

# Diagnostic Manager

Reference Guide

August 27, 2018

## Copyright

© Copyright Metalogix International GmbH and Idera, Inc., 2004-2018
All rights reserved. No part or section of the contents of this material may be reproduced or transmitted in any form or by any means without the written permission of Metalogix International GmbH.

Diagnostic Manager™ is a trademark of Metalogix International GmbH and Idera, Inc.

Windows SharePoint Services is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries. Other product and company names mentioned herein may be the trademarks of their respective owners.

## Technical Support

For information about Metalogix Technical support visit http://metalogix.com/support.

Technical support specialists can be reached by phone at +1-202-609-9100. The level of technical support provided depends upon the support package that you have purchased. Contact us to discuss your support requirements.

# Contents

**Welcome to Metalogix Diagnostic Manager ..... 5**

How does Metalogix Diagnostic Manager help me? ..... 5

**Performance Monitoring ..... 7**

SharePoint Server Performance Monitoring ..... 7

Page Availability and Performance Monitoring ..... 7

**Server Performance Information Retrieval ..... 8**

Information Collected ..... 8

WMI Scopes Analyzed ..... 8

Objects and Properties Accessed ..... 9

CPU ..... 9

Disk ..... 9

IIS ..... 10

Memory ..... 11

Network ..... 11

Paging File ..... 12

SharePoint Service Status ..... 12

Web Service ..... 13

SQL Server ..... 13

SQL Server Database ..... 14

SharePoint Excel Calculation Services ..... 14

SharePoint Excel Services ..... 16

SharePoint Search ..... 16

SharePoint Search Indexer ..... 17

**Page Component Performance Data Collection ..... 19**

**Page Component Data Collection ..... 20**

**Troubleshooting ..... 21**

Troubleshooting WMI Issues ..... 21

Checking for WMI, Performance Class, and Class Instance Availability ..... 22

Enabling a Disabled Performance Counter ..... 26

Changing the WMI Counter Timeout ..... 27

Resynchronizing and Repairing the WMI Repository ..... 27

Assigning Additional Memory to the WMI Pool ..... 28

Online Resources ..... 28
Customer Portal ..... 29

# Welcome to Metalogix Diagnostic Manager

Metalogix Diagnostic Manager is a powerful and intuitive monitoring solution that primarily helps administrators ensure the health of their SharePoint environment and the performance of their SharePoint applications. It saves time and money by dramatically reducing administrative overhead for IT and ensuring SharePoint business user productivity. Administrators can also use Diagnostic Manager to monitor servers, client machines, and Web pages not associated with a SharePoint environment.

Metalogix Diagnostic Manager provides real time monitoring from a single console, enabling you to quickly identify, diagnose, and resolve performance and availability problems. It also provides critical information for capacity planning by storing historical performance data of all the monitored servers. It monitors in real time, providing a comprehensive set of metrics that enable administrators to quickly pinpoint performance and availability issues related to server performance, resource usage and poorly performing HTML controls, solutions and Web controls.

Key features include the following:

- Continuous, automated monitoring from a central console.
- Intuitive dashboard user interface for an "at-a-glance" view of SharePoint and server health.
- Page component level performance analysis.
- Summary and detail performance metrics.
- Automated alerting of poorly performing pages, controls and server performance issues.
- Repository of historical performance for trending and analysis.
- Comprehensive reporting.
- Centralized SharePoint Unified Logging System (ULS) logging for all SharePoint servers in the farm.

## How does Metalogix Diagnostic Manager help me?

Metalogix Diagnostic Manager lets you monitor and analyze your Microsoft SharePoint farms, servers, ULS logs, and pages. The ways in which Diagnostic Manager can facilitate a number of tasks is described in the following table.

| Task | How |
|---|---|
| Offers easy installation and use. | Installs quickly, enabling immediate performance access to any page. A typical installation takes less than 30 minutes. Additionally, the easy-to-use UI makes it very easy to identify all performance issues on a single page. |
| Solve SharePoint performance problems even when SharePoint is not available. | Because Metalogix Diagnostic Manager runs outside of SharePoint, you have access to historical data, alerts issued, up to the minute performance information, and much more for all SharePoint servers in your farm. Metalogix Diagnostic Manager helps eliminate the risk of |

| Task | How |
|---|---|
| | spending hours trying to find out what might have caused your SharePoint environment to fail or perform badly. |
| Take control of your SharePoint health. | Continuously monitor, diagnose, discover, analyze and fix SharePoint performance and availability issues efficiently. Quickly understand why your SharePoint servers are not performing well or why pages are taking long to load. |
| Out-of-the-box monitoring. | Instant and automatic monitoring of all SharePoint farm servers without the need to develop any kind of script. Take advantage of built-in default options to start monitoring your SharePoint servers immediately. |
| Provides accurate and clean server-side performance metrics. | Understand the real impact of the server-side performance metrics without interference from browser type, location, operating systems and connection speed. Have control of how an application is behaving in its 'raw' state, and easily determine any application performance issues. |
| Quickly identifies worst performing page controls. | Immediately identify the worst performing HTML controls, solutions and Web controls by quickly analyzing their load time, size and type. |
| Collects component-level results. | Powerful collection of page control elements that clearly differentiates page loading, execution, or rendering of individual page elements. Quickly and easily identify what is impacting page performance, without guessing! |
| Provides true SharePoint-specific analysis. | Enables true visibility into SharePoint by exposing the inner workings of server controls, solutions, menus and other page elements. Using advanced methods, Metalogix Diagnostic Manager allows clear analysis of all the Web page controls and parts before they are rendered into HTML and delivered to the user. |
| Quickly identifies any potential problems for any servers in the farm. | Provides notifications of any potential problems found in any of the SharePoint ULS logs for any of the servers in the SharePoint farm. (SharePoint 2010, 2013, and 2016) |
| Monitor the health of Office 365 tenants | Provides status information for Office 365 services (including SharePoint Online, Exchange Online, and Skype for Business) and their features. |

# Performance Monitoring

Metalogix Diagnostic Manager helps you to monitor and analyze the performance of your SharePoint farms. Metalogix Diagnostic Manager lets you:

- Monitor the status of SharePoint farms.
- Monitor the status of specified SharePoint pages.
- Analyze the performance of the components that make up specified pages.

You can use these tools to help find and correct SharePoint performance and availability issues.

