



TIBCO Hawk[®]

Microagent Reference

*Software Release 6.2
September 2019*



Important Information

SOME TIBCO SOFTWARE EMBEDS OR BUNDLES OTHER TIBCO SOFTWARE. USE OF SUCH EMBEDDED OR BUNDLED TIBCO SOFTWARE IS SOLELY TO ENABLE THE FUNCTIONALITY (OR PROVIDE LIMITED ADD-ON FUNCTIONALITY) OF THE LICENSED TIBCO SOFTWARE. THE EMBEDDED OR BUNDLED SOFTWARE IS NOT LICENSED TO BE USED OR ACCESSED BY ANY OTHER TIBCO SOFTWARE OR FOR ANY OTHER PURPOSE.

USE OF TIBCO SOFTWARE AND THIS DOCUMENT IS SUBJECT TO THE TERMS AND CONDITIONS OF A LICENSE AGREEMENT FOUND IN EITHER A SEPARATELY EXECUTED SOFTWARE LICENSE AGREEMENT, OR, IF THERE IS NO SUCH SEPARATE AGREEMENT, THE CLICKWRAP END USER LICENSE AGREEMENT WHICH IS DISPLAYED DURING DOWNLOAD OR INSTALLATION OF THE SOFTWARE (AND WHICH IS DUPLICATED IN THE LICENSE FILE) OR IF THERE IS NO SUCH SOFTWARE LICENSE AGREEMENT OR CLICKWRAP END USER LICENSE AGREEMENT, THE LICENSE(S) LOCATED IN THE "LICENSE" FILE(S) OF THE SOFTWARE. USE OF THIS DOCUMENT IS SUBJECT TO THOSE TERMS AND CONDITIONS, AND YOUR USE HEREOF SHALL CONSTITUTE ACCEPTANCE OF AND AN AGREEMENT TO BE BOUND BY THE SAME.

This document contains confidential information that is subject to U.S. and international copyright laws and treaties. No part of this document may be reproduced in any form without the written authorization of TIBCO Software Inc.

TIBCO, the TIBCO logo, the TIBCO O logo, TIB, Information Bus, ActiveMatrix BusinessWorks, Enterprise Message Service, Hawk, Rendezvous, TIBCO Administrator, TIBCO Designer, and TIBCO Runtime Agent are either registered trademarks or trademarks of TIBCO Software Inc. in the United States and/or other countries.

Java and all Java based trademarks and logos are trademarks or registered trademarks of Oracle Corporation and/or its affiliates.

All other product and company names and marks mentioned in this document are the property of their respective owners and are mentioned for identification purposes only.

THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS DOCUMENT COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THIS DOCUMENT. TIBCO SOFTWARE INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS DOCUMENT AT ANY TIME.

THE CONTENTS OF THIS DOCUMENT MAY BE MODIFIED AND/OR QUALIFIED, DIRECTLY OR INDIRECTLY, BY OTHER DOCUMENTATION WHICH ACCOMPANIES THIS SOFTWARE, INCLUDING BUT NOT LIMITED TO ANY RELEASE NOTES AND "READ ME" FILES.

This and other products of TIBCO Software Inc. may be covered by registered patents. Please refer to TIBCO's Virtual Patent Marking document (<https://www.tibco.com/patents>) for details.

Copyright © 1996-2019. TIBCO Software Inc. All Rights Reserved.

Contents

Preface	xi
Changes from the Previous Release of This Guide	xii
Related Documentation	xiii
TIBCO Hawk Documentation	xiii
Other TIBCO Product Documentation	xiv
Typographical Conventions	xv
TIBCO Product Documentation and Support Services	xvii
How to Access TIBCO Documentation	xvii
How to Contact TIBCO Support	xvii
How to Join TIBCO Community	xvii
Chapter 1 Overview of TIBCO Hawk Microagents and Methods	1
About Microagents	2
Microagent Types	2
Method Returns	2
Additional Information	2
Chapter 2 Common Methods	3
getReleaseVersion	4
getTraceLevel	5
setTraceLevel	6
getTraceParameters	7
setTraceParameters	8
_onUnsolicitedMsg	9
Chapter 3 Platform-Independent Microagents and Methods	11
Summary of Platform-Independent Microagents	12
COM.TIBCO.hawk.microagent.Self	16
Self:getMicroAgentInfo	18
Self:getAgentTimeZone	19
Self:getSecurityInfo	20
Self:getComponentInfo	21
Self:turnDiagnosticsOn	22
Self:turnDiagnosticsOff	23
Self:getUptime	24
Self:doAMIDiscovery	25

Self:onMicroAgentEvent	26
COM.TIBCO.hawk.microagent.SysInfo	27
SysInfo:getOperatingSystem	28
SysInfo:getArchitecture	29
SysInfo:getHostName	30
SysInfo:getNetworkAddress	31
COM.TIBCO.hawk.microagent.HawkEventService	32
HawkEventService:onAgentAlive	33
HawkEventService:onAgentExpired	34
HawkEventService:onAlert	35
HawkEventService:onClear	37
HawkEventService:onMicroAgentChange	38
HawkEventService:onRulebaseChange	39
COM.TIBCO.hawk.microagent.RuleBaseEngine	40
RuleBaseEngine:deleteRuleBase	43
RuleBaseEngine:getExternalVariables	44
RuleBaseEngine:getRuleBaseNames	45
RuleBaseEngine:getScheduleNames	46
RuleBaseEngine:loadRuleBase	47
RuleBaseEngine:loadRuleBaseFromFile	48
RuleBaseEngine:unloadRuleBase	49
RuleBaseEngine:getConfigInfo	50
RuleBaseEngine:sendMail	51
RuleBaseEngine:onAlertCount	52
RuleBaseEngine:onAlertCountForRulebase	53
RuleBaseEngine:suspendAlert	54
RuleBaseEngine:suspendAllAlerts	55
RuleBaseEngine:resumeSuspendedAlerts	56
RuleBaseEngine:updateExternalVariables	57
COM.TIBCO.hawk.microagent.Messaging	58
Messaging:sendMessage	59
Messaging:onMessage	60
Messaging:onTimeout	61
Messaging:onNumber	62
COM.TIBCO.hawk.microagent.Repository	64
Repository:getName	66
Repository:getScheduleNames	67
Repository:getRuleBaseName	68
Repository:onRepositoryEvent	69
COM.TIBCO.hawk.microagent.Logfile	70
Logfile:onNewLine	71
Logfile:onXMLElement	72
Logfile:onNewLinewithPatternFile	73

Logfile:reEvaluateLogFileNames	75
COM.TIBCO.hawk.microagent.Custom	76
Custom:executeForString	77
Custom:executeForNumber	78
Custom:execute	79
COM.TIBCO.hawk.hma.FileStat	80
FileStat:getFileStatus	81
FileStat:getFileCount	85
COM.TIBCO.hawk.hma.TibRendezvous	86
TibRendezvous:onRvDaemonStatus	87
TibRendezvous:onRvDataLoss	91
TibRendezvous:onRvdDisconnectOrConnect	92
TibRendezvous:onRvLicenseExpire	93
TibRendezvous:onRvdRetransmissionSuppressed	94
COM.TIBCO.hawk.microagent.tcpdaemon.TcpClusterStatus	95
TcpClusterStatus:getDaemonStatus	96
TcpClusterStatus:getClusterStatus	97
TcpClusterStatus:getMemberCount	98
TcpClusterStatus:onMemberEvent	99
COM.TIBCO.hawk.microagent.TCPMessaging	100
TcpMessaging:sendMessage	101
TcpMessaging:onMessage	102
TcpMessaging:onNumber	103
TcpMessaging:onTimeout	104
Chapter 4 Microsoft Windows Microagents and Methods	105
Summary of Microsoft Windows Microagents	106
COM.TIBCO.hawk.hma.Performance	110
Performance:System	112
Performance:Processor	114
Performance:Memory	115
Performance:Cache	117
Performance:PhysicalDisk	119
Performance:LogicalDisk	121
Performance:Process	123
Performance:ProcessCount	125
Performance:Thread	126
Performance:Objects	127
Performance:Redirector	128
Performance:Server	130
Performance:Server Work Queues	132
Performance:Paging File	134
Performance:Browser	135

Performance:Telephony	137
Performance:NBT Connection	138
Performance:Network Interface	139
Performance:IP	141
Performance:ICMP	142
Performance:TCP	144
Performance:UDP	145
Performance:restart	146
COM.TIBCO.hawk.hma.Eventlog	147
Eventlog:getRecentApplicationEvents	148
Eventlog:getRecentSystemEvents	150
Eventlog:getRecentSecurityEvents	152
Eventlog:onApplicationEvent	154
Eventlog:onSecurityEvent	156
Eventlog:onSystemEvent	158
Eventlog:logEvent	160
COM.TIBCO.hawk.hma.Registry	161
Registry:setDWORD	162
Registry:setString	163
Registry:getDWORD	164
Registry:getString	165
Registry:getMultiString	166
Registry:setExpandString	167
Registry:enumerateKey	168
Registry:CreateKey	169
Registry:setQWORD	170
Registry:getQWORD	171
COM.TIBCO.hawk.hma.Services	172
Services:getServiceConfiguration	173
Services:getServiceStatus	176
Services:setServiceStartType	178
Services:startService	179
Services:stopService	180
Services:pauseService	181
Services:continueService	182
Services:controlService	183
COM.TIBCO.hawk.hma.Process	184
Process:getInstanceCount	185
Process:getInstanceCountByCommand	186
Process:getProcess	187
COM.TIBCO.hawk.hma.Network	189
Network:getConfig	190
COM.TIBCO.hawk.hma.System	192

System:getSystemInfo	193
System:getCpuInfo	194
Chapter 5 Solaris-SPARC Microagents and Methods	195
Summary of Solaris-SPARC Microagents	196
COM.TIBCO.hawk.hma.System	198
System:getSystemInfo	199
System:getCpuInfo	201
System:getSwapInfo	202
System:getTunableInfo	203
COM.TIBCO.hawk.hma.Process	204
Process:getProcess	205
Process:getInstanceCount	206
Process:getInstanceCountByCommand	207
COM.TIBCO.hawk.hma.FileSystem	208
FileSystem:getByPartition	209
FileSystem:getByMountPoint	210
FileSystem:getByFileSystem	212
COM.TIBCO.hawk.hma.Network	214
Network:getStatistics	215
Network:getConfig	216
Chapter 6 Solaris-10 Microagents and Methods	217
Summary of Solaris-10 Microagents	218
COM.TIBCO.hawk.hma.System	220
System:getSystemInfo	221
System:getCpuInfo	223
System:getSwapInfo	224
System:getTunableInfo	225
COM.TIBCO.hawk.hma.Process	226
Process:getProcess	227
Process:getInstanceCount	228
Process:getInstanceCountByCommand	229
COM.TIBCO.hawk.hma.FileSystem	230
FileSystem:getByPartition	231
FileSystem:getByMountPoint	232
FileSystem:getByFileSystem	233
COM.TIBCO.hawk.hma.Network	235
Network:getStatistics	236
Network:getConfig	238

Chapter 7 HP-UX Microagents and Methods	239
Summary of HP-UX Microagents	240
COM.TIBCO.hawk.hma.System	242
System:getSystemInfo	243
System:getCpuInfo	245
System:getSwapInfo	246
System:getTunableInfo	247
COM.TIBCO.hawk.hma.Process	248
Process:getProcess	249
Process:getInstanceCount	250
Process:getInstanceCountByCommand	251
COM.TIBCO.hawk.hma.FileSystem	252
FileSystem:getByPartition	253
FileSystem:getByMountPoint	255
FileSystem:getByFileSystem	257
COM.TIBCO.hawk.hma.Network	258
Network:getConfig	259
Network:getStatistics	260
Chapter 8 AIX Microagents and Methods	263
Summary of AIX Microagents	264
COM.TIBCO.hawk.hma.System	266
System:getCpuInfo	267
System:getSwapInfo	269
System:getSystemInfo	270
COM.TIBCO.hawk.hma.Process	271
Process:getProcess	272
Process:getInstanceCount	273
Process:getInstanceCountByCommand	274
COM.TIBCO.hawk.hma.FileSystem	275
FileSystem:getByPartition	276
FileSystem:getByMountPoint	277
FileSystem:getByFileSystem	278
COM.TIBCO.hawk.hma.Network	279
Network:getStatistics	280
Network:getConfig	282
Chapter 9 Linux Microagents and Methods	283
Summary of Linux Microagents	284
COM.TIBCO.hawk.hma.System	286
System:getSwapInfo	287

System:getSystemInfo	288
System:getCpuInfo	293
System:getTunableInfo	294
COM.TIBCO.hawk.hma.Process	295
Process:getProcess	296
Process:getInstanceCount	301
Process:getInstanceCountByCommand	302
COM.TIBCO.hawk.hma.FileSystem	303
FileSystem:getByPartition	304
FileSystem:getByMountPoint	305
FileSystem:getByFileSystem	306
COM.TIBCO.hawk.hma.Network	308
Network:getStatistics	309
Network:getConfig	310
Chapter 10 Mac OS X Methods	313
Summary of Mac OS X Microagents	314
COM.TIBCO.hawk.hma.System	316
System:getSystemInfo	317
System:getCpuInfo	318
System:getProcessorSetInfo	319
System:getSwapInfo	320
System:getTunableInfo	321
COM.TIBCO.hawk.hma.Process	322
Process:getProcess	323
Process:getInstanceCount	325
Process:getInstanceCountByCommand	326
COM.TIBCO.hawk.hma.FileSystem	327
FileSystem:getByPartition	328
FileSystem:getByMountPoint	329
FileSystem:getByFileSystem	330
COM.TIBCO.hawk.hma.Network	331
Network:getStatistics	332
Network:getConfig	333
Index	335

Preface

This manual covers TIBCO Hawk[®] microagents and their methods. TIBCO Hawk microagents are used to customize system monitoring by TIBCO Hawk agent. This manual is intended as a reference for users or administrators who are configuring TIBCO Hawk agent.

This manual assumes you are familiar with TIBCO Rendezvous architecture and the concepts of system monitoring.

Topics

- [Changes from the Previous Release of This Guide, page xii](#)
- [Related Documentation, page xiii](#)
- [Typographical Conventions, page xv](#)
- [TIBCO Product Documentation and Support Services, page xvii](#)

Changes from the Previous Release of This Guide

- Renamed the *TCP Daemon* component to *Cluster Manager*.

Related Documentation

This section lists documentation resources you may find useful.

TIBCO Hawk Documentation

The following documents form the TIBCO Hawk documentation set:

- *TIBCO Hawk Release Notes*: Read the release notes for a list of new and changed features. This document also contains lists of known issues and closed issues for this release.
- *TIBCO Hawk Concepts*: This manual includes basic descriptions of TIBCO Hawk concepts.
- *TIBCO Hawk Installation, Configuration, and Administration*: Read this book first. It contains step-by-step instructions for installing TIBCO Hawk software on various operating system platforms. It also describes how to configure the software for specific applications, once it is installed. An installation FAQ is included.
- *TIBCO Hawk Microagent Reference*: A reference to the microagents and methods used by a TIBCO Hawk Agent for system and application monitoring.
- *TIBCO Hawk WebConsole User's Guide*: This manual includes complete instructions for using TIBCO Hawk WebConsole.
- *TIBCO Hawk Programmer's Guide*: All programmers should read this manual. It contains detailed descriptions of Application Management Interface (AMI), Application Programming Interface (API) concepts, and the TIBCO Hawk security framework and its classes. It also contains detailed descriptions of each class and method for the following APIs:
 - AMI API
 - Java, C++ and C API
 - Console API
 - Java API
 - Configuration Object API
 - Java API

Programmers should refer to the appropriate language reference sections for the AMI API details. The TIBCO Hawk Application Management Interface (AMI) exposes internal application methods to TIBCO Hawk.

- *TIBCO Hawk Plug-in Reference Guide*: Contains details about the Enterprise Message Service, Messaging and JVM microagents methods that are used to administer and monitor the TIBCO Enterprise Message Service server.
- *TIBCO Hawk Plug-ins for TIBCO Administrator*: Contains detailed descriptions of the TIBCO Hawk plug-ins accessed via TIBCO Administrator.
- *TIBCO Hawk HTTP Adapter User's Guide*: Contains information about performing discovery, monitoring of agent status, monitoring of agent alerts, method invocation, method subscription, and many more activities on TIBCO Hawk and third-party products.
- *TIBCO Hawk Admin Agent Guide*: Contains basic configuration details for TIBCO Hawk Admin Agent and complete instructions for using the web interface of TIBCO Enterprise Administrator for TIBCO Hawk.
- *TIBCO Hawk Security Guide*: Provides guidelines to ensure security within the components of TIBCO Hawk and within the communication channels between the components.

Other TIBCO Product Documentation

You may find it useful to read the documentation for the following TIBCO products:

- TIBCO® Enterprise Administrator
- TIBCO ActiveSpaces®
- TIBCO Rendezvous®
- TIBCO Enterprise Message Service™

Typographical Conventions

The following typographical conventions are used in this manual.

Table 1 General Typographical Conventions

Convention	Use
<i>ENV_HOME</i>	<p>TIBCO products are installed into an installation environment. A product installed into an installation environment does not access components in other installation environments. Incompatible products and multiple instances of the same product must be installed into different installation environments.</p> <p>An installation environment consists of the following properties:</p> <ul style="list-style-type: none"> • Name Identifies the installation environment. This name is referenced in documentation as <i>ENV_NAME</i>. On Microsoft Windows, the name is appended to the name of Windows services created by the installer and is a component of the path to the product shortcut in the Windows Start > All Programs menu. • Path The folder into which the product is installed. This folder is referenced in documentation as <i>TIBCO_HOME</i>. <p>TIBCO Hawk installs into a directory within a <i>TIBCO_HOME</i>. This directory is referenced in documentation as <i>HAWK_HOME</i>. The default value of <i>HAWK_HOME</i> depends on the operating system. For example on Windows systems, the default value is <code>C:\tibco\hawk\6.0</code>.</p> <p>A TIBCO Hawk configuration folder stores configuration data generated by TIBCO Hawk. Configuration data can include sample scripts, session data, configured binaries, logs, and so on. This folder is referenced in documentation as <i>CONFIG_FOLDER</i>. For example, on Windows systems, the default value is <code>C:\ProgramData\tibco\cfgmgt\hawk</code>.</p>
<i>TIBCO_HOME</i>	
<i>HAWK_HOME</i>	
<i>CONFIG_FOLDER</i>	
code font	<p>Code font identifies commands, code examples, filenames, pathnames, and output displayed in a command window. For example:</p> <p>Use <code>MyCommand</code> to start the foo process.</p>

Table 1 General Typographical Conventions (Cont'd)

Convention	Use
bold code font	<p>Bold code font is used in the following ways:</p> <ul style="list-style-type: none"> • In procedures, to indicate what a user types. For example: Type admin. • In large code samples, to indicate the parts of the sample that are of particular interest. • In command syntax, to indicate the default parameter for a command. For example, if no parameter is specified, MyCommand is enabled: MyCommand [enable disable]
<i>italic font</i>	<p>Italic font is used in the following ways:</p> <ul style="list-style-type: none"> • To indicate a document title. For example: See <i>TIBCO BusinessWorks Concepts</i>. • To introduce new terms. For example: A portal page may contain several portlets. <i>Portlets</i> are mini-applications that run in a portal. • To indicate a variable in a command or code syntax that you must replace. For example: MyCommand <i>pathname</i>
Key combinations	<p>Key name separated by a plus sign indicate keys pressed simultaneously. For example: Ctrl+C.</p> <p>Key names separated by a comma and space indicate keys pressed one after the other. For example: Esc, Ctrl+Q.</p>
	The note icon indicates information that is of special interest or importance, for example, an additional action required only in certain circumstances.
	The tip icon indicates an idea that could be useful, for example, a way to apply the information provided in the current section to achieve a specific result.
	The warning icon indicates the potential for a damaging situation, for example, data loss or corruption if certain steps are taken or not taken.

TIBCO Product Documentation and Support Services

For information about this product, you can read the documentation, contact TIBCO Support, and join TIBCO Community.

How to Access TIBCO Documentation

Documentation for TIBCO products is available on the TIBCO Product Documentation website mainly in the HTML and PDF formats.

The TIBCO Product Documentation website is updated frequently and is more current than any other documentation included with the product. To access the latest documentation, visit <https://docs.tibco.com>.

Documentation for TIBCO Hawk is available on the [TIBCO Hawk Product Documentation](#) page.

How to Contact TIBCO Support

You can contact TIBCO Support in the following ways:

- For an overview of TIBCO Support, visit <https://www.tibco.com/services/support>.
- For accessing the Support Knowledge Base, viewing the latest product updates that were not available at the time of the release, and getting personalized content about products you are interested in, visit the TIBCO Support portal at <https://support.tibco.com>.
- For creating a Support case, you must have a valid maintenance or support contract with TIBCO. You also need a user name and password to log in to <https://support.tibco.com>. If you do not have a user name, you can request one by clicking **Register** on the website.

How to Join TIBCO Community

TIBCO Community is the official channel for TIBCO customers, partners, and employee subject matter experts to share and access their collective experience. TIBCO Community offers access to Q&A forums, product wikis, and best practices. It also offers access to extensions, adapters, solution accelerators, and tools that extend and enable customers to gain full value from TIBCO products. In addition, users can submit and vote on feature requests from within the [TIBCO Ideas Portal](#). For a free registration, go to <https://community.tibco.com>.

Chapter 1

Overview of TIBCO Hawk Microagents and Methods

This chapter describes microagents and gives information about their uses and functions.

Topics

- [About Microagents, page 2](#)
- [Microagent Types, page 2](#)
- [Method Returns, page 2](#)
- [Additional Information, page 2](#)

About Microagents

The TIBCO Hawk agent uses processes known as Hawk microagents to collect information and operate using that information. These microagents execute specific tasks known as methods. Methods collect information and perform tasks. Methods can be called directly or from within a TIBCO Hawk rulebase.

This manual describes how various Hawk microagents and respective methods can be used for system and application monitoring. It also provides a complete list of all the available methods for the respective microagents. You can use this guide as a reference to know all the available methods for any microagent.

Microagent Types

TIBCO Hawk microagents are classified by the functions they perform and can be one of three types:

- `IMPACT_INFO` returns information.
- `IMPACT_ACTION` performs an action on the TIBCO Hawk system.
- `IMPACT_ACTION_INFO` both returns information and takes an action based on it.

Method Returns

Methods can return information by invocation or subscription. Invocation returns information on a one-time basis. Subscription information is returned either synchronously, on a regular time schedule, or asynchronously, when data becomes available.

Method invocation returns a table of results, according to specified arguments (if any).

Additional Information

Help text for TIBCO Hawk microagent methods is also available in the microagent window in the TIBCO Hawk Display or Microagent Information in Hawk WebConsole. Selecting a microagent and method in this window will bring up help text in the window.

Chapter 2 **Common Methods**

Some TIBCO Hawk microagent methods are common across all platform-dependent as well as platform-independent microagents on all platforms. These common methods performs release and trace-related functions.

Topics

- [getReleaseVersion](#), page 4
- [getTraceLevel](#), page 5
- [setTraceLevel](#), page 6
- [getTraceParameters](#), page 7
- [setTraceParameters](#), page 8
- [_onUnsolicitedMsg](#), page 9

getReleaseVersion

Method

Purpose This method (on all platforms) returns version information about the current release of this microagent. The version is comprised of a major, minor, and update number separated by periods (for example, 3.1.1). Both a string representation of the version (`Version`) and an integer representation of the major, minor, and update components of the version (`Major`, `Minor`, and `Update`, respectively) is returned.

Type Synchronous, `INFO`.

Arguments None.

Returns

Name	Type	Description
Name	String	Name of the binary
Version	String	Version number
Date	String	Version date
Major	Integer	Release version major number
Minor	Integer	Release version minor number
Update	Integer	Release version update number

getTraceLevel

Method

Purpose This method (on all platforms) returns the trace level settings.

Remarks Invocation will return a table, with the setting for each trace level (Information, Warning, Error, Debug, or AMI) listed as being either on or off.

Type Synchronous, INFO.

Arguments None.

Returns

Name	Type	Description
Information	String	Information trace setting, on or off value
Warning	String	Warning trace setting, on or off value
Error	String	Error trace setting, on or off value
Debug	String	Debug trace setting, on or off value
AMI	String	AMI debug trace setting, on or off value

setTraceLevel

Method

Purpose This method (on all platforms) sets the trace level bits.

Remarks Each trace level can be set to On (enabled), Off (disabled) or Current (leave setting as is). Invocation will return a table, with the setting for each trace level (Information, Warning, Error, Debug, or AMI) listed as being either On or Off.

Type Synchronous, ACTION_INFO.

Arguments

Name	Type	Description
Information	String	Information trace setting, on or off value
Warning	String	Warning trace setting, on or off value
Error	String	Error trace setting, on or off value
Debug	String	Debug trace setting, on or off value
AMI	String	AMI debug trace setting, on or off value

Returns

Name	Type	Description
Information	String	Current information trace setting, on or off value
Warning	String	Current warning trace setting, on or off value
Error	String	Current error trace setting, on or off value
Debug	String	Current debug trace setting, on or off value
AMI	String	Current AMI debug trace setting, on or off value

getTraceParameters

Method

Purpose This method (on all platforms) returns tracing parameter information.

Type Synchronous, INFO.

Arguments None.

Returns

Name	Type	Description
Directory	String	Current trace file directory
File Name	String	Current trace file name
Max File Size	Integer	Current trace file maximum size (in kilobytes) before rollover
Max Trace File	Integer	Current maximum number of trace files to keep in the trace directory

setTraceParameters

Method

Purpose This method (on all platforms) sets tracing parameters.

Remarks: Use string `current` for keeping the current setting for string parameters and use integer value 0 for keeping the current setting for integer parameters.

Type Synchronous, ACTION_INFO.

Arguments

Name	Type	Description
Directory	String	Trace file directory where the trace log file should be kept.
File Name	String	Trace file name
Max File Size	Integer	Maximum size (in kilobytes) of trace file before rollover (kilobytes)
Max Trace File	Integer	Maximum number of trace files to keep in trace directory

Returns

Name	Type	Description
Directory	String	Current trace file directory
File Name	String	Current trace file name
Max File Size	Integer	Current trace file maximum size (in kilobytes) before rollover
Max Trace File	Integer	Current maximum number of trace files to keep in the trace directory

_onUnsolicitedMsg

Method

Purpose Subscribing to this method (on all platforms) returns any unsolicited notifications that are sent from this microagent. Invoking this method returns the last such message that was received (if any).

Remarks An unsolicited message is an information, warning, or error message that is sent directly to the manager.

Type Asynchronous, INFO.

Arguments None.

Returns

Name	Type	Description
TYPE	String	One of INFO, WARNING, or ERROR.
TEXT	String	The text of the message
ID	Integer	A number used by the application to identify the message
INBOX	String	An inbox address used to send the message

Platform-Independent Microagents and Methods

The TIBCO Hawk microagents described in this section are available on all platforms.

Topics

- [Summary of Platform-Independent Microagents](#), page 12
- [COM.TIBCO.hawk.microagent.Self](#), page 16
- [COM.TIBCO.hawk.microagent.SysInfo](#), page 27
- [COM.TIBCO.hawk.microagent.HawkEventService](#), page 32
- [COM.TIBCO.hawk.microagent.RuleBaseEngine](#), page 40
- [COM.TIBCO.hawk.microagent.Messaging](#), page 58
- [COM.TIBCO.hawk.microagent.Repository](#), page 64
- [COM.TIBCO.hawk.microagent.Logfile](#), page 70
- [COM.TIBCO.hawk.microagent.Custom](#), page 76
- [COM.TIBCO.hawk.hma.FileStat](#), page 80
- [COM.TIBCO.hawk.hma.TibRendezvous](#), page 86
- [COM.TIBCO.hawk.microagent.tcpdaemon.TcpClusterStatus](#), page 95
- [COM.TIBCO.hawk.microagent.TCPMessaging](#), page 100

Summary of Platform-Independent Microagents

The platform-independent microagents are as follows.

