



# **Windows Inspector Library**

**A Guide to the BigFix Windows Inspectors**

BigFix, Inc.  
Emeryville, CA

Last Modified: August 10, 2006

Compatible with  
BES 6.0

1998–2006 BigFix, Inc. All rights reserved.

BigFix<sup>®</sup>, Fixlet<sup>®</sup> and "Fix it before it fails"<sup>®</sup> are registered trademarks of BigFix, Inc. i-prevention, Powered by BigFix, Relevance Engine, and related BigFix logos are trademarks of BigFix, Inc. All other product names, trade names, trademarks, and logos used in this documentation are the property of their respective owners. BigFix's use of any other company's trademarks, trade names, product names and logos or images of the same does not necessarily constitute: (1) an endorsement by such company of BigFix and its products, and (2) an endorsement of the company or its products by BigFix.

No part of this documentation may be reproduced, transmitted, or otherwise distributed in any form or by any means (electronic or otherwise) without the prior written consent of BigFix, Inc. You may not use this documentation for any purpose except in connection with your use or evaluation of BigFix software and any other use, including for reverse engineering such software or creating compatible software, is prohibited. If the license to the software which this documentation accompanies is terminated, you must immediately return this documentation to BigFix, Inc. and destroy all copies you may have.

All inquiries regarding the foregoing should be addressed to:

BigFix, Inc.  
6121 Hollis Street  
Emeryville, CA 94608-2021

# Contents

---

## **PREFACE 1**

AUDIENCE .....	1
ORGANIZATION OF THIS MANUAL .....	1
CONVENTIONS USED IN THIS MANUAL .....	3
EXAMPLES .....	3
VERSIONS .....	4

## **INTRODUCTION 5**

## **PRIMITIVE OBJECTS 6**

BOOLEAN .....	6
INTEGER .....	10
INTEGER RANGE .....	16
INTEGER WITH MULTIPLICITY .....	17
FLOATING POINT .....	18
STRING .....	24
STRING POSITION .....	33
SUBSTRING .....	34
STRING WITH MULTIPLICITY .....	37
ROPE .....	38
BIT SET .....	39
REGULAR EXPRESSION .....	41
REGULAR EXPRESSION MATCH .....	43
UNDEFINED .....	44
HERTZ .....	44
TIME .....	46
TIME OF DAY .....	51
TIME ZONE .....	53
TIME OF DAY WITH TIME ZONE .....	54
TIME RANGE .....	56
TIME INTERVAL .....	59
DATE .....	62
DAY OF WEEK .....	66
DAY OF MONTH .....	68
DAY OF YEAR .....	70
MONTH .....	73
MONTH AND YEAR .....	77
NUMBER OF MONTHS .....	81
YEAR .....	82

## **WORLD OBJECTS 85**

WORLD .....	85
-------------	----

---

## **REGISTRY OBJECTS 114**

---

REGISTRY .....	114
REGISTRY KEY .....	116
REGISTRY KEY VALUE .....	121
REGISTRY KEY VALUE TYPE .....	124

---

## **FILESYSTEM OBJECTS 126**

---

FILESYSTEM OBJECT .....	126
FILE .....	128
APPLICATION .....	135
FOLDER .....	139
DRIVE .....	142
FILE SHORTCUT .....	146
FILE SECTION .....	147
FILE CONTENT .....	148
VERSION .....	149
FILE VERSION BLOCK .....	152
FILE LINE .....	157
XML DOM DOCUMENT .....	159
XML DOM NODE .....	160
APPLICATION USAGE SUMMARY .....	165

---

## **SYSTEM OBJECTS 167**

---

BIOS .....	167
OPERATING SYSTEM .....	168
PROCESSOR .....	172
RAM .....	174
ACTIVE DEVICE .....	175
LICENSE .....	177
LOCAL MSSQL DATABASE .....	178
SERVICE .....	179
LANGUAGE .....	182
PRIMARY LANGUAGE .....	183

---

## **FIREWALL OBJECTS 184**

---

FIREWALL .....	184
FIREWALL AUTHORIZED APPLICATION .....	185
FIREWALL PROFILE .....	186
FIREWALL PROFILE TYPE .....	188
FIREWALL POLICY .....	190
FIREWALL SCOPE .....	191
FIREWALL OPEN PORT .....	192
FIREWALL SERVICE .....	193
FIREWALL SERVICE TYPE .....	195
FIREWALL ICMP SETTINGS .....	196
FIREWALL REMOTE ADMIN SETTINGS .....	198
INTERNET PROTOCOL .....	199
IP VERSION .....	200

## **DMI OBJECTS 202**

DMI B32_BIT_MEMORY_ERROR_INFORMATION .....	202
DMI B64_BIT_MEMORY_ERROR_INFORMATION .....	203
DMI BASE_BOARD_INFORMATION .....	204
DMI BIOS_INFORMATION .....	204
DMI BIOS_LANGUAGE_INFORMATION .....	205
DMI BUILT_IN_POINTING_DEVICE .....	205
DMI CACHE_INFORMATION .....	206
DMI COOLING_DEVICE .....	207
DMI ELECTRICAL_CURRENT_PROBE .....	207
DMI END_OF_TABLE .....	208
DMI GROUP_ASSOCIATIONS .....	208
DMI HARDWARE_SECURITY .....	209
DMI INACTIVE .....	209
DMI MANAGEMENT_DEVICE .....	209
DMI MANAGEMENT_DEVICE_COMPONENT .....	210
DMI MANAGEMENT_DEVICE_THRESHOLD_DATA .....	211
DMI MEMORY_ARRAY_MAPPED_ADDRESS .....	211
DMI MEMORY_CONTROLLER_INFORMATION .....	212
DMI MEMORY_DEVICE .....	213
DMI MEMORY_DEVICE_MAPPED_ADDRESS .....	214
DMI MEMORY_MODULE_INFORMATION .....	215
DMI ON_BOARD_DEVICES_INFORMATION .....	216
DMI OUT_OF_BAND_REMOTE_ACCESS .....	216
DMI PHYSICAL_MEMORY_ARRAY .....	217
DMI PORT_CONNECTOR_INFORMATION .....	218
DMI PORTABLE_BATTERY .....	218
DMI PROCESSOR_INFORMATION .....	219
DMI SYSTEM_BOOT_INFORMATION .....	220
DMI SYSTEM_ENCLOSURE_OR_CHASSIS .....	221
DMI SYSTEM_INFORMATION .....	222
DMI SYSTEM_POWER_CONTROLS .....	222
DMI SYSTEM_RESET .....	223
DMI SYSTEM_SLOTS .....	223
DMI TEMPERATURE_PROBE .....	224
DMI VOLTAGE_PROBE .....	225

## **WMI OBJECTS 226**

WMI .....	226
WMI SELECT .....	227
WMI OBJECT .....	230

## **SITE OBJECTS 231**

SITE .....	231
SITE GROUP .....	234
FIXLET .....	234

---

## **CLIENT OBJECTS 236**

---

CLIENT .....	236
SETTING .....	237
SELECTED SERVER .....	238
OPERATING SYSTEM PRODUCT TYPE .....	240
OPERATING SYSTEM SUITE MASK .....	241
LOCAL GROUP .....	241
LOCAL GROUP MEMBER .....	242
EVENT LOG .....	243
EVENT LOG RECORD .....	245
EVENT LOG EVENT TYPE .....	248

---

## **ENVIRONMENT OBJECTS 250**

---

ENVIRONMENT .....	250
ENVIRONMENT VARIABLE .....	251

---

## **AUTHORIZATION OBJECTS 252**

---

ACCESS CONTROL ENTRY .....	252
ACCESS CONTROL LIST .....	256
SECURITY DESCRIPTOR .....	261
SECURITY IDENTIFIER .....	263

---

## **USER OBJECTS 264**

---

LOCAL USER .....	264
CURRENT USER .....	269

---

## **ACTION OBJECTS 270**

---

ACTION .....	270
--------------	-----

---

## **NETWORKING OBJECTS 273**

---

NETWORK .....	273
NETWORK INTERFACE .....	274
NETWORK IP INTERFACE .....	275
IPv4 ADDRESS .....	276
NETWORK ADAPTER .....	279
NETWORK ADDRESS LIST .....	281
INTERNET CONNECTION FIREWALL .....	282
PORT MAPPING .....	283
NETWORK SHARE .....	284
CONNECTION .....	286
CONNECTION STATUS .....	287
MEDIA TYPE .....	289
ACTIVE DIRECTORY LOCAL COMPUTER .....	291
ACTIVE DIRECTORY SERVER .....	292

---

<b>MICROSOFT IIS METABASE OBJECTS</b>	<b>293</b>
---------------------------------------	------------

METABASE .....	293
METABASE IDENTIFIER.....	293
METABASE KEY .....	294
METABASE TYPE .....	295
METABASE USER TYPE .....	296
METABASE VALUE.....	297

---

<b>INTROSPECTORS</b>	<b>299</b>
----------------------	------------

TYPE .....	299
PROPERTY .....	301
BINARY OPERATOR .....	303
UNARY OPERATOR.....	304
CAST .....	305

---

<b>KEY PHRASES (INSPECTORS)</b>	<b>307</b>
---------------------------------	------------

KEY PHRASES.....	307
CASTING OPERATORS .....	333

---

<b>APPENDIX</b>	<b>338</b>
-----------------	------------

FOLDERS ON WINDOWS MACHINES.....	338
PROCESSORS.....	339

---

<b>INDEX</b>	<b>341</b>
--------------	------------

---

## Preface

The *Windows Inspector Library* is a guide to the ordinary phrases (known as Inspectors) of the **Relevance Language™**. Using this guide, you can write your own Relevance Expressions and use them to target actions to exactly those computers that need them. Both the **BES Console** and the **BigFix Development Environment** allow you to write **Fixlet®** messages and post them to **Fixlet Sites**. For more information on how these programs support the Relevance language, see the *BigFix Enterprise Suite (BES) Console Operator's Guide* and the *BigFix Development Environment (BDE) Guide*.

---

### Audience

This guide is for IT managers, product support groups and other people who want to write Fixlet messages.

IT managers will use the BigFix Enterprise Suite (BES) to keep a network of computers up to date and running smoothly without interruption.

QA and other support teams will produce Fixlet messages for the BigFix Consumer Edition (BCE), designed to keep their users updated and their support calls to a minimum. Only those Inspectors marked with BCE are available for the consumer version. To get the most out of this manual, it helps to have some experience with the Windows Registry and the BigFix Relevance Language.

---

### Organization of this manual

For each Inspector in this library, there is a list of corresponding properties. The Inspectors are organized by category as follows:

- **Primitive Objects.** This chapter covers the basic data types supported by the language and describes the operations that can be applied to them.
- **World Objects.** This chapter covers the keywords used to create all the 'top' level objects of the world. The properties of these objects provide access to all levels of the machine state that can be inspected.
- **Registry Objects.** This chapter covers the keywords for dealing with the Windows registry. Particular attention is paid to registered applications and their associated file extensions.
- **File System Objects.** This chapter covers the keywords for extracting information from the file system, like applications, drives, pathnames, folders, versions, etc. It includes the keywords dealing with applications that have registered themselves in the Windows registry. It also includes the keywords needed to identify and compare version information of files and applications.

- **System Objects.** This chapter covers the keywords available for querying the name and version of the operating system. It also includes the version information of the system Bios. This chapter also covers the keywords used to describe the vendors and types of the various processors that coexist in a typical computer system.
- **Firewall Objects.** This chapter details the firewall Inspectors that examine the authorized applications, policies, services, settings and more.
- **DMI Objects.** This chapter covers the keywords that query the dmi data of the bios. This data, when present, provides detailed information about the properties and manufacture of the system.
- **WMI Objects.** This chapter covers WMI objects that provide access to the WMI (Windows Management Interface) query facility.
- **Site Objects.** This chapter covers the keywords that query the properties of Fixlet sites to which the client is subscribed.
- **Client Objects.** This chapter covers the client inspectors, which allow access to properties of the client application hosting the relevance evaluation.
- **Environment Objects.** An environment object is provided to access environment variables. These are the same variables you are used to seeing in a DOS shell when you type the 'set' command. Note that you are inspecting the environment of the application executing the relevance clause, which may or may not match the environment of other applications on the computer.
- **Authorization Objects.** This section covers inspectors that retrieve security and access settings.
- **User Objects.** This chapter covers the local and current user keywords. A Local User object is provided to access the user data of the local machine. Note that domain users are not available through this inspector.
- **Action Objects.** These are the keywords associated with properties available for inspection during the execution of BigFix Actions.
- **Network Objects.** This chapter covers the keywords used to query the local network configuration.
- **Microsoft IIS Metabase Objects.** This section lists the inspectors for the Microsoft IIS Metabase, which is a repository for most IIS configuration values.
- **Introspectors.** This chapter is concerned with Inspectors that query the Inspectors themselves, looking at types, properties, operators and casts.
- **Keywords (Inspector List).** This chapter provides an alphabetical list of all the Inspector keywords along with the form, context object type, and resulting object type.

---

## Conventions Used in this manual

This document makes use of the following conventions and nomenclature:

Convention	Use
<b>Bold Sans</b>	A bold sans-serif font is used for Inspector headers.
Mono-space	A mono-spaced font is used to indicate expressions in the Relevance Language.
{curly braces}	Braces are used to indicate the comparison {=, !=} or arithmetic operators {+, -} that are available for a binary operation.
<angle bracket>	Angle brackets are used to indicate an object type. For instance to indicate the creation and usage of a particular object, you might see “absolute value of <integer>” which indicates that an integer is to follow the “absolute value of” keyphrase.
<i>Italics</i>	An inspector form. Some inspectors are simple keywords. Others are a keyword in combination with another inspector. Still other forms allow iteration through object lists. Each form is defined below
Small print	The small print beneath the description of each Inspector lists the first implementation for every relevant operating system.

---

## Examples

Square bullets and a mono-spaced font denote examples of Inspectors as used in a Relevance Expression. If you have a color version of this file, these square bullets are also red:

- concatenation of "light" & "year"
- Returns "lightyear"

---

## Versions

Most Inspectors have equivalent implementations on other operating systems, allowing you to write cross-platform relevance expressions. There are exceptions, of course. To keep track of them for each Inspector and operating system, the debut BigFix version is listed at the end of the description, e.g.:

Win:1.2, RH:3.1, Sol:3.1, HPUX:4.0, AIX:4.1

These are the abbreviations for some of the current operating systems:

**Win:** the Windows version of the BigFix Enterprise Suite (BES).

**RH:** the Red Hat & Suse Linux version of BES.

**Sol:** the SUN Solaris operating system version of BES.

**HPUX:** the Hewlett-Packard Unix version of BES.

**AIX:** the AIX version of BES.

**Mac:** the Macintosh version of BES.

## Introduction

This manual details the properties and operators of the BigFix Inspector keywords. Inspectors are the basis of the Relevance Language. They can be thought of as object-oriented representations of the underlying computer system. With Inspectors, you can write Relevance expressions that query all aspects of the computer. Inspectors are also used to produce substituted variables in action buttons. In addition, they can be used to create human-readable descriptions of any given computer system.

You will notice that many of the keywords of the language are not unique; they get their meaning from their context. Accordingly, their definitions often include a phrase to define the context of each Inspector.

This document describes inspectors for Windows 95/98/ME as well as Windows NT, 2000, and XP. Only those Inspectors marked with BCE are available for the Bigfix Consumer Edition. Contact your BigFix sales representative for information about Inspector Guides for other operating systems, including Solaris, Mac, HP/UX, AIX, Red Hat and Suse Linux.

In the following pages, you will find tables defining the inspectors of the relevance language. The inspectors come in seven **forms** depending upon their context:

Form	Syntax required
<i>Cast</i>	<object> as keyword
<i>Global</i>	keyword
<i>Named</i>	keyword " <i>name</i> " of <object>
<i>NamedGlobal</i>	keyword " <i>name</i> "
<i>Numbered</i>	keyword <i>number</i> of <object>
<i>NumberedGlobal</i>	keyword <i>number</i>
<i>Plain</i>	keyword of <object>

These differ from one another in format and in the syntax they require. Except for *Cast*, these forms can be used to access both single objects and *lists* of objects by using the plural form of the keyword. The plurals are listed in the Keyword section later in this document.

**Creation Methods** are used to create objects of the specified type, and various **Properties** are available for each object.

**Operators** list the binary and unary operations that can be performed with the given object type. Binary operators take two inputs and generate one output. The integer '+' (addition) operator is an example of a binary operation. Unary operators take a single input and generate a single output. The boolean 'Not' operation is an example of a unary operation.

# Primitive Objects

The relevance language is based upon a comprehensive set of primitive objects. These primitives are the basic building blocks of the more complex objects to follow.

## Boolean

### Creation Methods

These boolean creation methods are in addition to the other properties that return the boolean type.

Key Phrase	Form	Description
<string> as boolean	<i>Cast</i>	Returns a boolean TRUE or FALSE from a string. The string must contain values of "TRUE" or "FALSE". Case is ignored. For example, "FalSe" as boolean = FALSE.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
bit <integer> of <integer>	<i>Numbered</i>	Return TRUE if the bit referenced by the integer is on. Bits are numbered starting with zero being the least significant. For example, bit 0 of 5 and bit 2 of 5 and not bit 1 of 5 = TRUE.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
boolean <string>	<i>NamedGlobal</i>	Creates the boolean value of the <string>, e.g., • boolean "False" = FALSE.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
conjunction of <boolean>	<i>Plain</i>	This inspector performs a serial AND on all its boolean arguments: • conjunction of (true; true; true) -> TRUE • conjunction of (true; true; false) -> FALSE.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
disjunction of <boolean>	<i>Plain</i>	This inspector performs a serial OR on all its boolean arguments: • disjunction of (false; false; false) -> FALSE • disjunction of (false; false; true) -> TRUE.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
false	<i>PlainGlobal</i>	Creates a boolean with value FALSE.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

Key Phrase	Form	Description
inexact of <floating point>	<i>Plain</i>	Returns TRUE if the calculation raised the inexact exception; that is, if some intermediate result could not be represented exactly.  Win:4.1, Mac:4.1
infinite of <floating point>	<i>Plain</i>	Returns TRUE if the floating point number is infinite.  Win:4.1, Mac:4.1
invalid of <floating point>	<i>Plain</i>	Returns TRUE if the calculation raised the invalid exception; that is, if some part of the calculation a function was applied to a value outside its domain.  Win:4.1, Mac:4.1
nan of <floating point>	<i>Plain</i>	Returns TRUE if the value is not a number.  Win:4.1, Mac:4.1
normal of <floating point>	<i>Plain</i>	Returns TRUE if the value is a valid floating point number.  Win:4.1, Mac:4.1
overflow of <floating point>	<i>Plain</i>	Returns TRUE if the calculation raised the overflow exception; that is, if some intermediate result was too large to be represented, but not an exact infinity.  Win:4.1, Mac:4.1
true	<i>PlainGlobal</i>	Creates a boolean with value TRUE.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
underflow of <floating point>	<i>Plain</i>	Returns TRUE if the calculation raised the underflow exception; that is, if some intermediate result was a nonzero value too small to be represented.  Win:4.1, Mac:4.1

## Properties

Key Phrase	Form	Return Type	Description
<boolean> as string	<i>Cast</i>	<string>	Converts the boolean value to a string. The possible values returned are "True" and "False" with this exact case, e.g., • TRUE as string = "True".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

Key Phrase	Form	Return Type	Description
conjunction of <boolean>	<i>Plain</i>	<boolean>	<p>This inspector performs a serial AND on all its boolean arguments:</p> <ul style="list-style-type: none"> <li>conjunction of (true; true; true) -&gt; TRUE</li> <li>conjunction of (true; true; false) -&gt; FALSE.</li> </ul> <p>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</p>
disjunction of <boolean>	<i>Plain</i>	<boolean>	<p>This inspector performs a serial OR on all its boolean arguments:</p> <ul style="list-style-type: none"> <li>disjunction of (false; false; false) -&gt; FALSE</li> <li>disjunction of (false; false; true) -&gt; TRUE.</li> </ul> <p>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</p>

## Operators

Key phrase	Return Type	Description
<boolean> * <time range>	<i>&lt;timed( time range, boolean )&gt;</i>	<p>Returns a time interval labeled with a boolean TRUE or FALSE.</p> <p>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</p>
<boolean> {cmp} <boolean>	<boolean>	<p>Compare two boolean expressions. Returns another boolean, depending on the evaluation of the comparison:</p> <ul style="list-style-type: none"> <li>{cmp} is one of: =, != .</li> </ul> <p>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</p>
<boolean> {op} <boolean>	<boolean>	<p>Operates on two boolean expressions. Returns another boolean, depending on the evaluation of the operation, e.g., (True And True) = True.</p> <ul style="list-style-type: none"> <li>{op} is one of: And, Or .</li> </ul> <p>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</p>
<floating point> {cmp} <floating point>	<boolean>	<p>Compares two floating point numbers, where:</p> <ul style="list-style-type: none"> <li>{cmp} is one of: =, &lt;, &lt;=.</li> </ul> <p>Win:4.1, Mac:4.1</p>
<floating point> {cmp} <integer>	<boolean>	<p>Compares a floating point number and an integer, where:</p> <ul style="list-style-type: none"> <li>{cmp} is one of: =, &lt;=, &lt;.</li> </ul> <p>Win:4.1, Mac:4.1</p>

Key phrase	Return Type	Description
<integer> {cmp} <floating point>	<boolean>	Compares an integer to a floating point number, where: • {cmp} is one of: =, <=, <.  Win:4.1, Mac:4.1
<time interval> {cmp} <time interval>	<boolean>	Compare two time intervals, where: • {cmp} is one of: =, !=, <, <=, >, >= .  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
<time range> * <boolean>	<timed( time range, boolean )>	Returns a time interval labeled with the specified boolean, in the form of: • (<date> to <date>), <boolean>.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Examples

- bit 0 of (least integer + 1)
- ▶ Returns the least significant bit of the smallest possible integer, plus one.
  
- conjunction of (current month = April; leap of year of current date)
- ▶ Returns TRUE during April of a leap year.
  
- disjunction of (current day\_of\_week = Monday ;current day\_of\_week = Wednesday; current day\_of\_week = Friday)
- ▶ Returns TRUE on either Monday, Wednesday or Friday.
  
- infinite of (floating point "1"/ 0)
- ▶ Returns TRUE.
  
- nan of (floating point "1.e-99999" \* floating point "1.e999999")
- ▶ Returns TRUE.
  
- overflow of (floating point "1.0e50000")
- ▶ Returns TRUE, since the number is too big to represent in floating point.

## Integer

Integers are represented internally as 64-bit signed values.

### Creation Methods

These integer creation methods are in addition to the other properties that return the integer type.

Key Phrase	Form	Description
<floating point> as integer	<i>Cast</i>	Rounds off and casts a floating point number as an integer.  Win:6.0
<integer> as integer	<i>Cast</i>	Integer casting for completeness.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
<string> as integer	<i>Cast</i>	Converts from a string to an integer.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
absolute value of <integer>	<i>Plain</i>	Creates the positive value of the <integer> object.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
greatest integer	<i>PlainGlobal</i>	Creates the value 9,223,372,036,854,775,807.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
hexadecimal integer <string>	<i>NamedGlobal</i>	Creates an integer from the provided hexadecimal value.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
integer <integer>	<i>NumberedGlobal</i>	Creates a global object with the given integer value, e.g., Integer 123.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
integer <string>	<i>NamedGlobal</i>	Creates a global object with the integer value given by a string, e.g., Integer "123" creates the value 123.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
integer ceiling of <floating point>	<i>Plain</i>	Returns the smallest integer not less than the floating point number. For example, ceiling of 2.1 = 3, ceiling of 2 = 2 and ceiling of -2.3 = -2.  Win:6.0

Key Phrase	Form	Description
integer floor of <floating point>	<i>Plain</i>	Returns the largest integer less than or equal to the floating point number. For example, floor of 2.8 = 2, floor of -2 = -2 and floor of -2.1 = -3. For nonnegative x, this is the same as the integer part of x.  Win:6.0
least integer	<i>PlainGlobal</i>	Creates the value -9,223,372,036,854,775,808.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
length of <rope>	<i>Plain</i>	Creates an integer object corresponding to the number of bytes in the rope.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
length of <string>	<i>Plain</i>	Creates an integer object corresponding to the number of bytes in the string.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
lower bound of <integer range>	<i>Plain</i>	The low end of the integer range.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
maximum of <integer>	<i>Plain</i>	Returns the maximum of a list of integers.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
minimum of <integer>	<i>Plain</i>	Returns the minimum of a list of integers.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
numeric value of <string>	<i>Plain</i>	Creates an integer object containing the value of the first number contained in a string.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1
product of <integer>	<i>Plain</i>	Multiplies a list of integers, returning the product.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
significant digits <integer> of <integer>	<i>Numbered</i>	Creates a number with <integer> significant digits (e.g.. significant digits 3 of 1235569 = 1240000).  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
sum of <integer>	<i>Plain</i>	Returns the sum of a list of integers.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
upper bound of <integer range>	<i>Plain</i>	The high end of the integer range.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1

## Properties

Integers are represented internally as 64-bit signed values.