## SharePoint Server Performance Monitoring

Metalogix Diagnostic Manager uses the Windows Management Interface (WMI) to collect performance and availability information (see Server Performance Information Retrieval) from the servers in your SharePoint farm. You can use the Management Console to review the data for the period that you specify. You can also use the Management Console to specify the conditions that trigger alerts, and the response to those alerts.

## Page Availability and Performance Monitoring

You can use Metalogix Diagnostic Manager to monitor and analyze pages that you specify.

When you monitor the page, Metalogix Diagnostic Manager tracks the load times and availability for the page at specified intervals. You can specify when Metalogix Diagnostic Manager generates alerts. You can monitor any page in a monitored SharePoint farm. You do not need to modify your SharePoint farm or the pages to monitor pages in this way. Monitoring page availability and load times has a low impact on your SharePoint farm.

Metalogix Diagnostic Manager can also analyze the performance of the components that make up a monitored SharePoint page. When you perform component analysis, you deploy a Metalogix Diagnostic Manager solution. The Collection Service uses the solution to analyze the performance of the components that make up the page. As with page load time monitoring, you can specify when Metalogix Diagnostic Manager generates alerts.

Normally, when SharePoint generates a monitored page, the solution is unused and has no impact on your SharePoint performance.

This comprehensive page analysis is separate from page availability monitoring. Because it is comprehensive, page component analysis takes additional time to perform. You should generally not perform comprehensive analysis on every monitored page every time you monitor the page. Instead, you should monitor page load times and availability on a routine basis. When routine monitoring locates a problem with a page, you can analyze the page components to further isolate the source of the problem.

# Server Performance Information Retrieval

The Collection Service uses the Windows Management Interface (WMI) to collect performance information from servers in the SharePoint farms that you monitor.

## Information Collected

The Collection Service retrieves information about the operating system, including the number of physical and logical processors in the CPU, and the physical and virtual memory. For Web front end (WFE) servers, the Collection Service also retrieves the installed IIS and ASP.NET versions.

In addition, the Collection Service retrieves SharePoint-specific information from the WFE servers and SQL Server-specific information from the database servers in your farm. The SharePoint specific information includes information about the SharePoint services and the performance of search, index, and other services. The SQL-specific information includes information about the database and log file sizes, and performance information for the server and the databases that it hosts.

The Metalogix Diagnostic Manager Collection Service includes the PageUtil.exe utility that lists all of the WMI metrics, arranged into groups and categories. By default, the utility is stored in the c:\Program Files (x86)\Metalogix\Diagnostic Manager\CollectionService\PageUtil.exe directory on the computer that hosts the Collection Service.

Open a command prompt in the Collection Service directory and type PageUtil.exe -m then press Enter to view the list.

## WMI Scopes Analyzed

The Collection Service accesses the following WMI scopes to collect information.

| Scope | Descriptions |
|---|---|
| \\<ComputerName>\root\default | Used to access the registry on the remote server. |
| \\<ComputerName>\root\cimv2 | Used to collect performance information. |
| \\<*ComputerName*>\root\Microsoft\SqlServer\ComputerManagement10 | Used to access the list of SQL Server instances. |
| \\<*ComputerName*>\root\Microsoft\SqlServer\ComputerManagement | Used to access the list of SQL Server 2003 instances. |

| Scope | Descriptions |
|---|---|
| \\<ComputerName>\root\mscluster | Used on SQL Server cluster nodes to determine the active/passive node in the cluster. |

## Objects and Properties Accessed

The Collection Service accesses the following WMI classes and fields on each server in the farm:

### CPU

| Item | WMI Class | Class Property |
|---|---|---|
| Processor Interrupts per Second (Count) | Win32_PerfRawData_PerfOS_Processor | InterruptsPersec |
| CPU Usage Total - Privileged Time (Percent) | Win32_PerfRawData_PerfOS_Processor | PercentPrivilegedTime |
| CPU Usage Total (Percent) | Win32_PerfRawData_PerfOS_Processor | PercentProcessorTime |
| CPU Usage Total - User Time (Percent) | Win32_PerfRawData_PerfOS_Processor | PercentUserTime |
| Processor Queue Length (Count) | Win32_PerfRawData_PerfOS_System | ProcessorQueueLength |
| CPU Usage - Per-Process Privileged Time (Percent) | Win32_PerfRawData_PerfProc_Process | PercentPrivilegedTime |
| CPU Usage - Per-Process Total (Percent) | Win32_PerfRawData_PerfProc_Process | PercentProcessorTime |
| CPU Usage - Per-Process User Time (Percent) | Win32_PerfRawData_PerfProc_Process | PercentUserTime |

### Disk

| Item | WMI Class | Class Property |
|---|---|---|
| Disk Free Space (MB) | Win32_PerfFormattedData_PerfDisk_LogicalDisk | FreeMegabytes |

| Item | WMI Class | Class Property |
|---|---|---|
| Disk Free Percent (Percent) | Win32_PerfFormattedData_PerfDisk_LogicalDisk | PercentFreeSpace |
| Disk Used (Percent) | Win32_PerfFormattedData_PerfDisk_LogicalDisk | PercentFreeSpace |
| Disk Idle Time (Percent) | Win32_PerfFormattedData_PerfDisk_LogicalDisk | PercentIdleTime |
| Average Disk Seconds Per Read (ms) | Win32_PerfRawData_PerfDisk_LogicalDisk | AvgDisksecPerRead |
| Average Disk Seconds Per Transfer (ms) | Win32_PerfRawData_PerfDisk_LogicalDisk | AvgDisksecPerTransfer |
| Average Disk Seconds Per Write (ms) | Win32_PerfRawData_PerfDisk_LogicalDisk | AvgDisksecPerWrite |
| Disk Size (MB) | Win32_PerfRawData_PerfDisk_LogicalDisk | PercentFreeSpace_Base |
| Average Disk Queue Length (Count) | Win32_PerfRawData_PerfDisk_PhysicalDisk | AvgDiskQueueLength |
| Disk Reads Per Second (Count) | Win32_PerfRawData_PerfDisk_PhysicalDisk | DiskReadsPersec |
| Disk Writes Per Second (Count) | Win32_PerfRawData_PerfDisk_PhysicalDisk | DiskWritesPersec |
| Disk Time (Percent) | Win32_PerfRawData_PerfDisk_PhysicalDisk | PercentDiskTime |