Table 2 *Platform-Independent Microagents*

Microagent	Description
Common Methods	<p>These methods are common to all microagents and provide release and trace-related functions. Methods are:</p> <ul style="list-style-type: none"> • getReleaseVersion • getTraceLevel • setTraceLevel • getTraceParameters • setTraceParameters • _onUnsolicitedMsg
Self	<p>Returns information from and takes actions that affect the TIBCO Hawk agent itself. Methods are:</p> <ul style="list-style-type: none"> • Self:getMicroAgentInfo • Self:getAgentTimeZone • Self:getComponentInfo • Self:turnDiagnosticsOn • Self:turnDiagnosticsOff
SysInfo	<p>Returns information on a network host: its name, address, type of computer and operating system. Methods are:</p> <ul style="list-style-type: none"> • SysInfo:getOperatingSystem • SysInfo:getArchitecture • SysInfo:getHostName • SysInfo:getNetworkAddress

Table 2 Platform-Independent Microagents (Cont'd)

Microagent	Description
HawkEventService	<p>This application reports on events generated by the TIBCO Hawk Agents across the network.</p> <ul style="list-style-type: none"> • HawkEventService:onAgentAlive • HawkEventService:onAgentExpired • HawkEventService:onAlert • HawkEventService:onClear • HawkEventService:onMicroAgentChange • HawkEventService:onRulebaseChange
RuleBaseEngine	<p>Returns information from rulebases and takes actions on rulebases that affect the TIBCO Hawk agent itself. The methods are:</p> <ul style="list-style-type: none"> • RuleBaseEngine:deleteRuleBase • RuleBaseEngine:getExternalVariables • RuleBaseEngine:getRuleBaseNames • RuleBaseEngine:getScheduleNames • RuleBaseEngine:loadRuleBase • RuleBaseEngine:loadRuleBaseFromFile • RuleBaseEngine:unloadRuleBase • RuleBaseEngine:getConfigInfo • RuleBaseEngine:sendMail • RuleBaseEngine:onAlertCount • RuleBaseEngine:onAlertCountForRulebase • RuleBaseEngine:suspendAlert • RuleBaseEngine:suspendAllAlerts • RuleBaseEngine:resumeSuspendedAlerts • RuleBaseEngine:updateExternalVariables

Table 2 Platform-Independent Microagents (Cont'd)

Microagent	Description
Messaging	<p>Responds to requests from TIBCO Hawk agents to send and receive messages using either TIBCO Rendezvous or TIBCO EMS. Methods are:</p> <ul style="list-style-type: none"> • Messaging:sendMessage • Messaging:onMessage • Messaging:onTimeout • Messaging:onNumber
Repository	<p>The Messaging microagent (on all platforms) responds to requests from TIBCO Hawk agents to send and receive messages using either TIBCO Rendezvous or TIBCO EMS.</p> <p>Responds to configuration requests from TIBCO Hawk agents. Methods are:</p> <ul style="list-style-type: none"> • Repository:getName • Repository:getScheduleNames • Repository:getRuleBaseName • Repository:onRepositoryEvent
Logfile	<p>Returns new lines added to a log file. Methods are:</p> <ul style="list-style-type: none"> • Logfile:onNewLine
Custom	<p>Runs executable programs and scripts from within the TIBCO Hawk system. Methods are:</p> <ul style="list-style-type: none"> • Custom:executeForString • Custom:executeForNumber • Custom:execute
FileStat	<p>Returns information on files. Methods are:</p> <ul style="list-style-type: none"> • FileStat:getFileStatus • FileStat:getFileCount

Table 2 Platform-Independent Microagents (Cont'd)

Microagent	Description
TibRendezvous	<p>Subscribes to TIBCO Rendezvous system messages. Methods are:</p> <ul style="list-style-type: none"> • TibRendezvous:onRvDaemonStatus • TibRendezvous:onRvDataLoss • TibRendezvous:onRvdDisconnectOrConnect • TibRendezvous:onRvLicenseExpire
TcpClusterStatus	<p>The <code>TcpClusterStatus</code> microagent provides methods to monitor the health of the TIBCO Hawk TCP Transport cluster and its daemons. Methods are:</p> <ul style="list-style-type: none"> • TcpClusterStatus:getDaemonStatus • TcpClusterStatus:getClusterStatus • TcpClusterStatus:getMemberCount • TcpClusterStatus:onMemberEvent
TcpMessaging	<p>The <code>TcpMessaging</code> microagent provides methods to send and receive messages by using the TCP Transport for TIBCO Hawk. Methods are:</p> <ul style="list-style-type: none"> • TcpMessaging:sendMessage • TcpMessaging:onMessage • TcpMessaging:onNumber • TcpMessaging:onTimeout

COM.TIBCO.hawk.microagent.Self

Microagent

Purpose The `Self` microagent (on all platforms) returns information from and takes actions that affect the TIBCO Hawk agent itself. You use its methods to examine release version and auto-configuration information, to examine active microagents, and to start and stop diagnostic tools you might need when speaking to technical support personnel.

Remarks Because the `Self:turnDiagnosticsOn` and `Self:turnDiagnosticsOff` methods can affect the agent's performance, you should only use these methods when asked to do so by TIBCO Hawk technical support.

Method	Description	Page
Self:getMicroAgentInfo	Returns information about the microagents currently active	18
Self:getAgentTimeZone	Returns information about the agent time zone	19
Self:getSecurityInfo	Returns information about the current security policy	20
Self:getComponentInfo	Returns version information about components in the current release	21
Self:turnDiagnosticsOn	Turns on agent diagnostics	22
Self:turnDiagnosticsOff	Turns off agent diagnostics	23
Self:getUptime	Returns how long the agent has been running	24
Self:doAMIDiscovery	Re-discovers all AMI instrumented applications	25
Self:onMicroAgentEvent	Reports the change in the status of the microagent	26

Proprietary Methods The following method of the `Self` microagent is proprietary. The TIBCO Hawk system does not support direct user invocation of this method from the TIBCO Hawk Display or Console API.

However, this method is used indirectly by the TIBCO Hawk Display when interacting with agents and thus has security implications.

Method Name	Impact	Description
describe	Info	This method retrieves microagent descriptors

Self:getMicroAgentInfo

Method

Purpose This method (on all platforms) returns information about the microagents currently active on a TIBCO Hawk agent.

Remarks Since applications instrumented with the TIBCO Hawk Application Management Interface (AMI) appear as microagents, this method can also gather information on AMI applications.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Name	String	The microagent name. If no name is specified, the method returns information on all microagents.

Returns

Name	Type	Description
Name	String	The microagent name. Table is indexed on the Name field
Display Name	String	The microagent display name
Count	Integer	The number of microagent instances with that name on the host
Help	String	A brief description of the microagent. Retrieved from the microagent

Self:getAgentTimeZone

Method

Purpose This method returns the Agent Time Zone information.

Type Open, Synchronous, IMPACT_INFO

Arguments None

Returns

Name	Type	Description
TimeZone ID	String	The time zone ID of the agent
TimeZone Name	String	The name of the agent's time zone suitable for presentation to the user in the default locale
TimeZone offset	Integer	The amount of time (considering daylight saving changes) in milliseconds to add to UTC to get standard time in the current time zone
TimeZone String	String	The textual representation of the current time zone

Self:getSecurityInfo

Method

Purpose This method (on all platforms) returns information about the security policy currently in effect.

Type Synchronous, `IMPACT_INFO`.

Arguments None.

Returns

Parameter	Type	Description
Policy Class	string	Name of security policy class
Description	string	Text description of security policy class

Self:getComponentInfo

Method

Purpose This method (on all platforms) returns version information about the components in the current release.

Remarks This is the same information displayed when you choose **Help > About TIBCO Hawk Components** from the TIBCO Hawk Display.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Component	String	The component name. If no component is specified, returns information for all components.

Returns

Name	Type	Description
Name	String	The component name. Table is indexed on the Name field.
Version	String	The component version
Date	String	The component date

Self:turnDiagnosticsOn

Method

Purpose This method (on all platforms) turns on agent diagnostics.

Remarks The `diagnostics` microagent diagnoses problems that appear in the TIBCO Hawk system. Because the `Self:turnDiagnosticsOn` and `Self:turnDiagnosticsOff` methods can affect the agent's performance, you should only use these methods when asked to do so by TIBCO Hawk technical support.

Diagnostic microagent methods are not described in this documentation set, but could be created as an application. For more information on creating applications, refer to the TIBCO Hawk programming manuals.

Type Synchronous, `IMPACT_ACTION`.

Arguments None.

Returns None.

Self:turnDiagnosticsOff

Method

Purpose	This method (on all platforms) turns off agent diagnostics.
Remarks	The <code>diagnostics</code> microagent is used in diagnosis of problems that appear in the TIBCO Hawk system. <code>Diagnostics</code> microagent methods are not described in this documentation set.
Type	Synchronous, <code>IMPACT_ACTION</code> .
Arguments	None.
Returns	None

Self:getUptime

Method

Purpose This method (on all platforms) returns how long the agent has been running.

Type Synchronous, `IMPACT_INFO`.

Arguments None.

Returns

Parameter	Type	Description
Uptime	String	Total days, hours, and minutes the agent has been running.
Total days	String	Total number of days the agent has been running.
Total hours	String	Total number of hours the agent has been running.
Total millisec	Integer	Total number of milliseconds the agent has been running.

Self:doAMIDiscovery

Method

Purpose	This method (on all platforms) re-discovers all AMI instrumented applications.
Remarks	<p>When TIBCO Hawk agent loses contact with an AMI instrumented application due to a network failure, the microagent corresponding to the application will be removed after 90 seconds and will not be re-discovered after network connection is restored. This method when invoked re-discovers all AMI instrumented applications. After network failure, AMI applications will not get rediscovered if using an earlier AMI implementation (prior to TIBCO Hawk 4.5) or if using a custom AMI implementation that does not provide rediscovery.</p> <p>It is recommended that this method is not invoked frequently, as this will lead to the TIBCO Hawk agent receiving a number of AMI application discovery messages thereby increasing local traffic.</p>
Type	Synchronous, <code>IMPACT_ACTION</code> .
Arguments	None.
Returns	None

Self:onMicroAgentEvent

Method

Purpose This method (on all platforms) reports the change in the status of the microagent.

Remarks If the TIBCO Hawk Agent is unable to communicate with the microagent, the status of the microagent is immediately changed to `Unreachable`. After 90 seconds of being inaccessible, the microagent status is changed to `Removed`, after which the TIBCO Hawk agent actually removes the microagent. The status of a newly discovered microagent is `Added`.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Parameter	Type	Description
Name	String	The name of the microagent

Returns

Parameter	Type	Description
Name	String	The name of the microagent
Display Name	String	The microagent display name
Status	String	Can be <code>Unreachable</code> , <code>Removed</code> , <code>Added</code>

COM.TIBCO.hawk.microagent.SysInfo

Microagent

Purpose The `sysinfo` microagent (on all platforms) identifies a network host. Its methods return name, address, type of computer and operating system.

Methods	Method	Description	Page
	SysInfo:getOperatingSystem	Returns operating system information	28
	SysInfo:getArchitecture	Returns hardware architecture	29
	SysInfo:getHostName	Returns the name of the host and the agent domain	30
	SysInfo:getNetworkAddress	Returns the IP address of the host	31

SysInfo:getOperatingSystem

Method

Purpose This method (on all platforms) returns operating system information.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
OS Name	String	The operating system name
OS Version	String	The operating system version

SysInfo:getArchitecture

Method

Purpose This method (on all platforms) returns hardware architecture.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Architecture	String	The architecture name

SysInfo:getHostName

Method

Purpose This method (on all platforms) returns the name of the host.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Host Name	String	The host name
Agent Domain	String	The name of the agent domain

SysInfo:getNetworkAddress

Method

Purpose This method (on all platforms) returns the IP address of the host.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
IP Address	String	The IP address of the host. For example: 160.101.26.126.

COM.TIBCO.hawk.microagent.HawkEventService

Microagent

Description This application reports on events generated by the TIBCO Hawk Agents across the network. Events reported include all instances of TIBCO Hawk Agent activation and expiration, add and remove operations for microagents and rulebases, all alerts generated and cleared.

Methods

Method	Description	Page
HawkEventService: onAgentAlive	Reports all instances of TIBCO Hawk Agent activation events across the network.	33
HawkEventService: onAgentExpired	Reports all instances of TIBCO Hawk Agent expiration events across the network.	34
HawkEventService: onAlert	Reports all alerts generated by TIBCO Hawk Agents across the network.	35
HawkEventService: onClear	Reports all clears generated by TIBCO Hawk Agents across the network.	37
HawkEventService: onMicroAgentChange	Reports all instances of microagent added and removed events across the network.	38
HawkEventService: onRulebaseChange	Reports all instances of rulebase added and removed events across the network.	39

HawkEventService:onAgentAlive

Method

Purpose Reports all instances of TIBCO Hawk Agent activation events across the network.

Type Open, Asynchronous, IMPACT_INFO

Arguments

Name	Type	Description
Agent Name	String	TIBCO Hawk Agent name. The symbol can be used to perform an OR operation. For example, abc xyz def will evaluate to true if Agent Name equals either abc or xyz or def.

Returns

Name	Type	Description
Hawk Domain	String	Value of the command line option "hawk_domain"
Agent Name	String	TIBCO Hawk Agent's name that generated this event
Agent Domain	String	TIBCO Hawk Agent's domain name (dns)
Agent IP Address	String	TIBCO Hawk Agent's IP address
Agent Start Time	String	TIBCO Hawk Agent's start time

HawkEventService:onAgentExpired

Method

Purpose Reports all instances of TIBCO Hawk Agent expiration events across the network.

Type Open, Asynchronous, IMPACT_INFO

Arguments

Name	Type	Description
Agent Name	String	TIBCO Hawk Agent name. The symbol can be used to perform an OR operation. For example, abc xyz def, will evaluate to true if Agent Name equals either abc or xyz or def.

Returns

Name	Type	Description
Hawk Domain	String	Value of the command line option "hawk_domain"
Agent Name	String	TIBCO Hawk Agent's name that generated this event
Agent Domain	String	TIBCO Hawk Agent's domain name (dns)
Agent IP Address	String	TIBCO Hawk Agent's IP address
Agent Start Time	String	TIBCO Hawk Agent's start time
Agent Expired Time	String	TIBCO Hawk Agent's expiry time

HawkEventService:onAlert

Method

Purpose Reports all alerts generated by TIBCO Hawk Agents across the network

Type Open, Asynchronous, IMPACT_INFO

Arguments	Name	Type	Description
	Agent Name	String	TIBCO Hawk Agent name. The symbol can be used to perform an OR operation. For example, abc xyz def will evaluate to true if Agent Name equals either abc or xyz or def.
	Rulebase	String	The name of the rulebase. The symbol can be used to perform an OR operation. For example, abc xyz def will evaluate to true if Rulebase Name equals either abc or xyz or def.

Returns	Name	Type	Description
	Hawk Domain	String	Value of the command line option "hawk_domain"
	Agent Name	String	TIBCO Hawk Agent's name that generated this event
	Agent Domain	String	TIBCO Hawk Agent's domain name (dns)
	Agent IP Address	String	TIBCO Hawk Agent's IP address
	Rulebase	String	The rulebase that generated this alert
	AlertID	Long	The numeric ID assigned to this alert by the TIBCO Hawk Agent
	Time	String	Time at which this alert was generated
	Text	String	User defined alert text value
	State	String	Can be ALERT_LOW, ALERT_MEDIUM, ALERT_HIGH or NO_ALERT
	Rule	String	The rule that generated this alert

Name	Type	Description (Cont'd)
Test	String	The test that generated this alert
Alert Properties	String	Miscellaneous alert properties defined by the user

HawkEventService:onClear

Method

Purpose Reports all clears generated by TIBCO Hawk Agents across the network

Type Open, Asynchronous, IMPACT_INFO

Arguments

Name	Type	Description
Agent Name	String	TIBCO Hawk Agent name. The symbol can be used to perform an OR operation. For example, abc xyz def, will evaluate to true if Agent Name equals either abc or xyz or def.
Rulebase	String	The name of the rulebase. The symbol can be used to perform an OR operation. For example, abc xyz def, will evaluate to true if Rulebase Name equals either abc or xyz or def.

Returns

Name	Type	Description
Hawk Domain	String	Value of the command line option "hawk_domain"
Agent Name	String	TIBCO Hawk Agent's name that generated this event
Agent Domain	String	TIBCO Hawk Agent's domain name (dns)
Agent IP Address	String	TIBCO Hawk Agent's IP address
Rulebase	String	The rulebase that generated this clear
AlertID	Long	The numeric ID assigned to this clear by the TIBCO Hawk Agent
Time	String	Time at which this alert was cleared
Clear Reason	String	Reason for the clear

HawkEventService:onMicroAgentChange

Method

Purpose Reports all instances of microagent added and removed events across the network.

Type Open, Asynchronous, `IMPACT_INFO`

Arguments

Name	Type	Description
Agent Name	String	TIBCO Hawk Agent name. The symbol can be used to perform an OR operation. For example, <code>abc xyz def</code> , will evaluate to true if Agent Name equals either abc or xyz or def.

Returns

Name	Type	Description
Hawk Domain	String	Value of the command line option "hawk_domain"
Agent Name	String	TIBCO Hawk Agent's name that generated this event
Agent Domain	String	TIBCO Hawk Agent's domain name (dns)
Agent IP Address	String	TIBCO Hawk Agent's IP address
Action	String	Type of event, could either be <code>onMicroAgentAdded</code> or <code>onMicroAgentRemoved</code>
MicroAgentID	String	Name and the instance of the microagent

HawkEventService:onRulebaseChange

Method

Purpose Reports all instances of rulebase added and removed events across the network.

Type Open, Asynchronous, IMPACT_INFO

Arguments

Name	Type	Description
Agent Name	String	TIBCO Hawk Agent name. The symbol can be used to perform an OR operation. For example, abc xyz def, will evaluate to true if Agent Name equals either abc or xyz or def.

Returns

Name	Type	Description
Hawk Domain	String	Value of the command line option "hawk_domain"
Agent Name	String	TIBCO Hawk Agent's name that generated this event
Agent Domain	String	TIBCO Hawk Agent's domain name (dns)
Agent IP Address	String	TIBCO Hawk Agent's IP address
Action	String	Type of event, could either be onRulebaseAdded or onRulebaseRemoved
Rulebase	String	Name of the rulebase

COM.TIBCO.hawk.microagent.RuleBaseEngine

Microagent

Purpose The `RuleBaseEngine` microagent (on all platforms) returns information from rulebases and takes actions on rulebases that affect the TIBCO Hawk agent itself. `RuleBaseEngine` methods are used to examine auto-configuration information, to examine active microagents, and to load and unload rulebases (for example in a rulebase-management rulebase).

Remarks In prior releases of the TIBCO Hawk software, many of the `RuleBaseEngine` functions were handled by the `Self` microagent.

Some methods of the `RuleBaseEngine` microagent are proprietary. They are listed here for completeness, but cannot be invoked directly.

Methods

Method	Description	Page
RuleBaseEngine:deleteRuleBase	Deletes the specified rulebase.	43
RuleBaseEngine:getExternalVariables	Returns the currently loaded external variables as tabular data	44
RuleBaseEngine:getRuleBaseNames	Returns the names of the currently loaded rulebases	45
RuleBaseEngine:getScheduleNames	Returns names of currently loaded schedules	46
RuleBaseEngine:loadRuleBase	Attempts to find and load the named rulebase	47
RuleBaseEngine:loadRuleBaseFromFile	Attempts to find and load a rulebase from a file path	48
RuleBaseEngine:unloadRuleBase	Attempts to unload the named rulebase	49
RuleBaseEngine:getConfigInfo	Returns configuration information for the rulebase	50

Method	Description	Page
RuleBaseEngine:sendMail	Sends email notification upon a rulebase event	51
RuleBaseEngine:onAlertCount	Provides current and average rulebase engine alert rate statistics	52
RuleBaseEngine:onAlertCountForRulebase	Provides the rulebase engine alert rate statistics	53
RuleBaseEngine:suspendAlert	Suspends an alert	54
RuleBaseEngine:suspendAllAlerts	Suspends alerts for the specified number of milliseconds	55
RuleBaseEngine:resumeSuspendedAlerts	Restarts alerts after alert suspension	56
RuleBaseEngine:updateExternalVariables	Updates the external variables and saves them to the variable file	57

Proprietary Methods

The following methods of the `RuleBaseEngine` microagent are proprietary. The TIBCO Hawk system does not support direct user invocation of these methods from the TIBCO Hawk Display or Console API.

However, these methods are used indirectly by TIBCO Hawk Display when interacting with agents and thus have security implications. Access to proprietary methods with an impact of Action should only be granted to trusted users who are allowed access to all Action methods.

Proprietary Method Summary

Method Name	Impact	Description
<code>getCommands</code>	Info	Returns the rulebase commands
<code>getRuleBases</code>	Info	Retrieves rulebases from the agent
<code>getSchedules</code>	Info	Returns the Schedules List
<code>getRBMap</code>	Info	Returns the <code>RuleBaseMap</code> file
<code>addRuleBase</code>	Action	Sends rulebases to the agent

Method Name	Impact	Description
updateRuleBase	Action	Updates rulebases on the agent
setSchedules	Action	Sets the schedules list for an agent
setRBMap	Action	This method sets the RuleBaseMap file for an agent
runTest	Action	Runs tests for network queries
onAlertPosted	Info	Provides data for alerts posted by the Rulebase Engine
onAlertCleared	Info	Provides data for alerts cleared by the Rulebase Engine

RuleBaseEngine:deleteRuleBase

Method

Purpose This method (on all platforms) deletes the specified rulebase.

Remarks The agent will delete the rulebase from its cache. If the agent is configured to run under `auto config` mode it will delete this rulebase from the `auto config` directory as well.

Type Synchronous, `IMPACT_ACTION`.

Arguments

Name	Type	Description
RuleBaseName	String	The name of the rulebase to delete

Returns None.

RuleBaseEngine:getExternalVariables

Method

Purpose This method (on all platforms) returns the currently loaded external variables as tabular data in two fields, variable name and value.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Variable Name	String	The name of the external variable
Value	String	The value of the external variable

RuleBaseEngine:getRuleBaseNames

Method

Purpose This method (on all platforms) returns the names of the currently loaded rulebases.

Remarks Each record of the table identifies one rulebase. Therefore, tests that search for strings within a name (such as NT) will result in separate actions for each rulebase name that matches the criterion.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
RuleBaseName	String	The name of each rulebase. Table is indexed on the RuleBaseName field

RuleBaseEngine:getScheduleNames

Method

Purpose This method (on all platforms) returns the names of the currently loaded schedules.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
ScheduleName	String	The name of each schedule. Table is indexed on the ScheduleName field.

RuleBaseEngine:loadRuleBase

Method

Purpose This method (on all platforms) attempts to find and load the named rulebase.

Remarks The agent will search all directories in its manual configuration path for the rulebase file, whether manual or automatic configuration is being used.

Type Synchronous, IMPACT_ACTION.

Arguments

Name	Type	Description
RuleBaseName	String	The name of the rulebase to load

Returns None.

RuleBaseEngine:loadRuleBaseFromFile

Method

Purpose This method (on all platforms) attempts to find and load the rulebase in the named file. An absolute file path is required.

Type Synchronous, `IMPACT_ACTION`.

Arguments

Name	Type	Description
File	String	The absolute path name of the file from which to load rulebases

Returns None.

RuleBaseEngine:unloadRuleBase

Method

Purpose This method (on all platforms) attempts to unload the named rulebase.

Remarks If the rulebase is not currently loaded then no error is returned.

Type Synchronous, IMPACT_ACTION.

Arguments

Name	Type	Description
RuleBaseName	String	The name of the rulebase to unload

Returns None.

RuleBaseEngine:getConfigInfo

Method

Purpose This method (on all platforms) returns information on how the rulebase is currently configured.

Type Synchronous, `IMPACT_INFO`.

Arguments None.

Returns

Name	Type	Description
Name	String	Name of rulebase
Reason Loaded	String	Reason for loading rulebase, such as <code>auto-config startup</code>
Loaded From	String	Source of rulebase
Load Time	String	Time rulebase was loaded, in format of day of week, month, day, time, year

RuleBaseEngine:sendMail

Method

Purpose This method (on all platforms) sends an email notification.

Remarks The `To` and `Subject` fields are only mandatory fields and all other fields are optional. If `From` is not specified, the current host ID is used. If the `Content` field is blank, the text of the `Subject` line is used. If the `Mail Server` is not specified, then SMTP server configured in the agent sends email.

Rulebases can send mail upon detecting a specified condition.

Type Synchronous, `IMPACT_ACTION`.

Arguments

Name	Type	Description
To	String	Address of the receiver
CC	String	CC (carbon copy) recipients of email
BCC	String	BCC (blind carbon copy) recipients of email
Subject	String	Subject of email
Content	String	Content of email
Mail Server	String	SMTP mail server used to send message
From	String	Address of the sender

Returns None.

RuleBaseEngine:onAlertCount

Method

Purpose Provides the rulebase engine alert rate statistics every 5 seconds. It provides alert rate averages over the previous 5, 10, 15, 30 and 60 seconds.

Remarks When using this method as a datasource in a rulebase, the test's clear condition will default to "Clear Timer" with a value of 900 seconds. However, because this async method provides statistics every 5 sec, choosing clear condition "First False" may be more useful.

`RuleBaseEngine:onAlertCount` and `RuleBaseEngine:suspendAllAlerts` can be used together to create an alert-rate circuit breaker that insulates the network from agents that produce alerts at message high rates that are too high. For example, using `onAlertCount` as a data source, if (5sec > 10) then `suspendAllAlerts` for 60,000 msec. This would suspend all alerts for 1min if the average rate of alerts over the last 5sec was greater than 2 alerts per sec.

Type Asynchronous, `IMPACT_INFO`

Arguments None

Returns

Name	Type	Description
5sec	Integer	The alert rate averaged over the previous 5 seconds
10sec	Integer	The alert rate averaged over the previous 10 seconds
15sec	Integer	The alert rate averaged over the previous 15 seconds
30sec	Integer	The alert rate averaged over the previous 30 seconds
60sec	Integer	The alert rate averaged over the previous 60 seconds

RuleBaseEngine:onAlertCountForRulebase

Method

- Purpose** Provides the rulebase engine alert rate statistics for a given rulebase every five seconds. It provides alert rate averages over the previous 5, 10, 15, 30 and 60 seconds.
- Remarks** When using this method as a datasource in a rulebase, the clear condition of the test defaults to "Clear Timer" with a value of 900 seconds. However, because this async method provides statistics every five seconds, choosing clear condition "First False" may be more useful.
- Type** Open, Asynchronous, IMPACT_INFO

Arguments

Name	Type	Description
RuleBaseName	String	The name of the rulebase.

Returns

Name	Type	Description
5sec	Integer	The alert rate averaged over the previous 5 seconds
10sec	Integer	The alert rate averaged over the previous 10 seconds
15sec	Integer	The alert rate averaged over the previous 15 seconds
30sec	Integer	The alert rate averaged over the previous 30 seconds
60sec	Integer	The alert rate averaged over the previous 60 seconds

RuleBaseEngine:suspendAlert

Method

Purpose This method suspends an alert for the specified number of milliseconds. The default is 10000 milliseconds.

Type Open, Synchronous, IMPACT_ACTION

Arguments

Name	Type	Description
AlertID	Long	Alert ID
Time	Long	Time in milliseconds
Reason	String	Reason to suspend alert

Returns None.

RuleBaseEngine:suspendAllAlerts

Method

Purpose Suspends all current and future alerts in the rulebase engine for the specified number of milliseconds.

Remarks If a specific rulebase is provided as an argument, this method applies only the alerts generated by that rulebase.

This method is generally used as an Action to a Rule using `RBE:onAlertRate` as a data source to create an alert-rate circuit breaker.

`RuleBaseEngine:suspendAllAlerts` takes a rulebase argument. If empty, it suspends all alerts on all rulebases. After suspending all alerts, a high level alert (one per rulebase) is issued to indicate that the rulebase has been suspended. That alert will be cleared when the suspension period expires.

When a rulebase is suspended, all its current and future alerts are suspended for the specified interval. Note that all other monitoring activity continues. For example, actions that send email or invoke scripts or other methods continue to operate normally.

Type Open, Synchronous, `IMPACT_ACTION`

Arguments

Name	Type	Description
RuleBaseName	String	The name of the rulebase whose alerts will be suspended. If omitted, this method applies to all rulebases.
Time	Long	The suspension period in milliseconds. Cannot be zero or negative.
Reason	String	The reason for this suspension. It is useful to also provide the time and source of suspension in this string. Cannot be empty string.

Returns None

RuleBaseEngine:resumeSuspendedAlerts

Method

- Purpose** This method (on all platforms) restarts all alerts after alert suspension.
- Type** Asynchronous, `IMPACT_ACTION`.
- Arguments** None.
- Returns** None.

RuleBaseEngine:updateExternalVariables

Method

Purpose	This method (on all platforms) reads the variables file and re-loads all rulebases that use the variables listed in the variable file.
Type	Synchronous, <code>IMPACT_ACTION</code> .
Arguments	None.
Returns	None.

COM.TIBCO.hawk.microagent.Messaging

Microagent



This microagent is available only when Hawk Message Transport is TIBCO Rendezvous.

Purpose The `Messaging` microagent (on all platforms) responds to requests from TIBCO Hawk agents to send and receive messages using either TIBCO Rendezvous or TIBCO EMS.

Methods

Method	Description	Page
Messaging:sendMessage	Sends a simple message as a string using the configured transport	59
Messaging:onMessage	Receives a message	60
Messaging:onTimeout	Returns the number of timeout intervals that have elapsed since the arrival of the previous message or since the start of the subscription to this method.	61
Messaging:onNumber	Receives a message with a <code>_data_</code> field having a numeric value	62

Messaging:sendMessage

Method

Purpose Sends a simple message as a string (in the field `_data_`) using the configured transport.

Type Synchronous, `IMPACT_ACTION`.

Arguments

Name	Type	Description
Subject	String	Message subject. Any leading and trailing spaces in the subject are ignored.
Message	String	Content of the message

Returns None.

Messaging:onMessage

Method

Purpose Receives a message using the configured transport.

Type Asynchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Subject	String	Message subject. Any leading and trailing spaces in the subject are ignored.

Returns

Name	Type	Description
Subscription Subject	String	Subscription subject
Message Subject	String	Message subject specified by the sender
MSG_CONTENT	String	Content of the message

Messaging:onTimeout

Method

Purpose Returns the number of timeout intervals that have elapsed since the arrival of the previous message or since the start of the subscription to this method.