Key Phrase	Form	Return Type	Description
<integer> as bit set	<i>Cast</i>	<bit set>	Returns the bits of the binary representation of the integer; bit zero is the least-significant bit. <small>Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1</small>
<integer> as bits	<i>Cast</i>	<bit set>	Returns the bits of the binary representation of the integer; bit zero is the least-significant bit. <small>Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1</small>
<integer> as day_of_month	<i>Cast</i>	<day of month>	Cast an integer as a day of the month type. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<integer> as floating point	<i>Cast</i>	<floating point>	Converts an integer into a floating point number. <small>Win:4.1, Mac:4.1</small>
<integer> as hexadecimal	<i>Cast</i>	<string>	Converts an integer into a hexadecimal string. <small>Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1</small>
<integer> as integer	<i>Cast</i>	<integer>	Reflective cast for completeness. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<integer> as month	<i>Cast</i>	<month>	Returns the name of the nth month of the year. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<integer> as string	<i>Cast</i>	<string>	Converts an integer to a string. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<integer> as year	<i>Cast</i>	<year>	Casts an integer as a year type. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
absolute value of <integer>	<i>Plain</i>	<integer>	Returns the positive value of the integer. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
bit <integer> of <integer>	<i>Numbered</i>	<boolean>	Returns TRUE if the numbered bit is on. Bits are numbered starting at zero. Bit 0 is the least significant bit. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
maximum of <integer>	<i>Plain</i>	<integer>	Returns the maximum of a list of integers. <small>Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>

Key Phrase	Form	Return Type	Description
mean of <integer>	<i>Plain</i>	<floating point>	The mean of the integer(s). <small>Win:5.1, Mac:4.1</small>
minimum of <integer>	<i>Plain</i>	<integer>	Returns the minimum of a list of integers. <small>Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
product of <integer>	<i>Plain</i>	<integer>	Multiplies a list of integers, returning the product. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
significant digits <integer> of <integer>	<i>Numbered</i>	<integer>	Returns a number with <integer> significant digits (e.g., significant digits 3 of 1235569 = 1240000). <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
standard deviation of <integer>	<i>Plain</i>	<floating point>	The standard deviation of the integer(s). <small>Win:5.1, Mac:4.1</small>
sum of <integer>	<i>Plain</i>	<integer>	Returns the sum of a list of integers. <small>Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
unique value of <integer>	<i>Plain</i>	<integer>	Returns the same integers, but with duplicates removed. • Note: As of version 6.0 of BES, this Inspector returns an integer with multiplicity. <small>Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
unique value of <integer>	<i>Plain</i>	<integer with multiplicity>	Given a set of integers, returns the number of instances of each integer. Given (1,2,2,2,3), returns (1,3,1). Earlier versions of this Inspector returned the unique set of integers. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Operators

Key phrase	Return Type	Description
<floating point> {cmp} <integer>	<boolean>	Compares a floating point number and an integer, where: <ul style="list-style-type: none"> <li>{cmp} is one of: =, &lt;=, &lt;.</li> </ul> Win:4.1, Mac:4.1
<floating point> {op} <integer>	<floating point>	Operates on a floating point number and an integer, returning a floating point number, where: <ul style="list-style-type: none"> <li>{op} is one of: +, -, *, /, And .</li> </ul> Win:4.1, Mac:4.1
<hertz> {op} <integer>	<hertz>	Returns a hertz object operated on by the given integer, where: <ul style="list-style-type: none"> <li>{op} is one of: *, / .</li> </ul> Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
<integer> * <number of months>	<number of months>	Multiply a number of months by an integer, producing a new number of months. This is a typical technique to create a value of this type. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<integer> * <time range>	<timed( time range, integer )>	Multiply a time range by an integer, producing a new time range. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<integer> {cmp} <floating point>	<boolean>	Compares an integer to a floating point number, where: <ul style="list-style-type: none"> <li>{cmp} is one of: =, &lt;=, &lt;.</li> </ul> Win:4.1, Mac:4.1
<integer> {cmp} <integer>	<boolean>	Returns boolean TRUE or FALSE, depending on the comparison operator, where: <ul style="list-style-type: none"> <li>{cmp} is one of: =, !=, &lt;, &lt;=, &gt;, &gt;= .</li> </ul> Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
<integer> {cmp} <registry key value type>	<boolean>	Returns boolean TRUE or FALSE, depending on the comparison operator, where: <ul style="list-style-type: none"> <li>{cmp} is one of: =, !=, &lt;, &lt;=, &gt;, &gt;= .</li> </ul> Win:1.2
<integer> {cmp} <registry key value>	<boolean>	Returns boolean TRUE or FALSE, depending on the comparison operator, where: <ul style="list-style-type: none"> <li>{cmp} is one of: =, !=, &lt;, &lt;=, &gt;, &gt;= .</li> </ul> Win:1.2

Key phrase	Return Type	Description
<code>&lt;integer&gt; {op} &lt;floating point&gt;</code>	<code>&lt;floating point&gt;</code>	Operates on an integer and a floating point number, returning a floating point number, where: • {op} is one of: -, +, *, /.  Win:4.1, Mac:4.1
<code>&lt;integer&gt; {op} &lt;integer&gt;</code>	<code>&lt;integer&gt;</code>	Returns the integer solution to the equation, depending on the operator, where: • {op} is one of: +, -, *, /, mod .  Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
<code>&lt;number of months&gt; {op} &lt;integer&gt;</code>	<code>&lt;number of months&gt;</code>	Where {op} is one of: *, /.  Win:6.0
<code>&lt;time range&gt; * &lt;integer&gt;</code>	<code>&lt;timed( time range, integer )&gt;</code>	Returns a time interval labeled with the specified integer, in the form of: • (<date> to <date>), <integer>.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Examples

- hexadecimal integer "A0"  
▶ Returns 160.
- integer ceiling of (15/8 as floating point)  
▶ Returns 2.
- integer floor of ("1.9" as floating point)  
▶ Returns 1.
- maximum of (sizes of files of folder "c:\")  
▶ Returns the size of the largest file in the indicated folder.
- minimum of (sizes of files of folder "c:\")  
▶ Returns the size of the smallest file in the indicated folder.
- numeric value of "string 123 xyz 45" = 123  
▶ Returns TRUE.
- product of (1;2;3)  
▶ Returns 6.
- sum of (sizes of files of folder "c:\")  
▶ Returns the sum of the sizes of all files in the specified folder.

- 255 as hexadecimal
- ▶ Returns the string "ff".
  
- maximum of (7;2;4;5)
- ▶ Returns 7.
  
- minimum of (sizes of files of folder "c:\")
- ▶ Returns the size of the smallest file in the indicated folder.
  
- significant digits 3 of 1235569
- ▶ Returns 1240000.
  
- sum of (sizes of files of folder "c:\")
- ▶ Returns the sum of the sizes of all files in the specified folder.
  
- (July-current month) < 2\*month
- ▶ Returns TRUE when the current date is between June and July.
  
- 21 mod 5
- ▶ Returns 1.

---

## Integer Range

Specifies a range between two 64-bit signed integers.

### Creation Methods

Key Phrase	Form	Description
distance of <selected server>	<i>Plain</i>	The distance, in IP gateway hops, to the server. Among servers with the same priority, closer servers are preferred. Returns an integer range, since the exact distance may not be known.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1

### Properties

Key Phrase	Form	Return Type	Description
lower bound of <integer range>	<i>Plain</i>	<integer>	The low end of the integer range.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
upper bound of <integer range>	<i>Plain</i>	<integer>	The high end of the integer range.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1

## Integer With Multiplicity

These Inspectors deal with arrays of integers, allowing you to pluck out unique numbers and count them. These objects are derived from integer types.

### Creation Methods

Key Phrase	Form	Description
unique value of <integer>	<i>Plain</i>	Given a set of integers, returns the number of instances of each integer. Given (1,2,2,2,3), returns (1,3,1). Earlier versions of this Inspector returned the unique set of integers.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

### Properties

Key Phrase	Form	Return Type	Description
multiplicity of <integer with multiplicity>	<i>Plain</i>	<integer>	Returns the multiplicity (quantity) of each element in a multiple integer list.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

### Examples

- `unique values of (1;2;3;3)`
  - Returns a list of the count of each integer, namely 1,1,2.
- `multiplicities of unique values of (1;2;3;3)`
  - Returns the multiplicity of (the number of times) each number in the list is used, namely, 1,1,2.

## Floating Point

The point type holds a floating-point number, with precision dependent on the computer. It also keeps track of the IEEE floating-point exceptions raised in a calculation and an estimate of the significance with which the number should be expressed when it is converted to a string. All arithmetic operations are carried out to the full precision of the computer; only conversions to string are affected by the estimated significance.

### Creation Methods

Key Phrase	Form	Description
<integer> as floating point	<i>Cast</i>	Converts an integer into a floating point number. <small>Win:4.1, Mac:4.1</small>
<string> as floating point	<i>Cast</i>	Converts the contents of a string into a floating point number. <small>Win:4.1, Mac:4.1</small>
floating point <string>	<i>NamedGlobal</i>	Creates a floating point number from the provided string. <small>Win:4.1, Mac:4.1</small>
less significance <integer> of <floating point>	<i>Numbered</i>	Removes <integer> number of digits of significance from the floating point value. <small>Win:4.1, Mac:4.1</small>
mean of <floating point>	<i>Plain</i>	The mean of the floating point number(s). <small>Win:5.1, Mac:4.1</small>
mean of <integer>	<i>Plain</i>	The mean of the integer(s). <small>Win:5.1, Mac:4.1</small>
more significance <integer> of <floating point>	<i>Numbered</i>	Adds <integer> number of digits of significance to the floating point value. <small>Win:4.1, Mac:4.1</small>
relative significance place <integer> of <floating point>	<i>Numbered</i>	The same floating point value, to be expressed to the given number of significant digits. <small>Win:4.1, Mac:4.1</small>
relative significance place of <floating point>	<i>Plain</i>	The base 10 logarithm of the quotient of the value and its significance place; approximately the number of significant digits to which the number should be expressed. <small>Win:4.1, Mac:4.1</small>

Key Phrase	Form	Description
significance place <integer> of <floating point>	<i>Numbered</i>	The same floating point value, to be expressed to the given decimal place.  Win:4.1, Mac:4.1
significance place of <floating point>	<i>Plain</i>	The base 10 logarithm of the significance threshold; approximately the number of digits to the left (positive) or right (negative) of the ones place to which the number should be expressed.  Win:4.1, Mac:4.1
significance threshold of <floating point>	<i>Plain</i>	The difference between the given value and the next number expressed to the same significance level. For example, the significance threshold of 3 is 1, the significance threshold of 3.0 is 0.1, and the significance threshold of 3000 is 1000.  Win:4.1, Mac:4.1
standard deviation of <floating point>	<i>Plain</i>	The standard deviation of the floating point number(s).  Win:5.1, Mac:4.1
standard deviation of <integer>	<i>Plain</i>	The standard deviation of the integer(s).  Win:5.1, Mac:4.1

## Properties

Key Phrase	Form	Return Type	Description
<floating point> as integer	<i>Cast</i>	<integer>	Rounds off and casts a floating point number as an integer.  Win:6.0
<floating point> as scientific notation	<i>Cast</i>	<string>	Converts a floating point number into a string with scientific notation.  Win:4.1, Mac:4.1
<floating point> as standard notation	<i>Cast</i>	<string>	Converts a floating point number into a string with standard notation.  Win:4.1, Mac:4.1
<floating point> as string	<i>Cast</i>	<string>	Converts a floating point number into a string with standard notation.  Win:4.1, Mac:4.1

Key Phrase	Form	Return Type	Description
divided by zero of <floating point>	<i>Plain</i>	<boolean>	Returns TRUE if the calculation raised the divide-by-zero exception; that is, if some part of the calculation produced an exact infinity.  Win:4.1, Mac:4.1
finite of <floating point>	<i>Plain</i>	<boolean>	Returns TRUE if the floating point number is finite.  Win:4.1, Mac:4.1
inexact of <floating point>	<i>Plain</i>	<boolean>	Returns TRUE if the calculation raised the inexact exception; that is, if some intermediate result could not be represented exactly.  Win:4.1, Mac:4.1
infinite of <floating point>	<i>Plain</i>	<boolean>	Returns TRUE if the floating point number is infinite.  Win:4.1, Mac:4.1
integer ceiling of <floating point>	<i>Plain</i>	<integer>	Returns the smallest integer not less than the floating point number. For example, ceiling of 2.1 = 3, ceiling of 2 = 2 and ceiling of -2.3 = -2.  Win:6.0
integer floor of <floating point>	<i>Plain</i>	<integer>	Returns the largest integer less than or equal to the floating point number. For example, floor of 2.8 = 2, floor of -2 = -2 and floor of -2.1 = -3. For nonnegative x, this is the same as the integer part of x.  Win:6.0
invalid of <floating point>	<i>Plain</i>	<boolean>	Returns TRUE if the calculation raised the invalid exception; that is, if some part of the calculation a function was applied to a value outside its domain.  Win:4.1, Mac:4.1
less significance <integer> of <floating point>	<i>Numbered</i>	<floating point>	Removes <integer> number of digits of significance from the floating point value.  Win:4.1, Mac:4.1
mean of <floating point>	<i>Plain</i>	<floating point>	The mean of the floating point number(s).  Win:5.1, Mac:4.1

Key Phrase	Form	Return Type	Description
more significance <integer> of <floating point>	<i>Numbered</i>	<floating point>	Adds <integer> number of digits of significance to the floating point value.  Win:4.1, Mac:4.1
nan of <floating point>	<i>Plain</i>	<boolean>	Returns TRUE if the value is not a number.  Win:4.1, Mac:4.1
normal of <floating point>	<i>Plain</i>	<boolean>	Returns TRUE if the value is a valid floating point number.  Win:4.1, Mac:4.1
overflow of <floating point>	<i>Plain</i>	<boolean>	Returns TRUE if the calculation raised the overflow exception; that is, if some intermediate result was too large to be represented, but not an exact infinity.  Win:4.1, Mac:4.1
relative significance place <integer> of <floating point>	<i>Numbered</i>	<floating point>	The same floating point value, to be expressed to the given number of significant digits.  Win:4.1, Mac:4.1
relative significance place of <floating point>	<i>Plain</i>	<floating point>	The base 10 logarithm of the quotient of the value and its significance place; approximately the number of significant digits to which the number should be expressed.  Win:4.1, Mac:4.1
significance place <integer> of <floating point>	<i>Numbered</i>	<floating point>	The same floating point value, to be expressed to the given decimal place.  Win:4.1, Mac:4.1
significance place of <floating point>	<i>Plain</i>	<floating point>	The base 10 logarithm of the significance threshold; approximately the number of digits to the left (positive) or right (negative) of the ones place to which the number should be expressed.  Win:4.1, Mac:4.1
significance threshold of <floating point>	<i>Plain</i>	<floating point>	The difference between the given value and the next number expressed to the same significance level. For example, the significance threshold of 3 is 1, the significance threshold of 3.0 is 0.1, and the significance threshold of 3000 is 1000.  Win:4.1, Mac:4.1

Key Phrase	Form	Return Type	Description
standard deviation of <floating point>	<i>Plain</i>	<floating point>	The standard deviation of the floating point number(s).  Win:5.1, Mac:4.1
underflow of <floating point>	<i>Plain</i>	<boolean>	Returns TRUE if the calculation raised the underflow exception; that is, if some intermediate result was a nonzero value too small to be represented.  Win:4.1, Mac:4.1

## Operators

Key phrase	Return Type	Description
<floating point> {op} <floating point>	<floating point>	Operates on two floating point numbers, returning another floating point number, where: • {op} is one of: +, -, *, /.  Win:4.1, Mac:4.1
<floating point> {op} <integer>	<floating point>	Operates on a floating point number and an integer, returning a floating point number, where: • {op} is one of: +, -, *, /.  Win:4.1, Mac:4.1
<integer> {cmp} <floating point>	<boolean>	Compares an integer to a floating point number, where: • {cmp} is one of: =, <=, <, >, >=, !=.  Win:4.1, Mac:4.1
<integer> {op} <floating point>	<floating point>	Operates on an integer and a floating point number, returning a floating point number, where: • {op} is one of: -, +, *, /.  Win:4.1, Mac:4.1

## Examples

- 4.5 as floating point
  - Returns 4.5.
- less significance 2 of floating point "5.115"
  - Returns 5.1.

- `mean of floating points( "1.3";"2.5")`
  - ▶ Returns 1.90.
- `mean of integers(1;2;3;4;5)`
  - ▶ Returns 3.0.
- `more significance 2 of floating point "5.2"`
  - ▶ Returns 5.200.
- `significance place 2 of floating point "9123"`
  - ▶ Returns 9100.
- `significance place of floating point "9000"`
  - ▶ Returns 3.00.
- `standard deviation of integers(1;2;3;4;5)`
  - ▶ Returns 1.4.
- `15/2 as integer`
  - ▶ Returns 7.
- `floating point "600987.9" as scientific notation`
  - ▶ Returns 6.009879e+5.
- `floating point "6.009e8" as standard notation`
  - ▶ Returns 600900000.
- `finite of (floating point "1"/ 0)`
  - ▶ Returns FALSE.
- `infinite of (floating point "1"/ 0)`
  - ▶ Returns TRUE.
- `integer floor of ("-2.1" as floating point)`
  - ▶ Returns -3.
- `less significance 2 of floating point "5.115"`
  - ▶ Returns 5.1.
- `mean of floating points( "1.3";"2.5")`
  - ▶ Returns 1.90.
- `more significance 2 of floating point "5.2"`
  - ▶ Returns 5.200.

- `nan of (floating point "1.e-99999" * floating point "1.e999999")`  
▶ Returns TRUE.
- `overflow of (floating point "1.0e50000")`  
▶ Returns TRUE, since the number is too big to represent in floating point.
- `significance place 2 of floating point "9123"`  
▶ Returns 9100.
- `significance place of floating point "9000"`  
▶ Returns 3.00.

---

## String

A string literal is written within double quotes. Special characters must be inserted by using the percent sign followed by 2 hex digits. Special characters include those characters with ASCII codes less than the 'space' character (hex 20) or greater than 'tilde' character (hex 7f) as well as the percent character itself (25 hex). For example, to create a string containing a null character and a percent character use "a null is %00, the percent itself is %25". Conversion to upper and lower case is also provided. String works in combination with the string position and substring data types. A string position is a point within a string. It can be compared to an integer, but it also acts as a pointer within a string so that the preceding and following text can be extracted. A substring is a part of a larger string. All operations allowed on a string can be performed on a substring. There are two substrings "be" in the string "To be or not to be". The substrings only differ in their positions within the string.

### Creation Methods

These string creation methods are in addition to the other properties that return the string type.

Key Phrase	Form	Description
<boolean> as string	<i>Cast</i>	Operates on a boolean to return a string. Possible values are "True" and "False". <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<date> as string	<i>Cast</i>	Cast a date type as a string. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<floating point> as scientific notation	<i>Cast</i>	Converts a floating point number into a string with scientific notation. <small>Win:4.1, Mac:4.1</small>
<floating point> as standard notation	<i>Cast</i>	Converts a floating point number into a string with standard notation. <small>Win:4.1, Mac:4.1</small>

Key Phrase	Form	Description
<floating point> as string	<i>Cast</i>	Converts a floating point number into a string with standard notation.  Win:4.1, Mac:4.1
<hertz> as string	<i>Cast</i>	Creates a string containing the number of hertz and the word hertz, e.g., (3 * hz) as string = "3 hertz".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
<integer> as hexadecimal	<i>Cast</i>	Converts an integer into a hexadecimal string.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
<integer> as string	<i>Cast</i>	Creates a string formatted with the integer provided. (-22) as string = "-22".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
<string> as hexadecimal	<i>Cast</i>	Converts a string to a hexadecimal number.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
<string> as left trimmed string	<i>Cast</i>	Trims the leading spaces from a string.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<string> as lowercase	<i>Cast</i>	Creates a lowercase version of the string provided.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
<string> as right trimmed string	<i>Cast</i>	Trims the trailing spaces from a string.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<string> as string	<i>Cast</i>	Reflexive cast of string to string.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
<string> as trimmed string	<i>Cast</i>	Trims the leading and trailing spaces off of the specified string.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<string> as uppercase	<i>Cast</i>	Creates an uppercase version of the string provided.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
<time interval> as string	<i>Cast</i>	Returns a string formatted as <ul style="list-style-type: none"> <li>• ddd days, HH:MM:SS.mmmmmm</li> <li>• For example, millisecond as string = "00:00:00.001".</li> </ul> Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
<time zone> as string	<i>Cast</i>	Creates a string containing a time zone. See <time zone>.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

Key Phrase	Form	Description
<time> as local string	<i>Cast</i>	Creates a string containing a time. See <time>. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
<time> as string	<i>Cast</i>	Creates a string containing a time. See <time>. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
<time> as universal string	<i>Cast</i>	Creates a string containing a time. See <time>. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
character <integer>	<i>NumberedGlobal</i>	Creates a string containing the single ASCII character for the decimal number provided. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
concatenation <string> of <string>	<i>Named</i>	Combines the second set of strings into a single string, separated by the first string. Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
concatenation of <string>	<i>Plain</i>	Combines the supplied strings into a single string, end-to-end. Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
hexadecimal string <string>	<i>NamedGlobal</i>	Creates a string from the given hexadecimal value. Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
parameter <string>	<i>NamedGlobal</i>	This Inspector is a synonym for the parameter <string> of <action>. It looks up the value of the action parameter specified by <string>. This is used in conjunction with the parameter set command. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
string <string>	<i>NamedGlobal</i>	Creates a string matching the name provided. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

## Properties

Key Phrase	Form	Return Type	Description
<string> as boolean	<i>Cast</i>	<boolean>	Returns a boolean value for the string. All possible capitalization's of "TRUE" and "FALSE" will convert successfully. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
<string> as date	<i>Cast</i>	<date>	Casts a string as a date type. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

Key Phrase	Form	Return Type	Description
<string> as day_of_month	Cast	<day of month>	Casts a string as a day of the month (eg. 28). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<string> as day_of_week	Cast	<day of week>	Casts a string as a day of the week. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<string> as floating point	Cast	<floating point>	Converts the contents of a string into a floating point number. <small>Win:4.1, Mac:4.1</small>
<string> as hexadecimal	Cast	<string>	Converts a string to a hexadecimal number. <small>Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1</small>
<string> as integer	Cast	<integer>	Returns an integer value for the string provided. If the string contains anything but ASCII digits, the conversion will fail. Use numeric value for more liberal parsing rules. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> as left trimmed string	Cast	<string>	Trims the leading spaces from a string. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<string> as local time	Cast	<time>	Returns a local time object from a properly formatted string. See <time>. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> as local zoned time_of_day	Cast	<time of day with time zone>	Converts a string to a time of day with local time zone. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<string> as lowercase	Cast	<string>	Returns a lowercase version of the string provided. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> as month	Cast	<month>	Converts a string into a month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<string> as right trimmed string	Cast	<string>	Trims the trailing spaces from a string. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<string> as string	Cast	<string>	Returns the string provided. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

Key Phrase	Form	Return Type	Description
<string> as time	<i>Cast</i>	<time>	Returns a time object from a properly formatted string. See <time>. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> as time interval	<i>Cast</i>	<time interval>	Returns a time interval object from a properly formatted string. Expects strings formatted as • ddd days, HH:MM:SS.mmmmmm. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> as time zone	<i>Cast</i>	<time zone>	Returns a time zone object from a properly formatted string. See <time zone>. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> as time_of_day	<i>Cast</i>	<time of day>	Converts a string to a time_of_day type. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<string> as trimmed string	<i>Cast</i>	<string>	Trims the leading and trailing spaces off of the specified string. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<string> as universal time	<i>Cast</i>	<time>	Returns a universal time object from a properly formatted string. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> as universal zoned time_of_day	<i>Cast</i>	<time of day with time zone>	Converts a string into a universal zoned time of day. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<string> as uppercase	<i>Cast</i>	<string>	Returns an uppercase version of the string provided. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> as version	<i>Cast</i>	<version>	Returns a version if the string can be parsed as a version. The first numeric set of characters delimited with period, comma or comma-space is returned. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1</small>
<string> as windows display time	<i>Cast</i>	<time>	Returns a Windows display time object from a properly formatted string. See <Time>. <small>Win:1.2</small>
<string> as year	<i>Cast</i>	<year>	Converts a string into a year. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