## IIS

| Item | WMI Class | Class Property |
|---|---|---|
| IIS Current Connections (Count) | Win32_PerfFormattedDat a_ W3SVC_WebService | CurrentConnections |
| IIS Maximum Con- nections (Count) | Win32_PerfFormattedDat a_ W3SVC_WebService | MaximumConnections |
| Application Pool Recycles | Win32_PerfRawData_ APPPOOLCountersProv APPPOOLWAS | TotalApplicationPoolRecycles ider_ |

| Item | WMI Class | Class Property |
|---|---|---|
| (Count) | | |
| IIS Server Services Not Running | Win32_Service | Display Name, Started, StartMode, StartName, State, Status |

## Memory

| Item | WMI Class | Class Property |
|---|---|---|
| Total Physical Memory (KB) | Win32_ComputerSystem | TotalPhysicalMemory |
| Free Physical Memory (KB) | Win32_OperatingSystem | FreePhysicalMemory |
| Free Virtual Memory (KB) | Win32_OperatingSystem | FreeVirtualMemory |
| Total Virtual Memory (KB) | Win32_OperatingSystem | TotalVirtualMemorySize |
| Memory Pages Per Second (Count) | Win32_PerfRawData_PerfOS_Memory | PagesPersec |
| Per-Process Handle Count (Count) | Win32_PerfRawData_PerfProc_Process | HandleCount |
| Per-Process Page Faults Per Second (Count) | Win32_PerfRawData_PerfProc_Process | PageFaultsPersec |
| Per-Process Pool Non-paged Bytes (Count) | Win32_PerfRawData_PerfProc_Process | PoolNonpagedBytes |
| Per-Process Memory Usage(MB) | Win32_PerfRawData_PerfProc_Process | PrivateBytes |
| Per-Process Thread Count (Count) | Win32_PerfRawData_PerfProc_Process | ThreadCount |
| Per-Process Working Set (MB) | Win32_PerfRawData_PerfProc_Process | WorkingSet |

## Network

| Item | WMI Class | Class Property |
|---|---|---|
| RedirectorSessions (Count) | Win32_PerfRawData_PerfNet_Redirector | ServerSessions |

| Item | WMI Class | Class Property |
|---|---|---|
| RedirectorSessions Hung (Count) | Win32_PerfRawData_PerfNet_Redirector | ServerSessionsHung |
| Network Bytes Received PerSecond (Count) | Win32_PerfRawData_ Tcpip_NetworkInterface | BytesReceivedPersec |
| Network Bytes Sent PerSecond (Count) | Win32_PerfRawData_ Tcpip_NetworkInterface | BytesSentPersec |
| Network Bytes Total PerSecond (Count) | Win32_PerfRawData_ Tcpip_NetworkInterface | BytesTotalPersec |
| Network Packets Per Second (Count) | Win32_PerfRawData_ Tcpip_NetworkInterface | PacketsPersec |

## Paging File

| Item | WMI Class | Class Property |
|---|---|---|
| Paging File Percent Usage (Percent) | Win32_Per-fFormattedData_PerfOS_ PagingFile | PercentUsage |
| Paging File Percent Usage Peak (Percent) | Win32_Per-fFormattedData_PerfOS_ PagingFile | PercentUsagePeak |
| Paging File Size (KB) | Win32_PerfRawData_PerfOS_PagingFile | PercentUsage_ Base |
| Per-Process Page File Bytes (MB) | Win32_PerfRawData_PerfProc_Process | PageFileBytes |

## SharePoint Service Status

| Item | WMI Class | Class Property |
|---|---|---|
| SharePoint Services Not Running | Win32_Service | DisplayName,Started,StartMode,StartName,State,Status |

## Web Service

| Item | WMI Class | Class Property |
|---|---|---|
| Web Svc Bytes Received Per Second (Count) | Win32_PerfRawData_W3SVC_WebService | BytesReceivedPersec |
| Web Svc Bytes Sent Per Second (Count) | Win32_PerfRawData_W3SVC_WebService | BytesSentPersec |
| Web Svc Bytes Total Per Second (Count) | Win32_PerfRawData_W3SVC_WebService | BytesTotalPersec |
| Web Svc Connection Attempts Per Second (Count) | Win32_PerfRawData_W3SVC_WebService | ConnectionAttemptsPersec |
| Web Svc Current Connections (Count) | Win32_PerfRawData_W3SVC_WebService | CurrentConnections |
| Web Svc Total Method Requests (Count) | Win32_PerfRawData_W3SVC_WebService | TotalMethodRequests |

The Collection Service queries the following WMI classes and fields on each server where it detects a running IIS installation:

## SQL Server

| Item | WMI Class | Class Property |
|---|---|---|
| Active Transactions/Total (Count) | Win32_PerfFormattedData_?Databases | ActiveTransactions |
| Data Files Size/Total (KB) | Win32_PerfFormattedData_?Databases | DataFilesSizeKB |
| Log Files Size/Total (KB) | Win32_PerfFormattedData_?Databases | LogFilesSizeKB |
| Transactions Per Second/Total (Count) | Win32_PerfFormattedData_?Databases | TransactionsPersec |
| User Connections (Count) | Win32_PerfFormattedData_?GeneralStatistics | UserConnections |
| Logins Per Second (Count) | Win32_PerfRawData_?GeneralStatistics | LoginsPersec |

| Item | WMI Class | Class Property |
|---|---|---|
| Logouts Per Second (Count) | Win32_PerfRawData_?GeneralStatistics | LogoutsPersec |
| Latch Waits Per Second (Count) | Win32_PerfRawData_?Latches | LatchWaitsPersec |
| Lock Requests Per Second (Count) | Win32_PerfRawData_?Locks | LockRequestsPersec |
| Lock Waits Per Second (Count) | Win32_PerfRawData_?Locks | LockWaitsPersec |
| Number of Deadlocks Per Second (Count) | Win32_PerfRawData_?Locks | NumberofDeadlocksPersec |
| SQL Server Services Not Running | Win32_Service | DisplayName,Started,StartMode,StartName,State,Status |