Type Asynchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Subject	String	Subscription subject. Any leading and trailing spaces in the subject are ignored.
Timeout Interval	Integer	Timeout interval in seconds

Returns

Name	Type	Description
Timeout Interval Count	Integer	The count of consecutive timeout intervals that have passed. The return value is reset to 0 (zero) when a message is received with the specified subject or topic. The return value of 0 can be used as a clear test condition within rulebases.

Messaging:onNumber

Method

Purpose Receives a message with a `_data_` field having a numeric value. The actual value could be either a `string`, `integer`, `long`, `float`, or `double` data type which is converted to a `double` data type.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Subject	String	Subscription subject Any leading and trailing spaces in the subject are ignored.

Returns

Name	Type	Description
Subscription Subject	String	The subscription subject
Message Subject	String	The message subject
<code>_data_</code>	Double	The numeric data

Returns

Name	Type	Description
Virtual Machine Process Id	String	Process Id of Java Virtual Machine process
Virtual Machine Process Name	String	The name of representing the running Java Virtual Machine
Thread Id	String	The ID of the thread
Thread Name	String	The name of the thread
Thread State	String	The state of the thread
Suspended	String	Denotes whether the thread is suspended
isInNative	String	Denotes whether it is executing native code via the Java Native Interface (JNI)

Name	Type	Description
Blocked Count	String	The total number of attempts that the thread is blocked to enter or re-enter a monitor.
Blocked Time	String	The approximate accumulated elapsed time (in milliseconds) that the thread has blocked to enter or re-enter a monitor since the enabling of thread contention monitoring.
Waited Count	String	The total count of the number of times the thread waited for notification.
Waited Time	String	The approximate accumulated elapsed time (in milliseconds) that the thread has waited for notification since the enabling of thread contention monitoring.
Thread User Time	String	CPU time executed by the thread in user mode in nanoseconds
Lock Name	String	The string representation of the monitor lock that the thread is blocked to enter or waiting to be notified through the <code>Object.wait</code> method.
Lock Owner Id	String	The thread ID which holds the monitor lock of an object on which the thread is blocked.
Lock Owner Name	String	The thread name which holds the monitor lock of an object on which the thread is blocked.

COM.TIBCO.hawk.microagent.Repository

Microagent

Purpose The `repository` microagent (on all platforms) responds to configuration requests from TIBCO Hawk agents.

Remarks Some methods of the Repository microagent are proprietary. They are listed here for completeness, but cannot be invoked directly.

Methods

Method	Description	Page
Repository:getName	Returns the name of the TIBCO Hawk repository	66
Repository:getScheduleNames	Returns the list of schedule names currently in the repository	67
Repository:getRuleBaseName	Returns the list of rulebase names currently in the repository	68
Repository:onRepositoryEvent	Returns repository operation	69

Proprietary Methods The following methods of the `RuleBaseEngine` microagent are proprietary. The TIBCO Hawk system does not support direct user invocation of these methods from the TIBCO Hawk Display or Console API.

However, these methods are used indirectly by the TIBCO Hawk Display when interacting with agents and thus have security implications. Access to proprietary methods with an impact of Action should only be granted to trusted users who are allowed access to all Action methods

Proprietary Methods Summary.

Method Name	Impact	Description
<code>getRuleBases</code>	Info	This method retrieves rulebases from the repository
<code>getSchedules</code>	Info	This method returns the schedules list
<code>getRBMap</code>	Info	This method returns the RuleBaseMap file
<code>addRuleBase</code>	IMPACT_ACTION	This method send rulebases to the repository.

Method Name	Impact	Description
updateRuleBase	Action	This method updates repository rulebases.
setSchedules	Action	This method sets the schedules list for an repository
deleteRuleBase	IMPACT_ACTION	This method deletes a rulebase from the repository.
setRBMap	Action	This method sets the RuleBaseMap for a repository.

Repository:getName

Method

Purpose Returns name of the repository.

Type Asynchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Name	String	Name of the repository

Repository:getScheduleNames

Method

Purpose Returns a table of names of the schedules currently in this repository.

Type Asynchronous, `IMPACT_INFO`.

Arguments None.

Returns

Name	Type	Description
Schedule name	String	Name of the schedule in repository

Repository:getRuleBaseName

Method

Purpose Returns a table of names of the rulebases in this repository.

Type Asynchronous, `IMPACT_INFO`.

Arguments None.

Returns

Name	Type	Description
Rulebase name	String	Name of the rulebase

Repository:onRepositoryEvent

Method

Purpose Returns information on repository events.

Type Asynchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Object Type	String	Type of object associated with event

Returns

Name	Type	Description
Event Type	String	Type of event that occurred
Object Name	String	Name of object associated with event
Object Type	String	Type of object associated with event

COM.TIBCO.hawk.microagent.Logfile

Microagent

Purpose The `Logfile` microagent (on all platforms) monitors the contents of log files. It has only one method: `onNewLine`. When used as a data source in a rule, the `onNewLine` method monitors growing log files and provides each new line as a new data element for testing. The `Logfile` microagent can process rolling log files, and so it continues to monitor the correct data stream even when log files roll over.

The `Logfile` microagent assumes that the file being monitored is a constantly growing log file with all new data appended at the end. If it detects that a file has decreased in size, it assumes that the file has been overwritten (or rolled) and begins monitoring the new file from the beginning.

When the `Logfile` microagent is first directed to monitor a particular log file (such as when you first load a rulebase that uses it), monitoring begins from the end of the file. In other words, only new lines are reported and any lines that existed in the file before monitoring was initiated are ignored.

Methods

Method	Description	Page
Logfile:onNewLine	Returns the text of a new line as it is added to the named log file	71
Logfile:onNewLinewithPatternFile	Returns the test of a new line as it is added to the named log file using the specified pattern file as a filter.	73
Logfile:reEvaluateLogFileNames	Forces re-evaluation of all logfiles that are currently being monitored.	75

Logfile:onNewLine

Method

Purpose This method (on all platforms) returns the new line in a log file whenever a new line is appended to the log file.

The first time this method is invoked, the `previousLine` field will be blank. On subsequent invocations, the `previousLine` field will contain the value of the `nextLine` field of the previous method invocation.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Logfile	String	The log file to be monitored.
		Back quoted strings, external and internal variables can be used. The TIBCO Hawk agent will execute the command within the back quotes and return the string for substitution.
		For example, <code>'echo /tmp/foo.log'</code> will be evaluated to <code>/tmp/foo.log</code>

Returns

Name	Type	Description
<code>nextLine</code>	String	The next available line in the log file
<code>previousLine</code>	String	The previous line in the log file

Logfile: onXMLElement

Method

Purpose This method (on all platforms) monitors XML log files.

Type Asynchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Logfile	String	The log file to be monitored. Back quoted strings, external and internal variables can be used. The TIBCO Hawk agent will execute the command within the back quotes and return the string for substitution. For example, 'echo /tmp/foo.log' will be evaluated to /tmp/foo.log
element	String	The tag name of element to monitor
xpath	String	The XPATH expression to apply to the element

Returns

Name	Type	Description
nextResult	String	The result of the XPATH evaluation

Logfile:onNewLinewithPatternFile

Method

Purpose This method (on all platforms) applies the patterns from the pattern file to a new line from the log file. It returns the matching severity and pattern from the pattern file and the new line from the log file. The pattern file contains patterns that are used to match relevant log file entries. The pattern file is used as a filter to determine which lines are returned to the subscriber.

The first time this method is invoked, the `previousLine` field will be blank. On subsequent invocations, the `previousLine` field will contain the value of the `nextLine` field of the previous method invocation.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Logfile	String	The log file to be monitored. Back quoted strings, external and internal variables can be used. The TIBCO Hawk agent will execute the command within the back quotes and return the string for substitution. For example, 'echo /tmp/foo.log' will be evaluated to /tmp/foo.log
patternfile	String	The pattern file containing the severity values and the Perl5 patterns. The pattern file cannot be empty, it should contain atleast one valid entry. Each line in the pattern file should contain an integer severity level followed by a white space and a valid Perl5 pattern. A negative severity level (for example, -1) means that all lines that match this pattern are to be ignored. All other severity levels (positive or 0) indicate that lines matching the corresponding patterns are of interest

Returns

Name	Type	Description
severity	Integer	The severity value of the matching pattern
pattern	String	The Perl5 pattern string of the matching pattern
nextLine	String	The next available line in the log file

Name	Type	Description
previousLine	String	The previous line in the log file

Logfile:reEvaluateLogFileNames

Method

Purpose	This method (on all platforms) forces a re-evaluation of all log file names that are currently being monitored.
Type	Synchronous, IMPACT_ACTION.
Arguments	None.
Returns	None

COM.TIBCO.hawk.microagent.Custom

Microagent

Purpose The `Custom` microagent (on all platforms) runs executable programs and scripts from within the TIBCO Hawk system. You can retrieve information from a script or program. Custom methods also allow you to write rules that act on that information as though it came from a microagent method call.

Methods	Method	Description	Page
	Custom:executeForString	Executes a command and returns the output as a String	77
	Custom:executeForNumber	Executes a command and parses the string output to return both an Integer and a Double	78
	Custom:execute	Executes a command and ignores the result	79



For instructions on how to disable execution of this method, refer to *TIBCO Hawk Installation and Configuration*.

Custom:executeForString

Method

Purpose This method (on all platforms) executes a command and returns the output as a string.

Remarks The command that is executed can return information by sending strings to `stdout`. If the executed command returns more than one line, the lines are concatenated to produce one string.

When using this method on the OS/400 platform, only commands that start with `system` (when using a CL command) and `qsh` (when using Q shell utilities) are supported. Use the `-c` option when using the `qsh` command.

Examples,

```
qsh -c ls
```

```
system DSPDTAARA DTAARA (<LibName>/<Data Area Name>)
```

Type Synchronous, `IMPACT_ACTION_INFO`.

Arguments

Name	Type	Description
command	String	The command to execute. Should be exactly as would be read on the command line. External and internal variables can be used in the command string.

Returns

Name	Type	Description
returnString	String	The String representation of the command's output

Custom:executeForNumber

Method

Purpose This method (on all platforms) executes a command and parses the output as a number which is then returned as both an integer and a double value. The integer value of a true double truncates the remainder.

Remarks The command that is executed can return information by sending a number to `stdout`. If the executed command returns more than one number, an error will result.

The executable can return information by sending strings to `stdout`. If the executable returns more than one line, the lines are concatenated to produce one string.

Type Synchronous, `IMPACT_ACTION_INFO`.

Arguments

Name	Type	Description
command	String	The command to execute. External and internal variables can be used in the command string.

Returns

Name	Type	Description
returnInteger	Integer	The integer value of the return string
returnDouble	Double	The double value of the return string

Custom:execute

Method

Purpose This method (on all platforms) executes a command and ignores the result.

Remarks If this method call is performed within an action, you can provide parameters for a script or program by using variable substitution. Variable substitution will insert information into the command from the rule's data source.

When using this method on the OS/400 platform, only commands that start with `system` (when using a CL command) and `qsh` (when using Q shell utilities) are supported. Use the `-c` option when using the `qsh` command.

Examples,

```
qsh -c ls
```

```
system DSPDTAARA DTAARA (<LibName>/<Data Area Name>)
```

Type Synchronous, IMPACT_ACTION.

Arguments

Name	Type	Description
command	String	The command to execute

Returns None.



All shell scripts called from your TIBCO Hawk environment must define the shell, for example `#!/bin/csh`.

COM.TIBCO.hawk.hma.FileStat

Microagent

Purpose The `FileStat` microagent (on all platforms) monitors the status and operating statistics of files.

Methods	Method	Description	Page
	FileStat:getFileStatus	Returns file status information	81
	FileStat:getFileCount	Returns the number of files matching a regular expression	85

FileStat:getFileStatus

Method

Description: This method returns status information for the directory(s) or file(s) contained in the specified directory whose names match the specified regular expression. Use a regular expression to query the status of a set of files. The regular expression can only be used on the file name portion of the input parameter. For example
`c:/temp/.*`

This expression queries all the files in `c:/temp` directory.
`c:/temp/.*.tmp`

This queries all the files in `c:/temp` directory that have the `.tmp` suffix.

Remarks The arguments and return values for this method are different for Microsoft Windows and Unix platforms.

Type Synchronous, `IMPACT_INFO`.

Microsoft Windows Only Arguments

Name	Type	Description
Directory	String	Name of the directory to be searched. No entry indicates the current directory. Regular expressions are <i>not</i> allowed for this argument.
File Name	String	Name of the file to query

Unix Only Arguments

Name	Type	Description
File Name	String	Absolute path and name of the file to query

Common to Both Microsoft Windows and Unix Returns

Name	Type	Description
File Name	String	Name of the file to query. Table is indexed on the File Name
File Size	Long	Total size of the file in bytes
Is Directory	Boolean	The handle identifies a directory
Seconds Since Accessed	Integer	Number of seconds elapsed since last access
Seconds Since Modified	Integer	Number of seconds elapsed since last modified

Name	Type	Description
Seconds Since Creation	Integer	(Microsoft Windows only) Number of seconds elapsed since creation. (UNIX only) Number of seconds since last status change.

**Microsoft
Windows Only
Returns**

Name	Type	Description
Alt File Name	String	(Microsoft Windows only) The alternate file name of this file or directory. This name is in the classic 8.3 (filename.ext) file name format.
Is Read Only	Boolean	(Microsoft Windows only) The file or directory is read-only. Applications can read the file but cannot write to it or delete it. In the case of a directory, applications cannot delete it.
Is Hidden	Boolean	(Microsoft Windows only) The file or directory is hidden. It is not included in an ordinary directory listing.
Is System	Boolean	(Microsoft Windows only) The file or directory is part of the operating system or is used exclusively by the operating system.
Is Archive	Boolean	(Microsoft Windows only) The file or directory is an archive file or directory. Applications use this attribute to mark files for backup or removal.
Is Encrypted	Boolean	(Microsoft Windows only) The file or directory is encrypted. For a file, this means that all data in the file is encrypted. For a directory, this means that encryption is the default for newly created files and subdirectories.
Is Normal	Boolean	(Microsoft Windows only) The file or directory has no other attributes set. This attribute is valid only if used alone.

Name	Type	Description
Is Offline	Boolean	(Microsoft Windows only) The file data is not immediately available. This attribute indicates that the file data has been physically moved to offline storage. This attribute is used by Remote Storage, the hierarchical storage management software. Applications should not arbitrarily change this attribute.
Has Reparse Point	Boolean	(Microsoft Windows only) The file has an associated reparse point.
Is Sparse	Boolean	(Microsoft Windows only) The file is a sparse file.
Is Temporary	Boolean	(Microsoft Windows only) The file is being used for temporary storage. File systems attempt to keep all of the data in memory for quicker access, rather than flushing it back to mass storage. A temporary file should be deleted by the application as soon as it is no longer needed.

Unix Only Returns

Name	Type	Description
Major	Integer	Major number of the device on which the file resides
Minor	Integer	Minor number of the device on which the file resides
Inode	Integer	File inode number
Mode	String	File mode as octal mask
Is Character Special	Boolean	This file is a character special file
Is Pipe	Boolean	This file is a pipe (FIFO)
Is Regular	Boolean	This file is a regular file
Is Readable	Boolean	This file has readable permission set
Is Writable	Boolean	This file has writable permission set

Name	Type	Description
Is Executable	Boolean	This file has executable permission set
Links	Integer	Count of hard links to the file
User	String	User name of the owner
Group	String	Group name of the owner

FileStat:getFileCount

Method

Description This method returns the count of directory(s) or file(s) contained in the specified directory whose names match the specified regular expression. The regular expression syntax follows the same syntax as specified for the `getFileStatus` method.

Remarks The arguments for this method are different for Windows and Unix platforms.

Type Synchronous, `IMPACT_INFO`.

Windows Only Arguments

Name	Type	Description
Directory	String	Name of the directory to be searched. No entry indicates the current directory. Regular expressions are <i>not</i> allowed for this argument.
File Name	String	Name of the file to query

Unix Only Arguments

Name	Type	Description
File Name	String	Absolute path and name of the file to query

Returns

Name	Type	Description
File Count	Integer	Number of files that match the specified query. Table is indexed on the <code>File Count</code>

COM.TIBCO.hawk.hma.TibRendezvous

Microagent



This microagent is available only when Hawk Message Transport is TIBCO Rendezvous.

Purpose The `TibRendezvous` microagent (on all platforms) monitors the status and operating statistics of a TIBCO Rendezvous daemon.

Remarks Because the behavior and configuration of the `TibRendezvous` microagent is identical across platforms, it is included in the platform-independent section of this reference. However, this microagent runs within the HMA process, which is specific to each platform.

All TIBCO Rendezvous advisory messages of class `ERROR` and `WARN` will be monitored by the `TibRendezvous` microagent. If an `ERROR` or `WARN` advisory message is not monitored by any of the `TibRendezvous` microagent methods, the message is presented as an unsolicited message. The message text will include the TIBCO Rendezvous daemon parameters, which indicate the message origin.

Methods

Method	Description	Page
TibRendezvous:onRvDaemonStatus	Subscribes to TIBCO Rendezvous internal status messages.	87
TibRendezvous:onRvDataLoss	Subscribes to TIBCO Rendezvous data loss messages.	91
TibRendezvous:onRvdDisconnectOrConnect	Subscribes to TIBCO Rendezvous disconnected/connected advisory messages.	92
TibRendezvous:onRvLicenseExpire	Subscribes to TIBCO Rendezvous license expired advisory messages.	93
TibRendezvous:onRvdRetransmissionSuppressed	Subscribes to TIBCO Rendezvous daemon outbound retransmission suppressed advisory messages.	94

TibRendezvous:onRvDaemonStatus

Method

Purpose This method (on all platforms) subscribes to TIBCO Rendezvous internal status messages and reports TIBCO Rendezvous daemon status.

Remarks The method argument `Interval` specifies the interval in seconds at which daemon status is to be reported (that is, the rate at which asynchronous method `onRvDaemonStatus` is called.)

The TIBCO Rendezvous status messages being monitored are reported at fixed 90 second intervals. The TibRendezvous Microagent must collect numerous status messages in order to construct an accurate view of the daemon status being reported. It is recommended that an `Interval` be specified that is a multiple of 90 seconds and no less than 90 seconds. If you enter a value less than 90 seconds, the default value (90 seconds) will be used.

If you have more than one network interface on the machine where the TIBCO Rendezvous daemon is running, the method may not report the daemon status if the network parameter is not specified. To determine the network interface that the TIBCO Rendezvous daemon uses to publish the status, use the TIBCO Rendezvous HTTP interface and browse to the specific service.

Type Asynchronous, INFO.

Arguments

Name	Type	Description
Service	String	TIBCO Rendezvous service
Network	String	TIBCO Rendezvous network
Daemon	String	TIBCO Rendezvous daemon
Interval	Integer	Report interval in seconds. (See Remarks above.)

Returns

Name	Type	Description
IP Address	String	Daemon IP address in standard form, for example 255.100.10.30

Name	Type	Description
IP Address (bin)	String	Daemon IP address. 1. The return value is a string representation of the binary digits that make up the IP address. For example, the IP address 255.100.10.30 would be presented as "11111111.1100100.00001010.000 11110". The IP address in this form allows for more flexible rulebase tests to be constructed using the Perl5PatternMatch operator of the TIBCO Hawk rulebase editor
Uptime	Integer	The number of seconds since the start time on this service as reported by the TIBCO Rendezvous daemon.
Version	String	The version of the daemon executable formatted as a string. For example, 6.3.5.
HTTP IP	String	IP address of the TIBCO Rendezvous daemon HTTP interface
HTTP Port	Integer	The IP port used in TIBCO Rendezvous daemon HTTP interface
HTTP URL	String	URL to access the TIBCO Rendezvous daemon HTTP interface
Messages Sent	Long	Number of messages sent to the network since the last interval
Messages Received	Long	Number of messages received from the network since the last interval
Bytes Sent	Long	Number of bytes in messages sent to the network since the last interval
Bytes Received	Long	Number of bytes in messages received from the network since the last interval
Packets Sent	Long	The total number of data packets transmitted since the last interval. This total counts initial transmission and all retransmissions; that is, each packet counts separately each time it is transmitted or retransmitted.

Name	Type	Description
Retransmitted Packets	Long	Number of data packets retransmitted to other daemons since the last interval
% Retransmitted Packets	Double	Number of data packets retransmitted to other daemons since the last interval expressed as a percent. Note that it is normal to have a value of several percentage points. A value that is consistently greater than 10 percent for more than 10 to 20 minutes is abnormal.
Packets Received	Long	The number of data packets received by this daemon process since the last interval.
Missed Packets	Long	The number of data packets missed since the previous data sample. This count includes only the initial transmission of a packet from another daemon; missed retransmissions are not counted.
% Missed Packets	Double	The number of data packets missed since the last interval expressed as a percent. Note that it is normal to have a value of several percentage points. A value that is consistently greater than 10 percent for more than 10 to 20 minutes is abnormal.
Time	Double	The time at which the host status message was generated.
Messages Sent/Per Second	Double	Messages sent per second in the last interval
Messages Received/Per Second	Double	Messages received per second in the last interval
Bytes Sent/Per Second	Double	Bytes sent per second in the last interval
Bytes Received/Per Second	Double	Bytes received per second in the last interval

Name	Type	Description
Packets Sent/Per Second	Double	Packets sent per second in the last interval
Packets Received/Per Second	Double	Packets received per second in the last interval
Inbound Data Loss	Long	Inbound data loss (in packets)
Outbound Data Loss	Long	Outbound data loss (in packets)
OS Type	Integer	A code number denoting the operating system of the daemon's host computer

TibRendezvous:onRvDataLoss

Method

Purpose This method (on all platforms) subscribes to TIBCO Rendezvous data loss messages and reports their occurrence.

Type Asynchronous, INFO.

Arguments

Name	Type	Description
Service	String	TIBCO Rendezvous service
Network	String	TIBCO Rendezvous network
Daemon	String	TIBCO Rendezvous daemon

Returns

Name	Type	Description
Subject	String	Advisory message subject name
IP Address	String	IP address of the <i>other</i> daemon where this daemon is experiencing data loss. The address is returned in standard form, for example 255.100.10.30
IP Address (bin)	String	IP address of the <i>other</i> daemon where this daemon is experiencing data loss. The address is returned as a string representation of the binary digits that make up the IP address. For example, the IP address 255.100.10.30 would be presented as "11111111.1100100.00001010.000 11110". The IP address in this form allows for more flexible rulebase tests to be constructed using the Perl5PatternMatch operator of the TIBCO Hawk rulebase editor
Advisory Description	String	The description of the advisory message

TibRendezvous:onRvdDisconnectOrConnect

Method

Purpose This method (on all platforms) subscribes to TIBCO Rendezvous disconnected/connected advisory messages and reports their occurrence.

Type Asynchronous, INFO.

Arguments

Name	Type	Description
Service	String	TIBCO Rendezvous service
Network	String	TIBCO Rendezvous network
Daemon	String	TIBCO Rendezvous daemon

Returns

Name	Type	Description
Subject	String	Advisory message subject name that indicates whether the TIBCO Rendezvous daemon is disconnected or connected.

TibRendezvous:onRvLicenseExpire

Method

Purpose This method (on all platforms) subscribes to TIBCO Rendezvous license expired advisory messages and reports their occurrence.

Type Asynchronous, INFO.

Arguments

Name	Type	Description
Service	String	TIBCO Rendezvous service
Network	String	TIBCO Rendezvous network
Daemon	String	TIBCO Rendezvous daemon

Returns

Name	Type	Description
Subject	String	Subject name
Expire Time	String	License expiration time

TibRendezvous:onRvdRetransmissionSuppressed

Method

Purpose This method (on all platforms) subscribes to TIBCO Rendezvous daemon outbound retransmission suppressed advisory messages.

Type Asynchronous, INFO.

Arguments

Name	Type	Description
Service	String	TIBCO Rendezvous service
Network	String	TIBCO Rendezvous network
Daemon	String	TIBCO Rendezvous daemon

Returns

Name	Type	Description
Subject	String	Subject name
IP Address	String	IP Address of the host computer where the chronically-lossy receiver (rvd) is running
Lost	Long	The number of packets requested by the host, but not retransmitted by the sending daemon (during the interval since the last advisory of this type for the receiving host and service).

COM.TIBCO.hawk.microagent.tcpdaemon.TcpClusterStatus

Microagent



This microagent is available only when TCP Transport for TIBCO Hawk is used as the Hawk messaging transport.

Purpose The `TcpClusterStatus` microagent provides methods to monitor the health of the TCP transport cluster and Cluster Managers.

Methods

Method	Description	Page
TcpClusterStatus:getDaemonStatus	Returns list of Cluster Managers that are connected to the cluster.	96
TcpClusterStatus:getClusterStatus	Returns the following statistics for the cluster: <ul style="list-style-type: none"> Cluster Manager connect URL Total number of configured Cluster Managers Member removal strategy Minimum number of Cluster Managers to keep the cluster alive Number of reachable Cluster Managers. 	97
TcpClusterStatus:getMemberCount	Returns total number of members in the cluster (including Cluster Managers, Hawk agents, and console applications).	98
TcpClusterStatus:onMemberEvent	Subscribes for members (Cluster Managers, Hawk agents, and console applications) joining and leaving cluster.	99

TcpClusterStatus:getDaemonStatus

Method

Purpose This method (on all platforms) returns list of Cluster Managers that are connected to the cluster.

Type Open, Synchronous, IMPACT_INFO

Arguments

Name	Type	Description
Address	String	(Optional) Socket address of the Cluster Manager

Returns

Name	Type	Description
Daemon Address	String	Socket address of the Cluster Manager. The socket address is the combination of IP and port number.
Daemon Status	String	Status of the Cluster Manager

TcpClusterStatus:getClusterStatus

Method

Purpose This method (on all platforms) returns statistics for the cluster.

Type Open, Synchronous, IMPACT_INFO.

Arguments None

Returns

Name	Type	Description
Daemon URL	String	Comma-separated socket address of the Cluster Manager. The socket address is the combination of IP and port number.
Total Daemon Count	Integer	Total number of Cluster Managers configured for the cluster.
Strategy	String	The member removal strategy applied for the cluster. The values are Quorum and Majority.
Required Daemon Count	Integer	Minimum number of Cluster Managers that are required to keep the cluster alive.
Reachable Daemon Count	Integer	Number of reachable Cluster Managers in the cluster.

TcpClusterStatus:getMemberCount

Method

Purpose This method (on all platforms) returns total number of members in the cluster (including Cluster Managers, Hawk agents, and console applications).

Type Open, Synchronous, IMPACT_INFO.

Arguments None

Returns

Name	Type	Description
Daemon Count	Integer	Number of Cluster Managers in the cluster.
Agent Count	Integer	Number of Hawk agents present in the cluster.
Console Count	Integer	Number of Console applications in the cluster.
Total Count	Integer	Total number of members in the cluster.

TcpClusterStatus:onMemberEvent

Method

Purpose Subscribes for members (Cluster Managers, Hawk agents, and console applications) joining and leaving the cluster.

Type Open, Asynchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Event Type	String	Types of event, such as, members joining and leaving cluster.

Returns

Name	Type	Description
Member Address	String	Socket address of the cluster member. The socket address is the combination of IP and port number.
Member Status	Status	Status of the cluster member
Member Type	String	Type of the cluster member

COM.TIBCO.hawk.microagent.TCPMessaging

Microagent



This microagent is available only when TCP Transport for TIBCO Hawk is used as the Hawk messaging transport.

Purpose The TCPMessaging microagent provides methods to send and receive messages by using the TCP Transport for TIBCO Hawk.

Methods

Method	Description	Page
TcpMessaging:sendMessage	Sends a message by using the TCP Transport for TIBCO Hawk.	101
TcpMessaging:onMessage	Receives the message by using TCP Transport for TIBCO Hawk.	102
TcpMessaging:onNumber	Receives a TCP message with numeric value.	103
TcpMessaging:onTimeout	Returns the number of timeout intervals that have elapsed because of the arrival of the previous message or because of the start of the subscription to this method.	104

TcpMessaging:sendMessage

Method

Purpose Sends a simple message as a string by using the TCP Transport for TIBCO Hawk.

Type Synchronous, `IMPACT_ACTION`.

Arguments

Name	Type	Description
Topic	String	Message topic
Message	String	Content of the message

Returns None.

TcpMessaging:onMessage

Method

Purpose Receives a message by using the TCP Transport for TIBCO Hawk.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Topic	String	Message topic.

Returns

Name	Type	Description
Topic	String	Message topic
Message	String	Content of the message

TcpMessaging:onNumber

Method

Purpose Receives a message having a numeric value.

Type Asynchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Topic	String	Message topic.

Returns

Name	Type	Description
Topic	String	The message topic
Message	Long	The numeric data

TcpMessaging:onTimeout

Method

Purpose Returns the number of timeout intervals that have elapsed because of the arrival of the previous message or because of the start of the subscription to this method.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Topic	String	Message topic.
Timeout Interval	Long	Timeout interval in seconds

Returns

Name	Type	Description
Timeout Interval Count	Integer	The count of consecutive timeout intervals that have passed. The return value is reset to 0 (zero) when a message is received with the specified subject or topic.

Chapter 4

Microsoft Windows Microagents and Methods

The TIBCO Hawk microagents described in this chapter run on the Microsoft Windows platform in the TIBCO Hawk microagent (HMA) process. They use the diagnostic and information-management tools of the Microsoft Windows platform (the Performance Monitor, Event Log, and Registry) to link to the Microsoft Windows system. For more specific information on monitoring under Microsoft Windows, see the *TIBCO Hawk Installation, Configuration, and Administration* guide.