Key Phrase	Form	Return Type	Description
<string> as zoned time_of_day	<i>Cast</i>	<time of day with time zone>	Converts a string into a zoned time of day. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
character <integer> of <string>	<i>Numbered</i>	<substring>	Returns a string of length 1 made by taking the character identified by <integer> from the string. Numbering begins at zero. Example, Character 1 of "HI" is "I". <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
character of <string>	<i>Plain</i>	<substring>	Returns the characters from the string. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
concatenation <string> of <string>	<i>Named</i>	<string>	Concatenation <string1> of <string2> concatenates a list of strings indicated by string2, placing string1 between each. <small>Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
concatenation of <string>	<i>Plain</i>	<string>	Combines the supplied strings into a single string, end-to-end. <small>Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
escape of <string>	<i>Plain</i>	<string>	Returns a string containing a \\ for every \ character found. Useful for setting registry key values to strings in regset action commands. <small>Win:1.2</small>
expand environment string of <string>	<i>Plain</i>	<string>	Uses the Windows ExpandEnvironmentStrings API to translate a string containing special Windows environment variables. For example, %windir%\my.dll might expand to c:\winnt\my.dll. <small>Win:1.2</small>
first <integer> of <string>	<i>Numbered</i>	<substring>	Returns a substring containing the number of characters specified from the given string. For example, First 5 of "To be or not to be" is "To be". <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
first <string> of <string>	<i>Named</i>	<substring>	Returns a substring containing the first occurrence of the name provided. See substring. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

Key Phrase	Form	Return Type	Description
last <integer> of <string>	<i>Numbered</i>	<substring>	Returns a substring containing the number of characters specified. For example, Last 5 of "To be or not to be" is "to be".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
last <string> of <string>	<i>Named</i>	<substring>	Returns a substring containing the last occurrence of the name provided.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
length of <string>	<i>Plain</i>	<integer>	Returns the number of characters in the string.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
numeric value of <string>	<i>Plain</i>	<integer>	Returns an integer for the first numeric value in the string.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1
position <integer> of <string>	<i>Numbered</i>	<string position>	Returns a string position pointing to the character position specified. The first character is at position 0.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
position of <string>	<i>Plain</i>	<string position>	Returns the positions of the string.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
substring <string> of <string>	<i>Named</i>	<substring>	Iterates through the string returning all the substrings matching the name given. For example, number of substrings "be" of "to be or not to be" = 2.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
substring after <string> of <string>	<i>Named</i>	<substring>	Returns the substrings that come after the first string delimiter.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
substring before <string> of <string>	<i>Named</i>	<substring>	Returns the substrings that come before the first string delimiter.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
substring between <string> of <string>	<i>Named</i>	<substring>	Returns the substring in the second string found between two instances of the first string.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
substring separated by <string> of <string>	<i>Named</i>	<substring>	Returns a substring (or set of substrings) delimited by the first string.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1

Key Phrase	Form	Return Type	Description
unique value of <string>	<i>Plain</i>	<string>	Returns the same strings, but with duplicates removed. • Note: As of version 6.0 of BES, this Inspector returns a <string with multiplicity>. <small>Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
unique value of <string>	<i>Plain</i>	<string with multiplicity>	Given a list of strings, returns the count of each unique string. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Operators

Key phrase	Return Type	Description
<rope> & <string>	<rope>	Concatenates a rope and a string, producing a rope. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<rope> contains <string>	<boolean>	Returns boolean TRUE if the rope contains the string. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> & <rope>	<rope>	Concatenates a rope and a string, returning a new rope. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> & <string>	<string>	Concatenates two strings, producing a new string. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> {cmp} <string>	<boolean>	Returns a boolean TRUE/FALSE depending on the result of the comparison, where: • {cmp} is one of: =, !=, <, <=, >, >= . <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

## Note

Many Inspectors return string values from the operating system using a variety of APIs. For the most part, these strings are encoded as single-byte character sets (SBCSs) or multi-byte character sets (MBCSs) depending on the active code page. You can use the code page Inspectors to determine which page is currently active on the client.

## Examples

- floating point "600987.9" as scientific notation
- Returns 6.009879e+5.

- floating point "6.009e8" as standard notation
  - ▶ Returns 600900000.
- 255 as hexadecimal
  - ▶ Returns the string "ff".
- concatenation ":" of (names of files of folder "c:\")
  - ▶ Returns a single string with the names of each file in the specified path separated by a colon.
- concatenation of "light" & "year"
  - ▶ Returns "lightyear".
- 01 Apr 2020 as date
  - ▶ Returns Wed, 01 Apr 2020.
- Tue as day\_of\_week
  - ▶ Returns Tuesday.
- 4.5 as floating point
  - ▶ Returns 4.5.
- exists character whose (it is "z") of "Paul Cezanne"
  - ▶ Returns True.
- concatenation "/" of ("a" ; "b" ; "c" )
  - ▶ Returns "a/b/c".
- concatenation of (name of it & ":") of files of folder "c:\"
  - ▶ Returns a single string with the names of each file in the specified path separated by a colon.
- first 2 of pathname of regapp "bigfix.exe" as lowercase = "c:"
  - ▶ Returns true if BigFix is installed on drive C:.
- preceding text of last "ab" of "abracadabra" is "abracad"
  - ▶ Returns True.
- substrings after ":" of "definition: after the colon"
  - ▶ Returns " after the colon".
- substrings before "<--" of "the item pointed to <--"
  - ▶ Returns "the item pointed to".
- substrings between "\*" of "the item \*between\* asterisks"
  - ▶ Returns "between".

- substrings separated by "," of "1,2,3"
- Returns the list of numbers separated by commas in the specified string.
- multiplicities of unique values of ("steak"; "chop"; "rib"; "rib"; "rib")
- Returns the multiplicity of (the number of times) each string in the list is used, namely, 1,3,1.
- Note that the multiplicities are based on the alphabetic order of the strings (chop, rib, steak), not their position in the list.

## String Position

String position works in combination with the string and substring data types. A string position is a point within a string. It can be compared to an integer (which it is derived from), but it also acts as a pointer within a string so that the preceding and following text can be extracted. A substring (a part of a larger string) is derived from a string object.

### Creation Methods

Key Phrase	Form	Description
end of <substring>	<i>Plain</i>	Creates an object corresponding to the position in the string of the end of the substring. For example, end of first "be" of "To be or not to be" = 5.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
position <integer> of <string>	<i>Numbered</i>	Creates an index (zero based) into the string. For example, position 5 of "to be or not to be" = 5.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
position of <string>	<i>Plain</i>	Iterates through the string returning values for all possible positions within it. For example, number of positions of "hi" = 3. Note that the positions being counted here are 0, 1, and 2.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
start of <substring>	<i>Plain</i>	Creates the position of the substring within its containing string. For example, Start of substring "or" of "to be or not to be" = 6.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

## Properties

Key Phrase	Form	Return Type	Description
following text of <string position>	<i>Plain</i>	<substring>	Returns the substring following the position in the string. For example, following text of position 5 of "0123456789" = "567890".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
preceding text of <string position>	<i>Plain</i>	<substring>	Returns the substring preceding the position in the string. For example, preceding text of position 5 of "0123456789" = "01234".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

## Note

String positions can be compared and combined with integers using the same operations that are available for integers.

## Examples

- preceding text of position 5 of "Four score and seven..."
- Returns "Four".

---

## Substring

A substring object is derived from a string object, so it has all the properties of a string. Substrings also have these additional properties:

## Creation Methods

Key Phrase	Form	Description
character <integer> of <string>	<i>Numbered</i>	Creates the single character substring at the position given within the string. For example, character 2 of "abc" = "c". Note that numbering begins at zero.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
character of <string>	<i>Plain</i>	Iterates through the string (or substring) returning substrings that contain the individual characters of the string. For example, number of characters of string "abc" = 3.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

Key Phrase	Form	Description
first <integer> of <string>	<i>Numbered</i>	Creates a substring for the given number of characters at the start of the string.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
first <string> of <string>	<i>Named</i>	Creates an object containing the first match of the given string. For example, first "be" of "to be or not to be" = "be".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
following text of <string position>	<i>Plain</i>	Creates an object containing the substring following the position in the string. For example, following text of position 5 of "0123456789" = "567890".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
following text of <substring>	<i>Plain</i>	Creates an object containing the string following the substring. For example, following text of last "." of "log.txt" = "txt".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
last <integer> of <string>	<i>Numbered</i>	Creates an object containing a substring from the last part of the string containing the number of characters specified. For example, Last 5 of "To be or not to be" is "to be".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
last <string> of <string>	<i>Named</i>	Creates a substring containing the last occurrence of the name provided.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
preceding text of <string position>	<i>Plain</i>	Creates the substring preceding the position in the string. For example, preceding text of position 5 of "0123456789" = "01234".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
preceding text of <substring>	<i>Plain</i>	Creates an object containing the string preceding the substring. For example, preceding text of last "." of "log.txt" = "log".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
substring <string> of <string>	<i>Named</i>	Iterates through the string returning all the substrings matching the name given. For example, number of substrings "be" of "to be or not to be" = 2.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

Key Phrase	Form	Description
substring after <string> of <string>	<i>Named</i>	Returns the substrings that come after the first string delimiter.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
substring before <string> of <string>	<i>Named</i>	Returns the substrings that come before the first string delimiter.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
substring between <string> of <string>	<i>Named</i>	Returns the substring in the second string found between two instances of the first string.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
substring separated by <string> of <string>	<i>Named</i>	Returns a substring (or set of substrings) delimited by the first string.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1

## Properties

Key Phrase	Form	Return Type	Description
end of <substring>	<i>Plain</i>	<string position>	Returns the position of the substring within its containing string. For example, end of first "be" of "to be or not to be" = 5.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
following text of <substring>	<i>Plain</i>	<substring>	Returns the string following the substring. For example, following text of last "." of "log.txt" = "txt".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
preceding text of <substring>	<i>Plain</i>	<substring>	Returns the string preceding the substring. For example, preceding text of last "." of "log.txt" = "log".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
start of <substring>	<i>Plain</i>	<string position>	Returns the position within the string of the substring. For example, start of substring "or" of "to be or not to be" = 6.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

## Note

All the string operators can also be applied to substrings.

## Examples

- first 2 of pathname of regapp "bigfix.exe" as lowercase = "c:"
  - ▶ Returns true if BigFix is installed on drive C:.
- substrings after ":" of "definition: after the colon"
  - ▶ Returns " after the colon".
- substrings before "<--" of "the item pointed to <--"
  - ▶ Returns "the item pointed to".
- substrings between "\*" of "the item \*between\* asterisks"
  - ▶ Returns "between".
- substrings separated by "," of "1,2,3"
  - ▶ Returns the list of numbers separated by commas in the specified string.

---

## String With Multiplicity

These Inspectors deal with arrays of strings, allowing you to pluck out unique strings and count them.

### Creation Methods

Key Phrase	Form	Description
unique value of <string>	<i>Plain</i>	Given a list of strings, returns the count of each unique string.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

### Properties

Key Phrase	Form	Return Type	Description
multiplicity of <string with multiplicity>	<i>Plain</i>	<integer>	Returns the multiplicity (quantity) of each element in a multiple string list.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Examples

- unique values of ("steak"; "chop"; "rib"; "rib"; "rib")
  - ▶ Returns the unique values of the multiple strings in alphabetical order, namely chop, rib, steak.
- multiplicities of unique values of ("steak"; "chop"; "rib"; "rib"; "rib")
  - ▶ Returns the multiplicity of (the number of times) each string in the list is used, namely, 1,3,1.
  - Note that the multiplicities are based on the alphabetic order of the strings (chop, rib, steak), not their position in the list.

## Rope

The rope object is a way to efficiently concatenate long strings. String literals in the Relevance language are limited to 512 characters, but internally, they can be any length. Ropes provide a technique for concatenating string literals that is memory-efficient. In general, the Fixlet author will not need to worry about ropes, but they are useful for increasing efficiency.

### Creation Methods

Key Phrase	Form	Description
rope <string>	<i>NamedGlobal</i>	Creates a rope object from the given string. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

### Properties

Key Phrase	Form	Return Type	Description
<rope> as string	<i>Cast</i>	<string>	Converts a rope into a string object. Once converted, all the other string properties are available. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
length of <rope>	<i>Plain</i>	<integer>	Returns the number of bytes in the rope. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

## Operators

Key phrase	Return Type	Description
<rope> & <rope>	<rope>	Concatenates two ropes into a new rope. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<rope> & <string>	<rope>	Concatenates a rope and a string, producing a rope. <small>Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:5.1</small>
<rope> contains <string>	<boolean>	Returns TRUE if the rope contains the string. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> & <rope>	<rope>	Concatenates a rope and a string, returning a new rope. <small>Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:5.1</small>

## Bit Set

A small, numbered collection of bits that can be examined and manipulated.

## Creation Methods

Key Phrase	Form	Description
<integer> as bit set	<i>Cast</i>	Returns the bits of the binary representation of the integer; bit zero is the least-significant bit. <small>Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1</small>
<integer> as bits	<i>Cast</i>	Returns the bits of the binary representation of the integer; bit zero is the least-significant bit. <small>Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1</small>
bit <integer>	<i>NumberedGlobal</i>	Creates a <bit set> object representing the nth bit position as specified by the integer. The integer value must be between 0 and 63 corresponding to the bit position of interest. <small>Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1</small>
bit set <string>	<i>NamedGlobal</i>	Returns the bits of the binary number given by the string. <small>Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1</small>
left shift <integer> of <bit set>	<i>Numbered</i>	A bit set which, at each position $n \geq \text{delta}$ , holds bit $n - \text{delta}$ of the original bit set, where delta is the given integer. <small>Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1</small>

Key Phrase	Form	Description
right shift <integer> of <bit set>	<i>Numbered</i>	A bit set which, at each position n, holds bit n+delta of the original bit set, where delta is the given shift integer.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1

## Properties

Key Phrase	Form	Return Type	Description
<bit set> as integer	<i>Cast</i>	<integer>	Returns the integer whose binary representation matches the bit set.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
<bit set> as string	<i>Cast</i>	<string>	Returns the bits (0s and 1s) in a string format.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
bit <integer> of <bit set>	<i>Numbered</i>	<boolean>	Returns the value of the bit at the given <integer> position in the set.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
least significant one bit of <bit set>	<i>Plain</i>	<integer>	Returns the least n such that bit n of the set is true.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
left shift <integer> of <bit set>	<i>Numbered</i>	<bit set>	A bit set which, at each position n $\geq$ delta, holds bit n-delta of the original bit set, where delta is the given integer.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
most significant one bit of <bit set>	<i>Plain</i>	<integer>	Returns the greatest n such that bit n of the set is true.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
one bit of <bit set>	<i>Plain</i>	<integer>	Returns the numbers n for which bit n of the set is true.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
right shift <integer> of <bit set>	<i>Numbered</i>	<bit set>	A bit set which, at each position n, holds bit n+delta of the original bit set, where delta is the given shift integer.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1

## Operators

Key phrase	Return Type	Description
<bit set> - <bit set>	<bit set>	Returns the bits that are true in the left bit set and false in the right bit set.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
<bit set> * <bit set>	<bit set>	Returns the intersection of the two bit sets.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
<bit set> + <bit set>	<bit set>	Returns the union of the two sets.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
<bit set> = <bit set>	<boolean>	Returns true if the corresponding bits of the two sets are equal.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
<bit set> contains <bit set>	<boolean>	Returns false if -- for any n -- bit n of the left set is false, but bit n of the right set is true.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1

## Examples

- bit 0 of 5  
▶ Returns TRUE.
- bit 3 of bit 3  
▶ Returns TRUE.

---

## Regular Expression

These Inspectors let you use regular expressions (or regexes) in relevance statements. They use the boost library implementation of the 'POSIX-Extended' regular expression syntax, as documented at:

- [http://www.boost.org/libs/regex/doc/syntax\\_extended.html](http://www.boost.org/libs/regex/doc/syntax_extended.html).

## Creation Methods

Key Phrase	Form	Description
case insensitive regex <string>	<i>NamedGlobal</i>	Creates a case-insensitive regular expression (regex) from the specified string.  Win:6.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
case insensitive regular expression <string>	<i>NamedGlobal</i>	Same as case insensitive regex <string>.  Win:6.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

Key Phrase	Form	Description
regex <string>	<i>NamedGlobal</i>	Creates a regex object from the given string. <small>Win:6.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
regular expression <string>	<i>NamedGlobal</i>	Same as regex <string>. <small>Win:6.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

## Operators

Key phrase	Return Type	Description
<regular expression> = <string>	<boolean>	Returns TRUE if the regular expression is equal to the specified string. <small>Win:6.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> = <regular expression>	<boolean>	Returns TRUE if the regular expression is equal to the specified string. <small>Win:6.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> contains <regular expression>	<boolean>	Returns TRUE if the specified string contains the contents of the regular expression. <small>Win:6.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> ends with <regular expression>	<boolean>	Returns TRUE if the string ends with the contents of the regular expression. <small>Win:6.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> starts with <regular expression>	<boolean>	Returns TRUE if the string starts with the contents of the regular expression. <small>Win:6.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

## Examples

- `regex ".*match.*" = "We will win the match tonight"`  
► Returns TRUE.
- `regex ".*PATH.*" = line 1 of file "/etc/profile"`  
► Returns TRUE if the word "PATH" exists in line 1 of the given file.

## Regular Expression Match

These Inspectors let you match regular expressions (or regexes) in relevance statements. They use the boost library implementation of the 'POSIX-Extended' regular expression syntax, as documented at:

- [http://www.boost.org/libs/regex/doc/syntax\\_extended.html](http://www.boost.org/libs/regex/doc/syntax_extended.html). These objects are derived from substring objects.

### Creation Methods

Key Phrase	Form	Description
first match <regular expression> of <string>	<i>Indexed</i>	Creates an object containing the first match to the regular expression in the given string. <small>Win:6.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
match <regular expression> of <string>	<i>Indexed</i>	Creates an object containing all the matches to the regular expression in the given string. <small>Win:6.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

### Properties

Key Phrase	Form	Return Type	Description
parenthesized part <integer> of <regular expression match>	<i>Numbered</i>	<substring>	Returns the nth parenthetical (given by <integer>) in the specified regular expression match. <small>Win:6.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
parenthesized part of <regular expression match>	<i>Plain</i>	<substring>	Returns the parenthetical part of the specified regular expression match. <small>Win:6.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

### Examples

- first match (regex "to.+") of "just too hot to handle"
- Returns "too hot to handle".

## Undefined

The "undefined" type is used as the result type of Inspectors that never return a value.

### Creation Methods

Key Phrase	Form	Description
error <string>	<i>NamedGlobal</i>	Always fails; if an error message is generated, it is based on the given string.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1

### Examples

- if FALSE then 1 else error "my error message"
- Returns the string: User-defined error: my error message.

## Hertz

The hertz object is useful to measure clock cycles. It is used primarily to measure clock frequency by the speed of the processor Inspector. Hertz objects have a resolution of 1 hertz and are stored internally as a 64 bit signed integer.

### Creation Methods

Key Phrase	Form	Description
absolute value of <hertz>	<i>Plain</i>	Creates a hertz object with a positive value.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
ghz	<i>PlainGlobal</i>	Creates a hertz object corresponding to 1 giga-hertz. For example, ghz = 1000*mhz.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
greatest hz	<i>PlainGlobal</i>	Creates the largest hertz object that can be represented on the current machine. It returns the value 9,223,372,036,854,775,807 hertz.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
hz	<i>PlainGlobal</i>	Creates a hertz object corresponding to 1 hertz.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
khz	<i>PlainGlobal</i>	Creates a hertz object corresponding to 1 kilohertz. For example, khz = 1000*hz.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

Key Phrase	Form	Description
least hz	<i>PlainGlobal</i>	Creates the largest negative hertz object that can be represented on the current machine. It returns the value -9,223,372,036,854,775,808 hertz. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
mhz	<i>PlainGlobal</i>	Creates a hertz object corresponding to 1 megahertz. For example, mhz = 1000*khz. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
significant digits <integer> of <hertz>	<i>Numbered</i>	Rounds up the value of a hertz object with <integer> significant digits. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

## Properties

Key Phrase	Form	Return Type	Description
<hertz> as string	<i>Cast</i>	<string>	Returns a string formatted "##### hertz". <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
absolute value of <hertz>	<i>Plain</i>	<hertz>	Returns the positive value of the hertz object. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
significant digits <integer> of <hertz>	<i>Numbered</i>	<hertz>	Returns the value of a hertz object with <integer> significant digits (e.g.. significant digits 3 of 1235569 = 1240000). <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

## Operators

Key phrase	Return Type	Description
- <hertz>	<hertz>	Returns the negative of the <hertz> value. <small>Win:2.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:5.1</small>
<hertz> {cmp} <hertz>	<boolean>	Returns a boolean TRUE/FALSE depending on the result of the comparison, where: <ul style="list-style-type: none"> <li>{cmp} is one of: =, !=, &lt;, &lt;=, &gt;, &gt;= .</li> </ul> <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<hertz> {op} <hertz>	<hertz>	Returns a hertz object equal to the result of the operation, where: <ul style="list-style-type: none"> <li>{op} is one of: +, -, mod .</li> </ul> <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

Key phrase	Return Type	Description
<hertz> {op} <integer>	<hertz>	Returns a hertz object equal to the result of the operation, where: <ul style="list-style-type: none"> <li>• {op} is one of: *, / .</li> </ul> Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

## Examples

- `speed of processor > 3*ghz`
  - ▶ Returns TRUE on machines faster than 3Ghz.
- `greatest hz`
  - ▶ Returns a large positive value, such as 9223372036854775807 hertz.
- `least hz`
  - ▶ Returns a large negative value, such as -9223372036854775808 hertz.
- `significant digits 2 of speed of processor/mhz/ 1000 as floating point`
  - ▶ Returns a floating point representation of the processor speed in GHz, such as 3.4 ghz.
- `significant digits 3 of 1235569`
  - ▶ Returns 1240000.
- `speed of processor`
  - ▶ Returns the speed of the processor in hz, such as 3394000000 hertz for a 3.4 GHz computer.

---

## Time

A time object is used to identify a point in time. Time objects are used to represent important properties of objects such as the modification time of a file. You can create time objects from literal strings. The format of the string is defined by the MIME standard. The difference between two Time objects may be calculated by subtracting them and yields time intervals. Time intervals may be added or subtracted from time objects to obtain time objects.