## SQL Server Database

| Item | WMI Class | Class Property |
|---|---|---|
| Active Transactions (Count) | Win32_PerfFormattedData_?Databases | ActiveTransactions |
| Data Files Size (KB) | Win32_PerfFormattedData_?Databases | DataFilesSizeKB |
| Log Files Size (KB) | Win32_PerfFormattedData_?Databases | LogFilesSizeKB |
| Transactions Per Second (Count) | Win32_PerfFormattedData_?Databases | TransactionsPersec |

The Collection Service queries the following WMI classes and fields on each server where it detects a SharePoint installation:

## SharePoint Excel Calculation Services

| Item | WMI Class | Class Property |
|---|---|---|
| Active Requests (Count) (2007) | Win32_PerfRawData_OfficeServerPerformanceMonitoring_ExcelCalculationServices | ActiveRequests |
| Active Requests (Count) (2010) | Win32_PerfRawData_MicrosoftWindowsSharePointSharePointServer_ExcelCalc | ActiveRequests |

| Item | WMI Class | Class Property |
|---|---|---|
| | ulationServices | |
| Average Request Processing Time (Seconds) (2007) | Win32_PerfRawData_OfficeServerPerformanceMonitoring_ExcelCalculationServices | AverageRequestProcessingTime |
| Average Request Processing Time (Seconds) (2010) | Win32_PerfRawData_MicrosoftWindowsSharePointSharePointServer_ExcelCalculationServices | AverageRequestProcessingTime |
| Average Session Time (Seconds) (2007) | Win32_PerfRawData_OfficeServerPerformanceMonitoring_ExcelCalculationServices | AverageSessionTime |
| Average Session Time (Seconds) (2010) | Win32_PerfRawData_MicrosoftWindowsSharePointSharePointServer_ExcelCalculationServices | AverageSessionTime |
| Requests Received Per Second (Count) (2007) | Win32_PerfRawData_OfficeServerPerformanceMonitoring_ExcelCalculationServices | RequestsReceivedPerSecond |
| Requests Received Per Second (Count) (2010) | Win32_PerfRawData_MicrosoftWindowsSharePointSharePointServer_ExcelCalculationServices | RequestsReceivedPerSecond |
| Requests With Errors Per Second (Count) (2007) | Win32_PerfRawData_OfficeServerPerformanceMonitoring_ExcelCalculationServices | RequestsWithErrorsPerSecond |
| Requests With Errors Per Second (Count) (2010) | Win32_PerfRawData_MicrosoftWindowsSharePointSharePointServer_ExcelCalculationServices | RequestsWithErrorsPerSecond |
| Sessions Per Second (Count) (2007) | Win32_PerfRawData_OfficeServerPerformanceMonitoring_ExcelCalculationServices | SessionsPerSecond |
| Sessions Per Second (Count) (2010) | Win32_PerfRawData_MicrosoftWindowsSharePointSharePointServer_ExcelCalculationServices | SessionsPerSecond |

## SharePoint Excel Services

| Item | WMI Class | Class Property |
|---|---|---|
| Web Front End Active Requests (Count) (2007) | Win32_PerfRawData_OfficeServerPerformanceMonitoring_ExcelServicesWebFrontEnd | ActiveRequests |
| Web Front End Active Requests (Count) (2010) | Win32_PerfRawData_MicrosoftWindowsSharePointSharePointServer_ExcelServicesApplicationWebFrontEnd | ActiveRequests |
| Web Front End Average Request Processing Time (Seconds) (2007) | Win32_PerfRawData_OfficeServerPerformanceMonitoring_ExcelServicesWebFrontEnd | AverageRequestProcessingTime |
| Web Front End Average Request Processing Time (Seconds) (2010) | Win32_PerfRawData_MicrosoftWindowsSharePointSharePointServer_ExcelServicesApplicationWebFrontEnd | AverageRequestProcessingTime |
| Web Front End Requests Per Second (Count) (2007) | Win32_PerfRawData_OfficeServerPerformanceMonitoring_ExcelServicesWebFrontEnd | Requestspersecond |
| Web Front End Requests Per Second (Count) (2010) | Win32_PerfRawData_MicrosoftWindowsSharePointSharePointServer_ExcelServicesApplicationWebFrontEnd | Requestspersecond |

## SharePoint Search

| Item | WMI Class | Class Property |
|---|---|---|
| Blocked Documents (Count) (2007) | Win32_PerfFormattedData_WSSArpi_SharePointSearchArchivalPlugin | Blockeddocuments |
| Blocked Documents (Count) (2010) | Win32_PerfFormattedData_MicrosoftWindowsSharePointSharePointServer_OSSSearchArchivalPlugin | Blockeddocuments |
| Processed Documents per second (Count) (2007) | Win32_PerfFormattedData_WSSGatherer_SharePointSearchGathererProjects | ProcessedDocumentsRate |

| Item | WMI Class | Class Property |
|---|---|---|
| Processed Documents per second (Count) (2010) | Win32_PerfFormattedData_MicrosoftWindowsSharePointSharePointServer_OSSSearchGathererProjects | DocumentsProcessedRate |
| Retries per second (Count) (2007) | Win32_PerfFormattedData_WSSGatherer_SharePointSearchGathererProjects | RetriesRate |
| Retries per second (Count) (2010) | Win32_PerfFormattedData_MicrosoftWindowsSharePointSharePointServer_OSSSearchGathererProjects | RetriesRate |
| Waiting Documents (Count) (2007) | Win32_PerfFormattedData_WSSGatherer_SharePointSearchGathererProjects | WaitingDocuments |
| Waiting Documents (Count) (2010) | Win32_PerfFormattedData_MicrosoftWindowsSharePointSharePointServer_OSSSearchGathererProjects | TransactionsWaiting |
| Idle Threads (Count) (2007) | Win32_PerfFormattedData_WSSGTHRSVC_SharePointSearchGatherer | IdleThreads |
| Idle Threads (Count) (2010) | Win32_PerfFormattedData_MicrosoftWindowsSharePointSharePointServer_OSSSearchGatherer | IdleThreads |