Topics

- [Summary of Microsoft Windows Microagents, page 106](#)
- [COM.TIBCO.hawk.hma.Performance, page 110](#)
- [COM.TIBCO.hawk.hma.Eventlog, page 147](#)
- [COM.TIBCO.hawk.hma.Registry, page 161](#)
- [COM.TIBCO.hawk.hma.Services, page 172](#)
- [COM.TIBCO.hawk.hma.Process, page 184](#)
- [COM.TIBCO.hawk.hma.Network, page 189](#)
- [COM.TIBCO.hawk.hma.System, page 192](#)

Summary of Microsoft Windows Microagents

The Microsoft Windows microagents are described as follows.

Table 3 Microsoft Windows Microagents (Sheet 1 of 4)

Method	Description
Common Methods	These methods are common to all microagents and provide release and trace-related functions. Methods are: getReleaseVersion getTraceLevel setTraceLevel getTraceParameters setTraceParameters _onUnsolicitedMsg

Table 3 Microsoft Windows Microagents (Sheet 2 of 4)

Method	Description
<p>Performance</p> <p>(Methods listed here represent a typical installation. Depending on the implementation, more or fewer Performance methods may be available. Abbreviated guidelines for use are given in this manual; more complete help is available via the TIBCO Hawk Display.)</p>	<p>Returns information from the Microsoft Windows Performance monitor. Methods are:</p> <p>Performance:System</p> <p>Performance:Processor</p> <p>Performance:Memory</p> <p>Performance:Cache</p> <p>Performance:PhysicalDisk</p> <p>Performance:LogicalDisk</p> <p>Performance:Process</p> <p>Performance:ProcessCount</p> <p>Performance:Thread</p> <p>Performance:Objects</p> <p>Performance:Redirector</p> <p>Performance:Server</p> <p>Performance:Server Work Queues</p> <p>Performance:Paging File</p> <p>Performance:Browser</p> <p>Performance:Telephony</p> <p>Performance:NBT Connection</p> <p>Performance:Network Interface</p> <p>Performance:IP</p> <p>Performance:ICMP</p> <p>Performance:TCP</p> <p>Performance:UDP</p>

Table 3 Microsoft Windows Microagents (Sheet 3 of 4)

Method	Description
Eventlog	<p>Interfaces with the Microsoft Windows event log service to detect system events, security events and application events. Methods are:</p> <p>Eventlog:getRecentApplicationEvents</p> <p>Eventlog:getRecentSystemEvents</p> <p>Eventlog:getRecentSecurityEvents</p> <p>Eventlog:onApplicationEvent</p> <p>Eventlog:onSecurityEvent</p> <p>Eventlog:onSecurityEvent</p> <p>Eventlog:logEvent</p>
Registry	<p>Allows you to retrieve and edit Microsoft Windows Registry entries. Methods are:</p> <p>Registry:setDWORD</p> <p>Registry:setString</p> <p>Registry:getDWORD</p> <p>Registry:getString</p> <p>Registry:getMultiString</p> <p>Registry:setExpandString</p> <p>Registry:enumerateKey</p> <p>Registry:CreateKey</p>

Table 3 Microsoft Windows Microagents (Sheet 4 of 4)

Method	Description
Service	<p>Allows the monitoring and management of Microsoft Windows services and device drivers. Methods are:</p> <p>Services:getServiceConfiguration</p> <p>Services:getServiceStatus</p> <p>Services:setServiceStartType</p> <p>Services:startService</p> <p>Services:stopService</p> <p>Services:pauseService</p> <p>Services:continueService</p> <p>Services:controlService</p>
Process	<p>Process:getInstanceCount</p> <p>Process:getProcess</p>
Network	<p>Used to obtain information about network identification, communications and errors. Methods are:</p> <p>Network:getConfig</p>
System	<p>Used to collect information on system-wide usage of the CPU and memory. Methods are:</p> <p>System:getSystemInfo</p> <p>System:getCpuInfo</p>

COM.TIBCO.hawk.hma.Performance

Microagent

The Performance microagent (on Microsoft Windows) retrieves information from the Microsoft Windows using Performance Data Helper (PDH) library. Each Performance microagent method corresponds to a Performance Monitor Object, and each method result field corresponds to a Performance Monitor Counter in a one-to-one mapping.

Methods listed here represent a typical installation. Abbreviated guidelines for use are given in this manual; more complete help is available via TIBCO Hawk Display.



The actual number of methods in the Performance microagent varies depending on how your machine is configured. Depending on the implementation, more or fewer Performance methods may be available. For example, if you are using SQL server, Microsoft Exchange, or Microsoft Windows Server software, additional methods may be available.

Information on the result fields of a Performance microagent method is available in the test builder screen while building a rulebase. The result fields of the methods given here are representative of the objects discovered by the microagent, and do not necessarily represent a full set of discovered objects. Information on the result fields of a Performance microagent method is also available from the Microsoft Windows Performance Monitor.



The default value for `TimeInterval` is 30 seconds. The minimum value for `TimeInterval` is 5 seconds. If the specified value is less than 5 seconds, `TimeInterval` defaults to 5 seconds.

Methods

Method	Returns	Page
Performance:System	Information on all the activities on a computer, combining all of its processors	112
Performance:Processor	Information on each processor separately	114
Performance:Memory	Information on the use of physical and virtual memory and paging	115
Performance:Cache	Information on the Microsoft Windows file cache	117

Method	Returns	Page
Performance:PhysicalDisk	Information on hard drive partitions	119
Performance:LogicalDisk	Information on each logical disk instance	121
Performance:Process	Information on one process	123
Performance:ProcessCount	The process count	125
Performance:Thread	Information on one thread	126
Performance:Objects	Summary information on all objects	127
Performance:Redirector	Information on network communications	128
Performance:Server	Information on use of computer as server	130
Performance:Server Work Queues	Information on client queues	132
Performance:Paging File	Information on page file use	134
Performance:Browser	Information on browser requests	135
Performance:Telephony	Information on telephony services	137
Performance:NBT Connection	Information on NBT connection	138
Performance:Network Interface	Information on the network interface	139
Performance:IP	Information on the IP connection	141
Performance:ICMP	Information on ICMP messages	142
Performance:TCP	Information on TCP connection	144
Performance:UDP	Information on UDP datagrams	145

Performance: System

Method

Purpose This method (on Microsoft Windows) returns general system information.

Type Asynchronous, IMPACT_INFO .

Arguments

Name	Type	Description
TimeInterval	Integer	Time interval in seconds after which data will be returned.

Returns

Name	Type
System	String
File Read Operations/sec	Double
File Write Operations/sec	Double
File Control Operations/sec	Double
File Read Bytes/sec	Double
File Write Bytes/sec	Double
File Control Bytes/sec	Double
Context Switches/sec	Double
System Calls/sec	Double
File Data Operations	Double
System Up Time	Double
Processor Queue Length	Double
Processes	Double
Threads	Double
Alignment Fixups/sec	Double
Exception Dispatches/sec	Double

Name (Cont'd)	Type
Floating Emulations/sec	Double
% Registry Quota In Use	Double

Performance:Processor

Method

Purpose This method (on Microsoft Windows) returns information on the operation of each processor in the computer.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Processor	String	Processor instances are usually numbers, such as 0, 1, 2, etc. Empty argument returns all processors.
TimeInterval	Integer	Time interval in seconds after which data will be returned

Returns

Name	Type
Instance	String. Table is indexed on the Instance field.
% Processor Time	Double
% User Time	Double
% Privileged Time	Double
Interrupts/sec	Double
% DPC Time	Double
% Interrupt Time	Double
DPCs Queued/sec	Double
DPC Rate	Double
DPC Bypasses/sec	Double
APC Bypasses/sec	Double

Performance:Memory

Method

Purpose This method (on Microsoft Windows), returns info on the use of physical (real) and virtual memory.

Type Asynchronous, IMPACT_INFO.

Arguments

Name	Type	Description
TimeInterval	Integer	Time interval in seconds after which data is returned.

Returns

Name	Type
Instance	String. Table is indexed on the Instance
Available Bytes	Double
Committed Bytes	Double
Commit Limit	Double
Page Faults/sec	Double
Write Copies/sec	Double
Transition Faults/sec	Double
Cache Faults	Double
Demand Zero Faults	Double
Pages/sec	Double
Pages Input/sec	Double
Page Reads/sec	Double
System Code Resident Bytes	Double
System Driver Total Bytes	Double
System Driver Resident Bytes	Double
System Cache Resident Bytes	Double

Name	Type
% Committed Bytes In Use	Double

Performance:Cache

Method

Purpose This method (on Microsoft Windows) returns information on the Microsoft Windows file cache.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
TimeInterval	Integer	Time interval in seconds after which data will be returned.

Returns

Name	Type
Instance	String. Table is indexed the on Instance.
Data Maps/sec	Double
Sync Data Maps/sec	Double
Async Data Maps/sec	Double
Data Map Hits %	Double
Data Map Pins/sec	Double
Pin Reads/sec	Double
Sync Pin Reads/sec	Double
Async Pin Reads/sec	Double
Pin Read Hits %	Double
Copy Reads/sec	Double
Sync Copy Reads/sec	Double
Async Copy Reads/sec	Double
Copy Read Hits %	Double
MDL Reads/sec	Double

Name (Cont'd)	Type (Cont'd)
Sync MDL Reads/sec	Double
Async MDL Reads/sec	Double
MDL Read Hits %	Double
Read Aheads/sec	Double
Fast Reads/sec	Double
Sync Fast Reads/sec	Double
Async Fast Reads/sec	Double
Fast Read Resource Misses/sec	Double
Fast Read Not Possibles/sec	Double
Lazy Write Flushes/sec	Double
Lazy Write Pages/sec	Double
Data Flushes/sec	Double
Data Flush Pages/sec	Double

Performance:PhysicalDisk

Method

Purpose This method (on Microsoft Windows) returns information on each hard drive instance connected to the computer.

Type Asynchronous, IMPACT_INFO .

Arguments

Name	Type	Description
Physical Disk	String	Physical disks are usually denoted using numbers: 0, 1, 2, and so on. Empty argument returns all physical disks.
TimeInterval	Integer	Time interval in seconds after which data will be returned.

Returns

Name	Type
Instance	String. Table is indexed on the Instance field.
Current Disk Queue Length	Double
% Disk Time	Double
Avg Disk Queue Length	Double
% Disk Read Time	Double
Avg. Disk Read Queue Length	Double
% Disk Write Time	Double
Avg. Disk Write Queue Length	Double
Avg. Disk sec/Transfer	Double
Avg. Disk sec/Read	Double
Avg. Disk sec/Write	Double
Disk Transfers/sec	Double
Disk Reads/sec	Double
Disk Writes/sec	Double

Name	Type
Disk Bytes/sec	Double
Avg. Disk Bytes/Transfer	Double
Avg. Disk Bytes/Read	Double
Avg. Disk Bytes/Write	Double

Performance:LogicalDisk

Method

Purpose This method (on Microsoft Windows) returns information on each logical disk instance connected to the computer.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Logical Disk	String	Logical Disk. A list of available disk selections is provided. Empty argument returns all logical disks.
TimeInterval	Integer	Time interval in seconds after which data will be returned

Returns

Name	Type
Instance	String. Table is indexed on the Instance.
% Free Space	Double
Free Megabytes	Double
Current Disk Queue Length	Double
% Disk Time	Double
Avg. Disk Queue Length	Double
% Disk Read Time	Double
Avg. Disk Read Queue Length	Double
% Disk Write Time	Double
Avg. Disk Write Queue Length	Double
Avg. Disk sec/Transfer	Double
Avg. Disk sec/Read	Double
Avg. Disk sec/Write	Double
Disk Transfers/sec	Double

Name	Type
Disk Reads/sec	Double
Disk Writes/sec	Double
Disk Bytes/sec	Double
Avg. Disk Bytes/Transfer	Double
Avg Disk Bytes/Read	Double
Avg. Disk Bytes/Write	Double

Performance:Process

Method

Purpose This method (on Microsoft Windows) returns information on a process (a program or group of threads).

Remarks The methods `Performance:Process` and `Process:getProcess` can seem similar, but return different data. `Performance:Process` returns all Instances of `Process` Object from the Performance Data Helper (PDH) library.

Invoking `Performance:process` without any arguments retrieves all `Process` instances in effect at that point in time. In contrast, invoking `Process:getProcess` without any arguments returns all currently running processes from the `PSAPI.lib`. This method only lists processes that have the right to access this library. Do not confuse these methods.

For a full description of this method and accompanying help text, refer to the help text provided by method invocation.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Process	String	The exact process name or a regular expression using the <code>Perl5PatternMatch</code> syntax. Empty argument string returns all processes.
TimeInterval	Integer	Time interval in seconds after which data will be returned

Returns

Name	Type
Instance	String. Table is indexed on the Instance field.
% Processor Time	Double
% User Time	Double
% Privileged Time	Double
Virtual Bytes Peak	Double
Virtual Bytes	Double
Page Faults/sec	Double

Name	Type
Working Set Peak	Double
Page File Bytes Peak	Double
Page File Bytes	Double
Private Bytes	Double
Thread Count	Double
Priority Base	Double
Elapsed Time	Double
ID Process	Double
Pool Paged Bytes	Double
Pool Nonpaged Bytes	Double
Handle Count	Double

Performance:ProcessCount

Method

- Purpose** This method (on Microsoft Windows) returns the process count.
- Remarks** This method does not derive from the Microsoft Windows performance monitor.
- Type** Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Process Name	String	The exact process name or a regular expression using the <code>Perl5PatternMatch</code> syntax. Empty argument returns all processes currently running.

Returns

Name	Type
Process Name	String
Instance Count	Integer

Performance:Thread

Method

Purpose This method (on Microsoft Windows) returns information for a specified thread. If a thread is not specified, information on all threads is returned.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Thread	String	The exact name of the thread or a regular expression using the <code>Perl5PatternMatch</code> syntax. Empty argument string returns all threads.
TimeInterval	Integer	Time interval in seconds after which data will be returned

Returns

Name	Type
Instance	String. Table is indexed on the Instance field.
% Processor Time	Double
% User Time	Double
% Privileged Time	Double
Context Switches/sec	Double
Elapsed Time	Double
Priority Base	Double
Start Address	Double
Thread State	Double
Thread Wait Reason	Double
ID Process	Double
ID Thread	Double

Performance:Objects

Method

Purpose This method (on Microsoft Windows) returns summary information on all objects currently in existence on the computer, including processes and threads.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
TimeInterval	Integer	Time interval in seconds after which data will be returned.

Returns

Name	Type
Instance	String. Table is indexed on the Instance field.
Process	Double
Threads	Double
Events	Double
Semaphores	Double
Mutexes	Double
Sections	Double

Performance:Redirector

Method

Purpose The Redirector manages network connections that originate from a computer. This method (on Microsoft Windows) returns information on operations of the redirector.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
TimeInterval	Integer	Time interval in seconds after which data will be returned.

Returns

Name	Type
Instance	String. Table is indexed on the Instance field.
Bytes Total/sec	Double
File Data Operations/sec	Double
Packets/sec	Double
Bytes Received/sec	Double
Packets Received/sec	Double
Read Bytes Paging/sec	Double
Read Bytes Non-Paging/sec	Double
Read Bytes Cache/sec	Double
Read Bytes Network/sec	Double
Bytes Transmitted/sec	Double
Packets Transmitted/sec	Double
Write Bytes Paging/sec	Double
Write Bytes Non-Paging/sec	Double
Write Bytes Cache/sec	Double

Name	Type
Write Bytes Network/sec	Double
File Read Operations/sec	Double
Read Packets/sec	Double
Reads Large/sec	Double
Read Packets Small/sec	Double
File Write Operations/sec	Double
Write Operations Random/sec	Double
Write Packets/sec	Double
Writes Large/sec	Double
Write Packets Small/sec	Double
Writes Denied/sec	Double
Network Errors/sec	Double
Server Sessions	Double
Server Reconnects	Double
Connects Core	Double
Connects Lan Manager 2.0	Double
Connects Lan Manager 2.1	Double
Connects Windows NT	Double
Server Disconnects	Double
Server Sessions Hung	Double
Current Commands	Double

Performance:Server

Method

Purpose This method (on Microsoft Windows) returns information on the process that interfaces services from a computer to network servers.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
TimeInterval	Integer	Time interval in seconds after which data will be returned.

Returns

Name	Type
Instance	String. Table is indexed on the Instance field.
Bytes Total/sec	Double
Bytes Received/sec	Double
Bytes Transmitted/sec	Double
Sessions Timed Out	Double
Sessions Errored Out	Double
Sessions Logged Off	Double
Sessions Forced Off	Double
Errors Logon	Double
Errors Access Permissions	Double
Errors Granted Access	Double
Errors System	Double
Blocking Requests Rejected	Double
Work Item Shortages	Double
Files Opened Total	Double

Name	Type
Files Open	Double
Server Sessions	Double
File Directory Searches	Double
Pool Nonpaged Bytes	Double
Pool Nonpaged Failures	Double
Pool Nonpaged Peak	Double
Pool Paged Bytes	Double
Pool Paged Failures	Double
Pool Paged Peak	Double
Content Blocks Queued/sec	Double
Logon/sec	Double
Logon Total	Double

Performance:Server Work Queues

Method

Purpose This method (on Microsoft Windows) returns information on queues that form when client computers are interacting with this computer as a server.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Server Work Queues	String	Server Work Queues. Empty argument returns all queues.
TimeInterval	Integer	Time interval in seconds after which data will be returned.

Returns

Name	Type
Instance	String. Table is indexed on the Instance field.
Queue Length	Double
Available Threads	Double
Available Work Items	Double
Borrowed Work Items	Double
Work Item Shortages	Double
Current Clients	Double
Bytes Received/sec	Double
Bytes Sent/sec	Double
Bytes Transferred/sec	Double
Read Operations/sec	Double
Read Bytes/sec	Double
Write Operations/sec	Double
Write Bytes/sec	Double

Name	Type
Total Bytes/sec	Double
Total Operations/sec	Double
Context Blocks Queued/sec	Double

Performance:Paging File

Method

Purpose This method (on Microsoft Windows) returns information on the use of Microsoft Windows paging files to provide virtual memory.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Paging File	String	Page file name. Exact match. Empty argument string returns information on all paging files.
TimeInterval	Integer	Time interval in seconds after which data will be returned.

Returns

Name	Type
Instance	String. Table is indexed on the Instance field.
% Usage	Double
% Usage Peak	Double

Performance:Browser

Method

Purpose This method (on Microsoft Windows) returns information on the use of a computer as a server for browser requests.

Type Asynchronous, IMPACT_INFO .

Arguments

Name	Type	Description
TimeInterval	Integer	Time interval in seconds after which data will be returned.

Returns

Name	Type
Instance	String. Table is indexed on the Instance field.
Announcements Server/sec	Double
Announcements Domain/sec	Double
Announcements Total/sec	Double
Election Packets/sec	Double
Mailslot Writes/sec	Double
Server List Requests/sec	Double
Enumerations Server/sec	Double
Enumerations Domain/sec	Double
Enumerations Other/sec	Double
Enumerations Total/sec	Double
Missed Server Announcements	Double
Missed Mailslot Datagrams	Double
Missed Server List Requests	Double
Server Announce Allocations Failed/sec	Double

Name	Type
Mailslot Allocations Failed	Double
Mailslot Receives Failed	Double
Mailslot Writes Failed	Double
Mailslot Opens Failed/sec	Double
Duplicate Master Announcements	Double
Illegal Datagrams/sec	Double

Performance:Telephony

Method

Purpose This method (on Microsoft Windows) returns information on the use of this computer for interacting with telephone devices and maintaining telephone services such as answering services and automated calling services.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
TimeInterval	Integer	Time interval in seconds after which data will be returned.

Returns

Name	Type	Description
Instance	String	Instance. Table is indexed on the Instance field.
Lines	Double	Lines
Telephone Devices	Double	Telephone Devices
Active Lines	Double	Active Lines
Active Telephones	Double	Active Telephones
Outgoing Calls/sec	Double	Outgoing calls per second
Incoming Calls/sec	Double	Incoming calls per second
Client Apps	Double	Client Applications
Current Outgoing Calls	Double	Current outgoing calls

Performance:NBT Connection

Method

Purpose This method (on Microsoft Windows) returns information for data sent over an NBT (NetBIOS over TCP/IP) connection.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
NBT Connection	String	NBT Connection. The name of the remote computer that is connected to the NBT connection connects.
TimeInterval	Integer	Time interval in seconds after which data will be returned.

Returns

Name	Type
Instance	String. Table is indexed on the Instance field.
Bytes Received/sec	Double
Bytes Sent/sec	Double
Total Bytes/sec	Double

Performance:Network Interface

Method

Purpose This method (on Microsoft Windows) returns statistics regarding the specified network interface. If a network interface is not specified, statistics for all network interfaces are returned.

Type Asynchronous, IMPACT_INFO .

Arguments	Name	Type	Description
	Network Interface	String	Network Interface.
	TimeInterval	Integer	Time interval in seconds after which data will be returned.

Returns	Name	Type
	Instance	String. Table is indexed on Instance.
	Bytes Total/sec	Double
	Packets/sec	Double
	Packets Received/sec	Double
	Packets Sent/sec	Double
	Current Bandwidth	Double
	Bytes Received/sec	Double
	Packets Received Unicast/sec	Double
	Packets Received Non-Unicast/sec	Double
	Packets Received Discarded	Double
	Packets Received Errors	Double
	Packets Received Unknown	Double
	Bytes Sent/sec	Double
	Packets Sent Unicast/sec	Double

Name	Type
Packets Sent Non-Unicast/sec	Double
Packets Outbound Discarded	Double
Packets Outbound Errors	Double
Output Queue Length	Double

Performance:IP

Method

Purpose This method (on Microsoft Windows) returns statistics for all IP processes.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
TimeInterval	Integer	Time interval in seconds after which data is returned.

Returns

Name	Type
Instance	String. Table is indexed on Instance.
Datagrams/sec	Double
Datagrams Received/sec	Double
Datagrams Received Header Errors	Double
Datagrams Received Address Errors	Double
Datagrams Received Unknown	Double
Datagrams Received Discarded	Double
Datagrams Received Delivered/sec	Double
Datagrams Sent/sec	Double
Datagrams Outbound Discarded	Double
Datagrams Outbound No Route	Double
Fragments Received/sec	Double
Fragments Re-assembled/sec	Double
Fragment Re-assembly Failures	Double
Fragmented Datagrams/sec	Double
Fragmentation Failures	Double
Fragments Created/sec	Double

Performance:ICMP

Method

Purpose This method (on Microsoft Windows) returns statistics for the ICMP processes.

Type Asynchronous, IMPACT_INFO .

Arguments

Name	Type	Description
TimeInterval	Integer	Time interval in seconds after which data will be returned.

Returns

Name	Type
Instance	String. Table is indexed on Instance.
Messages/sec	Double
Messages Received/sec	Double
Messages Received Errors	Double
Received Destination Unreachable	Double
Received Time Exceeded	Double
Received Parameter Problem	Double
Received Source Quench	Double
Received Redirect/sec	Double
Received Echo/sec	Double
Received Timestamp/sec	Double
Received Timestamp Reply/sec	Double
Received Address Mask	Double
Received Address Mask Reply	Double
Messages Sent/sec	Double
Messages Outbound Errors	Double

Name	Type
Sent Destination Unreachable	Double
Sent Time Exceeded	Double
Sent Parameter Problem	Double
Sent Source Quench	Double
Sent Redirect/sec	Double
Sent Echo/sec	Double
Sent Echo Reply/sec	Double
Sent Timestamp/sec	Double
Sent Timestamp Reply/sec	Double
Sent Address Mask	Double
Sent Address Mask Reply	Double

Performance:TCP

Method

Purpose This method (on Microsoft Windows) returns statistics for the TCP processes.

Type Asynchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
TimeInterval	Integer	Time interval in seconds after which data will be returned.

Returns

Name	Type
Instance	String. Table is indexed on Instance.
Segments/sec	Double
Connections Established	Double
Connections Active	Double
Connections Passive	Double
Connection Failures	Double
Connections Reset	Double
Segments Received/sec	Double
Segments Sent/sec	Double
Segments Retransmitted/sec	Double

Performance:UDP

Method

Purpose This method (on Microsoft Windows) returns statistics for the UDP processes.

Type Asynchronous, IMPACT_INFO .

Arguments

Name	Type	Description
TimeInterval	Integer	Time interval in seconds after which data will be returned.

Returns

Name	Type
Instance	String. Table is indexed on Instance.
Datagrams/sec	Double
Datagrams Received/sec	Double
Datagrams No Port/sec	Double
Datagrams Received Errors	Double
Datagrams Sent/sec	Double

Performance:restart

Method

- Purpose** This method causes the Performance microagent to restart in order to pick up Windows performance objects that were added after the Performance microagent was originally started. The Performance microagent snapshots the Windows performance objects at startup and announces methods for each object to the Hawk agent. Performance objects dynamically added to the system after the Performance microagent has announced are not included. Certain applications, such as Microsoft SQL Server, dynamically add performance objects when they are running and remove them when they are not running. Hawk rulebases can now be created that start applications, such as Microsoft SQL Server, restart the Performance microagent to pick up the new performance objects, and then load rulebases designed to monitor these new performance objects.
- Type** Synchronous, `IMPACT_ACTION`.
- Arguments** None.
- Returns** None.
- Remarks** The restart method only restarts the Performance microagent. All other HMA microagents and the HMA itself are unaffected.

COM.TIBCO.hawk.hma.Eventlog

Microagent

Purpose The Eventlog microagent (on Microsoft Windows) interfaces with the Microsoft Windows event log service, which records three types of events: system events, security events and application events. You can use this microagent's methods to respond to events as they happen or to retrieve events in any of the three event logs.

Methods	Method	Description	Page
	Eventlog:getRecentApplicationEvents	Returns most recent events in the application event log	148
	Eventlog:getRecentSystemEvents	Returns most recent events in the system event log	150
	Eventlog:getRecentSecurityEvents	Returns most recent events in the security event log	152
	Eventlog:onApplicationEvent	Returns event just added to the application event log	154
	Eventlog:onSecurityEvent	Returns event just added to the security event log	156
	Eventlog:onSystemEvent	Reports events posted to the Microsoft Windows event log	158
	Eventlog:logEvent	Adds the event to the event log	160

Eventlog:getRecentApplicationEvents

Method

Purpose This method (on Microsoft Windows) provides the 30 most recent events in the Microsoft Windows application event log.

Remarks This method should not be used as a data source for a rule, because subsequent invocation returns the same data. Use `onApplicationEvent` instead.

Composite event identifiers are represented in the result descriptions by the use of double colons between event parameters (`source::event`).

Type Synchronous, `IMPACT_INFO`.

Arguments None.

Returns

Name	Type	Description
Date	String	Date the event was generated
Time	String	Time the event was generated
Record	Integer	Record number of the event. Table is indexed on Record
Event	Integer	Source-specific ID of the event
Event Category	Integer	Source-specific category of the event. The category may be any value between zero and 65535 where zero represents "no category."
Type	String	Event type. Can be one of the following: <ul style="list-style-type: none"> Information Warning Error Audit-Success Audit-Failure
Source	String	Event source

Name	Type	Description
Strings	String	Message template merge strings. These strings are merged into the associated message template to form the complete textual description of the event. These strings are always returned even when the associated message template cannot be found.
Text	String	Complete textual description of the event

Eventlog:getRecentSystemEvents

Method

Purpose This method (on Microsoft Windows) provides the 30 most recent in the Microsoft Windows system event log.

Remarks This method should not be used as a data source for a rule, because subsequent invocation returns the same data. For data sources for rules, use [Eventlog:onSecurityEvent](#).

Composite event identifiers are represented in the result descriptions by the use of double colons between event parameters (source : : event).

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Date	String	Date the event was generated
Time	String	Time the event was generated
Record	Integer	Record number of the event. Table is indexed on Record
Event	Integer	Source-specific ID of the event
Event Category	Integer	Source-specific category of the event. The category may be any value between zero and 65535 where zero represents "no category."
Type	String	Event type. Can be one of the following: <ul style="list-style-type: none"> • Information • Warning • Error • Audit-Success • Audit-Failure
Source	String	Event source

Name	Type	Description
Strings	String	Message template merge strings. These strings are merged into the associated message template to form the complete textual description of the event. These strings are always returned even when the associated message template cannot be found.
Text	String	Complete textual description of the event

Eventlog:getRecentSecurityEvents

Method

Purpose This method (on Microsoft Windows) provides the 30 most recent events in the Microsoft Windows security event log.

Remarks This method should not be used as a data source for a rule, because subsequent invocation returns the same data. For data sources for rules, use [Eventlog:onSecurityEvent](#) instead.

Composite event identifiers are represented in the result descriptions by the use of double colons between event parameters (source : :event).

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Date	String	Date the event was generated
Time	String	Time the event was generated
Record	Integer	Record number of the event. Table is indexed on Record
Event	Integer	Source-specific ID of the event
Event Category	Integer	Source-specific category of the event. The category may be any value between zero and 65535 where zero represents "no category."
Type	String	Event type. Can be one of the following: <ul style="list-style-type: none"> • Information • Warning • Error • Audit-Success • Audit-Failure
Source	String	Event source

Name	Type	Description
Strings	String	Message template merge strings. These strings are merged into the associated message template to form the complete textual description of the event. These strings are always returned even when the associated message template cannot be found.
Text	String	Complete textual description of the event

Eventlog:onApplicationEvent

Method

Purpose This method (on Microsoft Windows) reports events as they are asynchronously posted to the Microsoft Windows application event log.