## Creation Methods

Key Phrase	Form	Description
<string> as local time	<i>Cast</i>	Local time creates a time object by parsing the string literal provided. The time zone is optional. If not present, the local time zone is assumed. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

Key Phrase	Form	Description
<string> as time	<i>Cast</i>	Parses the string. Time zone information must be provided. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> as universal time	<i>Cast</i>	Parses the string. If time zone is not provided in the string, the universal time zone is assumed. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<string> as windows display time	<i>Cast</i>	Parses the string. If time zone is not provided in the string, the current time zone in effect at the given time is assumed. <small>Win:1.2</small>
maximum of <time>	<i>Plain</i>	Returns the maximum time from a list of times. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
minimum of <time>	<i>Plain</i>	Returns the minimum time from a list of times. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
now	<i>PlainGlobal</i>	Creates an object for the current time. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
time <string>	<i>NamedGlobal</i>	The time inspector creates a time object by parsing the string literal provided. The zone info is required. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
universal time <string>	<i>NamedGlobal</i>	The universal time inspector returns a time object by parsing the string literal provided. The time zone is optional. If not present, universal time is assumed. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
windows display time <string>	<i>NamedGlobal</i>	Creates an object for a string that may match the time shown in the Windows file system. <small>Win:1.2</small>

## Properties

Key Phrase	Form	Return Type	Description
<time> as local string	<i>Cast</i>	<string>	Returns a string in MIME format of the given time object. The format is: ddd, DD mmm YYYY HH:MM:SS sZZZZ. The string is formatted using the local time zone. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

Key Phrase	Form	Return Type	Description
<time> as string	<i>Cast</i>	<string>	Same as above. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<time> as universal string	<i>Cast</i>	<string>	Returns a string in MIME format of the given time object. The format is: <ul style="list-style-type: none"> <li>• ddd, DD mmm YYYY HH:MM:SS +0000</li> <li>• The string is formatted using the universal time zone.</li> </ul> <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
date <time zone> of <time>	<i>Indexed</i>	<date>	Returns the date adjusted for the specified time zone. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
maximum of <time>	<i>Plain</i>	<time>	Returns the maximum time from a list of times. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
minimum of <time>	<i>Plain</i>	<time>	Returns the minimum time from a list of times. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
time <time zone> of <time>	<i>Indexed</i>	<time of day with time zone>	Adjusts the specified time to the given time zone. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Operators

Key phrase	Return Type	Description
<time interval> & <time>	<time range>	Concatenates a time interval with a time, returning a time range of the form time1 to time2. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<time range> & <time>	<time range>	Concatenates a time with a time range, producing a new time range, in the form of: <ul style="list-style-type: none"> <li>• &lt;date&gt; to &lt;date&gt;.</li> </ul> <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<time> & <time interval>	<time range>	Concatenates a time and a time interval, producing a time range object. <small>Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<time> & <time range>	<time range>	Concatenates a time and a time range, producing a new time range. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

Key phrase	Return Type	Description
<time> & <time>	<time range>	Concatenates two times into a time range, with the earliest date first and the latest date last.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<time> {cmp} <time>	<boolean>	Returns a boolean TRUE/FALSE depending on the result of the comparison, where: <ul style="list-style-type: none"><li>• {cmp} is one of: =, !=, &lt;, &lt;=, &gt;, &gt;= .</li></ul> Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
<time> {op} <time interval>	<time>	Returns a <time> corresponding to the operator, where: <ul style="list-style-type: none"><li>• {op} is one of: +, -.</li></ul> Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

## Note

The string format for a time object is given by the MIME standard. When output as a string, the format used is:

**ddd, DD mmm YYYY HH:MM:SS sZZZZ**

where:

<b>ddd</b>	The day of the week. Abbreviations are Mon, Tue, Wed, Thu, Fri, Sat, Sun.
<b>DD</b>	The day of the month. A leading zero will be applied to make it two characters wide.
<b>mmm</b>	The Month. Abbreviations are Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec.
<b>YYYY</b>	The year.
<b>HH</b>	The hour of the day. It is always output at two digits. Possible values run from 0 to 23. The digits 00 are used to designate midnight.
<b>MM</b>	The minutes of the hour. It is always output as two digits. Possible values run from 0 to 59.
<b>SS</b>	The seconds of the minute.
<b>s</b>	A single character representing whether the time is east or west of Greenwich. The value of + means east of Greenwich while the value - means west of Greenwich.
<b>ZZZZ</b>	The number of minutes east or west of Greenwich.

To create a string from a literal, use the format:**ddd,DD mmm YYYY HH:MM:SS zoneinfo**

where:

<b>ddd</b>	The optional day of the week. Abbreviations are Mon, Tue, Wed, Thu, Fri, Sat, Sun. Case is not important in these names. If provided it must be correct. For example, time "Sat, 19 jun 1998 00:00:00 +0000" will fail since June 19, 1998 was a Friday.
<b>DD</b>	The day of the month. One or two digits are allowed.
<b>mmm</b>	The Month. Abbreviations are Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec. Case is not important.
<b>YYYY</b>	The year. A two, three or four digit year. If two digits are given a base of 1900 is assumed.
<b>HH</b>	The hour of the day. It is always input at two digits. Possible values run from 0 to 23. The digits 00 are used to designate midnight.
<b>MM</b>	The minutes of the hour. It is always output as two digits. Possible values run from 0 to 59.
<b>SS</b>	The seconds of the minute. Range from 0 to 59. This is optional. If seconds are not present, the preceding colon should also not be present
<b>Zoneinfo</b>	The time zone information. It is provided in one of these formats: Single character + or - followed by 4 digits. The 4 digits are interpreted as HHMM two digits of hours and two digits of minutes. Plus designates east of universal time while minus designates west of universal time. Three letters for the civilian name of the time zone. cdt, edt, mdt, pdt are the designations for central, eastern, mountain and pacific daylight savings time while cst, est, mst, pst are the designations for central, eastern, mountain and pacific standard time. gmt designates Greenwich mean time. A single letter military name of the time zone. Military time zones use single letters from a to z, excepting j. a-m represent offsets from universal time of -1 to -12 hours respectively. z represents 0 offset.

## Examples

- maximum of (modification times of files of folder "temp" of windows folder)
  - ▶ Returns the latest time stamp from the files in the windows temporary folder.
- minimum of (modification times of files of folder "temp" of windows folder)
  - ▶ Returns the latest earliest stamp from the files in the windows temporary folder.
- now
  - ▶ Returns the current time.
- time "Sat, 01 Jan 2000 00:00:00 -0400" & now
  - ▶ Returns a time range from the beginning of the millennia to now, eg:
    - Sat, 01 Jan 2000 00:00:00 -0400 to Sat, 08 Apr 2006 20:39:51 -0400.

---

## Time Of Day

These Inspectors provide tools for dealing and calculating with time-of-day types, which are of the form HH:MM:SS, as in 12:59:59.

### Creation Methods

Key Phrase	Form	Description
<string> as time_of_day	<i>Cast</i>	Converts a string to a time_of_day type. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
midnight	<i>PlainGlobal</i>	Returns 00:00:00 as a time of day object. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
noon	<i>PlainGlobal</i>	Returns 12:00:00 as a time of day object. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
time of <time of day with time zone>	<i>Plain</i>	Returns the time of day, without the time zone information. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
time_of_day <string>	<i>NamedGlobal</i>	Creates a time of day object out of the given string. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Properties

Key Phrase	Form	Return Type	Description
<time of day> as string	<i>Cast</i>	<string>	Casts the time of day as a string type. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
hour_of_day of <time of day>	<i>Plain</i>	<integer>	Returns the hour section of the 'time of day' object. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
minute_of_hour of <time of day>	<i>Plain</i>	<integer>	Returns the 'minutes after the hour' section of the 'time of day' object. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
second_of_minute of <time of day>	<i>Plain</i>	<integer>	Extracts the 'seconds after the minute' section of the 'time of day' object. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
two digit hour of <time of day>	<i>Plain</i>	<string>	Extracts the 2-digit hour from the time of day. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
two digit minute of <time of day>	<i>Plain</i>	<string>	Extracts the 2-digit minute from the time of day. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
two digit second of <time of day>	<i>Plain</i>	<string>	Extracts the 2-digit second from the time of day. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Operators

Key phrase	Return Type	Description
<time of day> - <time of day>	<time interval>	Subtracts two times of day, returning a time interval. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<time of day> & <time zone>	<time of day with time zone>	Concatenates a time of day with a time zone, returning a time of day with time zone type. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<time of day> {cmp} <time of day>	<boolean>	Compares two times of day, where {cmp} is one of: <, <=, =. <small>Win:6.0</small>
<time zone> & <time of day>	<time of day with time zone>	Concatenates a time of day with a time zone, returning a time of day with time zone type. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Time Zone

Time zones are used in conjunction with the time object. Time zones have a resolution of 1 minute.

### Creation Methods

Key Phrase	Form	Description
<string> as time zone	<i>Cast</i>	Creates a time zone object corresponding to the string provided. For example, "pst" as time zone.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
local time zone	<i>PlainGlobal</i>	Creates a time zone object corresponding to the local time zone.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
time zone <string>	<i>NamedGlobal</i>	Creates a time zone object corresponding to the string provided. For example, time zone "edt" as string = "-0400".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
universal time zone	<i>PlainGlobal</i>	Creates a time zone object corresponding to the universal time zone.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

### Properties

Key Phrase	Form	Return Type	Description
<time zone> as string	<i>Cast</i>	<string>	Returns a string corresponding to the time zone object provided.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

### Operators

Key phrase	Return Type	Description
<time of day> & <time zone>	< <i>time of day with time zone</i> >	Concatenates a time of day with a time zone, returning a time of day with time zone type.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<time zone> & <time of day with time zone>	< <i>time of day with time zone</i> >	Converts a 'time of day with time zone' to the time in the specified time zone.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

Key phrase	Return Type	Description
<time zone> & <time of day>	<time of day with time zone>	Concatenates a time of day with a time zone, returning a time of day with time zone type.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<time zone> {op} <time interval>	<time zone>	Returns a time zone object offset by a time interval, where: <ul style="list-style-type: none"> <li>• {op} is one of: +, - .</li> </ul> Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

## Examples

- local time zone - 2 \* hour
- Returns the time zone two hours away.

---

## Time Of Day With Time Zone

These Inspectors provide tools for dealing and calculating with time-of-day-with-time-zone types, which are of the form HH:MM:SS +ZZZZ, as in 12:59:59 -0400.

## Creation Methods

Key Phrase	Form	Description
<string> as local zoned time_of_day	Cast	Converts a string to a time of day with local time zone.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<string> as universal zoned time_of_day	Cast	Converts a string into a universal zoned time of day.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<string> as zoned time_of_day	Cast	Converts a string into a zoned time of day.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
time <time zone> of <time>	Indexed	Converts the specified time to the given time zone.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
zoned time_of_day <string>	NamedGlobal	Creates a 'zoned time of day' out of a string object in the form of HH:MM:SS +ZZZZ.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Properties

Key Phrase	Form	Return Type	Description
<time of day with time zone> as string	<i>Cast</i>	<string>	Converts a 'time of day with time zone' object into a string.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
hour_of_day of <time of day with time zone>	<i>Plain</i>	<integer>	Returns the hour section of the 'time of day with time zone' object.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
minute_of_hour of <time of day with time zone>	<i>Plain</i>	<integer>	Returns the 'minutes after the hour' section of the 'time of day with time zone' object.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
second_of_minute of <time of day with time zone>	<i>Plain</i>	<integer>	Returns the 'seconds after the minute' section of the 'time of day with time zone' object.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
time of <time of day with time zone>	<i>Plain</i>	<time of day>	Returns the time of day, without the time zone information.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
zone of <time of day with time zone>	<i>Plain</i>	<time zone>	Returns the zone associated with the specified time.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Operators

Key phrase	Return Type	Description
<date> & <time of day with time zone>	<time>	Concatenates a date with a time and a time zone for a complete time stamp.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<time of day with time zone> - <time of day with time zone>	<time interval>	Subtracts two times of day (including time zones), returning a time interval.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<time of day with time zone> & <time zone>	<time of day with time zone>	Concatenates a 'time of day with a time zone' and another time zone. The 'time of day with time zone' object that is produced is adjusted to fit the appended time zone.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

Key phrase	Return Type	Description
<time of day with time zone> = <time of day with time zone>	<boolean>	Compares two times of day with time zone.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<time zone> & <time of day with time zone>	<time of day with time zone>	Converts a 'time of day with time zone' to the time in the specified time zone.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Examples

- 12:00:00 -4000 as universal zoned time\_of\_day
  - ▶ Returns 04:00:00 +0000.
- time (time zone "+0000") of now
  - ▶ Returns the time in Greenwich, England.
- hour\_of\_day of time (universal time zone) of now
  - ▶ Returns the hour of day in Greenwich, England.
- minute\_of\_hour of time (local time zone) of now
  - ▶ Returns the current minute past the hour.
- time zone "+0000" & time (universal time zone) of now
  - ▶ Returns the time in Greenwich, England.

## Time Range

These Inspectors provide tools for dealing and calculating with time-range types, which are of the form <time> to <time>, such as Tue, 18 Apr 2006 16:46:07 -0400 to Wed, 19 Apr 2006 16:46:07 -0400

## Creation Methods

Key Phrase	Form	Description
final part <time interval> of <time range>	<i>Indexed</i>	Returns a time range with the specified interval, but ending on the final date of the time range.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
initial part <time interval> of <time range>	<i>Indexed</i>	Returns a time range starting with the first date of the time range and lasting for the specified interval.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

Key Phrase	Form	Description
range after <time> of <time range>	<i>Indexed</i>	Returns a new time range, starting from the specified time and continuing through the end of the original range. The time must be within the range, or an error will result.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
range before <time> of <time range>	<i>Indexed</i>	Returns a new time range, starting from the original time in the specified range and continuing to the specified time. The time must be within the range, or an error will result.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Properties

Key Phrase	Form	Return Type	Description
<time range> as string	<i>Cast</i>	<string>	Casts a time range as a string.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
end of <time range>	<i>Plain</i>	<time>	Returns the end date of a time range.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
final part <time interval> of <time range>	<i>Indexed</i>	<time range>	Returns a time range with the specified interval, but ending on the final date of the time range.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
initial part <time interval> of <time range>	<i>Indexed</i>	<time range>	Returns a time range starting with the first date of the time range and lasting for the specified interval.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
length of <time range>	<i>Plain</i>	<time interval>	Returns the time interval (in days, hours, minutes, seconds) between the start and end date of a time range.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
range after <time> of <time range>	<i>Indexed</i>	<time range>	Returns a new time range, starting from the specified time and continuing through the end of the original range. The time must be within the range, or an error will result.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

Key Phrase	Form	Return Type	Description
range before <time> of <time range>	<i>Indexed</i>	<time range>	Returns a new time range, starting from the original time in the specified range and continuing to the specified time. The time must be within the range, or an error will result.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
start of <time range>	<i>Plain</i>	<time>	Returns the starting date of a time range.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Operators

Key phrase	Return Type	Description
<time range> & <time range>	<time range>	Returns the smallest range that contains both of the specified ranges (same as <time range> + <time range>).  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<time range> & <time>	<time range>	Concatenates a time with a time range, producing a new time range, in the form of: • <date> to <date>.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<time range> * <time range>	<time range>	Returns the intersection of the two specified time ranges, if one exists.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<time range> + <time range>	<time range>	Returns the smallest range that contains both of the specified ranges (same as <time range> & <time range>).  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<time> & <time range>	<time range>	Concatenates a time and a time range, producing a new time range.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Examples

- (week & now) \* (day & now)
- ▶ Returns a one-day time range (from yesterday to today).
  
- (week & now) + (day & now)
- ▶ Returns a one-week time range (from a week ago to today).

## Time Interval

Time intervals are used in conjunction with the time object. Time intervals have a resolution of 1 microsecond.

### Creation Methods

Key Phrase	Form	Description
<string> as time interval	<i>Cast</i>	Returns a time interval object from a properly formatted string. Expects strings formatted as <ul style="list-style-type: none"> <li>• ddd days, HH:MM:SS.mmmmmm.</li> </ul> Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
absolute value of <time interval>	<i>Plain</i>	Creates the positive value of a time interval. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
day	<i>PlainGlobal</i>	Creates a time interval corresponding to 1 day. For example, 2 * day = 48 * hour. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
greatest time interval	<i>PlainGlobal</i>	Creates the largest time interval that can be represented on the current machine. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
hour	<i>PlainGlobal</i>	Creates a time interval corresponding to 1 hour. For example, day = 24 * hour. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
least time interval	<i>PlainGlobal</i>	Creates the largest negative time interval that can be represented on the current machine. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
maximum of <time interval>	<i>Plain</i>	Returns the maximum interval from a list of time intervals. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
microsecond	<i>PlainGlobal</i>	Creates a time interval corresponding to 1 microsecond. For example, 1000 * microsecond = 1 * millisecond. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
millisecond	<i>PlainGlobal</i>	Creates a time interval corresponding to 1 millisecond. For example, 1000 * millisecond = 1 * second. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

Key Phrase	Form	Description
minimum of <time interval>	<i>Plain</i>	Returns the minimum interval from a list of time intervals. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
minute	<i>PlainGlobal</i>	Creates a time interval corresponding to 1 minute. For example, minute = 60 * second. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
second	<i>PlainGlobal</i>	Creates a time interval corresponding to 1 second. For example, 1000000 * microsecond = second. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
time interval <string>	<i>NamedGlobal</i>	Creates a time interval from the string. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
week	<i>PlainGlobal</i>	Creates a time interval corresponding to 1 week. For example, 7*day = 1*week. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

## Properties

Key Phrase	Form	Return Type	Description
<time interval> as string	<i>Cast</i>	<string>	Returns a string formatted as <ul style="list-style-type: none"> <li>• ddd days, HH:MM:SS.mmmmmm</li> <li>• For example, millisecond as string = "00:00:00.001".</li> </ul> <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
absolute value of <time interval>	<i>Plain</i>	<time interval>	Returns positive value of the time interval. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
maximum of <time interval>	<i>Plain</i>	<time interval>	Returns the maximum interval from a list of time intervals. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
minimum of <time interval>	<i>Plain</i>	<time interval>	Returns the minimum interval from a list of time intervals. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Operators

Key phrase	Return Type	Description
- <time interval>	<time interval>	The negative of a time interval. <small>Win:2.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:5.1</small>
<time interval> & <time>	<time range>	Concatenates a time interval with a time, returning a time range of the form time1 to time2. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<time interval> {op} <integer>	<time interval>	Creates a time interval calculated as an integer operation on another time interval, where: • {op} is one of: *, / . <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<time interval> {op} <time interval>	<time interval>	Returns a calculated time interval, where: • {op} is one of: +, -, mod, / . <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<time interval> {op} <time zone>	<time interval>	Returns a calculated time interval, where: • {op} is one of: +, - . <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
<time interval> + <time of day with time zone>	<time of day with time zone>	Adds a time interval (days, hours, minutes, seconds) to a time of the day with time zone to create a new time of the day with time zone. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<time interval> + <time of day>	<time of day>	Adds a time interval (days, hours, minutes, seconds) to a time of the day to create a new time of the day. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<time of day with time zone> {op} <time interval>	<time of day with time zone>	Adds or subtracts a time interval and a specified 'time of day with time zone' object, where {op} is one of: -, +. <small>Win:6.0</small>
<time of day> {op} <time interval>	<time of day>	Adds or subtracts a time interval to provide a new time of day. Here {op} is one of: -, +. <small>Win:6.0</small>
<time> & <time interval>	<time range>	Concatenates a time and a time interval, producing a time range object. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Examples

- maximum of ("00:00:00" as time interval; "01:01:01"as time interval)  
▶ Returns 01:01:10.
  
- minimum of ("00:00:00" as time interval; "01:01:01"as time interval)  
▶ Returns 00:00:00.
  
- time interval "01:00:00" & now  
▶ Returns a one-hour time range ending now, eg. Fri, 07 Apr 2006 12:36:10 -0400 to Fri, 07 Apr 2006 13:36:10 -0400.

---

## Date

These are the various Inspectors that access the date types.

### Creation Methods

Key Phrase	Form	Description
<string> as date	<i>Cast</i>	Casts a string as a date type. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
april <integer> of <integer>	<i>Numbered</i>	Returns the nth day of april and the specified year as a date (day of week, month day year). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
august <integer> of <integer>	<i>Numbered</i>	Returns the nth day of August and the specified year as a date (day of week, month day year). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
current date	<i>PlainGlobal</i>	Returns the current date in the format: • Day of week, Day Month Year. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
date <string>	<i>NamedGlobal</i>	Converts the given string into a date. The string should be of the form 'Day Month Year' and the returned date will be of the form 'Day of week, Day Month Year'. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
date <time zone> of <time>	<i>Indexed</i>	Returns the date adjusted for the specified time zone. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

Key Phrase	Form	Description
december <integer> of <integer>	<i>Numbered</i>	Returns the nth day of December and the specified year as a date (day of week, month day year). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
february <integer> of <integer>	<i>Numbered</i>	Returns the nth day of February and the specified year as a date (day of week, month day year). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
january <integer> of <integer>	<i>Numbered</i>	Returns the nth day of January and the specified year as a date (day of week, month day year). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
july <integer> of <integer>	<i>Numbered</i>	Returns the nth day of July and the specified year as a date (day of week, month day year). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
june <integer> of <integer>	<i>Numbered</i>	Returns the nth day of June and the specified year as a date (day of week, month day year). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
march <integer> of <integer>	<i>Numbered</i>	Returns the nth day of March and the specified year as a date (day of week, month day year). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
may <integer> of <integer>	<i>Numbered</i>	Returns the nth day of May and the specified year as a date (day of week, month day year). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
november <integer> of <integer>	<i>Numbered</i>	Returns the nth day of November and the specified year as a date (day of week, month day year). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
october <integer> of <integer>	<i>Numbered</i>	Returns the nth day of October and the specified year as a date (day of week, month day year). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
september <integer> of <integer>	<i>Numbered</i>	Returns the nth day of September and the specified year as a date (day of week, month day year). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Properties

Key Phrase	Form	Return Type	Description
<date> as string	<i>Cast</i>	<string>	Cast a date type as a string. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
day_of_month of <date>	<i>Plain</i>	<day of month>	Extracts the day of the month from the specified date. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
day_of_week of <date>	<i>Plain</i>	<day of week>	Extracts the day of the week (Monday, Tuesday, etc.) from the specified date. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
day_of_year of <date>	<i>Plain</i>	<day of year>	Extracts the day of year from the specified date, in the 'Month Day' format. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
month of <date>	<i>Plain</i>	<month>	Returns the month derived from the given date. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
month_and_year of <date>	<i>Plain</i>	<month and year>	Formats the specified date in month year format, eg. March 2012. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
year of <date>	<i>Plain</i>	<year>	Returns the year, extracted from the given date. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Operators

Key phrase	Return Type	Description
<date> - <date>	<time interval>	Subtracts two dates to produce a time interval. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<date> & <time of day with time zone>	<time>	Concatenates a date with a time and a time zone for a complete time stamp. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<date> {op} <number of months>	<date>	Adds or subtracts a specified number of months to a given date, where {op} is one of: -, +. <small>Win:6.0</small>
<date> {op} <time interval>	<date>	Add or subtract a time interval to a date, producing a new date, where {op} is one of: -, +. <small>Win:6.0</small>

Key phrase	Return Type	Description
<number of months> + <date>	<date>	Adds a number of months to a date, returning a new date.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<time interval> + <date>	<date>	Adds a time interval (days, hours, minutes, seconds) to a date to create a new date.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<time of day with time zone> & <date>	<time>	Concatenates a 'time of day with time zone' object with a date object to produce a time object.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Examples

■ 01 Apr 2020 as date

► Returns Wed, 01 Apr 2020.

■ april 1 of 2020

► Returns Wed, 01 Apr 2020..

■ date "09 Apr 2006"

► Returns Sun, 09 Apr 2006.

■ now - time "Sat, 01 Jan 2000 00:00:00 -0000"

► Returns the number of days, hours, minutes and seconds since the turn of the millennia, eg: 2288 days, 17:53:06.

■ current date + 14\*month

► Returns the date 14 months from today, eg. Mon, 07 Jan 2008.

■ current date + time interval "7 days"

► Returns the date a week from now.

## Day Of Week

These Inspectors provide tools for dealing and calculating with day-of-week types, which include Monday, Tuesday, etc. This set of Inspectors includes each day as a self-named object.

### Creation Methods

Key Phrase	Form	Description
<string> as day_of_week	<i>Cast</i>	Casts a string as a day of the week. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
current day_of_week	<i>PlainGlobal</i>	Retruns the current day of the week, eg. Monday, Tuesday, etc. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
day_of_week <string>	<i>NamedGlobal</i>	Converts the given string value to a day of week type, eg. Monday, Tuesday, etc. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
day_of_week of <date>	<i>Plain</i>	Extracts the day of the week from the specified date. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
friday	<i>PlainGlobal</i>	Returns Friday as a day of week object. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
monday	<i>PlainGlobal</i>	Returns the day of week object for Monday. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
saturday	<i>PlainGlobal</i>	Returns Saturday as a day of week object. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
sunday	<i>PlainGlobal</i>	Returns Sunday as a day of week object. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
thursday	<i>PlainGlobal</i>	Returns Thursday as a day of week object. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
tuesday	<i>PlainGlobal</i>	Returns Tuesday as a day of week object. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
wednesday	<i>PlainGlobal</i>	Returns Wednesday as a day of week object. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Properties

Key Phrase	Form	Return Type	Description
<day of week> as string	<i>Cast</i>	<string>	Casts the day of week as a string. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<day of week> as three letters	<i>Cast</i>	<string>	Casts the day of week as a three-letter abbreviation (Mon, Tue, etc.). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Operators

Key phrase	Return Type	Description
<day of week> - <day of week>	<time interval>	Subtract two day of week types (Monday, Tuesday, etc.) to produce a time interval. The answer cannot exceed 6 days. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<day of week> {op} <time interval>	<day of week>	Add or subtract a time interval from a day of the week to produce a new day of week. Here {op} is one of: -, +. <small>Win:6.0</small>
<day of week> = <day of week>	<boolean>	Compares two days of the week and returns a boolean TRUE or FALSE. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<time interval> + <day of week>	<day of week>	Adds a time interval (days, hours, minutes, seconds) to a day of the week to create a new day of the week. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Examples

- Tue as day\_of\_week
  - Returns Tuesday.
- day\_of\_week "Tuesday"
  - Returns Tuesday as a 'day of week' object.
- Saturday as three letters
  - Returns Sat.
- Friday - Wednesday
  - Returns 2.