## SharePoint Search Indexer

| Item | WMI Class | Class Property |
|---|---|---|
| Active Connections (Count) (2007) | Win32_PerfFormattedData_WSSIndex_SharePointSearchIndexerCatalogs | ActiveConnections |
| Active Connections (Count) (2010) | Win32_PerfFormattedData_MicrosoftWindowsSharePointSharePointServer_OSSSearchIndexerPlugin | ActiveConnections |

| Item | WMI Class | Class Property |
|---|---|---|
| Documents Filtered (Count) (2007) | Win32_PerfFormattedData_WSSIndex_SharePointSearchIndexerCatalogs | DocumentsFiltered |
| Documents Filtered (Count) (2010) | Win32_PerfFormattedData_MicrosoftWindowsSharePointSharePointServer_OSSSearchIndexerPlugin | DocumentsFiltered |
| Index Size (Count) (2007) | Win32_PerfFormattedData_WSSIndex_SharePointSearchIndexerCatalogs | IndexSize |
| Index Size (Count) (2010) | Win32_PerfFormattedData_MicrosoftWindowsSharePointSharePointServer_OSSSearchIndexerPlugin | IndexSize |
| Queries Failed (Count) (2007) | Win32_PerfFormattedData_WSSIndex_SharePointSearchIndexerCatalogs | QueriesFailed |
| Queries Failed (Count) (2010) | Win32_PerfFormattedData_MicrosoftWindowsSharePointSharePointServer_OSSSearchIndexerPlugin | QueriesFailed |
| Queries Succeeded (Count) (2007) | Win32_PerfFormattedData_WSSIndex_SharePointSearchIndexerCatalogs | QueriesSucceeded |
| Queries Succeeded (Count) (2010) | Win32_PerfFormattedData_MicrosoftWindowsSharePointSharePointServer_OSSSearchIndexerPlugin | QueriesSucceeded |

# Page Component Performance Data Collection

The Metalogix Diagnostic Manager Collection Service can analyze the load performance of individual page components on the SharePoint pages that you specify. The Collection Service uses the solution to analyze the page components that make up the pages that you specify.

When you analyze page performance, the analysis itself can change the page performance; the more comprehensive the analysis, the more changes the analysis introduces. The Metalogix Diagnostic Manager solution imposes the minimum possible performance penalty consistent with accurate analysis. In addition, performance analysis components are only used when performance analysis is actually in progress. When Metalogix Diagnostic Manager is not analyzing page performance, there is no performance impact on your SharePoint pages.

Metalogix Diagnostic Manager uses a custom HTTP module to collect page performance data. The solution can use the HTTP module to analyze any page that is based on a master page template that includes the AdditionalPageHead placeholder. The solution overrides the default AdditionalPageHead to direct the request to the analysis components.

You can examine the HTML that makes up the Master Page template for the SharePoint site to determine if the page includes the AdditionalPageHead placeholder. The Master Page header should include a line similar to the following:

<SharePoint:DelegateControl runat="server" ControlId="AdditionalPageHead" AllowMultipleControls="true"/>

The default Master Page templates include the placeholder.

# Page Component Data Collection

The methods that the solution uses to collect data vary. The solution uses a different data collection method for each type of control that makes up the page.

The solution uses the individual object event handlers to insert a listener into the execution process of ASP.NET controls, including web controls, server controls, user controls, and other ASP.NET elements. The solution overrides the load events and the unload events for each control. The solution uses the timers that it inserts to track component start and end times. Many controls are unloaded in batches by garbage collection processes on the Web front end (WFE) server. That is, the WFE server loads the ASP.NET controls, then keeps them resident until the garbage collector removes items in its queue. Resident items are technically still loaded until the garbage collector removes items in the queue.

SharePoint solutions and ASP.NET solutions are a type of server control but each has unique behaviors. The information that the solutions expose is different from that exposed by a base control. The Metalogix Diagnostic Manager solution treats the SharePoint and ASP.NET solutions as server controls to collect solution performance information from them. Metalogix Diagnostic Manager creates a solution group that is based on the type value of the solutions that it analyzes. Metalogix Diagnostic Manager also uses the SPWebPartManager object for the page to access the data.

Each page in your SharePoint deployment has a single Solution Manager object. SharePoint uses this Solution Manager object to keep track of all of the solutions on the page. The Solution Manager object is accessible within the page context using the this.page.WebPartManager command. Metalogix Diagnostic Manager also accesses the Solution Manager object out-of-band with the SPLimitedWebPartManager object from the GetLimitedWebPartManager method.

The method used to collect the data from the Solution Manager object varies, depending on the calling context. Metalogix Diagnostic Manager can access the SPWebPartManager object inside a solution. Metalogix Diagnostic Manager uses SPLimitedWebPartManager to get solution data from an HTTP module or handler assembly without access to the post-rendered context of a page.

For HTML controls, linked objects, or embedded objects, Metalogix Diagnostic Manager must request the object URI and time the response to measure performance. This method is simple for pages which do not utilize authentication mechanisms. When the page requires valid user credentials for a remote server, the task is more complex. If remote host is also a SharePoint server, timing complexities ensue. In the worst-case scenario, the SharePoint farm includes a complicated domain architecture that requires impersonation techniques to pass credentials to the remote host. As with ASP.NET controls, the solution performs careful filtering to determine which HTML controls have a potentially high-value performance impact and which are simply filler or formatting objects.

Metalogix Diagnostic Manager uses a DOM-parsing method to obtain and analyze the HTML components. In addition, it uses other mechanisms, including RegularExpressions, to build an array of controls for further processing. When the array is populated, the Metalogix Diagnostic Manager solution passes the URLs to the Data Collection service protocol analyzer object to retrieve the objects and generate performance statistics.

# Troubleshooting

## Troubleshooting WMI Issues

When Metalogix Diagnostic Manager is unable to collect WMI information, it generates an alert. The alert includes an error message that indicates the underlying problem. The message that appears is similar to the following:

```
WMI Object Counter names '<Counter Name>' encountered an Invalid Class error in
the WMI query for object "<Object Name>'. This usually indicates that the WMI
objects are not available on the target machine.
```

Diagnostic Manager may be unable to retrieve the WMI information for one or more of the following reasons:

- The WMI Counters are disabled.
- The performance library is overloaded.
- The WMI repository is corrupt.

When you troubleshoot WMI availability, there are several procedures that Metalogix recommends to correct the issue. You should perform the following checks in the specified order. If you encounter an error when you perform any of these checks, please contact Metalogix Technical Support for additional assistance.

1) Verify that the WMI system is available.

2) Verify that the performance counter class is accessible.

3) Verify that the performance counter instance is accessible.