Remarks Composite event identifiers are represented in the result descriptions by the use of double colons between event parameters (source : :event).

Type Asynchronous IMPACT_INFO .

Arguments

Name	Type	Description
source	String	Event source

Returns

Name	Type	Description
Date	String	Date the event was generated
Time	String	Time the event was generated
Record	Integer	Record number of the event. Table is indexed on Record
Event	Integer	Source-specific ID of the event
Event Category	Integer	Source-specific category of the event. The category may be any value between zero and 65535 where zero represents "no category."
Type	String	Event type. Can be one of the following: <ul style="list-style-type: none"> • Information • Warning • Error • Audit-Success • Audit-Failure
Source	String	Event source.

Name	Type	Description
Strings	String	Message template merge strings. These strings are merged into the associated message template to form the complete textual description of the event. These strings are always returned even when the associated message template cannot be found.
Text	String	Complete textual description of the event

Eventlog:onSecurityEvent

Method

Purpose This method (on Microsoft Windows) reports events as they are asynchronously posted to the Microsoft Windows security event log.

Remarks Use this method as a data source for rules. Composite event identifiers are represented in the result descriptions by the use of double colons between event parameters (source::event).

Type Asynchronous IMPACT_INFO.

Arguments

Name	Type	Description
source	String	Event source.

Returns

Name	Type	Description
Date	String	Date the event was generated
Time	String	Time the event was generated
Record	Integer	Record number of the event. Table is indexed on Record
Event	Integer	Source-specific ID of the event
Event Category	Integer	Source-specific category of the event. The category may be any value between zero and 65535 where zero represents "no category."
Type	String	Event type. Can be one of the following: <ul style="list-style-type: none"> • Information • Warning • Error • Audit-Success • Audit-Failure
Source	String	Event source

Name	Type	Description
Strings	String	Message template merge strings. These strings are merged into the associated message template to form the complete textual description of the event. These strings are always returned even when the associated message template cannot be found.
Text	String	Complete textual description of the event

Eventlog:onSystemEvent

Method

Purpose This method (on Microsoft Windows) reports events as they are being asynchronously posted to the Microsoft Windows system event log.

Remarks Composite event identifiers are represented in the result descriptions by the use of double colons between event parameters (source : :event).

Type Asynchronous IMPACT_INFO.

Arguments

Name	Type	Description
source	String	Event source

Returns

Name	Type	Description
Date	String	Date the event was generated
Time	String	Time the event was generated
Record	Integer	Record number of the event. Table is indexed on Record
Event	Integer	Source-specific ID of the event
Event Category	Integer	Source-specific category of the event. The category may be any value between zero and 65535 where zero represents "no category."
Type	String	Event type. Can be one of the following: <ul style="list-style-type: none"> • Information • Warning • Error • Audit-Success • Audit-Failure
Source	String	Event source

Name	Type	Description
Strings	String	Message template merge strings. These strings are merged into the associated message template to form the complete textual description of the event. These strings are always returned even when the associated message template cannot be found.
Text	String	Complete textual description of the event

Eventlog:logEvent

Method

Purpose This method adds Event to the Microsoft Windows Application Event Log, using source name TIBHawk.

Remarks The event type is indicated by its event ID in the Microsoft Windows Event Log.

Type Synchronous, IMPACT_ACTION

Arguments

Name	Type	Description
Event Type	String	The type of event, indicated by the event ID in the Microsoft Windows Event Log: <ul style="list-style-type: none"> Information (event ID is 16) Error (event ID is 17) Warning (event ID is 18)
Event Category	Integer	Specifies the source-specific category of this event. The category may be any value between zero and 65535 where zero represents "no category."
Description	String	Complete textual description of this event

Returns None.

COM.TIBCO.hawk.hma.Registry

Microagent

The `Registry` microagent (on Microsoft Windows) allows you to interact with the Microsoft Windows Registry.

Methods	Method	Description	Page
	Registry:setDWORD	Creates or overwrites a DWORD entry in the Registry	162
	Registry:setString	Creates or overwrites a string entry in the Registry	163
	Registry:getDWORD	Returns a DWORD value for a key and entry name in the Registry	164
	Registry:getString	Returns a string value for a key and entry name in the Registry	165
	Registry:getMultiString	Returns a multi-line string value for a key and entry name in the Registry	166
	Registry:setExpandString	Creates or overwrites an expandable string entry in the Registry	167
	Registry:enumerateKey	Enumerates specified key	168
	Registry:CreateKey	Adds key to Registry	169
	Registry:setQWORD	Creates or overwrites a QWORD entry in the Registry	170
	Registry:getQWORD	Returns a QWORD value for a key and entry name in the Registry	171

Registry:setDWORD

Method

Purpose This method (on Microsoft Windows) writes a DWORD entry to the Registry, given a key and value.

Remarks This method works exactly as if you were using the Microsoft Windows Registry editor and chose **Add Value**, selecting the data type REG_DWORD. The `Entry` argument of the method corresponds to the `Value Name` you enter, and the `Value` argument corresponds to what you type in the DWORD editor.

The `Key` must exist in the Registry. If the `Entry` argument is not already in the Registry, a new entry is created and its value set to `Value`. If the `Entry` argument exists in the Registry for the specified key, the value of the entry is set to the new value.

Type Synchronous, `IMPACT_ACTION`.

Arguments

Name	Type	Description
Key	String	Specifies key to change
Entry	String	Specifies entry to add or change
Value	Integer	Value to set for a specified entry

Returns None.

Registry:setString

Method

Purpose This method (on Microsoft Windows) writes a string entry to the Registry.

Remarks This method works exactly as if you were using the Microsoft Windows Registry editor and chose **Add Value**, selecting the data type REG_SZ. The `Entry` argument of the method corresponds to the `Value Name` you enter, and the `Value` argument corresponds to what you type in the String editor.

The key must exist in the Registry. If the `Entry` argument is not already in the Registry, a new entry is created and its value set to `Value`. If the `Entry` argument exist in the Registry for the specified key, the value of the entry is set to the new value.

Type Synchronous, IMPACT_ACTION.

Arguments

Name	Type	Description
Key	String	Specifies key to change
Entry	String	Specifies entry to add or change
Value	Integer	Value to set for a specified entry

Returns None.

Registry:getDWORD

Method

Purpose This method (on Microsoft Windows) returns a DWORD from the Microsoft Windows Registry (of the type REG_DWORD), given a key and entry.

Remarks If no entry is found for the key and entry name, an error is returned.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Key	String	Specifies key name to look up
Entry	String	Specifies name of entry to look up

Returns

Name	Type	Description
Value	Integer	Specifies value of entry

Registry.getString

Method

Purpose This method (on Microsoft Windows) returns a string (of the type REG_SZ) from the Microsoft Windows Registry by looking up the key and entry names you supply.

Remarks The method processes REG_EXPAND_SZ strings by expanding any environment variable substitutions.

If no entry is found for the key and entry name, an error is returned.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Key	String	Key to look up
Entry	String	Entry to look up

Returns

Name	Type	Description
Value	String	Value found

Registry:getMultiString

Method

Purpose This method (on Microsoft Windows) returns a multi string entry from the Registry (of the type REG_MULTI_SZ) by looking up key and entry names you supply.

Remarks The result is a column table of values, each row being a line of the multi string entry. If no entry is found for the key and entry name, an error is returned.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Key	String	Key to look up
Entry	String	Entry to look up

Returns

Name	Type	Description
Value	String	Value found. (Returns values as a table.) Table is indexed on Value
Index	String	The ordinal position of the individual strings in the multistring(1-n). Guarantees unique indexes in cases where the string values of the multistring are not unique.

Registry:setExpandString

Method

Purpose This method (on Microsoft Windows) writes a string with embedded environment variables (of the type REG_EXPAND_SZ) to the Microsoft Windows Registry. Embedded variables are in the form %VARIABLE% and are expanded by Microsoft Windows to fill out the string when it is needed.

Remarks The `Key` entry must exist in the Registry. If the `Entry` argument is not already in the Registry, a new entry is created and its value set. If the `Entry` argument exists in the Registry for the `Key` specified, the value of the entry is set to the new value.

Type Synchronous, IMPACT_ACTION.

Arguments

Name	Type	Description
Key	String	Key to change
Entry	String	Entry to change
Value	String	Value to set

Returns None.

Registry:enumerateKey

Method

Purpose Enumerate specified key. This method returns a table of the top-level key entries, with one row per entry.

Remarks The method processes REG_EXPAND_SZ strings by expanding any environment variable substitutions.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Key	String	Specifies key name

Returns

Name	Type	Description
Entry	String	Specifies name of returned entries
Type	String	Specifies type of entry. Entry can be of type key, REG_DWORD, REG_SZ, REG_MULTI_SZ, or REG_EXPAND_SZ.
Value	String	Specifies value of entry

Registry:CreateKey

Method

Purpose Adds a key to the Registry. This function adds a subkey to the Registry under a specified key.

Type Synchronous, IMPACT_ACTION.

Arguments

Name	Type	Description
RootKey	String	Parent key. Should be chosen from the following options: HKEY_CLASSES_ROOT HKEY_CURRENT_CONFIG HKEY_CURRENT_USER HKEY_LOCAL_MACHINE HKEY_USERS
Subkey	String	Subkey to be created

Returns None.

Registry:setQWORD

Method

Purpose This method (on Microsoft Windows) writes a QWORD entry to the Registry, given a key and value.

Remarks This method works exactly as if you were using the Microsoft Windows Registry editor and chose **Add Value**, selecting the data type REG_QWORD. The `Entry` argument of the method corresponds to the `Value Name` you enter, and the `Value` argument corresponds to what you type in the QWORD editor.

The `Key` must exist in the Registry. If the `Entry` argument is not already in the Registry, a new entry is created and its value set to `Value`. If the `Entry` argument exists in the Registry for the specified key, the value of the entry is set to the new value.

Type Synchronous, `IMPACT_ACTION`.

Arguments

Name	Type	Description
Key	String	Specifies key to change
Entry	String	Specifies entry to add or change
Value	Integer	Value to set for a specified entry

Returns None.

Registry:getQWORD

Method

Purpose This method (on Microsoft Windows) returns a QWORD from the Microsoft Windows Registry (of the type REG_QWORD), given a key and entry.

Remarks If no entry is found for the key and entry name, an error is returned.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Key	String	Specifies key name to look up
Entry	String	Specifies name of entry to look up

Returns

Name	Type	Description
Value	Integer	Specifies value of entry

COM.TIBCO.hawk.hma.Services

Microagent

Purpose This microagent provides for the monitoring and management of Microsoft Windows services and device drivers.

Methods	Method	Description	Page
	Services:getServiceConfiguration	This method gets the current configuration of the specified Microsoft Windows service(s).	173
	Services:getServiceStatus	This method gets the current status of the specified Microsoft Windows service(s).	176
	Services:setServiceStartType	This method sets the current start type of the specified Microsoft Windows service.	178
	Services:startService	This method starts the specified Microsoft Windows service.	179
	Services:stopService	This method stops the specified Microsoft Windows service.	180
	Services:pauseService	This method pauses the specified Microsoft Windows service.	181
	Services:continueService	This method continues the specified Microsoft Windows service.	182
	Services:controlService	This method invokes the specified user-defined service control command for the specified Microsoft Windows service.	183

Services:getServiceConfiguration

Method

Purpose This method gets the current configuration of the specified Microsoft Windows service(s).

Remarks If the Service parameter is blank or set to ALL then all installed Microsoft Windows services are returned. If a specific service is specified, only that Microsoft Windows service is returned

Type Synchronous, IMPACT_INFO.

Arguments	Name	Type	Description
	Service	String	Initial name of Microsoft Windows service

Returns	Name	Type	Description
	Service	String	Microsoft Windows service internal name. *Table is indexed on Service.
	Display Name	String	Microsoft Windows external display name
	Service Type	String	<p>The type of the Microsoft Windows service. The service types are:</p> <p>OWN_PROCESS: Indicates a Win32 service that runs in its own process.</p> <p>INTERACTIVE_OWN_PROCESS: Indicates a Win32 service that runs in its own process and that can interact with the desktop.</p> <p>SHARE_PROCESS: Indicates a Win32 service that shares a process with other services.</p> <p>INTERACTIVE_SHARE_PROCESS: Indicates a Win32 service that shares a process with other services and that can interact with the desktop.</p> <p>KERNEL_DRIVER: Indicates a device driver.</p> <p>FILE_SYSTEM_DRIVER: Indicates a file system driver.</p>

Name	Type	Description
Service Start Type	String	<p>The start type of the Microsoft Windows service. The service types are:</p> <p>DEMAND_START: Specifies a device driver or Win32 service started by the service control manager when a process calls the StartService function (e.g. Services Control Panel Applet).</p> <p>AUTO_START: Specifies a device driver or Win32 service started by the service control manager automatically during system startup.</p> <p>DISABLED: Specifies a device driver or Win32 service that can no longer be started.</p> <p>BOOT_START: Specifies a device driver started by the system loader.</p> <p>SYSTEM_START: Specifies a device driver started by the IoInitSystem function.</p>
Service Error Control	String	<p>Specifies the severity of the error if this Microsoft Windows service fails to start during startup, and determines the action taken by the startup program if failure occurs. The error controls are:</p> <p>IGNORE: The startup (boot) program logs the error but continues the startup operation.</p> <p>NORMAL: The startup program logs the error and displays a message box pop-up but continues the startup operation.</p> <p>SEVERE: The startup program logs the error. If the last-known good configuration is being started, the startup operation continues. Otherwise, the system is restarted with the last known good configuration.</p> <p>CRITICAL: The startup program logs the error, if possible. If the last-known good configuration is being started, the startup operation fails. Otherwise, the system is restarted with the last-known good configuration.</p>
Binary Path	String	The fully qualified path to the Microsoft Windows service binary file

Name	Type	Description
Load Order Group	String	The names of the load ordering group of which this Microsoft Windows service is a member. If the field is blank the service does not belong to a group.
Tag ID	Integer	Specifies a unique tag value for this Microsoft Windows service in the group specified by the Load Order Group value. A value of zero indicates that the service has not been assigned a tag.
Account	String	The account name, in the form of DomainName\Username, which the log-on name of the service process when it is run.

Services:getServiceStatus

Method

Purpose This method gets the current status of the specified Microsoft Windows service(s). If the Service parameter is blank or set to ALL then all installed Microsoft Windows services are returned. If a specific service is specified then only that Microsoft Windows service is returned.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Service	String	Internal name of Microsoft Windows service

Returns

Name	Type	Description
Service	String	Microsoft Windows service internal name. *Table is indexed on Service.
Display Name	String	Microsoft Windows external display name
Service Type	String	<p>The type of the Microsoft Windows service. The service types are:</p> <p>OWN_PROCESS: Indicates a Win32 service that runs in its own process.</p> <p>INTERACTIVE_OWN_PROCESS: Indicates a Win32 service that runs in its own process and that can interact with the desktop.</p> <p>SHARE_PROCESS: Indicates a Win32 service that shares a process with other services.</p> <p>INTERACTIVE_SHARE_PROCESS: Indicates a Win32 service that shares a process with other services and that can interact with the desktop.</p> <p>KERNEL_DRIVER: Indicates a device driver.</p> <p>FILE_SYSTEM_DRIVER: Indicates a file system driver.</p>

Name	Type	Description
Current State	String	<p>The current state of the Microsoft Windows service. The service states are:</p> <p>STOPPED: The service is not running.</p> <p>START_PENDING: The service is starting.</p> <p>STOP_PENDING: The service is stopping.</p> <p>RUNNING: The service is running.</p> <p>CONTINUE_PENDING: The service continue is pending.</p> <p>PAUSE_PENDING: The service pause is pending.</p> <p>PAUSED: The service is paused.</p>
Win32 Exit Code	Integer	The error code of the last error returned by the Microsoft Windows service.
Exit Code Description	String	The description of the last error returned by the Microsoft Windows service.
Service Exit Code	Integer	The service specific return code from the Microsoft Windows service.
Controls Accepted	String	<p>The control commands accepted by the Microsoft Windows service. If more than one control is specified they are listed in blank-delimited format. The controls accepted may be any combination of the following:</p> <p>STOP: The service can be stopped.</p> <p>PAUSE_CONTINUE: The service can be paused and continued.</p> <p>SHUTDOWN: The service is notified when system shutdown occurs allowing the system to shut down this service.</p>
Check Point	Integer	The last check point of the Microsoft Windows service
Wait Hint	Integer	The last wait hint suggested by the Microsoft Windows service

Services:setServiceStartType

Method

Purpose This method sets the current start type of the specified Microsoft Windows service.

Type Synchronous, `IMPACT_ACTION`.

Arguments

Name	Type	Description
Service	String	Microsoft Windows service internal name
Service Start Type	String	The start type of the Microsoft Windows service. Specifies when to start the service. System furnishes a popup window with the following options: DEMAND_START AUTO_START DISABLED BOOT_START SYSTEM_START

Returns None.

Services:startService

Method

Purpose This method starts the specified type of Microsoft Windows service.

Type Synchronous, `IMPACT_ACTION`.

Arguments

Name	Type	Description
Service	String	Microsoft Windows service internal name

Returns None.

Services:stopService

Method

Purpose This method stops the specified type of Microsoft Windows service.

Type Synchronous, `IMPACT_ACTION`.

Arguments

Name	Type	Description
Service	String	Microsoft Windows service internal name

Returns None.

Services:pauseService

Method

Purpose This method pauses the specified type of Microsoft Windows service.

Type Synchronous, `IMPACT_ACTION`.

Arguments

Name	Type	Description
Service	String	Microsoft Windows service internal name

Returns None.

Services:continueService

Method

Purpose This method continues the specified type of Microsoft Windows service.

Type Synchronous, IMPACT_ACTION.

Arguments

Name	Type	Description
Service	String	Microsoft Windows service internal name

Returns None.

Services:controlService

Method

Purpose This method invokes the specified user-defined service control command for the specified Microsoft Windows service.

Remarks User-defined control command codes must be in the range 128-255.

Type Synchronous, `IMPACT_ACTION`.

Arguments

Name	Type	Description
Service	String	Microsoft Windows service internal name
User Control	Integer	The Microsoft Windows service user-defined control command code

Returns None.

COM.TIBCO.hawk.hma.Process

Microagent

Purpose The `process` microagent (on Microsoft Windows) returns information about processes running on the system.

Methods	Method	Description	Page
	Process:getInstanceCount	Returns process count for a specified process	185
	Process:getInstanceCountByCommand	Returns process count filtered by command argument	186
	Process:getProcess	Returns information on a process	187

Process:getInstanceCount

Method

Purpose This method (on Microsoft Windows) returns an instance count for a specified process. If the Instance field is empty, the total process count of the system is returned.

Remarks Test any regular expressions you plan to use in rulebases by first using them interactively to insure they return the desired results.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Process Name	String	Name of process instance

Returns

Name	Type	Description
Process Name	String	Name of process instance. Table is indexed on Process Name
Process Count	Integer	Process count for specified process

Process:getInstanceCountByCommand

Method

Purpose This method (on Microsoft Windows) returns a count of processes, filtered by command argument.

Remarks Regular expressions can be used to query status of a group of processes.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Command	String	Command-line argument

Returns

Name	Type	Description
Command	String	Command-line argument
Process Count	Integer	Process count for specified command

Process:getProcess

Method

Purpose This method (on Microsoft Windows) returns information about a specified process.

Remarks If the argument field is empty, data is returned for all processes running on the system.

Test any regular expressions you plan to use in rulebases. Use them interactively first, to ensure they return the desired results.

The methods `Performance:Process` and `Process:getProcess` can seem similar, but return different data. `Performance:Process` returns all Instances of Process Object from the Performance Data Helper (PDH) library. Invoking `Performance:process` without any arguments retrieves all Process instances in effect at that point in time. Invoking `Process:getProcess` without any arguments returns all currently running processes from the `PSAPI.lib`. This method only lists processes that have the right to access this library. Do not confuse these methods.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Process Name	String	Name of process instance. Empty argument returns information for all processes.

Returns

Name	Type	Description
Process Name	String	Name of process instance
ID Process	Integer	Process ID for current process. Table is indexed on ID Process
Parent Process ID	Integer	Parent process ID of the current process
Command	String	Command line arguments
CPU Time	Integer	Total amount of time the process has spent in User Mode plus Kernel Mode (in milliseconds).
Class	String	Process priority class

Name	Type	Description
User Name	String	Account name for the process
Mem Usage	Integer	Memory usage in kilobytes (working set size)
Peak Working SetSize	Integer	Peak working set size in kilobytes
Page File Usage	Integer	Page File usage in kilobytes
Page Fault Count	Integer	Page Fault count
Start Time	Integer	The number of seconds since the process started

COM.TIBCO.hawk.hma.Network

Microagent

Purpose The `network` microagent (on Microsoft Windows) returns information about networks configuration as well as performance statistics in the system.

Methods	Method	Description	Page
	Network:getConfig	Returns network configuration information on a per-interface basis	190

Network:getConfig

Method

Purpose This method (on Microsoft Windows) returns network configuration information on a per-interface basis in a manner similar to that returned by the `ifconfig` command.

Type Open, Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Interface Name	String	Name of Network Adapter

Returns

Name	Type	Description
Interface Name	String	Name of Network Adapter
Network Address	String	Network Address in dotted format, 32 bit Value
Hardware Address	String	Hardware Address in colon format
Description	String	Description of Network Interface
Network Mask	String	Network Mask in dotted format, 32 bit Value
MTU	String	Maximum transmission unit
RX Bytes	Integer	Number of octets of data received through the interface
RX Packets	Integer	The number of packets received through the interface
RX Errors	Integer	The number of incoming packets that were discarded because of errors.
RX Dropped	Integer	The number of incoming packets that were discarded even though they did not have errors.
TX Bytes	Integer	The number of octets of data sent through this interface.

Name	Type	Description
TX Packets	Integer	The number of unicast packets sent through this interface.
TX Errors	Integer	The number of outgoing packets that were discarded because of errors
TX Dropped	Integer	The number of outgoing packets that were discarded even though they did not have errors

COM.TIBCO.hawk.hma.System

Microagent

Purpose The `system` microagent (on Microsoft Windows) returns process management and CPU utilization information about the system.

Methods	Method	Description	Page
	System:getSystemInfo	Returns process count for a specified process	193
	System:getCpuInfo	Returns process count filtered by command argument	194

System:getSystemInfo

Method

Purpose This method (on Microsoft Windows) returns the system process management information.

Type Open, Synchronous, IMPACT_INFO.

Arguments None

Returns

Name	Type	Description
Up Time	Integer	System up time in seconds
Free Real Memory	Long	Free Real Memory in KBytes
% Free Real Memory	Integer	% Free Real Memory in KBytes
Total Real Memory	Long	Total Real Memory in KBytes
Used Real Memory	Long	Used Real Memory in KBytes
% Used Real Memory	Integer	% Used Real Memory in KBytes

System:getCpuInfo

Method

Purpose This method (on Microsoft Windows) returns operating system CPU utilization information.

Type Open, Synchronous, IMPACT_INFO.

Arguments None

Returns

Name	Type	Description
Total CPUs	Integer	Total number of CPUs
% User Time	Integer	Percent of the time spent in "user" mode for this processor
% System Time	Integer	Percent of the time spent in "system" mode for this processor.
% Time Idle	Integer	Percent of the time spent in "idle" mode for this processor
% Time Usage	Integer	% Time Usage

Solaris-SPARC Microagents and Methods

The TIBCO Hawk microagents described in this chapter run on the Solaris-SPARC platform in the HMA process. For more specific information on monitoring on Solaris-SPARC, see the *TIBCO Hawk Installation, Configuration, and Administration* guide.

Topics

- [Summary of Solaris-SPARC Microagents, page 196](#)
- [COM.TIBCO.hawk.hma.System, page 198](#)
- [COM.TIBCO.hawk.hma.Process, page 204](#)
- [COM.TIBCO.hawk.hma.FileSystem, page 208](#)
- [COM.TIBCO.hawk.hma.Network, page 214](#)

Summary of Solaris-SPARC Microagents

The Solaris-SPARC microagents are described as follows.

Table 4 Solaris-SPARC Microagents (Sheet 1 of 2)

Microagent	Description
Common Methods	<p>These methods are common to all microagents and provide release and trace-related functions. Methods are:</p> <p>getReleaseVersion</p> <p>getTraceLevel</p> <p>setTraceLevel</p> <p>getTraceParameters</p> <p>setTraceParameters</p> <p>_onUnsolicitedMsg</p>
System	<p>Used to collect information on system-wide usage of the CPU and memory. Methods are:</p> <p>System:getCpuInfo</p> <p>System:getSwapInfo</p> <p>System:getTunableInfo</p> <p>System:getSystemInfo</p>
Process	<p>Used to obtain information about processes running on the system. Methods are:</p> <p>Process:getProcess</p> <p>Process:getInstanceCount</p> <p>Process:getInstanceCountByCommand</p>
FileSystem	<p>Used to obtain information about file systems connected to the computer. Methods are:</p> <p>FileSystem:getByPartition</p> <p>FileSystem:getByMountPoint</p> <p>FileSystem:getByFileSystem</p>

Table 4 Solaris-SPARC Microagents (Sheet 2 of 2)

Microagent	Description
Network	Used to obtain information about network identification, communications and errors. Methods are: Network:getStatistics Network:getConfig

COM.TIBCO.hawk.hma.System

Microagent

Purpose The `System` microagent (on Solaris-SPARC) collects information on system-wide usage of the CPU and memory, summarizing all processes on the computer.

Methods	Method	Description	Page
	System:getSystemInfo	Returns information on process management	199
	System:getCpuInfo	Returns information on CPU utilization	201
	System:getSwapInfo	Returns information on system swap space	202
	System:getTunableInfo	Returns tunable system information	203

System:getSystemInfo

Method

Purpose This method (on Solaris-SPARC) returns process management information, including process swapping and run queuing.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Up Time	Integer	The system up time in seconds (Time since last reboot)
Real Memory	Long	Total physical memory in KBytes
Free Memory	Long	Total free memory in KBytes
% Free Real Memory	Long	Percentage of free real memory
Run Queue Size	Integer	Average size of run queue multiplied by 100
Avg Running	Double	One minute average number of running processes
Avg Running (5 min)	Double	Average number of processes running over 5 minutes
Avg Running (15 min)	Double	Average number of processes running over 15 minutes
CPU Utilization	Integer	Percentage of time there is at least one runnable process
Avg Swapped Out	Integer	Average number of processes swapped out multiplied by 100
% Time Swapped Out	Integer	Percentage of time there is at least one process swapped out
Avg Wait I/O	Integer	Average number of processes waiting for I/O multiplied by 100
% User Time	Integer	Percentage of the time spent in user mode, averaged among all processors

Name	Type	Description
% System Time	Integer	Percentage of time spent in system mode, averaged among all processors
% Time Idle	Integer	Percentage of time spent in idle mode, averaged among all processors
% Time wait	Integer	Percentage of time spent in wait mode, averaged among all processors

System:getCpuInfo

Method

Purpose This method (on Solaris-SPARC) returns information on system CPU utilization.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Processor	Integer	Processor (slot) number. Table is indexed on Processor.
% User Time	Integer	Time spent when in the user mode
% System Time	Integer	Time spent when in the system
% Time Idle	Integer	Time spent in idle mode
% Time Wait	Integer	Time spent in wait mode
% Time Wait I/O	Integer	Time spent in wait I/O state
% Time Wait Swap	Integer	Time spent in wait swap state
% Time Wait Physical I/O	Integer	Time spent in wait physical I/O state
Physical Reads	Integer	Physical block reads
Physical Writes	Integer	Physical block writes
Logical Reads	Integer	Logical block reads
Logical Writes	Integer	Logical block writes
Raw Reads	Integer	Raw I/O reads
Raw Writes	Integer	Raw I/O writes
Context Switches	Integer	Context switches
Device Interrupts	Integer	Device interrupts

System:getSwapInfo

Method

Purpose This method (on Solaris-SPARC) returns information on the utilization of system swap space.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Descriptions
Avg KBytes Free	Double	Average free memory (kilobytes)
Avg KBytes Reserved	Double	Average reserved swap space (kilobytes)
Avg KBytes Allocated	Double	Average allocated swap space (kilobytes)
Avg KBytes Unreserved	Double	Average unreserved swap space (kilobytes)
Avg KBytes Unallocated	Double	Average unallocated swap space (kilobytes)
% Avg Reserved	Double	Average reserved swap space (%)
% Avg Available	Double	Average available swap space (%)
Swap Used	Long	Used swap space (Kilobytes)
Swap Free	Long	Free swap space (Kilobytes)
Swap Total	Long	Total swap space (Kilobytes)

System:getTunableInfo

Method

Purpose This method (on Solaris-SPARC) returns tunable system information.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
I/O Buffers	Integer	Number of I/O buffers
Max Processes System	Integer	Max number of processes system wide
Max User Processes System	Integer	Max user processes system wide
Max Processes User	Integer	Max number of processes per user
Hash Buffers	Integer	Number of hash buffers to allocate
Hash mask	Integer	Hash mask for buffers
Physical Memory	Integer	Maximum physical memory to use. If zero, use all available memory

COM.TIBCO.hawk.hma.Process

Microagent

Purpose The `Process` microagent (on Solaris-SPARC) furnishes information about processes running on the system.