## Day Of Month

These Inspectors provide tools for dealing and calculating with day-of-month types, which are numbers from 1-31.

### Creation Methods

Key Phrase	Form	Description
<integer> as day_of_month	<i>Cast</i>	Cast an integer as a day of the month type. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<string> as day_of_month	<i>Cast</i>	Casts a string as a day of month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
current day_of_month	<i>PlainGlobal</i>	Returns the current day of the month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
day of <day of year>	<i>Plain</i>	Returns the day of the month of the specified date. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
day_of_month <integer>	<i>NumberedGlobal</i>	Converts the given integer to a day of month type. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
day_of_month <string>	<i>NamedGlobal</i>	Converts the given string value (must be an integer from 1-31) to a day of month type. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
day_of_month of <date>	<i>Plain</i>	Extracts the day of the month from the specified date. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

### Properties

Key Phrase	Form	Return Type	Description
<day of month> as integer	<i>Cast</i>	<integer>	Cast a day of month type as an integer. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<day of month> as string	<i>Cast</i>	<string>	Cast a day of month type as a string. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<day of month> as two digits	<i>Cast</i>	<string>	Cast a day of month type as a two-digit number. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Operators

Key phrase	Return Type	Description
<day of month> - <day of month>	<time interval>	Subtract two day of month types, producing a time interval.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<day of month> & <month and year>	<date>	Concatenate a day of month with a month and year type to produce a complete date.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<day of month> & <month>	<day of year>	Concatenate a day of month with a month type to produce a day of year (eg. April 20).  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<day of month> {cmp} <day of month>	<boolean>	Compare two day of month types, where {cmp} is one of: <, <=, =.  Win:6.0
<day of month> {op} <time interval>	<day of month>	Add or subtract a time interval from a day of month to produce a new day of month. Here {op} is one of: -, +.  Win:6.0
<month and year> & <day of month>	<date>	Concatenates a month and year with a day of month to produce a complete date.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<month> & <day of month>	<day of year>	Concatenates a month and a day of the month to produce a day of year.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<time interval> + <day of month>	<day of month>	Adds a time interval (days, hours, minutes, seconds) to a day of the month to create a new day of the month.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Examples

- day of July 4
  - Returns 4.
- day\_of\_month of current date
  - Returns the current day of the month.
- current day\_of\_month as integer
  - Returns the day of the month as an integer.

■ (day\_of\_month 2) & june of 2008

► Returns Mon, 02 Jun 2008.

■ (day\_of\_month 2) & june

► Returns June 2.

## Day Of Year

These Inspectors provide tools for dealing and calculating with day-of-month types, which are of the form Sun, 01 Apr 2007.

### Creation Methods

Key Phrase	Form	Description
april <integer>	<i>NumberedGlobal</i>	Returns the nth day of april as a 'day of year' type (month day). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
august <integer>	<i>NumberedGlobal</i>	Returns the nth day of August as a 'day of year' type (month day). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
current day_of_year	<i>PlainGlobal</i>	Retruns the current day of the year, in a Month Day format. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
day_of_year of <date>	<i>Plain</i>	Extracts the day of year from the specified date, in the 'Month Day' format. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
december <integer>	<i>NumberedGlobal</i>	Returns the nth day of December as a 'day of year' type (month day). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
february <integer>	<i>NumberedGlobal</i>	Returns the nth day of February as a 'day of year' type (month day). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
january <integer>	<i>NumberedGlobal</i>	Returns the nth day of January as a 'day of year' type (month day). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
july <integer>	<i>NumberedGlobal</i>	Returns the nth day of July as a 'day of year' type (month day).

Key Phrase	Form	Description
		Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
june <integer>	<i>NumberedGlobal</i>	Returns the nth day of June as a 'day of year' type (month day). Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
march <integer>	<i>NumberedGlobal</i>	Returns the nth day of March as a 'day of year' type (month day). Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
may <integer>	<i>NumberedGlobal</i>	Returns the nth day of May as a 'day of year' type (month day). Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
november <integer>	<i>NumberedGlobal</i>	Returns the nth day of November as a 'day of year' type (month day). Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
october <integer>	<i>NumberedGlobal</i>	Returns the nth day of October as a 'day of year' type (month day). Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
september <integer>	<i>NumberedGlobal</i>	Returns the nth day of September as a 'day of year' type (month day). Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Properties

Key Phrase	Form	Return Type	Description
<day of year> as string	<i>Cast</i>	<string>	Casts a day of the year as a string type. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
day of <day of year>	<i>Plain</i>	<day of month>	Returns the day of the month of the specified date. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
month of <day of year>	<i>Plain</i>	<month>	Returns the month portion of the given date (in month day format). Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Operators

Key phrase	Return Type	Description
<day of year> - <day of year>	<time interval>	Subtracts two days of the year to produce a time interval.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<day of year> & <month and year>	<date>	Concatenates a day of the year with a month and year to create a complete date type.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<day of year> & <year>	<date>	Concatenates a day of the year with a year to create a complete date type.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<day of year> {cmp} <day of year>	<boolean>	Compares two days of the year, producing a boolean TRUE or FALSE, where {cmp} is one of: <, <=, =.  Win:6.0
<day of year> {op} <number of months>	<day of year>	Add or subtract a number of months to a day of the year to produce a new day of the year. Here {op} is one of: -, +.  Win:6.0
<day of year> {op} <time interval>	<day of year>	Add or subtract a time interval to a day of the year to produce a new day of the year. Here {op} is one of: -, +.  Win:6.0
<month and year> & <day of year>	<date>	Concatenates a month and year with a day of year to produce a complete date.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<number of months> + <day of year>	<day of year>	Adds a number of months to a day of the year (July 4, say) to produce another day of the year.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<time interval> + <day of year>	<day of year>	Adds a time interval (days, hours, minutes, seconds) to a day of the year to create a new day of the year.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<year> & <day of year>	<date>	Concatenates a year with the day of the year, returning a full date.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Examples

- year 2020 & april 1
- Returns Sun, 01 Apr 2007.

## Month

These Inspectors provide tools for dealing and calculating with month types, which are of the form January, February, etc. This set of Inspectors includes each month as a self-named object.

## Creation Methods

Key Phrase	Form	Description
<integer> as month	<i>Cast</i>	Returns the name of the nth month of the year. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<string> as month	<i>Cast</i>	Converts a string into a month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
april	<i>PlainGlobal</i>	Returns april as an object of type month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
august	<i>PlainGlobal</i>	Returns August as an object of type month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
current month	<i>PlainGlobal</i>	Returns the current month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
december	<i>PlainGlobal</i>	Returns December as an object of type month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
february	<i>PlainGlobal</i>	Returns February as an object of type month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
january	<i>PlainGlobal</i>	Returns January as an object of type month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
july	<i>PlainGlobal</i>	Returns July as an object of type month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
june	<i>PlainGlobal</i>	Returns June as an object of type month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
march	<i>PlainGlobal</i>	Returns March as an object of type month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
may	<i>PlainGlobal</i>	Returns May as an object of type month.

Key Phrase	Form	Description
		Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
month <integer>	<i>NumberedGlobal</i>	Returns the month type corresponding to the given <integer>. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
month <string>	<i>NamedGlobal</i>	Returns a month type corresponding to the given <string>. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
month of <date>	<i>Plain</i>	Returns the month of the given date. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
month of <day of year>	<i>Plain</i>	Returns the month portion of the given date (in month day format). Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
month of <month and year>	<i>Plain</i>	Returns the month portion of the given date (in month year format). Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
november	<i>PlainGlobal</i>	Returns November as an object of type month. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
october	<i>PlainGlobal</i>	Returns October as an object of type month. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
september	<i>PlainGlobal</i>	Returns September as an object of type month. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Properties

Key Phrase	Form	Return Type	Description
<month> as integer	<i>Cast</i>	<integer>	Converts the given month into an integer (1-12). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<month> as string	<i>Cast</i>	<string>	Converts the given month into a string value. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<month> as three letters	<i>Cast</i>	<string>	Converts the given month into a 3-letter string (Jan, Feb, etc.). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<month> as two digits	<i>Cast</i>	<string>	Converts the month into a two digit number (01 - 12). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Operators

Key phrase	Return Type	Description
<day of month> & <month>	<day of year>	Concatenate a day of month with a month type to produce a day of year (eg. April 20). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<month> - <month>	<number of months>	Subtracts two months, returning a positive number of months. If the first month is earlier than the second, it assumes the year has rolled over. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<month> & <day of month>	<day of year>	Concatenates a month and a day of the month to produce a day of year. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<month> & <year>	<month and year>	Returns a date (in month year format) from the concatenation of a month and a year. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<month> {cmp} <month>	<boolean>	Compares the values of two months, where {cmp} is one of: <, <=, =. <small>Win:6.0</small>
<month> {op} <number of months>	<month>	Adds or subtracts a number of months from the given month. Here {op} is one of: -, +. <small>Win:6.0</small>

Key phrase	Return Type	Description
<number of months> + <month>	<month>	Adds a number of months to the given month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<year> & <month>	<month and year>	Returns a date (in month year format) from the concatenation of a month and a year. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Examples

- 5 as month
  - ▶ Returns May.
- january as month - 1 \* month
  - ▶ Returns December (one month before January).
- current month + 2\*month
  - ▶ Returns the name of the month, two months from today.
- month 9
  - ▶ Returns September.
- month "jun"
  - ▶ Returns June.
- month of current date
  - ▶ Returns the current month, eg. September.
- month of (day\_of\_year of (current date + 40\*day))
  - ▶ Returns the name of the month 40 days from today, eg. October.
- january as three letters
  - ▶ Returns Jan.
- january as two digits
  - ▶ Returns 01.
- december - current month
  - ▶ Returns the number of months left until december. If the current month is April, it returns 8 months.
- December 3 & "2032" as year
  - ▶ Returns Fri, 03 Dec 2032.

- `July <= current month`
- ▶ Returns true in the second half of the year, when the month is greater than or equal to July.
  
- `current month + 2*month`
- ▶ Returns the name of the month two months from now. If it's currently January, this would return March.
  
- `year 2134 & april`
- ▶ Returns April 2134.

---

## Month And Year

These Inspectors provide tools for dealing and calculating with month-and-year types, which are of the form month of year, eg., January of 2007.

### Creation Methods

Key Phrase	Form	Description
april of <integer>	<i>Plain</i>	Creates a date (in month year format) corresponding to april of the specified year (as an <integer>). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
august of <integer>	<i>Plain</i>	Creates a date (in month year format) corresponding to August of the specified year (as an <integer>). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
current month_and_year	<i>PlainGlobal</i>	Returns the current date in month year format, eg. January 2012. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
december of <integer>	<i>Plain</i>	Creates a date (in month year format) corresponding to December of the specified year (as an <integer>). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
february of <integer>	<i>Plain</i>	Creates a date (in month year format) corresponding to February of the specified year (as an <integer>). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
january of <integer>	<i>Plain</i>	Creates a date (in month year format) corresponding to January of the specified year (as an <integer>). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
july of <integer>	<i>Plain</i>	Creates a date (in month year format) corresponding to July of the specified year (as an <integer>). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

Key Phrase	Form	Description
june of <integer>	<i>Plain</i>	Creates a date (in month year format) corresponding to June of the specified year (as an <integer>). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
march of <integer>	<i>Plain</i>	Creates a date (in month year format) corresponding to March of the specified year (as an <integer>). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
may of <integer>	<i>Plain</i>	Creates a date (in month year format) corresponding to May of the specified year (as an <integer>). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
month_and_year of <date>	<i>Plain</i>	Formats the specified date in month year format, eg. March 2012. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
november of <integer>	<i>Plain</i>	Creates a date (in month year format) corresponding to November of the specified year (as an <integer>). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
october of <integer>	<i>Plain</i>	Creates a date (in month year format) corresponding to October of the specified year (as an <integer>). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
september of <integer>	<i>Plain</i>	Creates a date (in month year format) corresponding to September of the specified year (as an <integer>). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Properties

Key Phrase	Form	Return Type	Description
<month and year> as string	<i>Cast</i>	<string>	Casts a date (in month year format) as a string. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
first <day of week> of <month and year>	<i>Indexed</i>	<date>	Finds the specific date corresponding to the first day of the week (eg. Friday) for a given month and year. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
first friday of <month and year>	<i>Plain</i>	<date>	Finds the date corresponding to the first Friday of any given month and year. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

Key Phrase	Form	Return Type	Description
first monday of <month and year>	<i>Plain</i>	<date>	Finds the date corresponding to the first Monday of any given month and year. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
first saturday of <month and year>	<i>Plain</i>	<date>	Finds the date corresponding to the first Saturday of any given month and year. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
first sunday of <month and year>	<i>Plain</i>	<date>	Finds the date corresponding to the first Sunday of any given month and year. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
first thursday of <month and year>	<i>Plain</i>	<date>	Finds the date corresponding to the first Thursday of any given month and year. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
first tuesday of <month and year>	<i>Plain</i>	<date>	Finds the date corresponding to the first Tuesday of any given month and year. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
first wednesday of <month and year>	<i>Plain</i>	<date>	Finds the date corresponding to the first Wednesday of any given month and year. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
length of <month and year>	<i>Plain</i>	<time interval>	Returns the number of days in the specified month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
month of <month and year>	<i>Plain</i>	<month>	Returns the name of the month corresponding to the given date. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
year of <month and year>	<i>Plain</i>	<year>	Returns the year portion of the specified date (in month year format). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Operators

Key phrase	Return Type	Description
<day of month> & <month and year>	<date>	Concatenate a day of month with a month and year type to produce a complete date.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<day of year> & <month and year>	<date>	Concatenates a day of the year with a month and year to create a complete date type.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<month and year> - <month and year>	<number of months>	Subtracts two dates (in month year format), returning a number of months.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<month and year> {cmp} <month and year>	<boolean>	Compares two dates (in month year format), where {cmp} is one of: <, <=, =.  Win:6.0
<month and year> {op} <number of months>	<month and year>	Adds or subtracts a number of months from a given date (in month year format), where {op} is one of: -, +.  Win:6.0
<number of months> + <month and year>	<month and year>	Adds a number of months to a given date (in month year format) producing a new date.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Examples

- `month_and_year of current date`
  - Returns the current date formatted as month year, eg. April, 2006.
- `first monday of april of 2020`
  - Returns the date of the first Monday in April 2020, which is Mon, 06 Apr 2020.
- `length of (month "February" & year "2004")`
  - Returns 29.
- `month of date "Sun, 02 Apr 2006" + 2*month`
  - Returns June.
- `year of current date`
  - Returns the current year, eg. 2006.

- `January of 2020 - current month_and_year`
  - ▶ Returns a time interval measured to the nearest month, such as 13 years, 9 months.
  
- `january of 2009 < current month_and_year`
  - ▶ Evaluates to TRUE when the current date is later than the specified date. This phrase could serve as an expiration flag.
  
- `current month_and_year + 18*month`
  - ▶ Gives a date 18 months ahead of the current date, eg. October 2007.

---

## Number Of Months

These Inspectors provide tools for dealing and calculating with number-of-month types, which are similar to integers, but with yearly roll-over.

### Creation Methods

Key Phrase	Form	Description
month	<i>PlainGlobal</i>	Returns the specified number of months. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
year	<i>PlainGlobal</i>	Returns the specified number of years as a <number of months> type. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

### Properties

Key Phrase	Form	Return Type	Description
<number of months> as string	<i>Cast</i>	<string>	Converts a number of months type into a string type. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

### Operators

Key phrase	Return Type	Description
- <number of months>	<number of months>	Creates the negative of the specified number of months. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<number of months> {cmp} <number of months>	<boolean>	Compare two numbers of months, where {cmp} is one of: <, <=, =. <small>Win:6.0</small>

Key phrase	Return Type	Description
<number of months> {op} <number of months>	<number of months>	Multiplies or divides a number of months by an integer. Here {op} is one of: *, /.  Win:6.0
<number of months> + <year>	<year>	Returns the year after adding the specified number of months.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<year> {op} <number of months>	<year>	Adds or subtracts the specified number of months to derive a new year. Here {op} is one of: -, +.  Win:6.0

## Examples

- 24\*month  
▶ Returns 2 years.
- year 1984 + 264\*month  
▶ Returns 2006.

---

## Year

These Inspectors provide tools for dealing and calculating with year types, which are of the form YYYY, as in 2008.

## Creation Methods

Key Phrase	Form	Description
<integer> as year	<i>Cast</i>	Casts an integer as a year type.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<string> as year	<i>Cast</i>	Converts a string into a year.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
current year	<i>PlainGlobal</i>	Returns the current year.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
year <integer>	<i>NumberedGlobal</i>	Creates a year object from the specified integer.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
year <string>	<i>NamedGlobal</i>	Creates a year object from the specified string.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

Key Phrase	Form	Description
year of <date>	<i>Plain</i>	The year derived from the given date. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
year of <month and year>	<i>Plain</i>	Returns the year portion of the specified date (in month year format). Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Properties

Key Phrase	Form	Return Type	Description
<year> as integer	<i>Cast</i>	<integer>	Casts a year as an integer. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<year> as string	<i>Cast</i>	<string>	Casts a year as a string. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
leap of <year>	<i>Plain</i>	<boolean>	Returns a flag indicating whether or not the specified year is a leap year. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
length of <year>	<i>Plain</i>	<time interval>	Returns the number of day in the specified year. Leap years have 366 days. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Operators

Key phrase	Return Type	Description
<day of year> & <year>	<date>	Concatenates a day of the year with a year to create a complete date type. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<month> & <year>	<month and year>	Returns a date (in month year format) from the concatenation of a month and a year. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<year> - <year>	<number of months>	Subtracts two years and produces a time interval marked in months and years. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
<year> & <day of year>	<date>	Concatenates a year with the day of the year, returning a full date. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

Key phrase	Return Type	Description
<code>&lt;year&gt; &amp; &lt;month&gt;</code>	<i>&lt;month and year&gt;</i>	Returns a date (in month year format) from the concatenation of a month and a year. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<code>&lt;year&gt; {cmp} &lt;year&gt;</code>	<i>&lt;boolean&gt;</i>	Compares two years, where {cmp} is one of: <, <=, =. <small>Win:6.0</small>

## Examples

- `length of year "2008"`
  - ▶ Returns 366.
- `year 2020 - year 2008`
  - ▶ Returns 12 years.
- `year 2080 > current year`
  - ▶ Returns TRUE until the year 2080.

# World Objects

## World

All objects created without context are known as 'properties of the world' in the relevance language. Below is a list of these global properties, sorted by key phrase.

### Properties

Key Phrase	Form	Return Type	Description
action	<i>PlainGlobal</i>	<action>	Returns the action currently being parsed. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
action <integer>	<i>NumberedGlobal</i>	<action>	Returns the action matching the <integer> id. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
action lock state	<i>PlainGlobal</i>	<action lock state>	Returns the client action lock state. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
active action	<i>PlainGlobal</i>	<action>	Returns the action currently executing. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
active device	<i>PlainGlobal</i>	<active device>	Returns a list of all active devices found using the Configuration Manager SetupDiGetClassDevs NT API. <small>Win:1.2</small>
active device file	<i>PlainGlobal</i>	<file>	Under Windows NT, returns a list of file objects corresponding the list returned from the Windows NT EnumDeviceDrivers() function. <small>Win:1.2</small>
active device file <string>	<i>NamedGlobal</i>	<file>	Under Windows NT, returns a file object corresponding to the name provided. See file. <small>Win:1.2</small>
all firewall scope	<i>PlainGlobal</i>	<firewall scope>	Returns the scope of computers that allow ALL traffic through the firewall, corresponding to the Microsoft enumerated type NET_FW_SCOPE_ALL. <small>Win:5.1</small>
ansi code page	<i>PlainGlobal</i>	<integer>	Returns an integer value of the Windows API GetACP. <small>Win:4.1</small>

Key Phrase	Form	Return Type	Description
any ip version	<i>PlainGlobal</i>	<ip version>	Returns a type corresponding to the Microsoft enumerated value NET_FW_IP_VERSION_ANY.  Win:5.1
apparent registration server time	<i>PlainGlobal</i>	<time>	Shorthand for 'now of registration server'. When the client registers with the server, the server passes its current time back to the client. The client starts a stop watch at that time. The apparent registration server time is the time the server passed back to the client, plus the elapsed time on the stop watch.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
application <string>	<i>NamedGlobal</i>	<application>	Returns an application for the name provided.  Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:4.1
application event log	<i>PlainGlobal</i>	<event log>	Returns the object corresponding to the application event log, which records certain application events, such as the failure of MS SQL to access a database.  Win:6.0
application usage summary	<i>PlainGlobal</i>	<application usage summary>	Returns an application usage summary containing information including the start time, duration and other statistics on client applications.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
application usage summary <string>	<i>NamedGlobal</i>	<application usage summary>	Returns the usage summary for the application specified in <string>.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
april	<i>PlainGlobal</i>	<month>	Returns april as an object of type month.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
april <integer>	<i>NumberedGlobal</i>	<day of year>	Returns the nth day of april as a 'day of year' type (month day).  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
audit failure event log event type	<i>PlainGlobal</i>	<event log event type>	Returns an object corresponding to an audit failure - an event related to the failed execution of an action.  Win:6.0
audit success event log event type	<i>PlainGlobal</i>	<event log event type>	Returns an object corresponding to an audit success in an event log.  Win:6.0

Key Phrase	Form	Return Type	Description
august	<i>PlainGlobal</i>	<month>	Returns August as an object of type month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
august <integer>	<i>NumberedGlobal</i>	<day of year>	Returns the nth day of August as a 'day of year' type (month day). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
backoffice bit <operating system suite mask>	<i>IndexedGlobal</i>	<boolean>	Returns TRUE if the backoffice bit of the Windows operating system suite mask is set. <small>Win:6.0</small>
binary operator <string>	<i>NamedGlobal</i>	<binary operator>	Typically used in the plural, returns the various possible binary inspectors that use the specified operators. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
binary operator returning <type>	<i>IndexedGlobal</i>	<binary operator>	Returns a list of binary operators that return the specified type. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
bit <integer>	<i>NumberedGlobal</i>	<bit set>	Returns TRUE or FALSE, corresponding to value of the bit specified by <integer>. <small>Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1</small>
bit set <string>	<i>NamedGlobal</i>	<bit set>	Returns the bits of the binary number given by the string. <small>Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1</small>
blade bit <operating system suite mask>	<i>IndexedGlobal</i>	<boolean>	Returns TRUE if the blade bit of the Suite Mask (a part of the Windows OS version) is set. <small>Win:6.0</small>
boolean <string>	<i>NamedGlobal</i>	<boolean>	Returns a boolean. For example, boolean "TRUE". <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
case insensitive regex <string>	<i>NamedGlobal</i>	<regular expression>	Returns a case-insensitive regular expression from the supplied string. <small>Win:6.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
case insensitive regular expression <string>	<i>NamedGlobal</i>	<regular expression>	Same as case insensitive regex <string>. <small>Win:6.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

Key Phrase	Form	Return Type	Description
cast <string>	<i>NamedGlobal</i>	<cast>	Returns a list of the objects that can be cast into the type specified by <string>.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
cast returning <type>	<i>IndexedGlobal</i>	<cast>	Returns a list of the objects that can be cast into the specified type.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
character <integer>	<i>NumberedGlobal</i>	<string>	Returns a string containing a single ASCII character. For example, character 90 = "Z".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
client	<i>PlainGlobal</i>	<client>	Returns the client object corresponding to the BigFix application evaluating the current relevance expression.  Win:3.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
client license	<i>PlainGlobal</i>	<license>	Global object containing client licensing information.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
communications bit <operating system suite mask>	<i>IndexedGlobal</i>	<boolean>	Returns TRUE if the communications bit of the Suite Mask (a part of the Windows OS version) is set.  Win:6.0
computer id	<i>PlainGlobal</i>	<integer>	This is a unique integer assigned to the computer by the BES system.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
computer name	<i>PlainGlobal</i>	<string>	Returns a string corresponding to the name of the computer as it appears on the network.  Win:3.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
connection status <integer>	<i>NumberedGlobal</i>	<connection status>	Returns the connection status based on its integer value. This Inspector is included to take advantage of new (or undocumented) additions to the status values.  Win:5.0
connection status authenticating	<i>PlainGlobal</i>	<connection status>	Returns the value NCS_AUTHENTICATING: The connection is waiting for authentication to occur.  Win:5.0

