### WT.tu

`WT.tu=1`

SDC generates this parameter and sets it to 1 if the URL was deemed too long and truncated by the JavaScript tag. If present, SDC writes an error and discards the hit.

You can configure the `logtruncatedhits` setting to log the truncated hit rather than discard it.

## HTTP Headers

You may want to access custom HTTP request headers. These headers can be inserted by third-party products such as load balancers, application servers, or web server plug-ins. This parameter is assigned the value of the specified HTTP header, which can then be referenced in a WebTrends custom report.

### WT.hdr. HTTP Header

`WT.hdr.HTTP Header=Value`

If the header is present in the incoming request, the header name is appended to `WT.hdr.` and the header value is assigned to the value. For example, suppose that a customer wants to log the `Accept:` header, and it comes in as `Accept: */*`. The resultant parameter would be `WT.hdr.Accept=*/*`. Note that values are URL encoded.

## JavaScript Tag Version

### WT.tv

`WT.tv=major.minor.revision`

The JavaScript tag contains this parameter, which specifies the version of the WebTrends JavaScript tag that is currently deployed. The value is passed as `major.minor.revision` where `major.minor` specifies the WebTrends Analytics version and `revision` specifies the version of the JavaScript tag.

Although this parameter is not used in reports, it can be useful for Support when troubleshooting a tagging problem.

## Site ID

### WT.site

`WT.site=siteid`



This Marketing Warehouse parameter enables the web site owner to supply a site ID for one or more events when multiple events occur on the same hit. The JavaScript tag generates and sets this parameter to the domain name/Site Id for cross-domain Warehouse use, enabling users to select only events for a certain ID when querying the database.

Currently, this parameter is not supported in Tag Builder and must be configured manually.

## Event Tracking

### WT. dl

**WT. dl =event /d**

Specifies the kind of event tracked. The WT. dl parameter passes a set of identifiers. Each identifier is associated with a given event which is typically mouse related. Identifiers are numeric, and are assigned inside an event handler. The WT. dl parameter can be used to filter event related traffic.

The JavaScript tag generates this parameter with the appropriate event id as shown in the following table:

Event Id	Event	Description
0	Page View	Generated when page is loaded.
20	Download	Generated when a download link is clicked. Download links are onsite links whose file type matches a configurable list of download file types.
21	Anchor	Generated when an anchor link is clicked. Anchor links are on-site links that contain anchor text.
22	Dynamic "javascript"	Generated when a link containing a JavaScript URL is clicked. Example: <a href="j avascr ipt: al ert(' Good Morni ng' )" >Good Morni ng</a>
23	Dynamic "mailto:"	Generated when a link containing a mailto URL is clicked. Example: <a href="mai l to: fi rstname. l astname@company. com? subj ect=Good%20Morni ng" >Send Emai l </a>
24	Off-site	Generated when an off-site link is clicked. Off-site links are page elements that lead to web sites that are not instrumented with the customers SmartSource tags. The list of on-site domains is user-configurable in WebTrends Administration.



Event Id	Event	Description
25	Right-click	Generated when a download link is right-clicked. Download links are on-site links whose file type matches a configurable list of download file types. The list of download file types is user-configurable in WebTrends Administration.
26	Form Button - Get method	Generated when a form button is clicked. Button is enclosed inside a form, and the method is GET.
27	Form Button - Post method	Generated when a form button is clicked. Button is enclosed inside a form, and the method is POST.
28	Form Button - Input tag	Generated when a form button is clicked. Button is not enclosed inside a form, but is enclosed in an <input> tag.
29	Form Button - Button tag	Generated when a form button is clicked. Button is not enclosed inside a form, but is enclosed in a <button> tag
30	Image Map	Generated when an image map is clicked.

## Parent DIV/Table ID

WT. nv

WT. nv=<i d| | class>

This parameter contains the id or class of the parent DIV or TABLE of the element that was clicked. The WT. nv value allows you to see the areas on a page to which an element was clicked belongs.

## Click-Based Tracking

WT. es

WT. es=hostname/page

This parameter permits tracking click-based events and the source page from which they originated. The JavaScript tag forms this parameter by concatenating the dcssi p and dcsuri parameters to determine what page a user was on when an "event" occurred. For example, when a user clicks on a link, the event source shows what page the user was viewing when the click occurred. Although in the case of a page view, these URLs are identical, in the case of a click the dcsuri is actually the "destination URL" of the click, while the event source is the "source URL" of the click.

## DCSID

WT. dcs\_id

WT. dcs\_id=DCSID



This parameter contains the value of the DCSID that generated the hit. The Standard Analysis Engine, Express Analysis Engine, and Event Database Loader pass the DCSID as value of for this parameter. This parameter becomes most useful when tracking multiple DCSIDs in SmartSource files from multiple sites. In this case, you can use this parameter to segment your report data by site. For example, you can use this parameter as a dimension in a custom report to report on activity for each site.