Methods	Method	Description	Page
	Process:getProcess	Returns information on a process	205
	Process:getInstanceCount	Returns the instance count	206
	Process:getInstanceCountByCommand	Returns the number of instances of a specific process	207

Process:getProcess

Method

Purpose This method (on Solaris-SPARC) returns a process table filtered by the `Process Name` argument.

Remarks The process name argument is a pattern match using regular expressions. Test any regular expressions you plan to use in rulebases by first using them interactively to insure they return the desired results.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Process Name	String	Process name. Empty argument string returns information for all processes.

Returns

Name	Type	Description
Process Name	String	Process name. Table is indexed on ID Process
ID Process	Integer	Process ID
Parent Process ID	Integer	Parent process ID of the current process
User Name	String	User name
Virtual KBytes	Integer	The size of virtual address space, in kilobytes
Stack KBytes	Integer	Stack size, in kilobytes
Heap KBytes	Integer	Heap size, in kilobytes
CPU Time	Integer	CPU usage (user plus system, in milliseconds)
% CPU	Integer	CPU usage in percent
% Memory	Integer	Memory usage in percent
Class	String	Scheduling class ID
Command	String	Full command line as given by UNIX <code>ps -ef</code> command. Truncated to 80 characters
Start Time	Integer	The number of seconds since the process started

Process:getInstanceCount

Method

Purpose This method (on Solaris-SPARC) returns an instance count for the process denoted by the `Process Name` argument.

Remarks Test any regular expressions you plan to use in rulebases by first using them interactively, to insure they return the desired results.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Process Name	String	Process name. Pattern match with regular expressions. Empty argument string returns combined instance count for all processes.

Returns

Name	Type	Description
Process Name	String	Process name. Table is indexed on Process Name
Process Count	Integer	Number of instances

Process:getInstanceCountByCommand

Method

Purpose This method (on Solaris-SPARC) returns a count of processes, filtered by command argument.

Remarks Regular expressions can be used to query status of a group of processes.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Command	String	Command-line arguments

Returns

Name	Type	Description
Command	String	Command-line argument
Process Count	Integer	Process count for specified command

COM.TIBCO.hawk.hma.FileSystem

Microagent

Purpose The `FileSystem` microagent (on Solaris-SPARC) furnishes information about file systems connected to the computer.

Methods	Method	Description	Page
	FileSystem:getByPartition	Returns information on a file system for its partition name	209
	FileSystem:getByMountPoint	Returns information on a file system for its mount point	210
	FileSystem:getByFileSystem	Returns utilization statistics for the currently mounted filesystems	212

FileSystem:getByPartition

Method

Purpose This method (on Solaris-SPARC) provides utilization statistics for currently mounted file systems, filtered by the device given as the `Partition` argument. Information will be listed for only local partitions with non-zero data blocks.

Remarks NFS mounted file systems are ignored and should be monitored by agents running on the file servers themselves. Invoke interactively with no argument to obtain valid arguments.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Partition	String	Device name (hard disk partition). Pattern match with regular expressions. Empty argument string returns information for all partitions.

Returns

Name	Type	Description
Partition	String	Device name of a hard disk partition. Table is indexed on Partition.
Mount Point	String	Mount point for device
% Free	Integer	Amount of disk space on the mounted file system that is currently free, expressed as percentage of total size
% Used	Integer	Amount of disk space on the mounted file system currently in use, expressed as percentage of total size
KBytes Free	Long	Amount of disk space on the mounted file system that is currently free, expressed as absolute value (kilobytes)
KBytes Used	Long	Amount of disk space on the mounted file system currently in use, expressed as absolute value (kilobytes)
KBytes Total	Long	Total amount of disk space on mounted file system (kilobytes)
% Minimum	Integer	Minimum Percentage free

FileSystem:getByMountPoint

Method

Purpose This method (on Solaris-SPARC) provides utilization statistics for currently mounted file systems filtered by the mount point given as the argument `Mount Point`. Information will be listed for only local partitions with non-zero data blocks.

Remarks NFS mounted file systems are ignored and should be monitored by agents running on the file servers themselves.

Invoke interactively with an empty string, to obtain valid arguments.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Mount Point	String	Name of directory where file system is mounted. Pattern match with regular expressions. Empty argument returns information for all mount points.

Returns

Name	Type	Description
Partition	String	Device name of a hard disk partition
Mount Point	String	Mount point of device. Table is indexed on Mount Point
% Free	Integer	Amount of disk space on the mounted file system that is currently free, expressed as percentage of total size
% Used	Integer	Amount of disk space on the mounted file system currently in use, expressed as percentage of total size
KBytes Free	Long	Amount of disk space on the mounted file system that is currently free, expressed as absolute value (kilobytes)
KBytes Used	Long	Amount of disk space on the mounted file system currently in use, expressed as absolute value (kilobytes)
KBytes Total	Long	Total amount of disk space on mounted file system (kilobytes)

Name	Type	Description
% Minimum	Integer	Minimum Percentage free

FileSystem:getByFileSystem

Method

Purpose This method (on Solaris-SPARC) provides utilization statistics for the currently mounted filesystems filtered by device name and file system type. The argument is treated as a pattern to match. If no argument is specified, the entire mount table is reported.

For any filesystem that does not respond within five seconds, the data values for % Free and % Used are considered stale and their values are set to -1.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
FileSystem Type	String	Name of the file system type.

Returns

Name	Type	Description
Partition	String	Device name of a hard disk partition
Mount Point	String	Mount point of device. Table is indexed on Mount Point.
File System Type	String	Name of the file system type
% Free	Integer	Amount of disk space on the mounted file system that is currently free, expressed as percentage of total size
% Used	Integer	Amount of disk space on the mounted file system currently in use, expressed as percentage of total size
KBytes Free	Long	Amount of disk space on the mounted file system that is currently free, expressed as absolute value (kilobytes)
KBytes Used	Long	Amount of disk space on the mounted file system currently in use, expressed as absolute value (kilobytes)
KBytes Total	Long	Total amount of disk space on mounted file system (kilobytes)

Name	Type	Description
% Minimum	Integer	Minimum Percentage free

COM.TIBCO.hawk.hma.Network

Microagent

Purpose The `Network` microagent (on Solaris-SPARC) furnishes information about network utilization statistics, identification, communications, and errors.

Methods	Method	Description	Page
	Network:getStatistics	Returns network performance information	215
	Network:getConfig	Returns information on network addresses and masks	216

Network:getStatistics

Method

Purpose This method (on Solaris-SPARC) returns network performance information similar to that returned by `netstat`.

Remarks All statistics are for the lifetime of the system (since system boot time).

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Interface Name	String	Name of the network interface

Returns

Name	Type	Description
Interface Name	String	Name of the network interface. Table is indexed on Interface Name.
Input Packets	Integer	Packets received on the interface, lifetime total
Input Packet Errors	Integer	Input errors on the interface, lifetime total
Output Packets	Integer	Packets sent on the interface, lifetime total
Output Packet Errors	Integer	Output errors on the interface, lifetime total
Network Collisions	Integer	Collisions, lifetime total
Framing Errors	Integer	Framing errors, lifetime total
CRC Errors	Integer	Checksum errors, lifetime total
Input Packets Missed	Integer	Receive missed, lifetime total
Collision Retry Errors	Integer	Collision retry errors, lifetime total
No Carrier Errors	Integer	No carrier, lifetime total
% Network Collisions	Double	Rate (percentage) of collisions against total number of output packets (lifetime totals)

Network:getConfig

Method

Purpose This method (on Solaris-SPARC) returns network configuration information in a manner similar to that returned by the `ifconfig` command.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Interface Name	String	Name of the network interface

Returns

Name	Type	Description
Interface Name	String	Name of the network interface. Table is indexed on Interface Name
Network Address	String	Network Address in dotted format, 32 bit Value
Broadcast Address	String	Broadcast address for the interface
Network Mask	String	Network mask for the interface

Chapter 6

Solaris-10 Microagents and Methods

The TIBCO Hawk microagents described in this chapter run on the Solaris-10 platform in the HMA process. For more specific information on monitoring on Solaris-10, see the *TIBCO Hawk Installation, Configuration, and Administration* guide.

Topics

- [Summary of Solaris-10 Microagents, page 218](#)
- [COM.TIBCO.hawk.hma.System, page 220](#)
- [COM.TIBCO.hawk.hma.Process, page 226](#)
- [COM.TIBCO.hawk.hma.FileSystem, page 230](#)
- [COM.TIBCO.hawk.hma.Network, page 235](#)

Summary of Solaris-10 Microagents

The Solaris-10 microagents are described as follows.

Table 5 Solaris-10 Microagents (Sheet 1 of 2)

Microagent	Description
Common Methods	<p>These methods are common to all microagents and provide release and trace-related functions. Methods are:</p> <p>getReleaseVersion</p> <p>getTraceLevel</p> <p>setTraceLevel</p> <p>getTraceParameters</p> <p>setTraceParameters</p> <p>_onUnsolicitedMsg</p>
System	<p>Used to collect information on system-wide usage of the CPU and memory. Methods are:</p> <p>System:getCpuInfo</p> <p>System:getSwapInfo</p> <p>System:getTunableInfo</p> <p>System:getSystemInfo</p>
Process	<p>Used to obtain information about processes running on the system. Methods are:</p> <p>Process:getProcess</p> <p>Process:getInstanceCount</p> <p>Process:getInstanceCountByCommand</p>
FileSystem	<p>Used to obtain information about file systems connected to the computer. Methods are:</p> <p>FileSystem:getByPartition</p> <p>FileSystem:getByMountPoint</p> <p>FileSystem:getByFileSystem</p>

Table 5 Solaris-10 Microagents (Sheet 2 of 2)

Microagent	Description
Network	Used to obtain information about network identification, communications and errors. Methods are: Network:getStatistics Network:getConfig

COM.TIBCO.hawk.hma.System

Microagent

Purpose The `System` microagent (on Solaris-10) collects information on system-wide usage of the CPU and memory, summarizing all processes on the computer.

Methods	Method	Description	Page
	System:getSystemInfo	Returns information on process management	223
	System:getCpuInfo	Returns information on CPU utilization	223
	System:getSwapInfo	Returns information on system swap space	224
	System:getTunableInfo	Returns tunable system information	225

System:getSystemInfo

Method

Purpose This method (on Solaris-10) returns process management information, including process swapping and run queuing.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Up Time	Integer	The system up time in seconds (Time since last reboot)
Real Memory	Long	Total physical memory in KBytes
Free Memory	Long	Total free memory in KBytes
% Free Real Memory	Long	Percentage of free real memory
Run Queue Size	Integer	Average size of run queue multiplied by 100
Avg Running	Double	Average number of processes running over one minute
Avg Running (5 min)	Double	Average number of processes running over five minutes
Avg Running (15 min)	Double	Average number of processes running over 15 minutes
% Time Usage	Integer	% Time Usage
CPU Utilization	Integer	Percentage of time there is at least one runnable process
Avg Swapped Out	Integer	Average number of processes swapped out multiplied by 100
% Time Swapped Out	Integer	Percentage of time there is at least one process swapped out
Avg Wait I/O	Integer	Average number of processes waiting for I/O multiplied by 100

Name	Type	Description
% User Time	Integer	Percentage of the time spent in user mode, averaged among all processors
% System Time	Integer	Percentage of time spent in system mode, averaged among all processors
% Time Idle	Integer	Percentage of time spent in idle mode, averaged among all processors
% Time wait	Integer	Percentage of time spent in wait mode, averaged among all processors

System:getCpuInfo

Method

Purpose This method (on Solaris-10) returns information on system CPU utilization.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Processor	Integer	Processor (slot) number. Table is indexed on Processor.
% User Time	Integer	Time spent when in the user mode
% System Time	Integer	Time spent when in the system
% Time Idle	Integer	Time spent in idle mode
% Time Wait	Integer	Time spent in wait mode
% Time Wait I/O	Integer	Time spent in wait I/O state
% Time Wait Swap	Integer	Time spent in wait swap state
% Time Wait Physical I/O	Integer	Time spent in wait physical I/O state
Physical Reads	Integer	Physical block reads
Physical Writes	Integer	Physical block writes
Logical Reads	Integer	Logical block reads
Logical Writes	Integer	Logical block writes
Raw Reads	Integer	Raw I/O reads
Raw Writes	Integer	Raw I/O writes
Context Switches	Integer	Context switches
Device Interrupts	Integer	Device interrupts

System:getSwapInfo

Method

Purpose This method (on Solaris-10) returns information on the utilization of system swap space.

Type Synchronous, `IMPACT_INFO`.

Arguments None.

Returns

Name	Type	Descriptions
Avg KBytes Free	Double	Average free memory (kilobytes)
Avg KBytes Reserved	Double	Average reserved swap space (kilobytes)
Avg KBytes Allocated	Double	Average allocated swap space (kilobytes)
Avg KBytes Unreserved	Double	Average unreserved swap space (kilobytes)
Avg KBytes Unallocated	Double	Average unallocated swap space (kilobytes)
% Avg Reserved	Double	Average reserved swap space (%)
% Avg Available	Double	Average available swap space (%)
Swap Used	Long	Used swap space (Kilobytes)
Swap Free	Long	Free swap space (Kilobytes)
Swap Total	Long	Total swap space (Kilobytes)

System:getTunableInfo

Method

Purpose This method (on Solaris-10) returns tunable system information.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
I/O Buffers	Integer	Number of I/O buffers
Max Processes System	Integer	Max number of processes system wide
Max User Processes System	Integer	Max user processes system wide
Max Processes User	Integer	Max number of processes per user
Hash Buffers	Integer	Number of hash buffers to allocate
Hash mask	Integer	Hash mask for buffers
Physical Memory	Integer	Maximum physical memory to use. If zero, use all available memory

COM.TIBCO.hawk.hma.Process

Microagent

Purpose The `Process` microagent (on Solaris-10) furnishes information about processes running on the system.

Methods	Method	Description	Page
	Process:getProcess	Returns information on a process	227
	Process:getInstanceCount	Returns the instance count	228
	Process:getInstanceCount ByCommand	Returns the number of instances of a specific process	229

Process:getProcess

Method

Purpose This method (on Solaris-10) returns a process table filtered by the `Process Name` argument.

Remarks The process name argument is a pattern match using regular expressions. Test any regular expressions you plan to use in rulebases by first using them interactively to insure they return the desired results.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Process Name	String	Process name. Empty argument string returns information for all processes.

Returns

Name	Type	Description
Process Name	String	Process name. Table is indexed on ID Process
ID Process	Integer	Process ID
Parent Process ID	Integer	Parent process ID of the current process
User Name	String	User name
Virtual KBytes	Integer	The size of virtual address space, in kilobytes
Stack KBytes	Integer	Stack size, in kilobytes
Heap KBytes	Integer	Heap size, in kilobytes
CPU Time	Integer	CPU usage (user plus system, in milliseconds)
% CPU	Integer	CPU usage in percent
% Memory	Integer	Memory usage in percent
Class	String	Scheduling class ID
Command	String	Full command line as given by UNIX <code>ps -ef</code> command. Truncated to 80 characters
Start Time	Integer	The number of seconds since the process started

Process:getInstanceCount

Method

Purpose This method (on Solaris-10) returns an instance count for the process denoted by the `Process Name` argument.

Remarks Test any regular expressions you plan to use in rulebases by first using them interactively, to insure they return the desired results.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Process Name	String	Process name. Pattern match with regular expressions. Empty argument string returns combined instance count for all processes.

Returns

Name	Type	Description
Process Name	String	Process name. Table is indexed on Process Name
Process Count	Integer	Number of instances

Process:getInstanceCountByCommand

Method

Purpose This method (on Solaris-10) returns a count of processes, filtered by command argument.

Remarks Regular expressions can be used to query status of a group of processes.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Command	String	Command-line arguments

Returns

Name	Type	Description
Command	String	Command-line argument
Process Count	Integer	Process count for specified command

COM.TIBCO.hawk.hma.FileSystem

Microagent

Purpose The `FileSystem` microagent (on Solaris-10) furnishes information about file systems connected to the computer.

Methods	Method	Description	Page
	FileSystem:getByPartition	Returns information on a file system for its partition name	231
	FileSystem:getByMountPoint	Returns information on a file system for its mount point	232
	FileSystem:getByFileSystem	Returns utilization statistics for the currently mounted filesystems	233

FileSystem:getByPartition

Method

Purpose This method (on Solaris-10) provides utilization statistics for currently mounted file systems, filtered by the device given as the `Partition` argument. Information will be listed for only local partitions with non-zero data blocks.

Remarks NFS mounted file systems are ignored and should be monitored by agents running on the file servers themselves. Invoke interactively with no argument to obtain valid arguments.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Partition	String	Device name (hard disk partition). Pattern match with regular expressions. Empty argument string returns information for all partitions.

Returns

Name	Type	Description
Partition	String	Device name of a hard disk partition. Table is indexed on Partition.
Mount Point	String	Mount point for device
% Free	Integer	Amount of disk space on the mounted file system that is currently free, expressed as percentage of total size
% Used	Integer	Amount of disk space on the mounted file system currently in use, expressed as percentage of total size
KBytes Free	Long	Amount of disk space on the mounted file system that is currently free, expressed as absolute value (kilobytes)
KBytes Used	Long	Amount of disk space on the mounted file system currently in use, expressed as absolute value (kilobytes)
KBytes Total	Long	Total amount of disk space on mounted file system (kilobytes)
% Minimum	Integer	Minimum Percentage free

FileSystem:getByMountPoint

Method

Purpose This method (on Solaris-10) provides utilization statistics for currently mounted file systems filtered by the mount point given as the argument `Mount Point`. Information will be listed for only local partitions with non-zero data blocks.

Remarks NFS mounted file systems are ignored and should be monitored by agents running on the file servers themselves.

Invoke interactively with an empty string, to obtain valid arguments.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Mount Point	String	Name of directory where file system is mounted. Pattern match with regular expressions. Empty argument returns information for all mount points.

Returns

Name	Type	Description
Partition	String	Device name of a hard disk partition
Mount Point	String	Mount point of device. Table is indexed on Mount Point
% Free	Integer	Amount of disk space on the mounted file system that is currently free, expressed as percentage of total size
% Used	Integer	Amount of disk space on the mounted file system currently in use, expressed as percentage of total size
KBytes Free	Long	Amount of disk space on the mounted file system that is currently free, expressed as absolute value (kilobytes)
KBytes Used	Long	Amount of disk space on the mounted file system currently in use, expressed as absolute value (kilobytes)
KBytes Total	Long	Total amount of disk space on mounted file system (kilobytes)
% Minimum	Integer	Minimum Percentage free

FileSystem:getByFileSystem

Method

Purpose This method (on Solaris-10) provides utilization statistics for the currently mounted filesystems filtered by device name and file system type. The argument is treated as a pattern to match. If no argument is specified, the entire mount table is reported.

For any filesystem that does not respond within five seconds, the data values for % Free and % Used are considered stale and their values are set to -1.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
FileSystem Type	String	Name of the file system type.

Returns

Name	Type	Description
Partition	String	Device name of a hard disk partition
Mount Point	String	Mount point of device. Table is indexed on Mount Point.
File System Type	String	Name of the file system type
% Free	Integer	Amount of disk space on the mounted file system that is currently free, expressed as percentage of total size
% Used	Integer	Amount of disk space on the mounted file system currently in use, expressed as percentage of total size
KBytes Free	Long	Amount of disk space on the mounted file system that is currently free, expressed as absolute value (kilobytes)
KBytes Used	Long	Amount of disk space on the mounted file system currently in use, expressed as absolute value (kilobytes)
KBytes Total	Long	Total amount of disk space on mounted file system (kilobytes)

Name	Type	Description
% Minimum	Integer	Minimum Percentage free

COM.TIBCO.hawk.hma.Network

Microagent

Purpose The `Network` microagent (on Solaris-10) furnishes information about network utilization statistics, identification, communications, and errors.

Methods	Method	Description	Page
	Network:getStatistics	Returns network performance information	236
	Network:getConfig	Returns information on network addresses and masks	238

Network:getStatistics

Method

Purpose This method (on Solaris-10) returns network performance information similar to that returned by `netstat`.

Remarks All statistics are for the lifetime of the system (since system boot time).

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Interface Name	String	Name of the network interface

Returns

Name	Type	Description
Interface Name	String	Name of the network interface. Table is indexed on Interface Name.
Input Bytes	Long	The number of octets of data received through the interface
Input Packets	Integer	Packets received on the interface, lifetime total
Input Packet Errors	Integer	Input errors on the interface, lifetime total
Output Bytes	Long	The number of octets of data sent through this interface
Output Packets	Integer	Packets sent on the interface, lifetime total
Output Packet Errors	Integer	Output errors on the interface, lifetime total
Network Collisions	Integer	Collisions, lifetime total
Framing Errors	Integer	Framing errors, lifetime total
CRC Errors	Integer	Checksum errors, lifetime total
Input Packets Missed	Integer	Receive missed, lifetime total
Collision Retry Errors	Integer	Collision retry errors, lifetime total
No Carrier Errors	Integer	No carrier, lifetime total

Name	Type	Description
% Network Collisions	Double	Rate (percentage) of collisions against total number of output packets (lifetime totals)

Network:getConfig

Method

Purpose This method (on Solaris-10) returns network configuration information in a manner similar to that returned by the `ifconfig` command.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Interface Name	String	Name of the network interface

Returns

Name	Type	Description
Interface Name	String	Name of the network interface. Table is indexed on Interface Name
Network Address	String	Network Address in dotted format, 32 bit Value
Hardware Address	String	Hardware Address in colon format
Broadcast Address	String	Broadcast address for the interface
Network Mask	String	Network mask for the interface

Chapter 7

HP-UX Microagents and Methods

The TIBCO Hawk microagents described in this chapter run on the HP-UX platform in the HMA process. For more specific information on monitoring on HP-UX, see the *TIBCO Hawk Installation, Configuration, and Administration* guide.

Topics

- [Summary of HP-UX Microagents, page 240](#)
- [COM.TIBCO.hawk.hma.System, page 242](#)
- [COM.TIBCO.hawk.hma.Process, page 248](#)
- [COM.TIBCO.hawk.hma.FileSystem, page 252](#)
- [COM.TIBCO.hawk.hma.Network, page 258](#)

Summary of HP-UX Microagents

The HP-UX microagents are described as follows.

Table 6 HP-UX Microagents (Sheet 1 of 2)

Microagent	Description
Common Methods	<p>These methods are common to all microagents and provide release and trace-related functions. Methods are:</p> <p>getReleaseVersion</p> <p>getTraceLevel</p> <p>setTraceLevel</p> <p>getTraceParameters</p> <p>setTraceParameters</p> <p>_onUnsolicitedMsg</p>
System	<p>Used to collect information on system-wide usage of the CPU and memory. Methods are:</p> <p>System:getSystemInfo</p> <p>System:getCpuInfo</p> <p>System:getSwapInfo</p> <p>System:getTunableInfo</p>
Process	<p>Used to obtain information about processes running on the system. Methods are:</p> <p>Process:getProcess</p> <p>Process:getInstanceCount</p> <p>Process:getInstanceCountByCommand</p>
FileSystem	<p>Used to obtain information about file systems connected to the computer. Methods are:</p> <p>FileSystem:getByPartition</p> <p>FileSystem:getByMountPoint</p> <p>FileSystem:getByFileSystem</p>

Table 6 HP-UX Microagents (Sheet 2 of 2)

Microagent	Description
Network	Used to obtain information about network identification, communications, and errors. Methods are: Network:getConfig Network:getStatistics

COM.TIBCO.hawk.hma.System

Microagent

Purpose The `System` microagent (on HP-UX) is used to collect information on system-wide usage of the CPU and memory, summarizing all processes on the computer.

Methods	Method	Description	Page
	System:getSystemInfo	Returns information on process managing	243
	System:getCpuInfo	Returns information on processor activity	245
	System:getSwapInfo	Returns information on system swap space	246
	System:getTunableInfo	Returns tunable system information	247

System:getSystemInfo

Method

Purpose This method (on HP-UX) returns information that pertains to multi-tasking process management by the system, including process swapping and run queuing. It also returns information on memory use (real and virtual).

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Up Time	Integer	System up time in seconds
Real Memory	Long	Total physical memory in KBytes
Free Memory	Long	Total free memory in KBytes
% Free Real Memory	Integer	Percent of the Free Real Memory
Processors	Integer	Number of active processors
Run Queue Size	Integer	Length of run queue
Disk Wait	Integer	Jobs in disk wait
Page Wait	Integer	Jobs in page wait
Sleeping	Integer	Jobs sleeping in core
Swapped Out	Integer	Swapped out runnable jobs
Avg Running	Double	Avg Running (1 min)
Avg Running (5 min)	Double	Avg Running (5 min)
Avg Running 15 min)	Double	Avg Running (15 min)
% User Time	Integer	Percent of the time currently run processes execute in user mode
% System Time	Integer	Percent of the time currently run processes execute in kernel mode

Name	Type	Description
% Time Idle	Integer	Percent of the time wait process is the currently- running process
% Time Wait	Integer	Percent of the time currently-running processes wait for block I/O to complete
Context Switches	Integer	Number of context switches
Device Interrupts	Integer	Number of device interrupts

System:getCpuInfo

Method

Purpose This method (on HP-UX) returns information on processor activity for all processors on the computer.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Processor	Integer	Processor (slot) number
Logical Reads	Integer	Logical Reads
Logical Writes	Integer	Logical Writes
NFS Reads	Integer	NFS Reads
NFS Writes	Integer	NFS Writes.
Raw Reads	Integer	Raw Reads
Raw Writes	Integer	Raw Writes
% User Time	Integer	Percent of the time currently run processes execute in user mode
% System Time	Integer	Percent of the time currently run processes execute in kernel mode
% Time Idle	Integer	Percent of the time wait process is the currently- running process
% Time Wait	Integer	Percent of the time currently-running processes wait for block I/O to complete

System:getSwapInfo

Method

Purpose This method (on HP-UX) returns information on the utilization of system swap space.

Type Synchronous, `IMPACT_INFO`.

Arguments None.

Returns

Name	Type	Description
Avg KBytes Free	Integer	Average free memory, in KB
Avg KBytes Unallocated	Integer	Average unallocated swap space, in KB
KBytes Used	Integer	Kilobytes used
KBytes Total	Integer	Kilobytes total
% Free	Double	Percentage free
% Used	Double	Percentage used

System:getTunableInfo

Method

Purpose This method (on HP-UX) returns tunable system information.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Maximum Processes	Integer	Maximum Processes
Boot Time	Integer	Boot Time
Physical Memory	Integer	Physical Memory
Page Size	Integer	Page Size
Maximum Memory	Integer	Maximum Memory
Maximum Inodes	Integer	Maximum Inodes
Maximum Files	Integer	Maximum Files

COM.TIBCO.hawk.hma.Process

Microagent

Purpose The `Process` microagent (on HP-UX) furnishes information about processes running on the system.

Methods	Method	Description	Page
	Process:getProcess	Returns information on a process	249
	Process:getInstanceCount	Returns the number of instances of a process	250
	Process:getInstanceCountByCommand	Returns a count of processes, filtered by command argument	251

Process:getProcess

Method

Purpose This method (on HP-UX) returns a process table filtered by the `Process Name` argument.

Remarks Test any regular expressions you plan to use in rulebases by first using them interactively to insure they return the desired results.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Process Name	String	Process name. Pattern match with regular expressions. Empty argument returns information for all processes.

Returns

Name	Type	Description
Process Name	String	Process name. Table is indexed on ID Process.
ID Process	Integer	The process ID of the process (this datum required in order to kill the process)
Parent Process ID	Integer	Parent process ID of the current process
User Name	String	Effective user name of the process
Virtual KBytes	Integer	The size of virtual address space (kilobytes)
Stack KBytes	Integer	The size of the process's stack (kilobytes)
% CPU	Integer	The ratio of CPU time used recently to CPU time available in the same period (the meaning of "recently" in this context is unspecified; the CPU time available is determined in an unspecified manner).
Command	String	The command with all its arguments as a string, as with <code>ps -ef</code> . Truncated to 58 characters.
CPU Time	Integer	CPU usage (user time + system time) in milliseconds

Process:getInstanceCount

Method

Purpose This method (on HP-UX) returns an instance count for the process denoted by the `Process Name` argument.

Remarks Test any regular expressions you plan to use in rulebases by first using them interactively to insure they return the desired results.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Process Name	String	Process name. Pattern match with regular expressions. Empty argument string returns combined instance count for all processes.

Returns

Name	Type	Description
Process Name	String	Process name
Process Count	Integer	Instance count for a currently running process

Process:getInstanceCountByCommand

Method

Purpose This method (on HP-UX) returns a count of processes, filtered by command argument.

Remarks Regular expressions can be used to query status of a group of processes.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Command	String	Command-line argument

Returns

Name	Type	Description
Command	String	Command-line argument
Process Count	Integer	Process count for specified command

COM.TIBCO.hawk.hma.FileSystem

Microagent

Purpose The `FileSystem` microagent (on HP-UX) furnishes information about file systems connected to the computer.

Methods	Method	Description	Page
	FileSystem:getByPartition	Returns information on a file system for its partition name	253
	FileSystem:getByMountPoint	Returns information on a file system for its mount point	255
	FileSystem:getByFileSystem	Returns utilization statistics for the currently mounted filesystems	257

FileSystem:getByPartition

Method

Purpose This method (on HP-UX) provides utilization statistics for currently mounted file systems filtered by the device given as the `Partition` argument. Information will be listed for only local partitions with non-zero data blocks.