Key Phrase	Form	Return Type	Description
connection status authentication failed	<i>PlainGlobal</i>	<connection status>	Returns the value NCS_AUTHENTICATION_FAILED: Authentication has failed on this connection.  Win:5.0
connection status authentication succeeded	<i>PlainGlobal</i>	<connection status>	Returns the value NCS_AUTHENTICATION_SUCCEEDED: Authentication has succeeded on this connection.  Win:5.0
connection status connected	<i>PlainGlobal</i>	<connection status>	Returns the value NCS_CONNECTED: The connection is in a connected state.  Win:5.0
connection status connecting	<i>PlainGlobal</i>	<connection status>	Returns the value NCS_CONNECTING: The connection is in the process of connecting.  Win:5.0
connection status disconnected	<i>PlainGlobal</i>	<connection status>	Returns the value NCS_DISCONNECTED: The connection is disconnected.  Win:5.0
connection status disconnecting	<i>PlainGlobal</i>	<connection status>	Returns the value NCS_DISCONNECTING: The connection is in the process of disconnecting.  Win:5.0
connection status hardware disabled	<i>PlainGlobal</i>	<connection status>	Returns the value NCS_HARDWARE_DISABLED: The hardware for the connection is present, but is not enabled.  Win:5.0
connection status hardware malfunction	<i>PlainGlobal</i>	<connection status>	Returns the value NCS_HARDWARE_MALFUNCTION: A malfunction has occurred in the hardware for the connection.  Win:5.0
connection status media disconnected	<i>PlainGlobal</i>	<connection status>	Returns the value NCS_MEDIA_DISCONNECTED: The media, for example the network cable, is disconnected.  Win:5.0

Key Phrase	Form	Return Type	Description
connection status no hardware present	<i>PlainGlobal</i>	<connection status>	Returns the value NCS_NO_HARDWARE_PRESENT: The hardware for the connection, for example network interface card (NIC), is not present.  Win:6.0
current date	<i>PlainGlobal</i>	<date>	Returns the current date in the format: • Day of week, Day Month Year.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
current day_of_month	<i>PlainGlobal</i>	<day of month>	Returns the current day of the month.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
current day_of_week	<i>PlainGlobal</i>	<day of week>	Retruns the current day of the week, eg. Monday, Tuesday, etc.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
current day_of_year	<i>PlainGlobal</i>	<day of year>	Retruns the current day of the year, in a Month Day format.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
current firewall profile type	<i>PlainGlobal</i>	<firewall profile type>	Retrieves the type of firewall profile that is currently in effect.  Win:5.1
current month	<i>PlainGlobal</i>	<month>	Returns the current month.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
current month_and_year	<i>PlainGlobal</i>	<month and year>	Returns the current date in month year format, eg. January 2012.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
current site	<i>PlainGlobal</i>	<site>	Returns the current site object. See site.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
current user	<i>PlainGlobal</i>	<current user>	Returns the current user if one is logged in to the desktop. • Note: For Unix, this returns a <user> type as of BES version 6.0.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
current year	<i>PlainGlobal</i>	<year>	Returns the current year.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

Key Phrase	Form	Return Type	Description
custom firewall scope	<i>PlainGlobal</i>	<firewall scope>	Returns the custom firewall scope, corresponding to the Microsoft enumerated type: NET_FW_SCOPE_CUSTOM.  Win:5.1
custom site subscription effective date <string>	<i>NamedGlobal</i>	<time>	Returns the date the custom site (specified by <string>) was last subscribed or unsubscribed. It is used internally by BES to manage custom site subscriptions.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
datacenter bit <operating system suite mask>	<i>IndexedGlobal</i>	<boolean>	Returns TRUE if the datacenter bit of the Suite Mask (a part of the Windows OS version) is set.  Win:6.0
date <string>	<i>NamedGlobal</i>	<date>	Converts the given string into a date. The string should be of the form 'Day Month Year' and the returned date will be of the form 'Day of week, Day Month Year'.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
day	<i>PlainGlobal</i>	<time interval>	Returns a time interval corresponding to 1 day.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
day_of_month <integer>	<i>NumberedGlobal</i>	<day of month>	Converts the given integer to a day of month type.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
day_of_month <string>	<i>NamedGlobal</i>	<day of month>	Converts the given string value (must be an integer from 1-31) to a day of month type.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
day_of_week <string>	<i>NamedGlobal</i>	<day of week>	Converts the given string value to a day of week type, eg. Monday, Tuesday, etc.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
december	<i>PlainGlobal</i>	<month>	Returns December as an object of type month.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
december <integer>	<i>NumberedGlobal</i>	<day of year>	Returns the nth day of December as a 'day of year' type (month day).  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

Key Phrase	Form	Return Type	Description
default web browser	<i>PlainGlobal</i>	<application>	Returns the application currently associated with HTML files. This is a Windows and Macintosh inspector; it will fail gracefully under other operating systems, rather than generate an error.  Win:3.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1
dmi	<i>PlainGlobal</i>	<dmi>	Creates the global dmi object. If no dmi information is available, creation of the object will fail.  Win:1.2, Lin:4.1
dns name	<i>PlainGlobal</i>	<string>	Returns the DNS name of the computer.  Win:3.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
domain firewall profile type	<i>PlainGlobal</i>	<firewall profile type>	Creates a domain firewall profile type for comparison.  Win:5.1
domain user	<i>PlainGlobal</i>	<local user>	Returns all of the users that are members of the domain for which the machine is a user.  Win:4.1
domain user <string>	<i>NamedGlobal</i>	<local user>	Returns the local user object corresponding to the specified name.  Win:4.1
drive	<i>PlainGlobal</i>	<drive>	Returns all the drive objects defined on the machine. • Note: For Unix, this Inspector returns a <filesystem> object as of version 6.0 of BES.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1
drive <string>	<i>NamedGlobal</i>	<drive>	Returns a drive object for the name provided. • Note: For Unix, this Inspector returns a <filesystem> object as of version 6.0 of BES.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1
embedded nt bit <operating system suite mask>	<i>IndexedGlobal</i>	<boolean>	Returns TRUE if the embedded nt bit of the Suite Mask (a part of the Windows OS version) is set.  Win:6.0
embedded restricted bit <operating system suite mask>	<i>IndexedGlobal</i>	<boolean>	Returns TRUE if the embedded restricted bit of the Suite Mask (a part of the Windows OS version) is set.  Win:6.0

Key Phrase	Form	Return Type	Description
enterprise bit <operating system suite mask>	<i>IndexedGlobal</i>	<boolean>	Returns TRUE if the enterprise bit of the Suite Mask (a part of the Windows OS version) is set.  Win:6.0
environment	<i>PlainGlobal</i>	<environment>	Returns an object corresponding to the currently defined set of environment variables.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1
error <string>	<i>NamedGlobal</i>	<undefined>	Always fails; if an error message is generated, it is based on the given string.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
error event log event type	<i>PlainGlobal</i>	<event log event type>	Returns an object corresponding to an error in the event log, such as the failure of a service to start.  Win:6.0
event log <string>	<i>NamedGlobal</i>	<event log>	Returns the named event log, which contains historical information that help to track down system and security problems. There are several distinct logs that you can specify, including: <ul style="list-style-type: none"> <li>• Application log: records application events</li> <li>• Security log: records global or local policy audit events</li> <li>• System log: records OS events.</li> </ul> Win:6.0
event log event type <integer>	<i>NumberedGlobal</i>	<event log event type>	Returns an event type object corresponding to the specified number. The enumerated types include: <ul style="list-style-type: none"> <li>• 1: error event</li> <li>• 2: warning event</li> <li>• 4: information event</li> <li>• 8: audit success event</li> <li>• 16: audit failure event.</li> </ul> Win:6.0
false	<i>PlainGlobal</i>	<boolean>	Returns the boolean FALSE.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
february	<i>PlainGlobal</i>	<month>	Returns February as an object of type month.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
february <integer>	<i>NumberedGlobal</i>	<day of year>	Returns the nth day of February as a 'day of year' type (month day).  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

Key Phrase	Form	Return Type	Description
file <string>	<i>NamedGlobal</i>	<file>	Returns a filesystem object corresponding to the full pathname provided in <string>.  Win:1.2, Lin:3.1, Sol:3.1, HP-UX:4.0, AIX:4.1, Mac:5.1
file_and_print firewall service type	<i>PlainGlobal</i>	<firewall service type>	Returns the global service type for file and print sharing, corresponding to the Microsoft enumerated type: NET_FW_SERVICE_FILE_AND_PRINT.  Win:6.0
firewall	<i>PlainGlobal</i>	<firewall>	Returns the global firewall object for this computer.  Win:5.1
firewall profile type <integer>	<i>NumberedGlobal</i>	<firewall profile type>	Returns the firewall profile type corresponding to the given integer: <ul style="list-style-type: none"> <li>• 0: Domain</li> <li>• 1: Standard</li> <li>• 2: Current.</li> </ul> Win:5.1
firewall scope <integer>	<i>NumberedGlobal</i>	<firewall scope>	Returns the scope of addresses from which a port can listen.  Win:5.1
firewall service type <integer>	<i>NumberedGlobal</i>	<firewall service type>	Returns the firewall service type specified by <integer>.  Win:5.1
floating point <string>	<i>NamedGlobal</i>	<floating point>	Creates a floating point number from the provided string.  Win:4.1, Mac:4.1
folder <string>	<i>NamedGlobal</i>	<folder>	Returns a folder object for the name provided. See drive.  Win:1.2, Lin:3.1, Sol:3.1, HP-UX:4.0, AIX:4.1, Mac:5.1
friday	<i>PlainGlobal</i>	<day of week>	Returns Friday as a day of week object.  Win:6.0, Lin:6.0, Sol:6.0, HP-UX:6.0, AIX:6.0
full wmi <string>	<i>NamedGlobal</i>	<wmi>	Returns a wmi object which can retrieve all values, including system values.  Win:3.0
ghz	<i>PlainGlobal</i>	<hertz>	Returns a Hertz object corresponding to 1 gigahertz. See hertz.  Win:1.2, Lin:3.1, Sol:3.1, HP-UX:4.0, AIX:4.1, Mac:4.1

Key Phrase	Form	Return Type	Description
greatest hz	<i>PlainGlobal</i>	<hertz>	Returns the largest hertz object that can be represented on this machine. See hertz.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
greatest integer	<i>PlainGlobal</i>	<integer>	Returns the largest integer that can be represented on this machine. See integer.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
greatest time interval	<i>PlainGlobal</i>	<time interval>	Returns the greatest time interval representable. The value corresponds to 106751991 days, 04:00:54.775807.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
hexadecimal integer <string>	<i>NamedGlobal</i>	<integer>	Creates an integer from the provided hexadecimal value.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
hexadecimal string <string>	<i>NamedGlobal</i>	<string>	Creates a string from the given hexadecimal value.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
hostname	<i>PlainGlobal</i>	<string>	Returns the standard host name, usually for the computer's network.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
hour	<i>PlainGlobal</i>	<time interval>	Returns a time interval corresponding to 1 hour.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
hyperthreading capable	<i>PlainGlobal</i>	<boolean>	Returns a boolean TRUE if the agent is able to detect that the processor is capable of running with hyperthreading enabled.  Win:6.0
hyperthreading enabled	<i>PlainGlobal</i>	<boolean>	Returns TRUE if the machine is running with hyperthreading enabled, a method in which each physical processor on the machine presents itself as multiple logical processors to the operating system.  Win:5.0
hz	<i>PlainGlobal</i>	<hertz>	Returns a hertz object corresponding to 1 hertz.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
information event log event type	<i>PlainGlobal</i>	<event log event type>	Returns an object corresponding to an information event -- An informational event which is generally related to a successful action.  Win:6.0

Key Phrase	Form	Return Type	Description
install folder <integer>	<i>NumberedGlobal</i>	<folder>	Returns a folder object corresponding to the number provided. The placement of some system folders can be found using numbers that have been associated with those folders. See folder.  Win:1.2
integer <integer>	<i>NumberedGlobal</i>	<integer>	Returns an integer. The keyword is optional.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
integer <string>	<i>NamedGlobal</i>	<integer>	Returns integer for name provided.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
internet protocol <integer>	<i>NumberedGlobal</i>	<internet protocol>	Returns the firewall internet protocol specified by the given integer. These correspond to the Microsoft enumerated types: <ul style="list-style-type: none"> <li>• NET_FW_IP_PROTOCOL_TCP</li> <li>• NET_FW_IP_PROTOCOL_UDP.</li> </ul> Win:5.1
ip version <integer>	<i>NumberedGlobal</i>	<ip version>	Returns the the IP version for the specified integer.  Win:5.1
ipv4	<i>PlainGlobal</i>	<ip version>	Provides a comparison value for a firewall ip version inspector.  Win:5.1
ipv4 address <string>	<i>NamedGlobal</i>	<ipv4 address>	Returns an ip address for the string provided.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
ipv6	<i>PlainGlobal</i>	<ip version>	Provides a comparison value for a firewall ip version inspector.  Win:5.1
january	<i>PlainGlobal</i>	<month>	Returns January as an object of type month.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
january <integer>	<i>NumberedGlobal</i>	<day of year>	Returns the nth day of January as a 'day of year' type (month day).  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
july	<i>PlainGlobal</i>	<month>	Returns July as an object of type month.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
july <integer>	<i>NumberedGlobal</i>	<day of year>	Returns the nth day of July as a 'day of year' type (month day).  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

Key Phrase	Form	Return Type	Description
june	<i>PlainGlobal</i>	<month>	Returns June as an object of type month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
june <integer>	<i>NumberedGlobal</i>	<day of year>	Returns the nth day of June as a 'day of year' type (month day). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
khz	<i>PlainGlobal</i>	<hertz>	Returns a hertz object corresponding to 1 kilohertz. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
least hz	<i>PlainGlobal</i>	<hertz>	Returns the least hertz value that can be represented on this machine. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
least integer	<i>PlainGlobal</i>	<integer>	Returns the least integer value that can be represented on this machine. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
least time interval	<i>PlainGlobal</i>	<time interval>	Returns the least time interval that can be represented on this machine. The value corresponds to -106751991 days, 04:00:54.775808. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
local administrator	<i>PlainGlobal</i>	<boolean>	Returns the boolean TRUE if the user belongs to the local administrator group. Also returns TRUE for Win9x and WinME. <small>Win:1.2</small>
local group	<i>PlainGlobal</i>	<local group>	Returns local groups defined on the local computer using the windows NetLocalGroupEnum API. Several local groups are defined simply by a default operating system install, and have names such as Administrators, Backup Operators, Guests, Network Configuration Operators, Power users, Users, etc. Some software applications also define local groups in order to help manage protections. <small>Win:6.0</small>
local group <string>	<i>NamedGlobal</i>	<local group>	Returns a local group corresponding to the given name, such as Administrator, Guests, etc. <small>Win:6.0</small>
local mssql database	<i>PlainGlobal</i>	<local mssql database>	Returns local MSSQL database objects. <small>Win:1.2</small>

Key Phrase	Form	Return Type	Description
local mssql database <string>	<i>NamedGlobal</i>	<local mssql database>	Returns the local Microsoft SQL (MSSQL) database object identified by the name provided.  Win:1.2
local subnet firewall scope	<i>PlainGlobal</i>	<firewall scope>	Returns the local subnet firewall scope, corresponding to the Microsoft enumerated type: NET_FW_SCOPE_LOCAL_SUBNET.  Win:5.1
local time <string>	<i>NamedGlobal</i>	<time>	Returns a time object for the name provided. See time.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
local time zone	<i>PlainGlobal</i>	<time zone>	Returns a time zone object corresponding to the local time zone.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
local user	<i>PlainGlobal</i>	<local user>	Returns all local users of the machine.  Win:1.2
local user <string>	<i>NamedGlobal</i>	<local user>	Returns the named local user.  Win:1.2
logical processor count	<i>PlainGlobal</i>	<integer>	Returns the number of logical processors available per physical processor. This can be interpreted as the number of hyperthreads that could be enabled on the machine. On a machine with 2 physical processors, each with 2 possible hyperthreads per processor, the 'physical processor count' and the 'logical processor count' would both return 2, while the 'number of processors' would return 4, since there are a total of 4 logical processors available for work. With hyperthreading turned off, the 'number of processors', 'logical processor count' and 'physical processor count' would all be 2. Disabling one of those processors will then give 'number of processors'=1, 'logical processor count'=2, and 'physical processor count'=1. If the number of processors / physical processor count != logical processor count, you can turn on hyperthreading.  Win:6.0
main gather service	<i>PlainGlobal</i>	<service>	Returns a service object for the main gathering service, typically located on the main server.  Win:3.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1

Key Phrase	Form	Return Type	Description
main processor	<i>PlainGlobal</i>	<processor>	Returns the processor object corresponding to the main processor.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
march	<i>PlainGlobal</i>	<month>	Returns March as an object of type month.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
march <integer>	<i>NumberedGlobal</i>	<day of year>	Returns the nth day of March as a 'day of year' type (month day).  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
may	<i>PlainGlobal</i>	<month>	Returns May as an object of type month.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
may <integer>	<i>NumberedGlobal</i>	<day of year>	Returns the nth day of May as a 'day of year' type (month day).  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
media type <integer>	<i>NumberedGlobal</i>	<media type>	Returns the media type based on its integer value. This Inspector is included to take advantage of new (or undocumented) additions to the media types.  Win:5.0
media type bridge	<i>PlainGlobal</i>	<media type>	Returns the value NCM_BRIDGE: Bridged connection.  Win:5.0
media type direct	<i>PlainGlobal</i>	<media type>	Returns the value NCM_DIRECT: Direct serial connection through a serial port.  Win:5.0
media type isdn	<i>PlainGlobal</i>	<media type>	Returns the value NCM_ISDN: Connection is through an integrated services digital network (ISDN) line.  Win:5.0
media type lan	<i>PlainGlobal</i>	<media type>	Returns the value NCM_LAN: Connection is to a local area network (LAN).  Win:5.0
media type phone	<i>PlainGlobal</i>	<media type>	Returns the value NCM_PHONE: Dial-up connection over a conventional phone line.  Win:5.0

Key Phrase	Form	Return Type	Description
media type pppoe	<i>PlainGlobal</i>	<media type>	Returns the value NCM_PPPOE: Point-to-Point protocol (PPP) over Ethernet.  Win:5.0
media type shared access host lan	<i>PlainGlobal</i>	<media type>	Returns the value NCM_SHAREDACCESSHOST_LAN: Shared connection to a LAN.  Win:5.0
media type shared access host ras	<i>PlainGlobal</i>	<media type>	Returns the value NCM_SHAREDACCESSHOST_RAS: Shared connection to a remote or wide area network (WAN).  Win:5.0
media type tunnel	<i>PlainGlobal</i>	<media type>	Returns the value NCM_TUNNEL: Virtual private network (VPN) connection.  Win:5.0
metabase	<i>PlainGlobal</i>	<metabase>	Returns the IIS metabase object.  Win:4.1
mhz	<i>PlainGlobal</i>	<hertz>	Returns a hertz object corresponding to 1 megahertz.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
microsecond	<i>PlainGlobal</i>	<time interval>	Returns a time interval corresponding to .000001 seconds.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
midnight	<i>PlainGlobal</i>	<time of day>	Returns 00:00:00 as a time of day object.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
millisecond	<i>PlainGlobal</i>	<time interval>	Returns a time interval corresponding to .001 seconds.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
minute	<i>PlainGlobal</i>	<time interval>	Returns a time interval corresponding to 1 minute.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
module <string>	<i>NamedGlobal</i>	<module>	For BigFix internal use only.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1
monday	<i>PlainGlobal</i>	<day of week>	Returns the day of week object for Monday.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

Key Phrase	Form	Return Type	Description
month	<i>PlainGlobal</i>	<number of months>	Returns the specified number of months. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
month <integer>	<i>NumberedGlobal</i>	<month>	Returns the month type corresponding to the given <integer>. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
month <string>	<i>NamedGlobal</i>	<month>	Returns a month type corresponding to the given <string>. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
native registry	<i>PlainGlobal</i>	<registry>	On 32 bit versions of windows, this returns the same as registry32 and registry. On 64 bit versions of windows, this returns the same as registry64. <small>Win:6.0</small>
network	<i>PlainGlobal</i>	<network>	Returns an object containing properties of the network. <small>Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1</small>
network share	<i>PlainGlobal</i>	<network share>	Creates a network shared object. <small>Win:4.1</small>
network share <string>	<i>NamedGlobal</i>	<network share>	Creates a named network shared object. <small>Win:4.1</small>
none firewall service type	<i>PlainGlobal</i>	<firewall service type>	Returns the no firewall service type, corresponding to the Microsoft enumerated type: NET_FW_SERVICE_NONE. <small>Win:6.0</small>
noon	<i>PlainGlobal</i>	<time of day>	Returns 12:00:00 as a time of day object. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
november	<i>PlainGlobal</i>	<month>	Returns November as an object of type month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
november <integer>	<i>NumberedGlobal</i>	<day of year>	Returns the nth day of November as a 'day of year' type (month day). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
now	<i>PlainGlobal</i>	<time>	Returns the current time as a time object. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

Key Phrase	Form	Return Type	Description
nt domain controller product type	<i>PlainGlobal</i>	<operating system product type>	Returns an object corresponding to OS product type of nt domain controller. <small>Win:6.0</small>
nt server product type	<i>PlainGlobal</i>	<operating system product type>	Returns an object corresponding to OS product type of nt server. <small>Win:6.0</small>
nt workstation product type	<i>PlainGlobal</i>	<operating system product type>	Returns an object corresponding to OS product type of nt workstation. <small>Win:6.0</small>
october	<i>PlainGlobal</i>	<month>	Returns October as an object of type month. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
october <integer>	<i>NumberedGlobal</i>	<day of year>	Returns the nth day of October as a 'day of year' type (month day). <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
oem code page	<i>PlainGlobal</i>	<integer>	Returns an integer value of the Windows API GetOEMCP. <small>Win:4.1</small>
operating system	<i>PlainGlobal</i>	<operating system>	Returns the operating system object. See operating system. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
operating system product type <integer>	<i>NumberedGlobal</i>	<operating system product type>	Returns an object corresponding to the numbered OS product type. <small>Win:6.0</small>
parameter <string>	<i>NamedGlobal</i>	<string>	This Inspector is a synonym for the parameter <string> of <action>. It looks up the value of the action parameter specified by <string>. This is used in conjunction with the parameter set command. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
pending login	<i>PlainGlobal</i>	<boolean>	Installers may leave values in the registry that the operating system will execute when the next user logs in. Pending login can detect these registry entries. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1</small>
pending restart	<i>PlainGlobal</i>	<boolean>	Returns TRUE if the operating system indicates that a restart needs to occur. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1</small>

Key Phrase	Form	Return Type	Description
pending restart <string>	<i>NamedGlobal</i>	<boolean>	Immediately after issuing a command like 'Action requires restart "PatchGroupX"', the expression 'Pending restart "PatchGroupX"' will be true until the next restart.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
personal bit <operating system suite mask>	<i>IndexedGlobal</i>	<boolean>	Returns TRUE if the personal bit of the Suite Mask (a part of the Windows OS version) is set.  Win:6.0
physical processor count	<i>PlainGlobal</i>	<integer>	Returns the number of physical processors on the machine. Note that 'number of processors' returns the number of logical processors. To determine the number of logical processors per physical processor, use 'number of processors / physical processor count'.  Win:5.0
processor	<i>PlainGlobal</i>	<processor>	Returns all the processor objects defined on the machine. See processor.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
processor <integer>	<i>NumberedGlobal</i>	<processor>	Returns a processor object for the numbered processor. Processors are numbered from 1.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1
property <string>	<i>NamedGlobal</i>	<property>	Typically used in the plural, returns the "line" Inspector properties.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
property returning <type>	<i>IndexedGlobal</i>	<property>	Produces a list of the Inspector properties that return the specified <type>.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
ram	<i>PlainGlobal</i>	<ram>	Returns a ram object for inspecting the properties of Random Access Memory installed on the machine. See ram.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
random access memory	<i>PlainGlobal</i>	<ram>	Same as above.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1
recent application	<i>PlainGlobal</i>	<application>	Returns all the application objects that have recently been executing on the machine. See application.  Win:1.2, Lin:6.0