4) Verify that the Windows performance counters are enabled on the server.

5) Change the WMI counter timeout.

6) Resynchronize the WMI Performance Library.

7) Update the WMI Registry and register the WMI DLL files.

8) Salvage the WMI Repository.

9) Rebuild the list of WMI counters.

10) Assign additional memory to the WMI pool..

# Checking for WMI, Performance Class, and Class Instance Availability

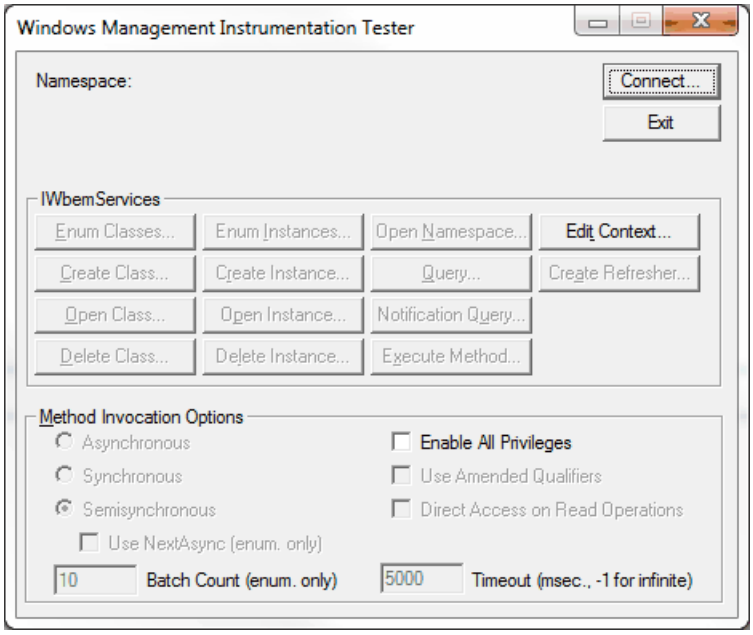
If WMI or the class or instance for a performance counter is not available, the Metalogix diagnostic Manager Collection Service cannot retrieve the counter data. If the Management Console does not include data for a performance counter, you can use WbemTest, the Microsoft Windows Management Instrumentation Tester utility, to determine if the counter is available.

You can use the WbemTest utility to test access to any WMI performance counter on a server that is part of the SharePoint farm. You can also use the utility to test access to a WMI performance counter remotely, from the computer that hosts the Collection Service. Since you can test access locally and remotely, you can use WbemTest to determine if network access issues are interfering with access to the WMI performance counters.

When you test access to a performance counter, you need to know the class name and the property name. (See Objects and Properties Accessed for lists of WMI properties and classes.)

### To use the WbemTest utility to test a machine's access to WMI:

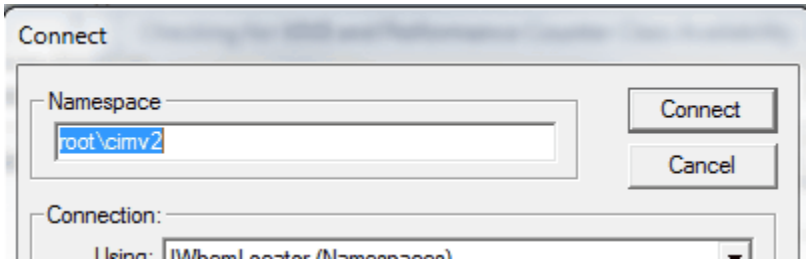
1 From the Windows Start menu Search box or a command prompt, type **WbemTest**.




2 Click **[Connect]**.

3 Make sure the Namespace field contains the appropriate value:

- If you started the utility locally - `root\cimv2`
- If you started the utility remotely - `\\<target server name>\root\cimv2`