Remarks NFS mounted file systems are ignored and should be monitored by agents running on the file servers themselves.

Invoke interactively with no argument to obtain valid arguments.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Partition	String	Device name (hard disk partition). Pattern match with regular expressions. Empty argument string returns information for all partitions.

Returns

Name	Type	Description
Partition	String	Device name of a hard disk partition. Table is indexed on Partition
Mount Point	String	Mount Point
% Free	Integer	Amount of disk space currently free on the mounted file system, expressed as percentage of total size
% Used	Integer	Amount of disk space on the mounted file system currently in use, expressed as percentage of total size
KBytes Free	Long	Amount of disk space on the mounted file system that is currently free, expressed as absolute value (KB)
KBytes Used	Long	Amount of disk space on the mounted file system currently in use, expressed as absolute value (KB)
KBytes Total	Long	Total amount of disk space on mounted file system (KB)

Name	Type	Description
% Minimum	Integer	Minimum percentage free

FileSystem:getByMountPoint

Method

Purpose This method (on HP-UX) provides utilization statistics for currently mounted file systems filtered by the mount point given as the argument `Mount Point`. Information will be listed for only local partitions with non-zero data blocks.

Remarks NFS mounted file systems are ignored and should be monitored by agents running on the file servers themselves. Invoke interactively with no argument to obtain valid arguments.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Mount Point	String	Name of the directory where file system is mounted. Pattern match with regular expressions. Empty argument string returns information for all mount points.

Returns

Name	Type	Description
Partition	String	“Device” name of a hard disk partition
Mount Point	String	Mount Point. Table is indexed on Mount Point
% Free	Integer	Amount of disk space on the mounted file system that is currently free, expressed as percentage of total size
% Used	Integer	Amount of disk space on the mounted file system currently in use, expressed as percentage of total size
KBytes Free	Long	Amount of disk space on the mounted file system that is currently free, expressed as absolute value (kilobytes)
KBytes Used	Long	Amount of disk space on the mounted file system currently in use, expressed as absolute value (kilobytes)
KBytes Total	Long	Total amount of disk space on mounted file system (kilobytes)

Name	Type	Description
% Minimum	Integer	Minimum percentage free

FileSystem:getByFileSystem

Method

Purpose This method (on HP-UX) provides utilization statistics for the currently mounted filesystems filtered by device name and file system type. The argument is treated as a pattern to match. If no argument is specified, the entire mount table is reported.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
FileSystem Type	String	Name of the file system type

Returns

Name	Type	Description
Partition	String	Device name of a hard disk partition
Mount Point	String	Mount point of device. Table is indexed on Mount Point.
File System Type	String	Name of the file system type
% Free	Integer	Amount of disk space on the mounted file system that is currently free, expressed as percentage of total size
% Used	Integer	Amount of disk space on the mounted file system currently in use, expressed as percentage of total size
KBytes Free	Long	Amount of disk space on the mounted file system that is currently free, expressed as absolute value (kilobytes)
KBytes Used	Long	Amount of disk space on the mounted file system currently in use, expressed as absolute value (kilobytes)
KBytes Total	Long	Total amount of disk space on mounted file system (kilobytes)
% Minimum	Integer	Minimum Percentage free

COM.TIBCO.hawk.hma.Network

Microagent

Purpose The `Network` microagent (on HP-UX) furnishes information about network identification.

Methods	Method	Description	Page
	Network:getConfig	Returns information on network addresses and masks	259
	Network:getStatistics	Returns network performance information	260

Network:getConfig

Method

Purpose This method (on HP-UX) returns network configuration information similar to that returned by `ifconfig`.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Interface Name	String	Name of the network interface

Returns

Name	Type	Description
Interface Name	String	Name of the network interface. Table is indexed on Interface Name.
Network Address	String	Network Address in dotted format, 32 bit Value
Broadcast Address	String	Broadcast address for the interface
Network Mask	String	Network mask for the interface

Network:getStatistics

Method

Purpose This method returns network performance information similar to that returned by `netstat`.

Remarks All statistics are for the lifetime of the system (since it was booted).

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Interface Name	String	Name of the network interface

Returns

Name	Type	Description
Interface Name	String	Name of the network interface
Input Packets	Integer	Packets received on the interface, lifetime total
Input Packet Errors	Integer	Input errors on the interface, lifetime total
Output Packets	Integer	Packets sent on the interface, lifetime total
Output Packet Errors	Integer	Output errors on the interface, lifetime total
Network Collisions	Integer	Collisions, lifetime total
Framing Errors	Long	Framing errors, lifetime total
% Network Collisions	Double	Rate (percentage) of collisions against total number of output packets, lifetime totals.
Abort Trans Errors	Long	IEEE 802.5 Abort Trans Errors
AC Errors	Long	IEEE 802.5 AC Errors
Burst Errors	Long	IEEE 802.5 Burst Errors
Carrier Sense Errors	Long	IEEE 802.3 Carrier Sense Errors
Control Field Errors	Long	IEEE 802.3 Control Field Errors
CRC Errors	Long	Checksum errors, lifetime total

Name	Type	Description
Deferred Transmissions	Long	IEEE 802.3 Deferred Transmissions
Discards	Integer	Discards, lifetime total
Excessive Collisions	Long	IEEE 802.3 Excessive Collisions
FCS Errors	Long	IEEE 802.3 FCS Errors
Frame Copied Errors	Long	IEEE 802.5 Frame Copied Errors
Frame Too Longs	Long	IEEE 802.3 Frame Too Longs
Hard Errors	Long	IEEE 802.5 Hard Errors
Internal Errors	Long	IEEE 802.5 Internal Errors
Internal Mac Receive Errors	Long	IEEE 802.3 Internal Mac Receive Errors
Internal Mac Transmit Errors	Long	IEEE 802.3 Internal Mac Transmit Errors
Late Collisions	Long	IEEE 802.3 Late Collisions
Line Errors	Long	IEEE 802.5 Line Errors
Lost Frame Errors	Long	IEEE 802.5 Lost Frame Errors
Multicasts Accepted	Long	IEEE 802.3 Multicasts Accepted
Multiple Collision Frames	Long	IEEE 802.3 Multiple Collision Frames
Receive Congestions	Long	IEEE 802.5 Receive Congestions
Signal Loss	Long	IEEE 802.5 Signal Loss
Single Collision Frames	Long	IEEE 802.3 Single Collision Frames
Soft Errors	Long	IEEE 802.5 Soft Errors
SQE Test Errors	Long	IEEE 802.3 SQE Test Errors
Stats Alignment Errors	Long	IEEE 802.3 Stats Alignment Errors

Name	Type	Description
Token Errors	Long	IEEE 802.5 Token Errors
Unknown Protos	Integer	Unknown Protocol frames, lifetime total

Chapter 8

AIX Microagents and Methods

The TIBCO Hawk microagents described in this chapter run on the IBM AIX platform in the HMA process. For more specific information on monitoring on AIX, see the *TIBCO Hawk Installation, Configuration, and Administration* guide.

Topics

- [Summary of AIX Microagents, page 264](#)
- [COM.TIBCO.hawk.hma.System, page 266](#)
- [COM.TIBCO.hawk.hma.Process, page 271](#)
- [COM.TIBCO.hawk.hma.FileSystem, page 275](#)
- [COM.TIBCO.hawk.hma.Network, page 279](#)

Summary of AIX Microagents

The AIX-specific microagents are described as follows.

Table 7 AIX Microagents

Microagent	Description
Common Methods	<p>These methods are common to all microagents and provide release and trace-related functions. Methods are:</p> <p>getReleaseVersion</p> <p>getTraceLevel</p> <p>setTraceLevel</p> <p>getTraceParameters</p> <p>setTraceParameters</p> <p>_onUnsolicitedMsg</p>
System	<p>Used to collect information on system-wide usage of the CPU. Methods are:</p> <p>System:getCpuInfo</p> <p>System:getSwapInfo</p> <p>System:getSystemInfo</p>
Process	<p>Used to obtain information about processes running on the system. Methods are:</p> <p>Process:getProcess</p> <p>Process:getInstanceCount</p> <p>Process:getInstanceCountByCommand</p>
FileSystem	<p>Used to obtain information about file systems connected to the computer. Methods are:</p> <p>FileSystem:getByPartition</p> <p>FileSystem:getByMountPoint</p> <p>FileSystem:getByFileSystem</p>

Table 7 AIX Microagents

Microagent	Description
Network	Used to obtain information about network identification, communications, and errors. Methods are: Network:getConfig Network:getStatistics

COM.TIBCO.hawk.hma.System

Microagent

Purpose The `System` microagent (on AIX) is used to collect information on system-wide usage of the CPU.

Methods	Method	Description	Page
	System:getCpuInfo	Returns information on CPU utilization	267
	System:getSwapInfo	Gets system Swap utilization information	269
	System:getSystemInfo	Returns operating system information	270

System:getCpuInfo

Method

Purpose This method (on AIX) returns information on system CPU utilization.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
% User Time	Double	Percent of the time currently run processes execute in user mode
% System Time	Double	Percent of the time currently run processes execute in kernel mode
% Time Idle	Double	Percent of the time wait process is the current running process
% Time Wait	Double	Percent of the time currently run processes wait for block I/O to complete
Physical Reads	Integer	Total count of physical reads performed by the system, lifetime total
Physical Writes	Integer	Total count of physical writes performed by the system, lifetime total
Logical Reads	Integer	Total count of logical reads performed by the system, lifetime total
Logical Writes	Integer	Total count of logical writes performed by the system, lifetime total
Raw Reads	Integer	Total count of block (raw) reads performed by the system, lifetime total
Raw Writes	Integer	Total count of block (raw) writes performed by the system, lifetime total
Remote Reads	Integer	Total count of remote read requests, lifetime total
Remote Writes	Integer	Total count of remote writes performed by the system, lifetime total

Name	Type	Description
Exec Total	Integer	Total count of <code>exec</code> calls performed by the system, lifetime total
Fork Total	Integer	Total count of <code>fork</code> calls performed by the system, lifetime total
Context Switches	Integer	Total count of the system changing to a different current running process, lifetime total (this counter may wrap)
Device Interrupts	Integer	Total count of device interrupts, lifetime total
Software Interrupts	Integer	Total count of software interrupts, lifetime total
Avg Running	Double	Average number of processes in the run queue
Avg Swapped Out	Double	Average number of processes that are inactive because they are waiting to be paged in.

System:getSwapInfo

Method

Purpose This method (on AIX) gets system Swap utilization information.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Avg KBytes Free	Long	Average free memory (kilobytes)
Avg KBytes Unallocated	Long	Average unallocated swap space (kilobytes)
KBytes Used	Long	Amount of disk space on the mounted file system currently in use, expressed as absolute value (kilobytes)
KBytes Total	Long	Total amount of disk space on mounted file system (kilobytes)
% Free	Double	Amount of disk space that is currently free, expressed as percentage of total size
% Used	Double	Amount of disk space that is currently used, expressed as percentage of total size

System:getSystemInfo

Method

Purpose This method (on AIX) returns operating system information.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Up Time	Integer	The system up time in seconds. (Time since last reboot.)
Real Memory	Integer	The total real memory (in kilobytes)
% Free Real Memory	Integer	The total percentage of free real memory
Avg Running	Double	Average number of processes running over 1 minute
Avg Running (5 min)	Double	Average number of processes running over 5 minutes
Avg Running (15 min)	Double	Average number of processes running over 15 minutes
% Time Usage	Integer	% Time Usage
Total Processes	Integer	The total processes currently on the system

COM.TIBCO.hawk.hma.Process

Microagent

Purpose The `Process` microagent (on AIX) furnishes information about processes running on the system.

Methods	Method	Description	Page
	Process:getProcess	Returns information on a process	272
	Process:getInstanceCount	Returns the number of instances of a process	273
	Process:getInstanceCount ByCommand	Returns the number of instances of a process, filtered by command	274

Process:getProcess

Method

Purpose This method (on AIX) returns a process table filtered by the `Process Name` argument.

Remarks Test any regular expressions you plan to use in rulebases by first using them interactively to insure they return the desired results.

If no arguments are given, the entire table is returned.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Process Name	String	Process name. Pattern match with regular expressions. Empty argument string returns information for all processes.

Returns

Name	Type	Description
Process Name	String	Process name
ID Process	Integer	The process ID of the process (this datum required in order to kill the process). Table is indexed on ID Process.
Parent Process ID	Integer	Parent process ID of the current process
User Name	String	Effective user name of the process
Status	String	Status of process
Virtual KBytes	Integer	The size of virtual address space (kilobytes)
CPU Time	Integer	CPU usage (user time + system time, milliseconds)
% CPU	Integer	The ratio of CPU time used to CPU time available computed for the lifetime of the process
% Memory	Integer	The ratio of process's resident set size to the physical memory on the machine
Command	String	The actual command, along with its argument

Process:getInstanceCount

Method

Purpose This method (on AIX) returns an instance count for the process denoted by the `Process Name` argument.

Remarks Test any regular expressions you plan to use in rulebases by first using them interactively to insure they return the desired results.

If no arguments are given, the total count of all running processes is returned.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Process Name	String	Process name. Pattern match with regular expressions. Empty argument string returns combined instance count for all processes.

Returns

Name	Type	Description
Process Name	String	Process name. Table is indexed on <code>Process Name</code> .
Process Count	Integer	Instance count for a currently running process

Process:getInstanceCountByCommand

Method

Purpose This method (on AIX) returns a count of processes, filtered by command.

Remarks Regular expressions can be used to query status of a group of processes.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Command	String	Command-line argument.

Returns

Name	Type	Description
Command	String	Command-line argument
Process Count	Integer	Process count for specified command

COM.TIBCO.hawk.hma.FileSystem

Microagent

Purpose The `FileSystem` microagent (on AIX) furnishes information about file systems connected to the computer.

Methods	Method	Description	Page
	FileSystem:getByPartition	Returns information on a file system for its partition name	276
	FileSystem:getByMountPoint	Returns information on a file system for its mount point	277
	FileSystem:getByFileSystem	Returns utilization statistics for the currently mounted filesystems	278

FileSystem:getByPartition

Method

Purpose This method (on AIX) provides utilization statistics for currently mounted file systems filtered by the device given as the `Partition` argument. Information will be listed for only local partitions with non-zero data blocks.

Remarks NFS mounted file systems are ignored and should be monitored by agents running on the file servers themselves.

If no argument is specified, the entire table is returned.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Partition	String	Device name (hard disk partition). Pattern match with regular expressions. Empty argument string returns information for all partitions.

Returns

Name	Type	Description
Partition	String	Device name of a hard disk partition. Table is indexed on Partition.
Mount Point	String	Mount Point
% Free	Integer	Amount of disk space on the mounted file system that is currently free, expressed as percentage of total size
% Used	Integer	Amount of disk space on the mounted file system currently in use, expressed as percentage of total size
KBytes Free	Long	Amount of disk space on the mounted file system that is currently free, expressed as absolute value (kilobytes)
KBytes Used	Long	Amount of disk space on the mounted file system currently in use, expressed as absolute value (kilobytes)
KBytes Total	Long	Total amount of disk space on mounted file system (kilobytes)

FileSystem:getByMountPoint

Method

Purpose This method (on AIX) provides utilization statistics for currently mounted file systems filtered by the mount point given as the argument `Mount Point`. Information will be listed for only local partitions with non-zero data blocks.

Remarks NFS mounted file systems are ignored and should be monitored by agents running on the file servers themselves. If no argument is specified, the entire table is returned.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Mount Point	String	Name of the directory where file system is mounted. Pattern match with regular expressions. Empty argument string returns information for all mount points.

Returns

Name	Type	Description
Partition	String	Device name of a hard disk partition
Mount Point	String	Mount point. Table is indexed on Mount Point.
% Free	Integer	Amount of disk space on the mounted file system that is currently free, expressed as percentage of total size
% Used	Integer	Amount of disk space on the mounted file system currently in use, expressed as percentage of total size
KBytes Free	Long	Amount of disk space on the mounted file system that is currently free, expressed as absolute value (kilobytes)
KBytes Used	Long	Amount of disk space on the mounted file system currently in use, expressed as absolute value (kilobytes)
KBytes Total	Long	Total amount of disk space on mounted file system (kilobytes)

FileSystem:getByFileSystem

Method

Purpose This method (on AIX) provides utilization statistics for the currently mounted filesystems filtered by device name and file system type. The argument is treated as a pattern to match. If no argument is specified, the entire mount table is reported.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
FileSystem Type	String	Name of the file system type

Returns

Name	Type	Description
Partition	String	Device name of a hard disk partition
Mount Point	String	Mount point of device. Table is indexed on Mount Point
File System Type	String	Name of the file system type
% Free	Integer	Amount of disk space on the mounted file system that is currently free, expressed as percentage of total size
% Used	Integer	Amount of disk space on the mounted file system currently in use, expressed as percentage of total size
KBytes Free	Long	Amount of disk space on the mounted file system that is currently free, expressed as absolute value (kilobytes)
KBytes Used	Long	Amount of disk space on the mounted file system currently in use, expressed as absolute value (kilobytes)
KBytes Total	Long	Total amount of disk space on mounted file system (kilobytes)
% Minimum	Integer	Minimum Percentage free

COM.TIBCO.hawk.hma.Network

Microagent

Purpose The Network microagent (on AIX) furnishes information about network utilization statistics, identification, communications, and errors.

Methods	Method	Description	Page
	Network:getStatistics	Returns network performance information	280
	Network:getConfig	Returns information on network addresses and masks	282

Network:getStatistics

Method

Purpose This method (on AIX) returns network performance information in a manner similar to using the `netstat` command. This method returns statistics only for the following device types

- Ethernet (IEEE 802.3)
- Ethernet (Standard, Version2).

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Interface Name	String	Name of the network interface

Returns

Name	Type	Description
Interface Name	String	Name of the network interface. Table is indexed on Interface Name.
Input Bytes	Long	The number of octets of data received through the interface
Input Packets	Integer	Packets received on the interface, lifetime total
Input Packet Errors	Integer	Packet errors received on the interface, lifetime total
Output Bytes	Long	The number of octets of data sent through this interface
Output Packets	Integer	Packets sent on the interface, lifetime total
Output Packet Errors	Integer	Packet errors sent on the interface, lifetime total
Network Collisions	Integer	Collisions, lifetime total
Framing Errors	Integer	Framing errors, lifetime total
No Carrier Errors	Integer	No carrier errors, lifetime total

Name	Type	Description
% Network Collisions	Double	Rate (percentage) of collisions against total number of output packets, lifetime total

Network:getConfig

Method

Purpose This method (on AIX) returns network configuration information in a manner similar to using the `ifconfig` command.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Interface Name	String	Name of the network interface

Returns

Name	Type	Description
Interface Name	String	Name of the network interface. Table is indexed on Interface Name
Network Address	String	Network Address in dotted format, 32 bit Value
Hardware Address	String	Hardware Address in colon format
Broadcast Address	String	Broadcast address for the interface
Network Mask	String	Network mask for the interface
MTU	String	Maximum transmission unit

Chapter 9

Linux Microagents and Methods

The TIBCO Hawk microagents described in this chapter run on the Linux platform in the HMA process. For more specific information on monitoring, see the *TIBCO Hawk Installation, Configuration, and Administration* guide.

Topics

- [Summary of Linux Microagents, page 284](#)
- [COM.TIBCO.hawk.hma.System, page 286](#)
- [COM.TIBCO.hawk.hma.Process, page 295](#)
- [COM.TIBCO.hawk.hma.FileSystem, page 303](#)
- [COM.TIBCO.hawk.hma.Network, page 308](#)

Summary of Linux Microagents

The Linux microagents are described as follows..

Table 8 Linux Microagents (Sheet 1 of 2)

Microagent	Description
Common Methods	<p>These methods are common to all microagents and provide release and trace-related functions. Methods are:</p> <p>getReleaseVersion</p> <p>getTraceLevel</p> <p>setTraceLevel</p> <p>getTraceParameters</p> <p>setTraceParameters</p> <p>_onUnsolicitedMsg</p>
System	<p>Collects information on system-wide usage of the CPU and memory. Methods are:</p> <p>System:getSwapInfo</p> <p>System:getSystemInfo</p> <p>System:getCpuInfo</p> <p>System:getTunableInfo</p>
Process	<p>Used to obtain information about processes running on the system. Methods are:</p> <p>Process:getProcess</p> <p>Process:getInstanceCount</p> <p>Process:getInstanceCountByCommand</p>
FileSystem	<p>Collects information about file systems associated with the computer. Methods are:</p> <p>FileSystem:getByPartition</p> <p>FileSystem:getByMountPoint</p> <p>FileSystem:getByFileSystem</p>

Table 8 Linux Microagents (Sheet 2 of 2)

Microagent	Description
Network	Collects information about network identification communications and errors. Methods are: Network:getStatistics Network:getConfig

COM.TIBCO.hawk.hma.System

Microagent

Purpose The System microagent (on Linux) is used to collect information on system-wide usage of the CPU and memory, summarizing all processes on the computer.

Methods	Method	Description	Page
	System:getSwapInfo	Returns information on system swap space	287
	System:getSystemInfo	Returns information on various operating system parameters	288
	System:getCpuInfo	Returns system CPU utilization information	293
	System:getTunableInfo	Returns information on system tunable parameters	294

System:getSwapInfo

Method

Purpose This method (on Linux) returns operating system swap space utilization information.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Avg KBytes Free	Integer	The average free memory, in KB
Avg KBytes Unallocated	Integer	Average unallocated swap space, in KB
KBytes Used	Integer	The number of kilobytes of swap space used
KBytes Total	Integer	The number of kilobytes total swap space
% Free	Double	The amount free swap space, expressed as a percentage
% Used	Double	The amount of swap space used, expressed as a percentage

System:getSystemInfo

Method

Purpose This method (on Linux) returns operating system information.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Up Time	Integer	The system up time in seconds. (Time since last reboot.)
Run Queue Length	Integer	The length of the run queue (number of processes)
Real Memory	Integer	The total real memory (in kilobytes)
Free Memory	Integer	The total free memory (in kilobytes)
% Free Real Memory	Integer	The total percentage of free real memory
% Buffers Memory	Integer	The total percentage of memory used to cache file or disk blocks
% Cached Memory	Integer	The total percentage of memory used to cache page files
Avg Running	Double	Average number of processes running over 1 minute
Avg Running (5 min)	Double	Average number of processes running over 5 minutes
Avg Running (15 min)	Double	Average number of processes running over 15 minutes
% Time Usage	Double	% Time Usage
Total Processes	Integer	The total processes currently on the system
Context Switches	Long	The total number of context switches of the CPU from one process to another

Name	Type	Description
Device Interrupts	Long	The total number of device interrupts currently on the computer system
Buffers	Long	The amount of physical RAM, in kilobytes, used for file buffer
Cache Memory	Long	The amount of physical RAM, in kilobytes, used as cache memory
SwapCached	Long	The amount of swap, in kilobytes, used as cache memory
Page Cache Memory Active	Long	The total amount of buffer or page cache memory, in kilobytes, that is in active use. This is memory that has been recently used and is usually not reclaimed for other purposes.
Page Cache Memory Free	Long	The total amount of buffer or page cache memory, in kilobytes, that are free and available. This is memory that has not been recently used and can be reclaimed for other purposes.
High Total Memory	Long	The total amount of memory, in kilobytes, that is not directly mapped into kernel space
High Free Memory	Long	The free amount of memory, in kilobytes, that is not directly mapped into kernel space
Low Total Memory	Long	The total amount of memory, in kilobytes, that is directly mapped into kernel space
Low Free Memory	Long	The free amount of memory, in kilobytes, that is directly mapped into kernel space
Dirty Memory	Long	The total amount of memory, in kilobytes, waiting to be written back to the disk

Name	Type	Description
WriteBack Memory	Long	The total amount of memory, in kilobytes, actively being written back to the disk
Mapped Memory	Long	The total amount of memory, in kilobytes, which have been used to map devices, files, or libraries using the mmap command
Slab Memory	Long	The total amount of memory, in kilobytes, used by the kernel to cache data structures for its own use
Committed_AS	Long	The total amount of memory, in kilobytes, estimated to complete the workload. This value represents the worst case scenario value, and also includes swap memory
PageTable Memory	Long	The total amount of memory, in kilobytes, dedicated to the lowest page table level
VmallocTotal	Long	The total amount of memory, in kilobytes, of total allocated virtual address space
VMallocUsed Memory	Long	The total amount of memory, in kilobytes, of used virtual address space
VmallocChunk Memory	Long	The largest contiguous block of memory, in kilobytes, of available virtual address space
HugePages_Total	Long	The total number of hugepages for the system
HugePages_Free	Long	The total number of hugepages available for the system
Hugepagesize	Long	The size for each hugepages unit in kilobytes
Active(anon)	Long	Active anonymous pages
Inactive(anon)	Long	Inactive anonymous pages
Active(file)	Long	Active page cache

Name	Type	Description
Inactive(file)	Long	Inactive page ache
Unevictable	Long	These pages aren't going anywhere, they are pinned, cannot be swapped or reclaimed. Includes kernel pages and Mlocked pages
Mlocked	Long	Just mlocked() pages from userspace, amount of allocated memory that's strictly prohibited from being paged out
AnonPages	Long	(Anonymous Pages) field denotes allocated pages which have no backing storage, will include anonymous hugepages
Shmem	Long	In-kernel data structures cache, depicts the amount of shared memory used by group(s) processes. Size of shared memory, anybody who did mmap() with MAP_SHARED MAP_ANONYMOUS
Sreclaimable	Long	Part of Slab, that might be reclaimed, such as caches
Sunreclaim	Long	Part of Slab, that cannot be reclaimed on memory pressure
KernelStack	Long	Pages used for kernel stack space
NFS_Unstable	Long	NFS pages sent to the server, but not yet committed to stable storage
Bounce	Long	Memory used for block device 'bounce buffers'
WritebackTmp	Long	Memory used by FUSE for temporary writeback buffers
CommitLimit	Long	This is the total amount of memory currently available to be allocated on the system, expressed in kilobytes. This limit is adhered to only if strict overcommit accounting is enabled

Name	Type	Description
HardwareCorrupted	Long	The amount of memory in "poisoned pages", for example, memory which has failed (as flagged by Error Correcting Code (ECC) typically). ECC memory is capable of correcting small errors and detecting larger ones. If an error that cannot be corrected is detected using ECC (in memory or cache, depending on the system's hardware support), then the Linux kernel marks the corresponding page as poisoned.
AnonHugePages	Long	Non-file backed huge pages mapped into user-space page tables
HugePages_Rsvd	Long	This is the number of huge pages for which a commitment to allocate from the pool has been made, but no allocation has yet been made. These reserved huge pages guarantee that an application will be able to allocate a huge page from the pool of huge pages at fault time
HugePages_Surp	Long	This is the number of huge pages in the pool
DirectMap4k	Long	Counts the number of pages mapped as 4kB pages. It gives an indication of the load for Translation Lookaside Buffer (TLB), cache used to store mappings between virtual addresses and physical pages in memory.
DirectMap2M	Long	Counts the number of pages mapped as 2MB pages. It gives an indication of the load for Translation Lookaside Buffer (TLB), cache used to store mappings between virtual addresses and physical pages in memory.

System:getCpuInfo

Method

Purpose This method (on Linux) returns operating system CPU utilization information. On multi-processor machines, CPU utilization for individual processors are returned.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Processor	Integer	Processor (slot) number
% User Time	Integer	The percent of time spent in user mode for the processor
% System Time	Integer	The percent of time spent in system mode for the processor
% Time Idle	Integer	The percent of time spent in idle state for the processor
IO Waits	Long	IO Waits
Hardware IRQ	Long	Hardware interrupts (IRQ)
Software IRQ	Long	Software interrupts (IRQ)

System:getTunableInfo

Method

Purpose This method (on Linux) returns operating system tunable parameter information.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Page Size	Integer	Page size (in bytes)
Inodes Allocated	Long	Denotes the number of inodes the system has allocated. This number will grow and shrink dynamically.
Inodes Free	Long	Represents the number of free inodes
Maximum Files	Integer	The maximum number of files a process can have open simultaneously

COM.TIBCO.hawk.hma.Process

Microagent

Purpose The `Process` microagent (on Linux) furnishes information about processes running on the system.

Methods	Method	Description	Page
	Process:getProcess	Returns information on a process	296
	Process:getInstanceCount	Returns the number of instances of a process	301
	Process:getInstanceCountByCommand	This method returns a count of processes, filtered by argument	302

Process:getProcess

Method

Purpose On Linux, returns a process table filtered by the `Process Name` argument.

Remarks The argument is treated as a pattern to match. If no argument string is given, all processes are returned.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Process Name	String	Process name. Empty argument returns information for all processes.