**Note**

If you look at your SmartSource files, you will not find this parameter in the query string. This is because WebTrends Analytics adds it to the query parameter set during analysis.

## SDC-Parameter Override Parameters

You can use the parameters in this section to override SDC parameters on the client side. Consider the following example:

If you want a specific page, `/xyz.html`, logged to the `cs-uri-stem` field, you can assign the page name to the `dcsuri` parameter in the JavaScript tag as shown in the Modified Tag.

Default JavaScript Tag:

```
DCS.dcsuri = "window.location.pathname;
```

Modified JavaScript Tag:

```
DCS.dcsuri = "/xyz.html ";
```

However, because modifying JavaScript is error-prone, you could instead use the `DCS.dcsuri` parameter in a META tag to override the `dcsuri` assignment in the JavaScript tag. Your META tag would look like this:

```
<META NAME="DCS.dcsuri " CONTENT="/xyz.html ">
```

Keep in mind that because these parameters simply override assignments in the JavaScript tag, the parameters themselves are not actually sent to SDC. The JavaScript tag contains a custom object named `DCS`. This object contains property name/value pairs that are used to form query parameters sent to SDC. To continue our example, the JavaScript tag first extracts the META tag information and performs the following assignment:

```
DCS.dcsuri = /xyz.html
```

Next, the JavaScript tag iterates through all name/value pairs in the `DCS` object and forms query parameters.

In our example, the following query parameter is formed:

```
&dcsuri = /xyz.html
```

Note that the custom object name itself (`DCS`) is not sent to SDC.

**DCS.dcsref**

```
DCS.dcsref = Referrer
```

This parameter is assigned to the `dcsref` parameter before the hit is sent to SDC. The value is included in the `cs(Referrer)` field of the log file.

**DCS.dcssip**

```
DCS.dcssip = Domain
```



This parameter is assigned to the `dcscip` parameter before the hit is sent to SDC. The value is included in the `cs-host` field of the log file.

**DCS. dcsua**

`DCS. dcsua=user agent`

This parameter is assigned to the `dcsua` parameter before the hit is sent to SDC. Value is included in the `cs(user agent)` field. Use a plug sign to encode spaces rather than `%20`.

**DCS. dcsuri**

`DCS. dcsuri=uri-stem`

This parameter is assigned to the `dcsuri` parameter before the hit is sent to SDC. The value is included in the `cs-uri-stem` field of the log file.

**DCS. dcspro**

`DCS. dcspro=Protocol`

This parameter is assigned to the `dcspro` parameter before the hit is sent. The value is included in the `cs(Version)` field of the log file.

**DCS. dcsqry**

`DCS. dcsqry=uri-query`

This parameter is assigned to the `dcsqry` parameter before the hit is sent. The value is included in the `cs-uri-query` field of the log file.

**DCS. dcsaut**

`DCS. dcsaut=authenticated username`

This parameter is assigned to the `dcsaut` parameter before the hit is sent to SDC. The value is included in the `cs-username` field.

**DCS. dcsmet**

`DCS. dcsmet=method`

This parameter is assigned to the `dcsmet` parameter before the hit is sent to SDC. The value is included in the `cs-method` field.

**DCS. dcssta**

`DCS. dcssta=status`

This parameter is assigned to the `dcssta` parameter before the hit is sent to SDC. The value is included in the `sc-status` field.

**DCS. dcsbyt**

`DCS. dcsbyt=bytes`

This parameter is assigned to the `dcsqry` parameter before the hit is sent to SDC. The value finally is included in the `sc-bytes` field.

**DCS. dcscip**

`DCS. dcscip=ip address`

This parameter is assigned to the `dcscip` parameter before the hit is sent to SDC. Value is included in the `c-ip` field.

**DCS. dcsua**

`DCS. dcsua=user agent`

This parameter is assigned to the `dcsua` parameter before the hit is sent to SDC. Value is included in the `cs(user agent)` field. Use a plug sign to encode spaces rather than `%20`.



## Conversion Plug-In Parameters

The WebTrends encoding conversion plug-in uses the parameters in this section during the character encoding conversion process. The WebTrends JavaScript tag generates these parameters when the `gi 18n` global variable is set to `true` and when a web page contains DCSext query parameters.

For more information about DCSext query parameters, see “Customizing the WebTrends JavaScript Tag” in *WebTrends Analytics On Demand Implementation Guide*. For more information about the encoding conversion plug-in see “Internationalization and WebTrends” in the *Administration User’s Guide*.

### WT. dep

`WT. dep=DCSext parameter1[; DCSext parameter2. . . ]`

Contains a semicolon-delimited list of the custom DCSext query parameters on a web page. The encoding conversion plug-in uses this information to identify the parameters that are known to be encoded in UTF-8. For example, a web page that contains `DCSext. abc=655` and `DCSext. xyz=889` would be captured by the JavaScript tag as `WT. dep=abc; xyz.`



### Note

Do not use this parameter for collecting data.