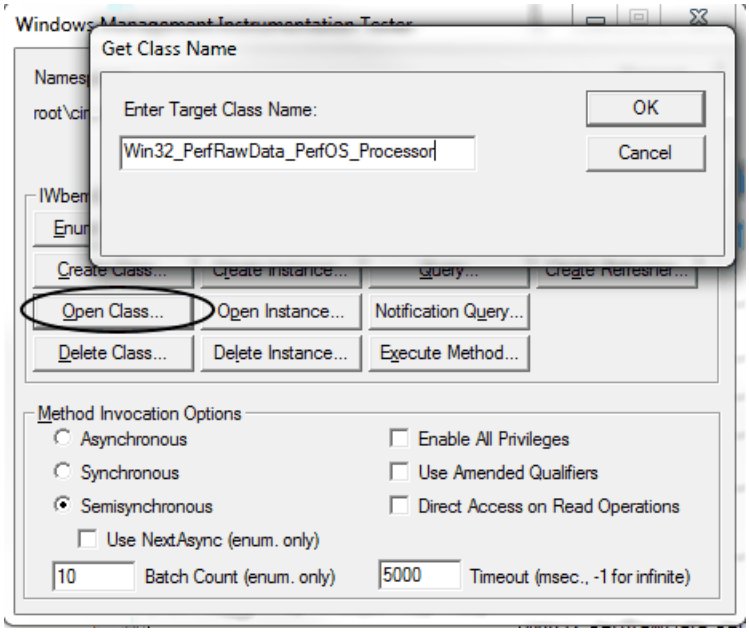



3 Click **[Connect]**.

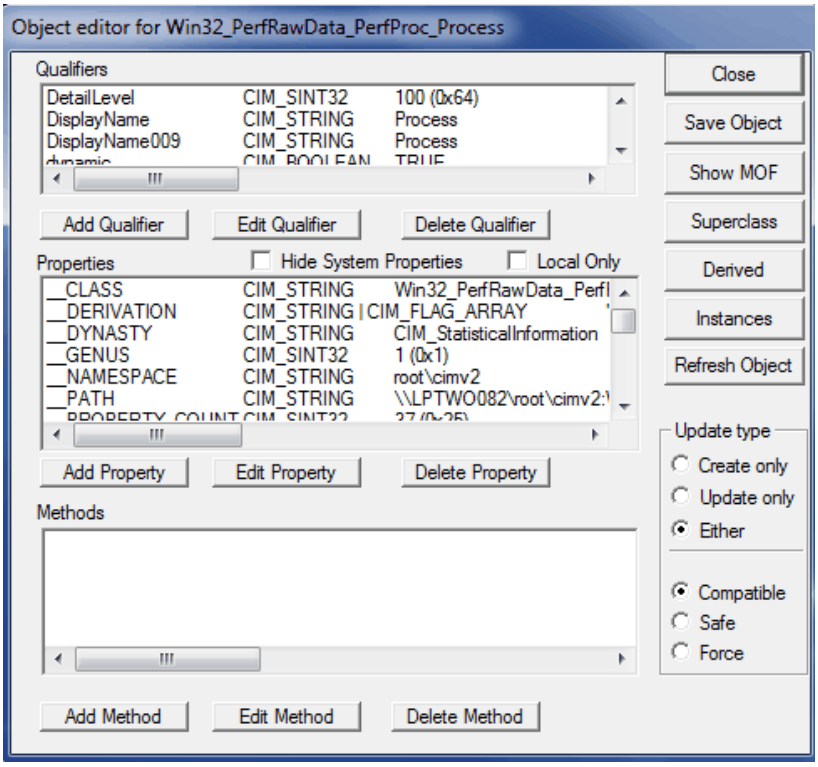
### To test access to a performance counter class:

1 Click **[Open Class]**.

2 In the enter Target Class Name field, enter the WMI Class name. See Objects and Properties Accessed for complete lists of WMI classes.



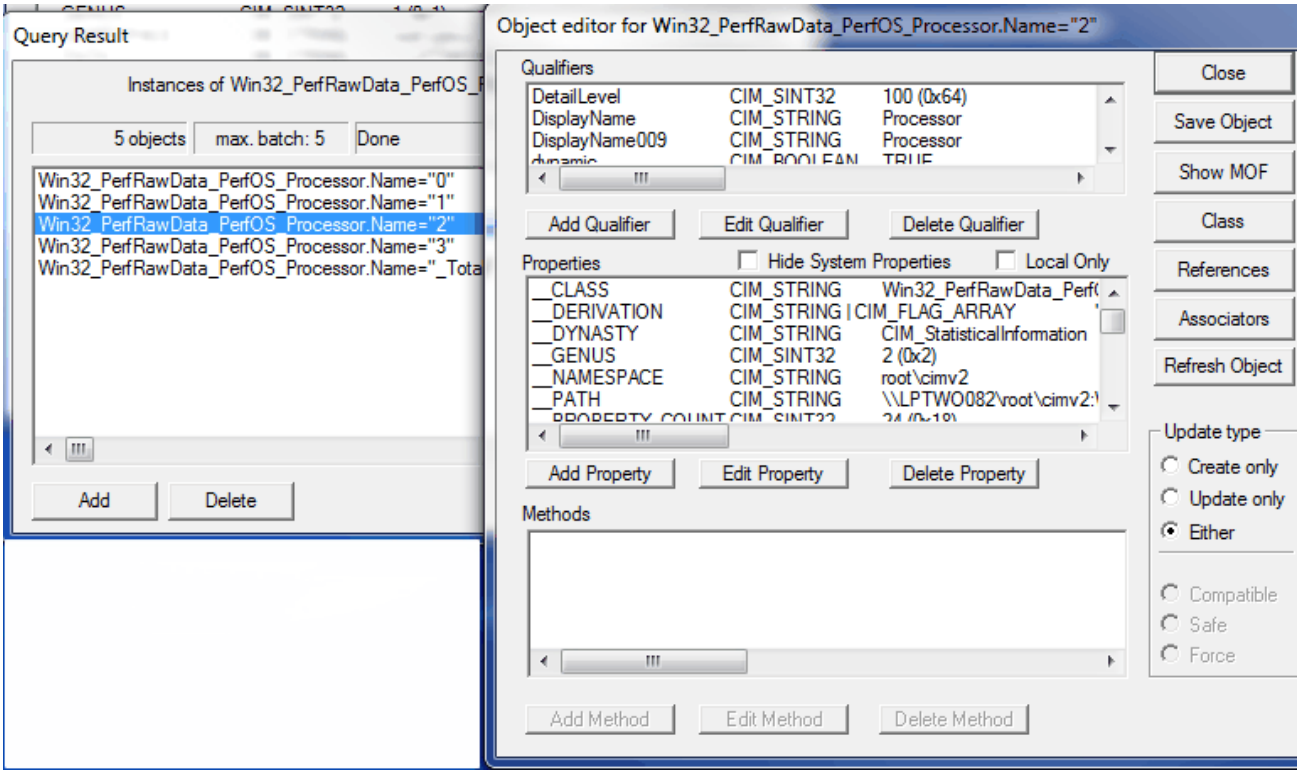

3 Click **[OK]** to open the Object Editor for the class.




**To test access to a performance counter instance for the class:**

In the Object Editor for the selected class, click **[Instances]**.

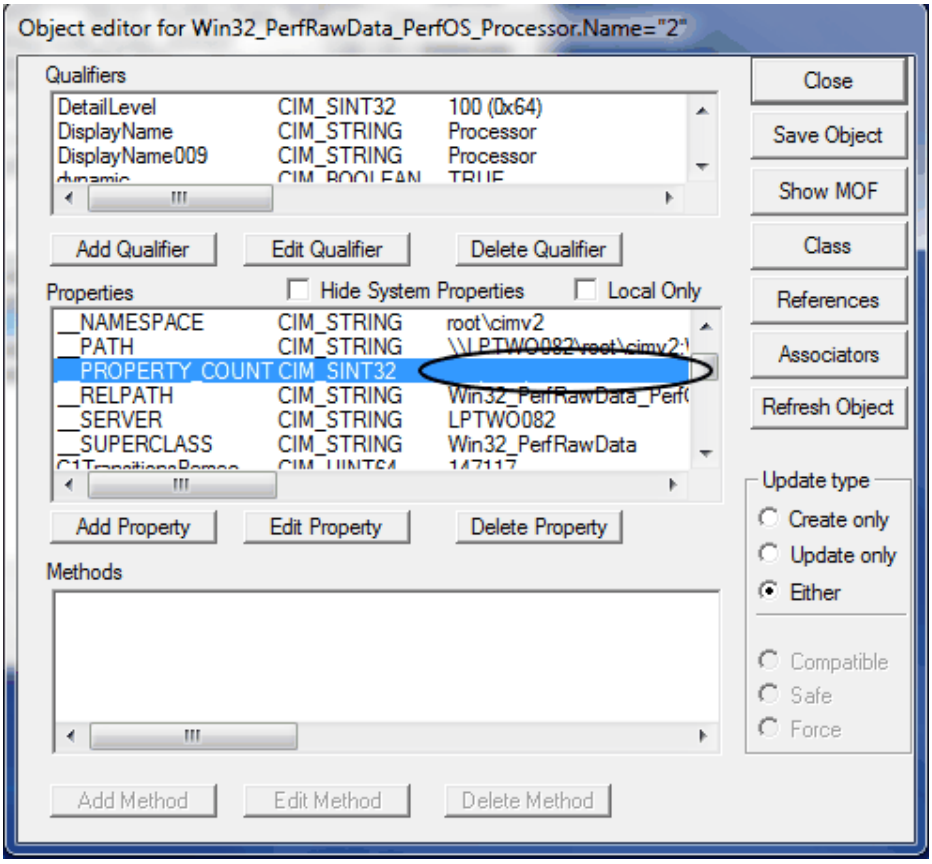
The Query Results dialog lists all of the instances for the class. Instance names indicate the element on the target computer that the represent.