Returns

Name	Type	Description
Process Name	String	Name of the process being executed
ID Process	Integer	Process ID of the process. Table is indexed on ID process.
Parent Process ID	Integer	Parent process ID of the current process
User Name	String	Effective user name of the process
Status	String	Status of process
Virtual KBytes	Integer	The size of virtual address space, in kilobytes
Stack KBytes	Integer	The size of stack space, in kilobytes
Heap KBytes	Integer	The size of heap space, in kilobytes
CPU Time	Integer	CPU usage: the user time plus system time, in milliseconds.
% CPU	Integer	Percent of CPU used
% Memory	Integer	Percent of memory used
Major Page Faults	Integer	Major page faults since process start
Minor Page Faults	Integer	Minor page faults since process start
Command	String	Full command line with all arguments

Name	Type	Description
Start Time	Integer	The number of seconds since the process started
Number of threads	Integer	Number of threads in this process
Characters Written	Long	The number of bytes which this task has caused, or shall cause to be written to disk
Characters Read	Long	The number of bytes which this process has caused to be read from storage
Read Syscalls	Long	Attempt to count the number of read I/O operations
Write Syscalls	Long	Attempt to count the number of write I/O operations
Bytes read	Long	Attempt to count the number of bytes which this process really did cause to be fetched from the storage layer.
Bytes written	Long	Attempt to count the number of bytes which this process caused to be sent to the storage layer
Cancelled_write_bytes	Long	This field represents the number of bytes which this process caused to not happen, by truncating pagecache
RSS	Double	Resident Set Size: number of pages the process has in real memory
VmPeak	Long	Peak virtual memory size in Kbytes
VmPin	Long	Pinned memory size. It is size of pinned pages that cannot be swapped. (available since Linux 3.2)
VmLCK	Long	Locked memory size in KBytes
VmHWM	Long	Peak resident set size in KBytes
VmPTE	Long	Page table entries size in KBytes

Name	Type	Description
VmRSS	Long	Resident set size in KBytes
VmData	Long	Size of data in KBytes
VmStk	Long	Size of stack in KBytes
VmExe	Long	Size of text segments in KBytes
VmLib	Long	Shared library code size in KBytes
VmSwap	Long	Swap Space Used in Kbytes
PGID	Long	The process group ID of the process
SID	Integer	The session ID of the process
TTY_NR	Integer	The controlling terminal of the process
TTY_PGRP	Integer	The ID of the foreground process group of the controlling terminal of the process
FLAGS	Integer	The kernel flags word of the process
CMAJ_FLT	Long	The number of major faults that the process's waited-for children have made
CMIN_FLT	Long	The number of minor faults that the process's waited-for children have made
STIME	Double	Amount of time that this process has been scheduled in kernel mode
CSTIME	Double	Amount of time that this process's waited-for children have been scheduled in kernel mode
UTIME	Double	Amount of time that this process has been scheduled in user mode
CUTIME	Double	Amount of time that this process's waited-for children have been scheduled in user mode
Priority	Long	Process scheduling Priority
NICE	Long	The nice value, a value in the range 19 (low priority) to -20 (high priority)

Name	Type	Description
IT_REAL_VALUE	Long	The time in jiffies before the next SIGALRM is sent to the process due to an interval timer
Start_Code	Long	The address above which program text can run
End_Code	Long	The address below which program text can run
Start_Stack	Long	The address of the start (bottom) of the stack
ESP	Long	The current value of ESP (stack pointer), as found in the kernel stack page for the process
EIP	Long	The current EIP (instruction pointer)
Pending	Long	The bitmap of pending signals, displayed as a decimal number
Blocked	Long	The bitmap of blocked signals, displayed as a decimal number
SIGIN	Long	The bitmap of ignored signals
SIGCATCH	Long	The bitmap of caught signals
WCHAN	Double	This is the channel in which the process is waiting. It is the address of a location in the kernel where the process is sleeping
NSWAP	Long	Number of pages swapped
CNSWAP	Long	Cumulative nswap for child processes
EXIT_SIGNAL	Long	Signal to send to parent thread on exit. (available in Linux 2.1.22 and later)
Processor	Long	CPU number last executed on. (available in Linux 2.2.8 and later)
RT_PRIORITY	Long	Real-time scheduling priority, a number in the range 1 to 99 for processes scheduled under a real-time policy, or 0, for non-real-time processes. (available in Linux 2.5.19 and later)

Name	Type	Description
Policy	Long	Scheduling policy. Decode using the SCHED_* constants in linux/sched.h. (available in Linux 2.5.19 and later)

Process:getInstanceCount

Method

Purpose This method (on Linux) returns an instance count for the process denoted by the `Process Name` argument.

Remarks The argument is treated as a pattern to match. If no argument string is given, the count of all processes is returned.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Process Name	String	Process name. Empty string returns combined instance count for all processes.

Returns

Name	Type	Description
Process Name	String	Process name. Table is indexed on Process Name.
Process Count	Integer	Number of instances

Process:getInstanceCountByCommand

Method

Purpose This method (on Linux) returns a count of processes, filtered by argument.

Remarks Regular expressions can be used to query status of a group of processes.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Command	String	Command-line argument

Returns

Name	Type	Description
Command	String	Command-line argument
Process Count	Integer	Process count for specified command

COM.TIBCO.hawk.hma.FileSystem

Microagent

Purpose The FileSystem microagent (on Linux) is used to collect information on file systems connected to this computer.

Methods	Method	Description	Page
	FileSystem:getByPartition	Returns utilization statistics for currently mounted file systems filtered by device name	304
	FileSystem:getByMountPoint	Returns utilization statistics for currently mounted file systems filtered by mount directory	305
	FileSystem:getByFileSystem	Returns utilization statistics for the currently mounted filesystems	306

FileSystem:getByPartition

Method

Purpose This method (on Linux) returns utilization statistics for currently mounted file systems, filtered by device name. Information will be listed for only local partitions with non-zero data blocks.

Remarks The argument is being treated as a pattern to match. If no argument string is given the whole mount table is reported.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Partition	String	Device name (hard disk partition)

Returns

Name	Type	Description
Partition	String	Device name of a hard disk partition. Table is indexed by Partition.
Mount Point	String	Mount point
% Free	Integer	The amount of disk space on the mounted file system that is currently free, expressed as a percentage of total size
% Used	Integer	The amount of disk space on the mounted file system that is currently in use, expressed as a percentage of total size
KBytes Free	Long	The amount of disk space on the mounted file system that is currently free, expressed as an absolute value in kilobytes
KBytes Used	Long	The amount of disk space on the mounted file system that is currently in use, expressed as an absolute value in kilobytes
KBytes Total	Long	The amount of disk space on the mounted file system, expressed in kilobytes
% Minimum	Integer	The minimum % free (for super user only)

FileSystem:getByMountPoint

Method

Purpose This method (on Linux) returns utilization statistics for currently mounted file systems filtered by the mount directory. Information will be listed for only local partitions with non-zero data blocks.

Remarks The argument is being treated as a pattern to match. If no argument is given, the entire mount table is reported.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Mount Point	String	Name of the directory where the file system is mounted

Returns

Name	Type	Description
Mount Point	String	Mount point. Table is indexed by Mount Point.
Partition	String	Device name of a hard disk partition
% Free	Integer	The amount of disk space on the mounted file system that is currently free, expressed as a percentage of total size
% Used	Integer	The amount of disk space on the mounted file system that is currently in use, expressed as a percentage of total size
KBytes Free	Long	The amount of disk space on the mounted file system that is currently free, expressed as an absolute value in kilobytes
KBytes Used	Long	The amount of disk space on the mounted file system that is currently in use, expressed as an absolute value in kilobytes
KBytes Total	Long	The amount of disk space on the mounted file system, expressed in kilobytes
% Minimum	Integer	The minimum % free (for super user only)

FileSystem:getByFileSystem

Method

Purpose This method (on Linux) provides utilization statistics for the currently mounted filesystems filtered by device name and file system type. The argument is treated as a pattern to match. If no argument is specified, the entire mount table is reported.

For any filesystem that does not respond within five seconds, the data values for % Free and % Used are considered stale and their values are set to -1.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
FileSystem Type	String	Name of the file system type

Returns

Name	Type	Description
Partition	String	Device name of a hard disk partition
Mount Point	String	Mount point of device. Table is indexed on Mount Point.
File System Type	String	Name of the file system type
% Free	Integer	Amount of disk space on the mounted file system that is currently free, expressed as percentage of total size
% Used	Integer	Amount of disk space on the mounted file system currently in use, expressed as percentage of total size
KBytes Free	Long	Amount of disk space on the mounted file system that is currently free, expressed as absolute value (kilobytes)
KBytes Used	Long	Amount of disk space on the mounted file system currently in use, expressed as absolute value (kilobytes)
KBytes Total	Long	Total amount of disk space on mounted file system (kilobytes)

Name	Type	Description
% Minimum	Integer	Minimum Percentage free

COM.TIBCO.hawk.hma.Network

Microagent

Purpose The `Network` microagent (on Linux) furnishes information about network utilization statistics, identification, communications, and errors.

Methods	Method	Description	Page
	Network:getStatistics	Returns network performance information	309
	Network:getConfig	Returns information on network addresses and masks	310

Network:getStatistics

Method

Purpose This method (on Linux) returns network performance information in a manner similar to using the `netstat` command.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Interface Name	String	Name of the network interface

Returns

Name	Type	Description
Interface Name	String	Name of the network interface. Table is indexed on Interface Name.
Input Bytes	Long	The number of octets of data received through the interface
Input Packets	Integer	Packets received on the interface, lifetime total
Input Packet Errors	Integer	Packet errors received on the interface, lifetime total
Output Bytes	Long	The number of octets of data sent through this interface
Output Packets	Integer	Packets sent on the interface, lifetime total
Output Packet Errors	Integer	Packet errors sent on the interface, lifetime total
Network Collisions	Integer	Collisions, lifetime total
Framing Errors	Integer	Framing errors, lifetime total
No Carrier Errors	Integer	No carrier errors, lifetime total
% Network Collisions	Double	Rate (percentage) of collisions against total number of output packets, lifetime total

Network:getConfig

Method

Purpose This method (on Linux) returns network configuration information in a manner similar to using the `ifconfig` command.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Interface Name	String	Name of the network interface

Returns

Name	Type	Description
Interface Name	String	Name of the network interface. Table is indexed on Interface Name
Network Address	String	Network Address in dotted format, 32 bit Value
Hardware Address	String	Hardware Address in colon format
Broadcast Address	String	Broadcast address for the interface
Network Mask	String	Network mask for the interface
MTU	String	Maximum transmission unit
Speed (Mb/s)	String	Speed of Network Interface

Chapter 10 **Mac OS X Methods**

The TIBCO Hawk microagents described in this chapter run on the Mac OS X platform in the HMA process. For more specific information on monitoring, see the *TIBCO Hawk Installation, Configuration, and Administration* guide.

Topics

- [Summary of Mac OS X Microagents, page 314](#)
- [COM.TIBCO.hawk.hma.System, page 316](#)
- [COM.TIBCO.hawk.hma.Process, page 322](#)
- [COM.TIBCO.hawk.hma.FileSystem, page 327](#)
- [COM.TIBCO.hawk.hma.Network, page 331](#)

Summary of Mac OS X Microagents

The Mac OS X microagents are described as follows.

Table 9 Mac OS X Microagents (Sheet 1 of 2)

Microagent	Description
Common Methods	<p>These methods are common to all microagents and provide release and trace-related functions. Methods are:</p> <p>getReleaseVersion</p> <p>getTraceLevel</p> <p>setTraceLevel</p> <p>getTraceParameters</p> <p>setTraceParameters</p> <p>_onUnsolicitedMsg</p>
System	<p>Used to collect information on system-wide usage of the CPU and memory. Methods are:</p> <p>System:getSystemInfo</p> <p>System:getCpuInfo</p> <p>System:getProcessorSetInfo</p> <p>System:getSwapInfo</p> <p>System:getTunableInfo</p>
Process	<p>Used to obtain information about processes running on the system. Methods are:</p> <p>Process:getProcess</p> <p>Process:getInstanceCount</p> <p>Process:getInstanceCountByCommand</p>

Table 9 Mac OS X Microagents (Sheet 2 of 2)

Microagent	Description
FileStat	Used to obtain information about file systems connected to the computer. Methods are: FileSystem:getByPartition FileSystem:getByMountPoint FileSystem:getByFileSystem
Network	Used to obtain information about network identification communications and errors. Methods are: Network:getStatistics Network:getConfig

COM.TIBCO.hawk.hma.System

Microagent

Purpose The `System` microagent (on Mac OS X) is used to collect information on system-wide usage of the CPU and memory, summarizing all processes on the computer.

Methods	Method	Description	Page
	System:getSystemInfo	Returns information on system resource utilization	317
	System:getCpuInfo	Returns information on CPU utilization	318
	System:getProcessorSetInfo	Returns information on processor set utilization	319
	System:getSwapInfo	Returns information on the utilization of system swap space	320
	System:getTunableInfo	Returns information on system tunable variables	321

System:getSystemInfo

Method

Purpose This method (on Mac OS X) returns information that pertains to multi-tasking process management by the system, including process swapping and run queuing. It also returns information on the memory usage.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Up Time	Long	System up time in seconds
Real Memory	Long	Total real memory in kilobytes
Free Memory	Long	Free Memory in kilobytes
% Free Real Memory	Double	% Free Real Memory
Avg Running	Double	Average number of processes running over 1 minute
Avg Running (5 min)	Double	Average number of processes running over 5 minutes
Avg Running (15 min)	Double	Average number of processes running over 15 minutes
% Time Usage	Integer	% Time Usage
% User Time	Double	Percent of the time currently run processes execute in user mode
% System Time	Double	Percent of the time currently run processes execute in kernel mode
% Time Idle	Double	Percent of the time wait process is the currently- running process

System:getCpuInfo

Method

Purpose This method (on Mac OS X) returns information on processor activity for all processors on the computer.

Type Synchronous, `IMPACT_INFO`.

Arguments None.

Returns

Name	Type	Description
Processor	Integer	Processor (slot) number
Running	Integer	Value of one (1) indicates the processor is running
Master	Integer	Value of one (1) indicates that the processor is the master processor
% User Time	Double	Percent of the time currently run processes execute in user mode
% System Time	Double	Percent of the time currently run processes execute in kernel mode
% Time Idle	Double	Percent of the time when the CPU is in idle state

System:getProcessorSetInfo

Method

Purpose This method (on Mac OS X) returns information on processor set utilization.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Processor Set	Integer	Processor (slot) number
CPU Count	Integer	Number of CPUs in the processor set.
Load Average	Double	Load average of the processor set is a measure of how much processing resources is being utilized
Mach Factor	Double	A variant of the Load Average which indicates how much processing resources is available
Total Processes	Integer	Total processes in the processor set
Total Threads	Integer	Total threads in the processor set

System:getSwapInfo

Method

Purpose This method (on Mac OS X) returns information on the utilization of system swap space.

Type Synchronous, `IMPACT_INFO`.

Arguments None.

Returns

Name	Type	Description
Page Size	Long	The size of a page in bytes
Page Free	Long	The total number of free pages in the system
Page Active	Long	The total number of pages currently in use and pageable
Page Inactive	Long	The total number of pages on the inactive list
Page Wired Down	Long	The total number of pages that cannot be paged out
Page Copy-On-Write	Long	The number of faults that caused a page to be copied
Page Reactivated	Long	The total number of pages that have been moved from the inactive list to the active list
Pageins	Long	The number of requests for pages from a pager
Pageouts	Long	The number of pages that have been paged out
Hit Rate	Double	The percentage of page hits over page lookup

System:getTunableInfo

Method

Purpose This method (on Mac OS X) returns tunable system information.

Type Synchronous, IMPACT_INFO.

Arguments None.

Returns

Name	Type	Description
Max. Vnodes	Integer	Maximum number of Vnodes
Max. Processes	Integer	Maximum number of processes
Maximum Files	Integer	Maximum Files
Maximum Files/Process	Integer	Maximum number of open files per process
Maximum Process/User	Integer	Maximum number of processes per user

COM.TIBCO.hawk.hma.Process

Microagent

Purpose The `Process` microagent (on Mac OS X) furnishes information about processes running on the system.

Methods	Method	Description	Page
	Process:getProcess	Returns information on a process	323
	Process:getInstanceCount	Returns the number of instances of a process	325
	Process:getInstanceCountByCommand	This method returns a count of processes, filtered by argument	326

Process:getProcess

Method

Purpose This method (on Mac OS X) returns a process table filtered by the `Process Name` argument.

Remarks The argument is treated as a pattern to match. If no argument string is given, all processes are returned.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Process Name	String	Process name. Empty argument returns information for all processes.

Returns

Name	Type	Description
Process Name	String	Name of the process being executed
ID Process	Integer	Process ID of the process. Table is indexed on ID process.
Parent Process ID	Integer	Parent process ID of the current process
User Name	String	Effective user name of the process
Status	String	Status of process
State	String	Process status
Virtual KBytes	Integer	The size of virtual address space, in kilobytes
Real KBytes	Integer	The size of the real address space, in kilobytes
Threads	Integer	The number of threads in the process
CPU Time	Double	CPU usage: the user time plus system time, in milliseconds.
% CPU	Double	Percent of CPU used
% Memory	Double	Percent of memory used
Page Faults	Integer	The number of page faults

Name	Type	Description
COW faults	Integer	The number of copy-on-write faults
Pageins	Integer	The number of actual pageins
System Calls	Long	The number of system calls made
Command	String	Command line used to start process
Start Time	Integer	The number of seconds since the process started

Process:getInstanceCount

Method

Purpose This method (on Mac OS X) returns an instance count for the process denoted by the `Process Name` argument.

Remarks The argument is treated as a pattern to match. If no argument string is given, the count of all processes is returned.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Process Name	String	Process name. Empty string returns combined instance count for all processes.

Returns

Name	Type	Description
Process Name	String	Process name. Table is indexed on Process Name.
Process Count	Integer	Number of instances

Process:getInstanceCountByCommand

Method

Purpose This method (on Mac OS X) returns a count of processes, filtered by argument.

Remarks Regular expressions can be used to query status of a group of processes.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Command	String	Command-line argument

Returns

Name	Type	Description
Command	String	Command-line argument
Process Count	Integer	Process count for specified command

COM.TIBCO.hawk.hma.FileSystem

Microagent

Purpose The FileSystem microagent (on Mac OS X) is used to collect information on file systems connected to this computer.

Methods	Method	Description	Page
	FileSystem:getByPartition	Returns utilization statistics for currently mounted file systems filtered by device name	328
	FileSystem:getByMountPoint	Returns utilization statistics for currently mounted file systems filtered by mount directory	329
	FileSystem:getByFileSystem	Returns utilization statistics for the currently mounted filesystems	330

FileSystem:getByPartition

Method

Purpose This method (on Mac OS X) returns utilization statistics for currently mounted file systems, filtered by device name. Information will be listed for only local partitions with non-zero data blocks.

Remarks The argument is being treated as a pattern to match. If no argument string is given the whole mount table is reported.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Partition	String	Device name (hard disk partition)

Returns

Name	Type	Description
Partition	String	Device name of a hard disk partition. Table is indexed by Partition.
Mount Point	String	Mount point
% Free	Integer	The amount of disk space on the mounted file system that is currently free, expressed as a percentage of total size
% Used	Integer	The amount of disk space on the mounted file system that is currently in use, expressed as a percentage of total size
KBytes Free	Long	The amount of disk space on the mounted file system that is currently free, expressed as an absolute value in kilobytes
KBytes Used	Long	The amount of disk space on the mounted file system that is currently in use, expressed as an absolute value in kilobytes
KBytes Total	Long	The amount of disk space on the mounted file system, expressed in kilobytes
% Minimum	Integer	The minimum % free (for super user only)

FileSystem:getByMountPoint

Method

Purpose This method (on Mac OS X) returns utilization statistics for currently mounted file systems filtered by the mount directory. Information will be listed for only local partitions with non-zero data blocks.

Remarks The argument is being treated as a pattern to match. If no argument is given, the entire mount table is reported.

Type Synchronous, IMPACT_INFO.

Arguments

Name	Type	Description
Mount Point	String	Name of the directory where the file system is mounted.

Returns

Name	Type	Description
Mount Point	String	Mount point. Table is indexed by Mount Point.
Partition	String	Device name of a hard disk partition
% Free	Integer	The amount of disk space on the mounted file system that is currently free, expressed as a percentage of total size
% Used	Integer	The amount of disk space on the mounted file system that is currently in use, expressed as a percentage of total size
KBytes Free	Long	The amount of disk space on the mounted file system that is currently free, expressed as an absolute value in kilobytes
KBytes Used	Long	The amount of disk space on the mounted file system that is currently in use, expressed as an absolute value in kilobytes
KBytes Total	Long	The amount of disk space on the mounted file system, expressed in kilobytes
% Minimum	Integer	The minimum % free (for super user only)

FileSystem:getByFileSystem

Method

Purpose This method (on Mac OS X) provides utilization statistics for the currently mounted filesystems filtered by device name and file system type. The argument is treated as a pattern to match. If no argument is specified, the entire mount table is reported.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
FileSystem Type	String	Name of the file system type

Returns

Name	Type	Description
Partition	String	Device name of a hard disk partition
Mount Point	String	Mount point of device. Table is indexed on Mount Point.
File System Type	String	Name of the file system type
% Free	Integer	Amount of disk space on the mounted file system that is currently free, expressed as percentage of total size
% Used	Integer	Amount of disk space on the mounted file system currently in use, expressed as percentage of total size
KBytes Free	Long	Amount of disk space on the mounted file system that is currently free, expressed as absolute value (kilobytes)
KBytes Used	Long	Amount of disk space on the mounted file system currently in use, expressed as absolute value (kilobytes)
KBytes Total	Long	Total amount of disk space on mounted file system (kilobytes)
% Minimum	Integer	Minimum Percentage free

COM.TIBCO.hawk.hma.Network

Microagent

Purpose The Network microagent (on Mac OS X) furnishes information about network utilization statistics, identification, communications, and errors.

Methods	Method	Description	Page
	Network:getStatistics	Returns network performance information	332
	Network:getConfig	Returns information on network addresses and masks	333

Network:getStatistics

Method

Purpose This method (on Mac OS X) returns network performance information in a manner similar to using the `netstat` command. All statistics are for the lifetime of the system since it was last booted.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Interface Name	String	Name of the network interface

Returns

Name	Type	Description
Interface Name	String	Name of the network interface. Table is indexed on Interface Name.
Input Bytes	Long	The number of octets of data received through the interface
Input Packets	Integer	Packets received on the interface, lifetime total
Input Packet Errors	Integer	Packet errors received on the interface, lifetime total
Output Bytes	Long	The number of octets of data sent through this interface
Output Packets	Integer	Packets sent on the interface, lifetime total
Output Packet Errors	Integer	Packet errors sent on the interface, lifetime total
Network Collisions	Integer	Collisions, lifetime total
Framing Errors	Integer	Framing errors, lifetime total
No Carrier Errors	Integer	No carrier errors, lifetime total
% Network Collisions	Double	Rate (percentage) of collisions against total number of output packets, lifetime total

Network:getConfig

Method

Purpose This method (on Mac OS X) returns network configuration information in a manner similar to using the `ifconfig` command.

Type Synchronous, `IMPACT_INFO`.

Arguments

Name	Type	Description
Interface Name	String	Name of the network interface

Returns

Name	Type	Description
Interface Name	String	Name of the network interface. Table is indexed on Interface Name
Network Address	String	Network Address in dotted format, 32 bit Value
Hardware Address	String	Hardware Address in colon format
Broadcast Address	String	Broadcast address for the interface
Network Mask	String	Network mask for the interface
MTU	String	Maximum transmission unit

Index

A

agent diagnostics, disabling 23
 agent diagnostics, enabling 22
 auto-configuration information 16

C

COM.TIBCO.hawk.commonlogging.event.CLEventPublisher 105
 COM.TIBCO.hawk.Custom 76
 COM.TIBCO.hawk.Eventlog 147
 COM.TIBCO.hawk.Logfile 70
 COM.TIBCO.hawk.microagent.Custom 76
 COM.TIBCO.hawk.microagent.Eventlog 147
 COM.TIBCO.hawk.microagent.FileStat 80
 COM.TIBCO.hawk.microagent.FileSystem 208, 230, 252, 275, 303
 COM.TIBCO.hawk.microagent.Logfile 70
 COM.TIBCO.hawk.microagent.Network 214, 235, 258, 279, 308, 331
 COM.TIBCO.hawk.microagent.Performance 110
 COM.TIBCO.hawk.microagent.Process 184, 204, 226, 248, 271, 295
 COM.TIBCO.hawk.microagent.Registry 161
 COM.TIBCO.hawk.microagent.RuleBaseEngine 40
 COM.TIBCO.hawk.microagent.Service 172
 COM.TIBCO.hawk.microagent.SysInfo 27
 COM.TIBCO.hawk.microagent.System 198, 220, 242, 266, 286, 316
 COM.TIBCO.hawk.microagent.tcpdaemon.TcpClusterStatus 95, 100
 COM.TIBCO.hawk.microagent.TibRendezvous 86
 COM.TIBCO.hawk.Performance 110
 COM.TIBCO.hawk.Process 184
 COM.TIBCO.hawk.Registry 161
 COM.TIBCO.hawk.Repository 58, 64

COM.TIBCO.hawk.RuleBaseEngine 40
 COM.TIBCO.hawk.Self 16
 COM.TIBCO.hawk.Service 172
 COM.TIBCO.hawk.SysInfo 27
 Common methods
 getReleaseVersion 4
 getTraceLevel 5
 getTraceParameters 7
 onUnsolicitedMsg 9
 setTraceLevel 6
 setTraceParameters 8
 configuration requests, responding to 64
 Custom
 execute() 79
 executeForNumber() 78
 executeForString() 77

D

diagnostic tools, starting and stopping 16

E

Eventlog
 getRecentApplicationEvents 148
 getRecentSecurityEvents 152
 getRecentSystemEvents 150
 logEvent() 160
 onApplicationEvent() 154
 onSecurityEvent() 156
 onSystemEvent() 158

F

FileStat

getFileCount() 85

getFileStatus() 81

FileSystem

getByMountPoint 210, 232, 255, 277, 305

getByPartition 209, 231, 253, 276, 304, 328

functions of microagents 2

H

HawkController

getFileStores 105

I

IMPACT_ACTION microagent type 2

IMPACT_ACTION_INFO microagent type 2

IMPACT_INFO microagent type 2

L

Linux Microagents 284

log files content, monitoring 70

Logfile onNewLine() 71

Logfile onXMLElement 72

M

microagent functions 2

microagent information 2

microagent types 2

N

Network

getConfig 216, 238, 259, 282, 310, 333

getStatistics 215, 236, 260, 280, 309, 332

network host, identifying 27

O

operating statistics of files 80

P

Performance

Browser 135

Cache 117

ICMP 142

IP 141

LogicalDisk() 121

Memory 115

NBT Connection() 138

Network Interface() 139

Objects 127

Paging File() 134

PhysicalDisk() 119

Process() 123

ProcessCount() 125

Processor() 114

Redirector 128

Server 130

Server Work Queues() 132

System 112

TCP 144

Telephony 137

Thread() 126

UDP 145

Process

getInstanceCount 185, 206, 228, 250, 273, 301, 325

getInstanceCountByCommand 207, 229, 251, 274,

302

getInstanceCountByCommand() 186
 getProcess 187, 205, 227, 249, 272, 296

R

Registry

CreateKey() 169
 EnumerateKey() 168
 getDWORD() 164, 171
 getMultiString() 166
 getString() 165
 setDWORD() 162
 setExpandString() 167
 setQWORD() 170
 setString() 163

release version information 16

Repository

getName 66
 getRuleBaseName 68
 getScheduleNames 67
 onRepositoryEvent() 69

retrieving information from a script or program 76

rulebase functions 40

RuleBaseEngine

deleteRuleBase 43
 getConfigInfo 50
 getRuleBaseNames 45
 getScheduleNames 46
 loadRuleBase 47
 loadRuleBaseFromFile() 48
 onAlertCountForRulebase 53
 ResumeSuspendedAlerts 56
 sendMail() 51
 unloadRuleBase 49

rulebases information 40

running time record 24

S

security policy information 20

Self

doAMIDiscovery 25
 getAgentTimeZone 19
 getComponentInfo() 21
 getMicroAgentInfo() 18
 getSecurityInfo 20
 getUptime 23, 24
 onMicroAgentEvent 26
 turnDiagnosticsOff 23
 turnDiagnosticsOn 22

Services

continueService 182
 controlService() 183
 getServiceConfiguration() 173
 getServiceStatus() 176
 pauseService() 181
 setServiceStartType() 178
 startService() 179
 stopService() 180

Summary of AIX Microagents 264

Summary of HP-UX Microagents 240

Summary of Microsoft Windows Microagents 106

Summary of Platform-Independent Microagents 12

Summary of Solaris Microagents 196, 218

SysInfo

getArchitecture 29
 getHostName 30
 getNetworkAddress 31
 getOperatingSystem 28

System

getCpuInfo 201, 223, 245, 267, 269, 293, 318
 getSwapInfo 202, 224, 246, 287, 320
 getSystemInfo 199, 221, 243, 270, 288
 getTunableInfo 203, 225, 247, 294, 321

T

TcpClusterStatus

getClusterStatus() 97
 getDaemonStatus 96, 101
 getMemberCount() 98
 onMemberEvent() 99

TIBCO Rendezvous daemon, monitoring 86

TIBCO support

TIBCOcommunity [xvii](#)

TIBCO_HOME [xv](#)

TibRendezvous

onRvDaemonStatus() [87](#)

onRvDataLoss() [91](#)

onRvdDisconnectOrConnect() [92](#)

onRvLicenseExpire() [93](#)

trace level bits, setting [6](#)

trace level settings [5](#)

tracing parameter information [7](#)

tracing parameters, setting [8](#)

U

unsolicited message defined [9](#)

unsolicited notifications [9](#)

V

version information [21](#)

W

writing custom rules [76](#)