Key Phrase	Form	Return Type	Description
recent application <string>	<i>NamedGlobal</i>	<application>	Returns an application for the name provided it has recently executed. The name is assumed to be the last part of an executable file name.  Win:1.2, Lin:6.0
regapp	<i>PlainGlobal</i>	<application>	Returns all the application objects defined under the 'App Paths' key of the registry. • Note: This Inspector returns a <filesystem> object on the Macintosh.  Win:1.2
regapp <string>	<i>NamedGlobal</i>	<application>	Returns an application object for the name provided. On a Macintosh, returns a file that has been registered with Launch Services. See application and regapp.  Win:1.2
regex <string>	<i>NamedGlobal</i>	<regular expression>	Creates a regex object from the given string.  Win:6.0, Lin:3.1, Sol:3.1, HP-UX:4.0, AIX:4.1, Mac:4.1
registry	<i>PlainGlobal</i>	<registry>	Returns a registry object.  Win:1.2
regular expression <string>	<i>NamedGlobal</i>	<regular expression>	Same as regex <string>.  Win:6.0, Lin:3.1, Sol:3.1, HP-UX:4.0, AIX:4.1, Mac:4.1
relay service	<i>PlainGlobal</i>	<service>	Returns a service object for the relay component of BES.  Win:3.0, Lin:3.1, Sol:3.1, HP-UX:4.0, AIX:4.1
remote desktop firewall service type	<i>PlainGlobal</i>	<firewall service type>	Returns the remote desktop firewall service type, corresponding to the Microsoft enumerated type: NET_FW_SERVICE_REMOTE_DESKTOP.  Win:5.1
rope <string>	<i>NamedGlobal</i>	<rope>	Creates a rope object from the given string.  Win:1.2, Lin:3.1, Sol:3.1, HP-UX:4.0, AIX:4.1, Mac:4.1
running application	<i>PlainGlobal</i>	<application>	Returns all the application objects that are currently executing on the machine. See application.  Win:1.2, Lin:6.0

Key Phrase	Form	Return Type	Description
running application <string>	<i>NamedGlobal</i>	<application>	Returns an application for the name provided it is currently executing. The name is assumed to be the last part of an executable file name.  Win:1.2, Lin:6.0
running service	<i>PlainGlobal</i>	<service>	Returns all the running service objects.  Win:1.2
running service <string>	<i>NamedGlobal</i>	<service>	Returns the running service object matching the name provided.  Win:1.2
saturday	<i>PlainGlobal</i>	<day of week>	Returns Saturday as a day of week object.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
second	<i>PlainGlobal</i>	<time interval>	Returns a time interval corresponding to 1 second.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
security event log	<i>PlainGlobal</i>	<event log>	Returns a security event log, which records global or local group policy events.  Win:6.0
selected server	<i>PlainGlobal</i>	<selected server>	The BES Server or BES Relay to which the agent reports. Returned as the "selected server" type.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
september	<i>PlainGlobal</i>	<month>	Returns September as an object of type month.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
september <integer>	<i>NumberedGlobal</i>	<day of year>	Returns the nth day of September as a 'day of year' type (month day).  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
service	<i>PlainGlobal</i>	<service>	Returns all the service objects.  Win:1.2
service <string>	<i>NamedGlobal</i>	<service>	Returns the service object matching the name provided regardless of its running state.  Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
single user ts bit <operating system suite mask>	<i>IndexedGlobal</i>	<boolean>	Returns TRUE if the single user ts bit of the Suite Mask (a part of the Windows OS version) is set.  Win:6.0

Key Phrase	Form	Return Type	Description
site	<i>PlainGlobal</i>	<site>	Returns all the site objects that are currently loaded into memory. See site.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
site <string>	<i>NamedGlobal</i>	<site>	Returns a site object for the name provided. The name is the URL of the site location. See site.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
small business bit <operating system suite mask>	<i>IndexedGlobal</i>	<boolean>	Returns TRUE if the small business bit of the Suite Mask (a part of the Windows OS version) is set.  Win:6.0
small business restricted bit <operating system suite mask>	<i>IndexedGlobal</i>	<boolean>	Returns TRUE if the small business restricted bit of the Suite Mask (a part of the Windows OS version) is set.  Win:6.0
standard firewall profile type	<i>PlainGlobal</i>	<firewall profile type>	Returns the Standard firewall profile type. This is a global property.  Win:5.1
string <string>	<i>NamedGlobal</i>	<string>	Returns a string for the name provided. The keyword string is optional. For example, string "hi" = "hi".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
sunday	<i>PlainGlobal</i>	<day of week>	Returns Sunday as a day of week object.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
system event log	<i>PlainGlobal</i>	<event log>	Returns a system event log, which records OS or component events, such as the failure of a bootstrap service.  Win:6.0
system file <string>	<i>NamedGlobal</i>	<file>	Returns a file object corresponding to the relative pathname provided.  Win:1.2
system ini device file	<i>PlainGlobal</i>	<file>	Returns a list of file objects corresponding to all the device files loaded as a result of a device= lines of the system.ini file. See file.  Win:1.2

Key Phrase	Form	Return Type	Description
system ini device file <string>	<i>NamedGlobal</i>	<file>	Returns a file object corresponding to a device file loaded as a result of a device= line of the system.ini file.  Win:1.2
system language	<i>PlainGlobal</i>	<string>	Returns the language of the system as a string. It is identified using the GetSystemDefaultLangID() system call. See the language keyword of the application object for a list of possible language value.  Win:1.2, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1
system locale	<i>PlainGlobal</i>	<language>	Determines which bitmap fonts, and OEM, ANSI, and MAC code pages are defaults for the system. This only affects applications that are not fully Unicode.  Win:4.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1
system ui language	<i>PlainGlobal</i>	<language>	Determines the default language of menus and dialogs, messages and help files.  Win:4.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1
system wow64 folder	<i>PlainGlobal</i>	<folder>	Returns a filesystem object corresponding to a "Windows On Windows 64" system folder, which does not exist on 32-bit Windows. You can find out more about the WOW64 system folder at the Microsoft site: <a href="http://msdn.microsoft.com/library/default.asp?url=/library/en-us/sysinfo/base/getsystemwow64directory.asp">http://msdn.microsoft.com/library/default.asp?url=/library/en-us/sysinfo/base/getsystemwow64directory.asp</a> .  Win:6.0
system x32 folder	<i>PlainGlobal</i>	<folder>	Returns a filesystem object corresponding to a 32-bit system folder. On a 32-bit machine, this is equivalent to the normal system folder.  Win:6.0

Key Phrase	Form	Return Type	Description
system x64 folder	<i>PlainGlobal</i>	<folder>	Returns a filesystem object corresponding to a 64-bit system folder. This is the same as the system folder, but with file system redirection disabled. For more information about file redirection, see the Microsoft site <a href="http://msdn.microsoft.com/library/default.asp?url=/library/en-us/win64/win64/file_system_redirector.asp">http://msdn.microsoft.com/library/default.asp?url=/library/en-us/win64/win64/file_system_redirector.asp</a> .  Win:6.0
tcp	<i>PlainGlobal</i>	<internet protocol>	Returns an internet protocol corresponding to the Microsoft enumerated type: NET_FW_IP_PROTOCOL_TCP.  Win:5.1
terminal bit <operating system suite mask>	<i>IndexedGlobal</i>	<boolean>	Returns TRUE if the terminal bit of the Suite Mask (a part of the Windows OS version) is set.  Win:6.0
thursday	<i>PlainGlobal</i>	<day of week>	Returns Thursday as a day of week object.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
time <string>	<i>NamedGlobal</i>	<time>	Returns a time object for the name provided. See time.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
time interval <string>	<i>NamedGlobal</i>	<time interval>	Creates a time interval from the string.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
time zone <string>	<i>NamedGlobal</i>	<time zone>	Returns a time zone object for the name provided.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
time_of_day <string>	<i>NamedGlobal</i>	<time of day>	Creates a time of day object out of the given string.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
total processor core count	<i>PlainGlobal</i>	<integer>	Returns an integer corresponding to the total number of processor cores.  Win:6.0
true	<i>PlainGlobal</i>	<boolean>	Returns the boolean TRUE.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
tuesday	<i>PlainGlobal</i>	<day of week>	Returns Tuesday as a day of week object.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

Key Phrase	Form	Return Type	Description
udp	<i>PlainGlobal</i>	<internet protocol>	Returns an internet protocol corresponding to the Microsoft enumerated type: NET_FW_IP_PROTOCOL_UDP.  Win:5.1
unary operator <string>	<i>NamedGlobal</i>	<unary operator>	Typically used in the plural, this inspector returns a list of objects that use the specified operator.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
unary operator returning <type>	<i>IndexedGlobal</i>	<unary operator>	Returns a list of the unary operator inspectors (such as negative) that return the specified type.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
universal time <string>	<i>NamedGlobal</i>	<time>	Returns a time object for the name provided. See time.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
universal time zone	<i>PlainGlobal</i>	<time zone>	Returns a time zone object corresponding to the universal time zone.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
upnp firewall service type	<i>PlainGlobal</i>	<firewall service type>	Returns the UPnP (Universal Plug and Play) firewall service type, corresponding to the Microsoft enumerated type: NET_FW_SERVICE_UPNP. • Note: UPnP is not the same as PnP. UPnP is used for network connectivity via TCP/IP to various devices (scanners, printers, etc.).  Win:5.1
user language	<i>PlainGlobal</i>	<string>	Returns the language of the system as a string. It is identified by using the GetUserDefaultLangId() system call. See the language keyword of the application object for a list of possible language value.  Win:1.2
user locale	<i>PlainGlobal</i>	<language>	Determines which settings are used for formatting dates, times, currency, and numbers as a default for each user. Also determines the sort order for sorting text.  Win:4.1

Key Phrase	Form	Return Type	Description
user ui language	<i>PlainGlobal</i>	<language>	Non-MUI: Same as system UI Language. • MUI: Determines the language of menus and dialogs, messages, and help files.  Win:4.1
version <string>	<i>NamedGlobal</i>	<version>	Short hand for 'file version'.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
warning event log event type	<i>PlainGlobal</i>	<event log event type>	Returns an object corresponding to a warning in the event log. Warnings can be used to prevent future system problems.  Win:6.0
wednesday	<i>PlainGlobal</i>	<day of week>	Returns Wednesday as a day of week object.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
week	<i>PlainGlobal</i>	<time interval>	Returns a time interval corresponding to 1 week.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
windows display time <string>	<i>NamedGlobal</i>	<time>	Returns a string that may match the time shown in the Windows file system.  Win:1.2
windows file <string>	<i>NamedGlobal</i>	<file>	Returns a file object corresponding to the relative pathname (within the Windows folder) provided. See file.  Win:1.2
windows folder	<i>PlainGlobal</i>	<folder>	Returns a folder object of the Windows folder This is operating system dependent. Under Win98 this is usually c:\Windows.  Win:1.2
wmi	<i>PlainGlobal</i>	<wmi>	Returns the wmi object corresponding to the "root\cimv2" namespace.  Win:3.0
wmi <string>	<i>NamedGlobal</i>	<wmi>	Returns the wmi object corresponding to the namespace string provided.  Win:3.0
x32 application <string>	<i>NamedGlobal</i>	<application>	Returns an object corresponding to the 32-bit application specified by <string>.  Win:6.0

Key Phrase	Form	Return Type	Description
x32 file <string>	<i>NamedGlobal</i>	<file>	Returns an object corresponding to a 32 bit file with name specified by <string>.  Win:6.0
x32 folder <string>	<i>NamedGlobal</i>	<folder>	Returns a filesystem object corresponding to a 32-bit folder.  Win:6.0
x32 registry	<i>PlainGlobal</i>	<registry>	Returns a 32-bit registry object. This Inspector is equivalent to the ordinary registry Inspector.  Win:6.0
x64 application <string>	<i>NamedGlobal</i>	<application>	Returns an object corresponding to the 64-bit application specified by <string>. On a 32-bit computer, this is equivalent to a normal application Inspector, but on a 64-bit machine, this Inspector returns an object that has filesystem redirection disabled.  Win:6.0
x64 file <string>	<i>NamedGlobal</i>	<file>	Returns an object corresponding to a 64 bit file with pathname specified by <string>. On a 32-bit computer, this is equivalent to a normal file Inspector, but on a 64-bit machine, this Inspector returns an object that has filesystem redirection disabled.  Win:6.0
x64 folder <string>	<i>NamedGlobal</i>	<folder>	Returns a filesystem object corresponding to a 64-bit folder with the given pathname. On a 32-bit computer, this is equivalent to a normal folder Inspector, but on a 64-bit machine, this Inspector returns an object that has filesystem redirection disabled. This action is transitive: any resulting filesystem objects will also have redirection disabled. For example, "pathnames of files of x64 folder <path>" will disable redirection when locating the folder, iterating over the files in the folder and calculating pathnames.  Win:6.0

Key Phrase	Form	Return Type	Description
x64 registry	<i>PlainGlobal</i>	<registry>	Returns a 64-bit registry object. This Inspector is for 64-bit computers only; there is no 64-bit registry on a 32-bit computer. • Note that "x64 registry" and "native registry" on 64-bit machines do NOT provide the same view as the 64-bit version of regedit (the "physical" view). If you try to access the physical location of the 32-bit view keys using a 64-bit view, it will be mapped back to the equivalent location in the 64-bit view. <small>Win:6.0</small>
year	<i>PlainGlobal</i>	<number of months>	Returns the specified number of years as a <number of months> type. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
year <integer>	<i>NumberedGlobal</i>	<year>	Creates a year object from the specified integer. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
year <string>	<i>NamedGlobal</i>	<year>	Creates a year object from the specified string. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
zoned time_of_day <string>	<i>NamedGlobal</i>	<time of day with time zone>	Returns a 'time of day with time zone' object from the specified string. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Examples

- `ansi code page = 1252`
- ▶ Returns TRUE on English Windows systems.
  
- `description of record (oldest record number of it) of application event log`
- ▶ Returns a description of the oldest record in the application event log.
  
- `binary operators "&"`
- ▶ Returns a list of all the concatenation inspectors available.
  
- `binary operators returning (type "integer")`
- ▶ Returns a list of binary operators that return an integer, including +, -, \*, /, %, and other combinations.
  
- `bit 0 of 5`
- ▶ Returns TRUE.

- `bit set "101" as integer`
  - ▶ Returns 5.
- `casts "integer"`
  - ▶ Returns a list of the objects that can be cast as integers, eg., `<string> as integer`, `<integer> as integer`, etc.
- `device names of connections whose (status of it = connection status connected) of network`
  - ▶ Returns the names of the connected network devices.
- `if FALSE then 1 else error "my error message"`
  - ▶ Returns the string: User-defined error: my error message.
- `exists event log "Application"`
  - ▶ Returns TRUE if the application log exists.
- `current firewall profile type = firewall profile type 3`
  - ▶ Returns TRUE.
- `hexadecimal integer "A0"`
  - ▶ Returns 160.
- `number of processors / physical processor count != logical processor count`
  - ▶ Returns TRUE if hyperthreading has not yet been turned on.
- `paths of network shares`
  - ▶ Returns a list of the paths currently being shared over the network.

## Registry Objects

These are the keywords for dealing with the Windows registry. Particular attention is paid to registered applications and their associated file extensions.

### Registry

These are the Inspectors that expose the Windows registry.

#### Creation Methods

Key Phrase	Form	Description
native registry	<i>PlainGlobal</i>	On 32 bit versions of windows, this returns the same as registry32 and registry. On 64 bit versions of windows, this returns the same as registry64.  Win:6.0
registry	<i>PlainGlobal</i>	Creates an object for accessing the registry.  Win:1.2
x32 registry	<i>PlainGlobal</i>	Returns a 32-bit registry object. This Inspector is equivalent to the ordinary registry Inspector.  Win:6.0
x64 registry	<i>PlainGlobal</i>	Returns a 64-bit registry object. This Inspector is for 64-bit computers only; there is no 64-bit registry on a 32-bit computer. <ul style="list-style-type: none"> <li>• Note that "x64 registry" and "native registry" on 64-bit machines do NOT provide the same view as the 64-bit version of regedit (the "physical" view). If you try to access the physical location of the 32-bit view keys using a 64-bit view, it will be mapped back to the equivalent location in the 64-bit view.</li> </ul> Win:6.0

## Properties

Key Phrase	Form	Return Type	Description
application <string> of <registry>	<i>Named</i>	<application>	Returns an application object matching name provided. See application.  Win:1.2
application folder <string> of <registry>	<i>Named</i>	<folder>	Returns the folder containing the matching name provided. See application. The application does not have to exist. The folder has to exist.  Win:1.2
application of <registry>	<i>Plain</i>	<application>	Iterates through the properly installed applications. See application.  Win:1.2
device key <string> of <registry>	<i>Named</i>	<registry key>	Iterates through all the keys known to the configuration manager for active devices whose "DeviceDesc" matches the name provided. See registry key.  Win:1.2
device key of <registry>	<i>Plain</i>	<registry key>	Iterates through all the keys known to the configuration manager for active devices. See registry key.  Win:1.2
file extension <string> of <registry>	<i>Named</i>	<registry key>	Returns a key associated with the named extension. See registry key.  Win:1.2
file type <string> of <registry>	<i>Named</i>	<registry key>	Returns a key associated the named file type. See registry key.  Win:1.2
key <string> of <registry>	<i>Named</i>	<registry key>	Returns a key associated with the name provided. See registry key.  Win:1.2

## Examples

- name of application of key ".txt" of key "HKEY\_CLASSES\_ROOT" of the registry = "NOTEPAD.EXE"
- True when text files are to be opened with notepad.exe on the current machine.

- value of file extension "bmp" of registry = "Paint.Picture"
- ▶ Returns TRUE if there is only one value of the key "HKEY\_CLASSES\_ROOT\.bmp" and it contains the string "Paint.Picture".
  
- file extension ".txt" of the registry
- ▶ Returns a key corresponding to the application designated to process files with this extension. The dot is optional in the name provided. Looks for the key under HKEY\_CLASSES\_ROOT.
  
- file type "txtfile" of the registry
- ▶ Returns a key whose existence indicates that there is an application designated to process text files. Looks for the key under HKEY\_CLASSES\_ROOT.
  
- key "HKEY\_CLASSES\_ROOT\txtfile" of the registry
- ▶ Returns a key whose existence indicates that there is an application designated to process text files. Looks for the key under HKEY\_CLASSES\_ROOT.
  
- key "txtfile" of key "HKEY\_CLASSES\_ROOT" of the registry
- ▶ Returns a key whose existence indicates that there is an application designated to process text files. Looks for the key under HKEY\_CLASSES\_ROOT.

---

## Registry Key

The registry key objects represent Windows registry keys whose existence and properties can be inspected. Keys can be identified by name. There are several Inspectors that return keys from parts of the registry that store file associations and active device drivers.

### Creation Methods

Key Phrase	Form	Description
device key <string> of <registry>	<i>Named</i>	Iterates through all the keys known to the configuration manager for active devices whose "DeviceDesc" matches the name provided.  Win:1.2
device key of <registry>	<i>Plain</i>	Iterates through all the keys known to the configuration manager for active devices.  Win:1.2
driver key of <active device>	<i>Plain</i>	The key identified by adding the value of 'driver key value name of active device' to HKLM\System\CurrentControlSet\Control\Class\.  Win:1.2

Key Phrase	Form	Description
driver key of <registry key>	<i>Plain</i>	Uses the value of "Driver" of the key to indirectly return a key corresponding to HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Class\<value of Driver>.  Win:1.2
file extension <string> of <registry>	<i>Named</i>	Creates a key object provided the registry indicates support for the named file extension.  Win:1.2
file type <string> of <registry>	<i>Named</i>	Creates a key object provided the registry indicates support for the named file type.  Win:1.2
key <string> of <registry key>	<i>Named</i>	Creates an object for the named sub-key of the key.  Win:1.2
key <string> of <registry>	<i>Named</i>	Creates an object for the named key. The name may be a full path to a key of the form "HKEY_CLASSES_ROOT\Fixlet.Pool\".  Win:1.2
key of <registry key>	<i>Plain</i>	Iterates through the sub-keys of a key.  Win:1.2

## Properties

Key Phrase	Form	Return Type	Description
application <string> of <registry key>	<i>Named</i>	<application>	Returns the application associated with the named command. Normally used with a sub-key of key HKEY_CLASSES_ROOT whose name is a file type.  Win:1.2
application folder <string> of <registry key>	<i>Named</i>	<folder>	Returns the parent folder associated with the named application. Normally used with a sub-key of key HKEY_CLASSES_ROOT whose name is a file type.  Win:1.2
application folder of <registry key>	<i>Plain</i>	<folder>	Returns the parent folder associated with the named application. Normally used with a sub-key of key HKEY_CLASSES_ROOT whose name is a file type.  Win:1.2
application of	<i>Plain</i>	<application>	Returns the application associated with the "open"

Key Phrase	Form	Return Type	Description
<registry key>			command. Normally used with a sub-key of key HKEY_CLASSES_ROOT whose name is a file extension.  Win: 1.2
default value of <registry key>	<i>Plain</i>	<registry key value>	Returns the unnamed value associated with a key as a string. It does not necessarily exist.  Win: 1.2
driver key of <registry key>	<i>Plain</i>	<registry key>	Normally used as a property of a device key. Looks up the value of "Driver" of the key provided to indirectly return another key corresponding to HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Class\<value of Driver>.  Win: 1.2
key <string> of <registry key>	<i>Named</i>	<registry key>	Returns a key for the named sub-key.  Win: 1.2
key of <registry key>	<i>Plain</i>	<registry key>	Iterates through the sub-keys of the key.  Win: 1.2
name of <registry key>	<i>Plain</i>	<string>	Returns the name of the key as a string.  Win: 1.2
security descriptor of <registry key>	<i>Plain</i>	<security descriptor>	Specifies the security descriptor associated with the specified registry key.  Win: 4.1
value <string> of <registry key>	<i>Named</i>	<registry key value>	Returns the named value stored under the key. See registry key value.  Win: 1.2
value of <registry key>	<i>Plain</i>	<registry key value>	Iterates through values stored under a key.  Win: 1.2

## Note

The terminology of keys, values, default values and values that have names and data is chosen to match the convention's used by the Windows registry editor as well as the API's provided by the Windows operating system for accessing this information.

Top branches of the Windows registry include:

HKEY\_CLASSES\_ROOT

HKEY\_LOCAL\_MACHINE

HKEY\_CURRENT\_USER

HKEY\_USERS

HKEY\_CURRENT\_CONFIG

HKEY\_DYN\_DATA

HKEY\_PERFORMANCE\_DATA (NT)

The trailing slashes on registry key names are optional.

## File extensions, File types, and associated applications:

The following table represents a small part of the registry. It illustrates the relationship between the notions of file extension, file type, and the shell commands associated with the inspector keywords

Description	HKEY_CLASSES_ROOT\	Default Value
File extension key	.txt	default value = txtfile
File type key	txtfile\shell\	
Named command	txtfile\shell\open\command	default value = c:\windows\NOTEPAD.EXE %1
Named command	txtfile\shell\print\command	default value = c:\windows\NOTEPAD.EXE /p %1

## Device Keys of the registry:

The Configuration Manager of the Windows 9x operating system maintains a list of active devices under the HKEY\_DYN\_DATA\Config Manager\Enum key of the registry. The items in the list contain values named "HardwareKey" which are the names of keys under

HKEY\_LOCAL\_MACHINE\Enum. The value "DeviceDesc" contains a description of the device.

The device key inspectors allow you to determine if a particular piece of hardware matching the Device Description is currently active.

Device key "Hardware ABC from Company XYZ" will only return a key if there is an entry under HKEY\_DYN\_DATA\Config Manager\Enum that points to it.

## Examples

■ file extension ".txt" of the registry

► Returns a key corresponding to the application that opens files with this extension. The dot is optional in the name provided.

- name of application of file extension "html" of the registry = "iexplore.exe"
- ▶ Verifies that the name of the application assigned to process html documents is Internet Explorer.
- file type "txtfile" of the registry
- ▶ Returns a key whose existence may indicate that there is an application designated to process files of this type. Looks for the key under HKEY\_CLASSES\_ROOT.
- key "HKEY\_CLASSES\_ROOT\txtfile" of the registry
- ▶ Returns a key whose existence indicates that there is an application designated to process text files.
- application "print" of key "HKEY\_CLASSES\_ROOT\.txt" of registry
- ▶ Returns the application designated to print the files with ".txt" extensions.
- application "bigfix.exe" of the registry as string
- ▶ Results in a string of the form "BigFix.exe" "1.0.32.0" "BigFix Client Application" "1.0.32.0" "BigFix Inc."
- name of application of key ".txt" of key "HKEY\_CLASSES\_ROOT" of the registry = "NOTEPAD.EXE"
- ▶ True when text files are to be opened with notepad.exe on the current machine.
- default value of key ".txt" of key "HKEY\_CLASSES\_ROOT" of the registry = "txtfile"
- ▶ True when the file extension is of type txtfile.
- exists default value of key "AppEvents\EventLabels\AppGPFault" of key "HKEY\_CURRENT\_USER" of registry
- ▶ Returns TRUE if the key exists and has a default value.
- names of keys of key "HKEY\_CLASSES\_ROOT\txtfile\shell" of the registry
- ▶ Iterates through all the sub-keys of the key provided. In this case, returning all the shell commands available to process the given file type.
- type of value "ProfileFlags" of key "HKEY\_CURRENT\_CONFIG" of registry = "REG\_BINARY"
- ▶ Returns TRUE when a value named ProfileFlags under the key "HKEY\_CURRENT\_CONFIG" exists and contains binary data.
- size of value whose (name of it = "ProfileFlags") of key "HKEY\_CURRENT\_CONFIG" of registry = 4
- ▶ Returns TRUE when a value named ProfileFlags exists as a child of the key "HKEY\_CURRENT\_CONFIG" and the size of it is 4.