**To view data for a single performance counter:**

In the object editor for the selected performance counter instance, select the property associated with the performance counter.

If no value displays in the third column, then the WMI data collection for the performance counter failed.




## Enabling a Disabled Performance Counter

If performance counters are disabled, the Diagnostic Manager Collection Service cannot collect WMI information from a server.

Performance counters may be disabled for one of the following reasons:

- An administrator may have disabled the counters in an attempt to maximize performance. However, performance counters do not have an appreciable effect on server performance.
- The Windows Perflib Performance Library can automatically disable the performance counters when it encounters a disabling error. If it does disable the counters, it records an event in the log and sets the value of the `DisablePerformanceCounters` registry key to 1.

**To enable the performance counter for a metric if it has been disabled:**

1 Open the Windows Registry Editor

2 Locate the following key:

`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\<service name>\Performance\Disable Performance Counters.`

3 Change the value of the registry key to 0.

## Changing the WMI Counter Timeout

If counters do not respond in a timely manner, the WMI call to retrieve the counter value fails. If this happens, the WMI system logs an error to the Windows Event Log. The event that appears in the log is:

```
WMI ADAP was unable to process the perfproc.dll performance library due to a
time violation in the open function.
```

If this error appears, you should modify the timeout for the counter, as described below.

**To change the WMI counter timeout:**

1 Open the Windows Registry Editor

2 Locate the following key:

`HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\<counterfile>\Performance`
.

3 Change the key to a higher value.

## Resynchronizing and Repairing the WMI Repository

Under certain circumstances, the WMI Repository can be corrupted. In addition, the WMI DLL files may be unregistered. If this happens, you can use the Windows command prompt to correct the issue. If this is unsuccessful, you can attempt to rebuild the Repository.

If necessary, you can rebuild the list of available WMI counters.

**To resynchronize the WMI Performance Library:**

1 From the Windows Control Panel, restart the Windows Management Instrumentation service.

2 Open a Windows command prompt.

3 To clear all WMI-related values from the registry, type `winmgmt/clearadap`, then press **Enter**.

4 To re-register all performance libraries with WMI, type `winmgt/resyncperf`, then press **Enter**.

5 To restart the WMI service:

a) Type `net stop winmgmt`, then press **Enter**.

b) Type `net start winmgmt`, then press **Enter**.

**To update the WMI Registry and register the WMI DLL files:**

1 At the Windows command prompt, type `cd %windir%\system32\wbem`, then press **Enter**.

2 Type `for /f %s in ('dir /b *.dll') do regsvr32 /s %s`, then press **Enter**.

3 Type `for /f %s in ('dir /b *.mof') do mofcomp %s`, then press **Enter**.

**To salvage the WMI Repository:**

At the Windows command prompt, type `Windmgmt /salvage repository`, then press **Enter**.

**To build the list of WMI counters:**

1 At the Windows command prompt, type `lodctr /r`, then press **Enter**.

2 Type `winmgmt /resyncperf`, then press **Enter**.

## Assigning Additional Memory to the WMI Pool

Under certain circumstances, WMI may encounter a memory limit and need additional memory assigned to its memory pool.

**To assign additional memory to the WMI pool:**

1 From the Windows Control Panel, restart the Windows Management Instrumentation service.

2 To re-register all performance libraries with WMI, open a Windows command prompt and type `winmgmt/resyncperf`, then press **Enter**.

3 Set the WMIHandlesPerHost value to 8192 (0x2000). For guidance, refer to http://blogs.technet.com/b/askperf/archive/2008/09/16/memory-and-handle-quotas-in-the-wmi-provider-serviceNew Link.aspx.

## Online Resources

The following online resources are available to help you troubleshoot WMI issues.

NOTE: Although many of these articles specifically reference Windows Server 2003, most of the information is relevant for Server 2008 and 2012 as well.

| Aid | URL |
|---|---|
| Microsoft WMI Diagnostic utility: | http://www.microsoft.com/en-us/download/details.aspx?id=7684 |
| Basic WMI Testing: | http://blogs.technet.com/b/askperf/archive/2007/06/22/basic-wmi-testing.aspx |
| Using WMIDiagnostic on Windows Server 2008: | http://blogs.technet.com/b/askperf/archive/2008/11/11/wmidiag-vbs-and-the-missing-wmi-files.aspx |

| Aid | URL |
| --- | --- |
| | http://blogs.technet.com/b/askperf/archive/2008/07/11/wmi-troubleshooting-the-repository-on-vista-server-2008.aspx |
| Microsoft Technet articles on WMI: | http://technet.microsoft.com/en-us/library/ee692942.aspx |
| WMI Secrets, troubleshooting, and tips: | http://technet.microsoft.com/en-us/library/ee692772.aspx<br>http://technet.microsoft.com/en-us/library/ff406382.aspx |

## Customer Portal

The following recommended articles, accessible via the Metalogix Customer Portal, provide additional information.

- What ports are used by Metalogix Diagnostic Manager?
- Alert for "Missing SharePoint Counters" received repeatedly for a newly added farm
- Metalogix Diagnostic Manager alert, "WMI Object Counter encountered an Invalid Class error in WMI query"