## Document Revision History

Table 15: Document Revision History contains a summary of changes made to this document beginning with the release of WebTrends Analytics, version 8.7.

**Table 15: Document Revision History**

Software Version	Date of Last Update	Summary of Changes
v8.7	March, 2009	Added Document Revision History section. Corrected event ID typo's (for Dynamic “Mail To” and “Off-site” events) for “WT.dl” on page 185
v8.7d	July, 2009	Added footer link to Documentation Center.





---

# Index

## A

- action rights
  - about 77
- adding users 79
- Advertising Click
  - META tags 69
  - parameters 151
- Advertising View parameters 150
- analysis
  - memory requirements 104
- Analysis Engine 39, 40
- archiving 103
- Auto-Configuration parameters 149

## B

- backing up WebTrends data 103
- browser size query parameter 158
- bugs
  - submitting to WebTrends vi

## C

- caching files from an FTP server 107
- connection type query parameter 157
- Content Group META tags 64
- cookie information 83
- cookie query parameters 183
- cookie rejection 83
- cookies
  - browser rejection 83
  - defined 83
- Counts parameters 171
- cross-domain tracking 89
- Customer Center vi

## D

- data
  - archiving 103
  - restoring from backup 103
- data repositories 129

- data storage 101
- data storage locations 129
- database
  - backing up MySQL 133
- DCSID query parameter 186
- distributed architecture 27
  - managing 34
- distributed installation
  - installing Main Application 30
  - managing 34
- DNS resolution 107

## E

- Elapsed Time parameters 170

## F

- feedback, sending to WebTrends vi
- first-party cookie 85
- Flash installed query parameter 158
- FTP caching 107

## G

- GeoTrends 39, 40

## H

- home page query parameter 158
- hosts, monitoring 34
- HTML page title lookups 108



## I

- installation
  - custom or distributed 27
- installation, unattended. See unattended installation 37
- Internet resolution 107
- IP addresses
  - caching 107
  - resolving 107

## L

- licensing
  - checking page view balance in WTOD 2
  - checking status in WTOD 2
- log file rotation
  - overview 101

## M

- marketing campaign META tags 65
- marketing campaign parameters 150
- Marketing Warehouse 39, 40
- memory limiting 104
- META tag syntax 69
- META tags
  - advertising clicks 69
  - content groups 64
  - customizing 63
  - deploying 71
  - for Parent-Child profiles 66
  - marketing campaigns 65
  - onsite advertising 68
  - page titles 70
  - revenue 67
  - servers 65
  - shopping cart activity 68
  - URLs 69–70
- Microsoft SQL Server
  - unattended installation, planning 41
- MySQL
  - backing up 133
- mysqldump.exe 133

## O

- On Demand Database (ODDB) 106
- onsite Advertising META tags 68

## P

- page title lookups 108
- page title META tags 70
- Parent-Child profiles
  - META tags 66

- performance 103
- performance issues 108
- permissions. See user rights
- physical data storage 101
- profile and template rights
  - about 77

## Q

- query parameter
  - DCSID 186

## R

- RAM limitations, managing 104
- reports
  - managing database storage 106
  - scheduling 105
  - storing 105
- restoring from backup 103
- revenue META tags 67
- rights
  - see user rights 77
- roles
  - see user roles 78
- rolling over log data 101
- rotating logs 101

## S

- Scenario Analysis parameters 151
- SDC-Generated Visitor parameters 180
- SDC-Parameter Override parameters 187
- Secure Sockets Layer 99
  - enabling for WebTrends 99
- server META tags 65
- Server parameters 151
- services
  - stopping 22
- Shopping Cart activity META tags 68
- Shopping Cart parameters 153
- simultaneous analysis 108
- Split parameters 155
- SQL Server
  - planning unattended installation 41
- storage issues 103
- System Database 38, 40

## T

- table limiting 104
- third-party cookie 84
- thrashing 104
- Title parameters 155
- titles, HTML pages 108
- Transactions/Purchases parameters 172



## U

- UI Server 39, 40
- unattended installation 37
  - defined 37
  - performing 41
  - planning for Microsoft SQL Server 41
  - preparing for 37
  - prerequisites 40
- URL META tags 69–70
- user rights 77
  - about 77
  - overwriting with a role 79
  - View Only 80
- user roles
  - about 78
  - applying to existing users 79
  - creating 78
  - predefined settings 81
- Users 79
- users, adding 79

## V

- View Only user rights 80
- Visitor Firsts parameters 169
- Visitor History parameters 167
- Visitor Tracking parameters 180

## W

- Web Client parameters 156
- WebTrends
  - data restoration 129
  - data storage 129
- WebTrends components, installation with SQL Server 50
- WebTrends Query parameters 145
- WebTrends services
  - stopping 22
- WT.bs query parameter 158
- WT.ct query parameter 157
- WT.dep 189
- WT.fi query parameter 158
- WT.hp query parameter 158