- value "AutoRewind" of key "HKEY\_CURRENT\_USER\Software\Microsoft\ActiveMovie\Control\Media Player" of registry = 1
- Returns TRUE if the Media Player is set to AutoRewind.

## Registry Key Value

This Inspector is used to access values stored within a registry key. All values have sizes and types. All of the values of a registry key have names except one, and it is called the 'default value'. The type of the data stored in the value determines what casting operations are allowed. We have implemented several casting Inspectors that you can use to extract values from the registry.

### Creation Methods

Key Phrase	Form	Description
default value of <registry key>	<i>Plain</i>	Every key may have a default or unnamed value. This inspector returns the default value of the key. This value has the same properties as any other registry key value except that it does not have a name property.  Win:1.2
value <string> of <registry key>	<i>Named</i>	Creates an object with the value of the key. The name property of the value will match the name provided.  Win:1.2
value of <registry key>	<i>Plain</i>	Creates an object with all the values of a key.  Win:1.2

### Properties

Key Phrase	Form	Return Type	Description
<registry key value> as application	<i>Cast</i>	<application>	If the data stored in the value is a string and it is the full pathname of an application that exists on disk, the application object is returned.  Win:1.2
<registry key value> as file	<i>Cast</i>	<file>	If the data stored in the value is a string and it is the full pathname of a file that exists on disk, the file object is returned.  Win:1.2
<registry key value> as folder	<i>Cast</i>	<folder>	If the data stored in the value is a string and it is the full pathname of a folder that exists on disk, the folder object is returned.  Win:1.2

Key Phrase	Form	Return Type	Description
<registry key value> as integer	<i>Cast</i>	<integer>	Returns the value stored in the registry entry provided it can be fully represented as an integer.  Win: 1.2
<registry key value> as string	<i>Cast</i>	<string>	Returns a string if the data of the value is of type REG_SZ.  Win: 1.2
<registry key value> as system file	<i>Cast</i>	<file>	If the data stored in the value is a string and it is a relative pathname from the system folder of a file that exists on disk, the corresponding file object is returned.  Win: 1.2
<registry key value> as time	<i>Cast</i>	<time>	If the data stored in the value is a string in MIME compliant date format, this property will return a time object. If the data stored is a binary value and is 16 or more bytes in length, its first 16 bytes are interpreted as a SYSTEMTIME and the corresponding time object is returned. See time.  Win: 1.2
name of <registry key value>	<i>Plain</i>	<string>	Returns the name of the value as a string. (see escape of <string> for more information).  Win: 1.2
size of <registry key value>	<i>Plain</i>	<integer>	Returns the size of the data as an integer.  Win: 1.2
type of <registry key value>	<i>Plain</i>	<registry key value type>	Returns the type of the data of the value. See type of value of key or registry.  Win: 1.2

## Operators

Key phrase	Return Type	Description
<registry key value> {cmp} <integer>	<boolean>	Returns a boolean TRUE/FALSE depending on the result of the comparison, where: <ul style="list-style-type: none"> <li>{cmp} is one of: =, !=, &lt;, &lt;=, &gt;, &gt;= .</li> </ul> Win: 1.2

Key phrase	Return Type	Description
<registry key value> {cmp} <registry key value>	<boolean>	Returns a boolean TRUE/FALSE depending on the result of the comparison, where: <ul style="list-style-type: none"> <li>{cmp} is one of: =, !=, &lt;, &lt;=, &gt;, &gt;= .</li> </ul> Win:1.2
<registry key value> {cmp} <string>	<boolean>	Returns a boolean TRUE/FALSE depending on the result of the comparison, where: <ul style="list-style-type: none"> <li>{cmp} is one of: =, !=, &lt;, &lt;=, &gt;, &gt;= .</li> </ul> Win:1.2

## Note

Eleven literal types are currently recognized. Future types may be handled as numeric types. The possible numeric values of each type and their string literal values include:

0	REG_NONE
1	REG_SZ
2	REG_EXPAND_SZ
3	REG_BINARY
4	REG_DWORD
5	REG_DWORD_BIG_ENDIAN
6	REG_LINK
7	REG_MULTI_SZ
8	REG_RESOURCE_LIST
9	REG_FULL_RESOURCE_DESCRIPTOR
10	REG_RESOURCE_REQUIREMENTS_LIST

## Examples

■ default value of key ".txt" of key "HKEY\_CLASSES\_ROOT" of the registry = "txtfile"

► True when the file extension is of type txtfile.

■ exists default value of key "AppEvents\EventLabels\AppGPFault" of key "HKEY\_CURRENT\_USER" of registry

► Returns TRUE if the key exists and has a default value.

- type of value "ProfileFlags" of key "HKEY\_CURRENT\_CONFIG" of registry = "REG\_BINARY"
    - ▶ Returns TRUE when a value named ProfileFlags under the key "HKEY\_CURRENT\_CONFIG" exists and contains binary data.
  
  - value "AutoRewind" of key "HKEY\_CURRENT\_USER\Software\Microsoft\ActiveMovie\Control\Media Player" of registry = 1
    - ▶ Returns TRUE when the specified value of the key equals 1.
  
  - size of value whose (name of it = "ProfileFlags") of key "HKEY\_CURRENT\_CONFIG" of registry = 4
    - ▶ Returns TRUE when a value named ProfileFlags exists as a child of the key "HKEY\_CURRENT\_CONFIG" and the size of it is 4.
  
  - type of default value of key "AppEvents\EventLabels\AppGPFault" of key "HKEY\_CURRENT\_USER" of registry = "REG\_SZ"
    - ▶ Returns TRUE if the default type of the specified key is REG\_SZ.
- 

## Registry Key Value Type

The type identifier of the data associated with a registry key value.

### Creation Methods

Key Phrase	Form	Description
type of <registry key value>	<i>Plain</i>	Creates an integer designating the type of data stored in the registry key value. See the registry MS documentation for these numeric values, which correspond to the enumerated constants discussed in the "<registry key value type> as string" property.  Win:1.2

## Properties

Key Phrase	Form	Return Type	Description
<registry key value type> as string	<i>Cast</i>	<string>	Returns the type of value as a string. One of REG_SZ, REG_NONE, REG_DWORD, REG_LINK, REG_BINARY, REG_MULTI_SZ, REG_EXPAND_SZ, REG_RESOURCE_LIST, REG_DWORD_LITTLE_ENDIAN, REG_DWORD_BIG_ENDIAN, REG_FULL_RESOURCE_DESCRIPTOR, REG_RESOURCE_REQUIREMENTS_LIST.  Win:1.2

## Operators

Key phrase	Return Type	Description
<registry key value type> {cmp} <integer>	<boolean>	Returns a boolean TRUE/FALSE depending on the result of the comparison, where: • {cmp} is one of: =, !=, <, <=, >, >= .  Win:1.2
<registry key value type> {cmp} <registry key value type>	<boolean>	Returns a boolean TRUE/FALSE depending on the result of the comparison, where: • {cmp} is one of: =, !=, <, <=, >, >= .  Win:1.2
<registry key value type> {cmp} <string>	<boolean>	Returns a boolean TRUE/FALSE depending on the result of the comparison, where: • {cmp} is one of: =, !=, <, <=, >, >= .  Win:1.2

# Filesystem Objects

This chapter covers the keywords for extracting information from the file system, like files, drives, pathnames, folders, etc. It also includes the keywords needed to identify and compare version information of files and patches.

## Filesystem Object

### Properties

Key Phrase	Form	Return Type	Description
accessed time of <filesystem object>	Plain	<time>	When the filesystem object (file or folder) was last accessed. Some file systems maintain this property.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
ancestor of <filesystem object>	Plain	<folder>	Returns all ancestor folders (recursive parent folders) of the given filesystem object (file or folder).  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
archive of <filesystem object>	Plain	<boolean>	Returns TRUE if the Archive bit is turned on for the specified file or folder (filesystem object). This bit is often used by backup software.  Win:6.0
compressed of <filesystem object>	Plain	<boolean>	Returns TRUE if the file or folder (filesystem object) has been compressed.  Win:6.0
creation time of <filesystem object>	Plain	<time>	The date and time of creation of the specified file or folder. This corresponds to what is shown in the "Get Info" box.  Win:6.0, Mac:4.1
drive of <filesystem object>	Plain	<drive>	Returns the drive associated with the specified file or folder (filesystem object).  Win:6.0
hidden of <filesystem object>	Plain	<boolean>	Returns TRUE if the file or folder (filesystem object) is marked as hidden.  Win:6.0

Key Phrase	Form	Return Type	Description
location of <filesystem object>	<i>Plain</i>	<string>	Returns the name of the directory in which the file or folder (filesystem object) is located.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
modification time of <filesystem object>	<i>Plain</i>	<time>	The date and time of latest modification of the file. This corresponds to what is shown in the "Get Info" box.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:4.1
name of <filesystem object>	<i>Plain</i>	<string>	This returns the name of the file or folder.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:4.1
normal of <filesystem object>	<i>Plain</i>	<boolean>	Returns TRUE if the file or folder (filesystem object) is 'normal'.  Win:6.0
offline of <filesystem object>	<i>Plain</i>	<boolean>	Returns TRUE if the file or folder (the filesystem object) is marked as 'offline'.  Win:6.0
parent folder of <filesystem object>	<i>Plain</i>	<folder>	The folder containing the specified file or folder.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:4.1
pathname of <filesystem object>	<i>Plain</i>	<string>	Returns the full pathname of the specified file or folder (filesystem object) as a string.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:5.1
readonly of <filesystem object>	<i>Plain</i>	<boolean>	Returns TRUE if the file or folder (the filesystem object) is marked as read-only.  Win:6.0
system of <filesystem object>	<i>Plain</i>	<boolean>	Returns TRUE if the file or folder (the filesystem object) is marked as a system folder.  Win:6.0
temporary of <filesystem object>	<i>Plain</i>	<boolean>	Returns TRUE if the file or folder (the filesystem object) is marked as a temporary folder.  Win:6.0

## Examples

■ creation time of file "System" of System Folder > time "3 jan 1998 00:00+0000"

► Returns TRUE if the creation time of the system file is newer than the specified date.

- name of object "iChat.app" of applications folder
- ▶ Returns iChat.app.
- posix paths of items whose (name of it starts with "i") of applications folder
- ▶ Returns a list of the paths of applications starting with "i", such as /Applications/iCal.app, /Applications/iChat.app or /Applications/iTunes.app.

## File

For each file in the file system, you can create a corresponding file object and inspect its properties. Inspectors are also provided to look at version data of executable files. File objects are derived from filesystem objects.

### Creation Methods

See application objects for additional creation methods

Key Phrase	Form	Description
<registry key value> as file	<i>Cast</i>	If the value contains a string and the string points to an existing file, a file object is returned.  Win:1.2
<registry key value> as system file	<i>Cast</i>	If the value contains a string and the string points to an file, a file object is returned. Relative paths are interpreted relative to the system folder.  Win:1.2
active device file	<i>PlainGlobal</i>	Under Windows NT, returns a list of file objects corresponding the list returned from the Windows NT EnumDeviceDrivers() function.  Win:1.2
active device file <string>	<i>NamedGlobal</i>	Under Windows NT, returns a file object corresponding to the name provided. Names provided need only match the last component of the file. This inspector uses the Windows NT EnumDeviceDrivers() function.  Win:1.2
descendant of <folder>	<i>Plain</i>	Returns a list of all the descendant files of the specified folder.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1

Key Phrase	Form	Description
file <string>	<i>NamedGlobal</i>	Returns a filesystem object corresponding to the full pathname provided in <string>.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1
file <string> of <folder>	<i>Named</i>	Creates the file objects corresponding to the named file within the folder.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
file of <folder>	<i>Plain</i>	Iterates through the files of a folder.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
file of <service>	<i>Plain</i>	This returns the executable file associated with the given <service> under Windows 2000 operating systems.  Win:3.0
find file <string> of <folder>	<i>Named</i>	Creates an object corresponding to the files of the folder that that match the wildcard <string> provided.  Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
masthead of <site>	<i>Plain</i>	A copy of the masthead is maintained with the site data. This inspector returns a file object for the copy.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
system file <string>	<i>NamedGlobal</i>	Creates the file objects corresponding to the named file within the system folder.  Win:1.2
system ini device file	<i>PlainGlobal</i>	Creates a list of file objects corresponding to all the device= lines of the system.ini file. Note that items whose pathnames start with '*' are not placed into this list.  Win:1.2
system ini device file <string>	<i>NamedGlobal</i>	Creates a file object corresponding to a device file loaded as a result of a device= line of the system.ini file. The name provided should match the last component of the full path.  Win:1.2
windows file <string>	<i>NamedGlobal</i>	Returns a file object corresponding to the relative pathname (within the Windows folder) provided. See file.  Win:1.2

Key Phrase	Form	Description
x32 file <string>	<i>NamedGlobal</i>	Returns an object corresponding to a 32 bit file with name specified by <string>.  Win:6.0
x64 file <string>	<i>NamedGlobal</i>	Returns an object corresponding to a 64 bit file with pathname specified by <string>. On a 32-bit computer, this is equivalent to a normal file Inspector, but on a 64-bit machine, this Inspector returns an object that has filesystem redirection disabled.  Win:6.0

## Note

File systems that do not maintain the creation or last accessed times will often return the last modification time when queried for the creation or last accessed times or files. Modification times are preserved when files are copied. Thus, it is not uncommon to see a file that appears to have been modified before it was created.

## Properties

Key Phrase	Form	Return Type	Description
<file> as string	<i>Cast</i>	<string>	Creates a string containing the full pathname of the specified file. See <file>.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1
byte <integer> of <file>	<i>Numbered</i>	<integer>	Returns the numeric value of the byte located at the offset specified by number within the file. Byte 0 of the file is the first byte.  Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
content of <file>	<i>Plain</i>	<file content>	Returns an object that can be used to search for a string in the file. See content.  Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
executable file format of <file>	<i>Plain</i>	<string>	Returns a four-byte string containing the format specifier for the specified file.  Win:4.1
file version of <file>	<i>Plain</i>	<version>	Returns the file version extracted from the file's resource block. See version.  Win:1.2

Key Phrase	Form	Return Type	Description
first raw version block of <file>	<i>Plain</i>	<file version block>	Returns the first version block directly from a PE file. If the first block is sufficient for your purposes, use this version inspector for best speed.  Win:4.1
key <string> of <file>	<i>Named</i>	<string>	Returns a key and its value from the given structured text file. It iterates over lines that start with the key name (as specified by <string>) followed by an = or : character. When searching, white space is ignored.  Win:4.1, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1
line <integer> of <file>	<i>Numbered</i>	<string>	Returns the nth line (specified by <integer>) from the given file. • Note: On Unix computers this Inspector returns a <file line> object as of BES 6.0.  Win:4.1, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1
line <integer> of <file>	<i>Numbered</i>	<file line>	Returns the nth line (specified by <integer>) from the given file.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:5.1
line containing <string> of <file>	<i>Named</i>	<string>	Returns all lines from the given file that contain the specified string. • Note: On Unix computers this Inspector returns a <file line> object as of BES 6.0.  Win:4.1, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1
line containing <string> of <file>	<i>Named</i>	<file line>	Returns all lines from the given file that contain the specified string.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:5.1
line of <file>	<i>Plain</i>	<string>	Iterates over all the lines of the specified file. Lines are truncated to 1023 characters. • Note: On Unix computers this Inspector returns a <file line> object as of BES 6.0.  Win:4.1, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1
line of <file>	<i>Plain</i>	<file line>	Iterates over all the lines of the specified file. NOTE: lines are truncated to 1023 characters.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:5.1

Key Phrase	Form	Return Type	Description
line starting with <string> of <file>	<i>Named</i>	<string>	Same as line <string> of <file>. • Note: On Unix computers this Inspector returns a <file line> object as of BES 6.0.  Win:4.1, Lin:3.1, Sol:3.1, HP-UX:4.0, AIX:4.1
line starting with <string> of <file>	<i>Named</i>	<file line>	Same as line <string> of <file>.  Win:6.0, Lin:6.0, Sol:6.0, HP-UX:6.0, AIX:6.0, Mac:5.1
only raw version block of <file>	<i>Plain</i>	<file version block>	Returns the only version block directly from a PE file.  Win:4.1
only version block of <file>	<i>Plain</i>	<file version block>	Most files only have 1 version block. This property allows language independent access when there is only one version block present. The result is the same as 'version block 1'.  Win:1.2
product version of <file>	<i>Plain</i>	<version>	Returns the product version extracted from the file's resource block. See version.  Win:1.2
raw file version of <file>	<i>Plain</i>	<version>	Returns the file version directly from a PE file.  Win:4.1
raw product version of <file>	<i>Plain</i>	<version>	Returns the product version directly from a PE file.  Win:4.1
raw version block <integer> of <file>	<i>Numbered</i>	<file version block>	Returns the numbered version block directly from a PE file.  Win:4.1
raw version block <string> of <file>	<i>Named</i>	<file version block>	Returns the named version block directly from a PE file.  Win:4.1
raw version block of <file>	<i>Plain</i>	<file version block>	Returns the version block directly from a PE file.  Win:4.1
raw version of <file>	<i>Plain</i>	<version>	Returns the version directly from a PE file.  Win:4.1

Key Phrase	Form	Return Type	Description
section <string> of <file>	<i>Named</i>	<file section>	Returns a named section of a file. Useful for locating sections of 'ini' files. Section names are delimited by square bracket characters '[section name]'. See examples below.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1
security descriptor of <file>	<i>Plain</i>	<security descriptor>	Specifies the security descriptor associated with the specified file.  Win:4.1
sha1 of <file>	<i>Plain</i>	<string>	Returns the sha1 checksum of the file hex encoded as a 40 character long string.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
shortcut of <file>	<i>Plain</i>	<file shortcut>	Returns the properties and locates the target of a file shortcut: <ul style="list-style-type: none"> <li>• pathname (string)</li> <li>• start in pathname (string)</li> <li>• argument string (string)</li> <li>• icon pathname (string)</li> <li>• icon index (integer).</li> </ul> Win:1.2
size of <file>	<i>Plain</i>	<integer>	Returns the size in bytes of a file.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1
variable of <file>	<i>Plain</i>	<string>	Returns the names of variables contained in an INF style file, in the format [section].name=value.  Win:4.1, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
version block <integer> of <file>	<i>Numbered</i>	<file version block>	You can identify the particular version block you want to access by ordinal number.  Win:1.2
version block <string> of <file>	<i>Named</i>	<file version block>	You can identify the particular version block you are looking up by name. The name you provide should match the id string of the version block.  Win:1.2
version block of <file>	<i>Plain</i>	<file version block>	Iterates through the version blocks of a file.  Win:1.2

Key Phrase	Form	Return Type	Description
version of <file>	Plain	<version>	Synonym for file version of <file>.  Win:1.2, Mac:4.1
xml document of <file>	Plain	<xml dom document>	Returns an XML Document Object Model (DOM) for the specified file.  Win:5.1

### Note

Folder and file names may be case sensitive. Use “as uppercase” or “as lowercase” if you don’t know the actual case when making comparisons. Iterating through folders with many files can be time consuming. Consider using the “find file” Inspector which allows you to filter set of files by using the wildcard.

### Note

The format of the string returned when casting a file using 'as string' is:

"<fileName>" "<version>" "<fileDesc>" "<fileVersion>" "<companyName>"

Where:

<fileName>	The name of the file
<version>	The 'Product Version' of the file.
<fileDesc>	The value 'FileDescription' of version block 1 of the file.
<fileVersion>	The value 'FileVersion' of version block 1 of the file.
<companyName>	The value 'CompanyName' of version block 1 of the file.

### Examples

■ Number of find files "siteico\*.bmp" of client folder of current site = 3

► Returns TRUE if there are 3 files matching the wildcard pattern siteico\*.bmp.

■ modification time of masthead of current site < time "4 Aug 1997 01:00 pdt"

► TRUE if the masthead is older than the specified date.

■ exists windows file "command.com"

► Verifies the existence of the named file in the Windows folder.

■ byte 0 of file "C:\test.txt"

► Returns the first byte in the specified file.

- content of file "oeminfo.ini" of system folder as lowercase contains "dell"
  - ▶ Returns TRUE if the specified file contains the string "dell" anywhere in the file.
- executable file format of client
  - ▶ Returns a string like "PE%00%00".
- file version of application "iexplore.exe" of the registry < "4"
  - ▶ Test for older version of IE -- returns TRUE is version is less than 4.
- line 5 of file "/usr/lib/foobar"
  - ▶ Returns the fifth line of the specified file.
- lines starting with "foo" of file "/usr/lib/foobar"
  - ▶ Iterates over lines of the given file that start with "foo".
- product version of file "qna.exe" of parent folder of regapp "bigfix.exe" = product version of regapp "bigfix.exe"
  - ▶ Verifies the existence a co-executable located in the same folder with the proper version.
- product version of regapp "bigfix.exe" > version "1.0.21"
  - ▶ Returns TRUE if the application has a version of 1.0.22 or higher, and FALSE if the application has a version of 1.0.21 or less.
- pathname of shortcut of file "BigFix.lnk" of (value "Common Desktop" of key "HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders" of registry as folder)
  - ▶ Returns the pathname associated with the shortcut.
- version of file (pathname of shortcut of file "BigFix.lnk" of folder "c:\Documents and Settings\All Users\Desktop")
  - ▶ Returns the version number of the application to which the shortcut points.

---

## Application

Application objects derive from file objects. Therefore, application objects inherit all of the properties of the file object. This means that you can inspect properties such as 'modification time' or 'Product Version' of an application just as you would a file. See the properties of a file object for a complete list of these.

- The real power and primary purpose of the application object is their creation. The creation methods are optimized in anticipation of the importance of this object.

## Creation Methods

Key Phrase	Form	Description
<registry key value> as application	<i>Cast</i>	If the value is of type string, and the string is a full pathname to an executable that exists on disk, an application object is created.  Win:1.2
application <string>	<i>NamedGlobal</i>	Creates an application object for the name provided.  Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
application <string> of <folder>	<i>Named</i>	As with the file object, you can create an application object by naming it relative to its parent folder.  Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
application <string> of <registry key>	<i>Named</i>	Creates the application object associated with the named command. Normally used with a sub-key of key HKEY_CLASSES_ROOT whose name is a file type.  Win:1.2
application <string> of <registry>	<i>Named</i>	Creates the application object associated with the name provided. The name provided must be the name of a sub-key of the 'App Paths' registry key. See notes.  Win:1.2
application of <registry key>	<i>Plain</i>	Creates the application object associated with the "open" command. Normally used with a sub-key of key HKEY_CLASSES_ROOT whose name is a file extension.  Win:1.2
application of <registry>	<i>Plain</i>	Iterates through the 'App Paths' registry key creating objects for the applications that exist. See notes.  Win:1.2
default web browser	<i>PlainGlobal</i>	Returns the application object (typically the web browser) associated with HTML documents.  Win:3.0
recent application	<i>PlainGlobal</i>	Iterates through the list of recently executed applications, creating application objects. This includes the list of all currently running applications.  Win:1.2, Lin:6.0
recent application <string>	<i>NamedGlobal</i>	If named application has been executed recently, this

Key Phrase	Form	Description
		inspector creates an application object. Only specify the last component of the filename.  Win:1.2, Lin:6.0
regapp	<i>PlainGlobal</i>	Iterates through the applications of the registry. The applications will be those associated with the sub-keys of the 'App Paths' registry key. See notes.  Win:1.2
regapp <string>	<i>NamedGlobal</i>	Returns an application object for the name provided. On a Macintosh, returns a file that has been registered with Launch Services. See application and regapp.  Win:1.2
running application	<i>PlainGlobal</i>	Iterates through the list of running applications.  Win:1.2, Lin:6.0
running application <string>	<i>NamedGlobal</i>	If the named application is currently executing then this inspector creates an application object. Only specify the last component of the file name.  Win:1.2, Lin:6.0
x32 application <string>	<i>NamedGlobal</i>	Returns an object corresponding to the 32-bit application specified by <string>.  Win:6.0
x64 application <string>	<i>NamedGlobal</i>	Returns an object corresponding to the 64-bit application specified by <string>. On a 32-bit computer, this is equivalent to a normal application Inspector, but on a 64-bit machine, this Inspector returns an object that has filesystem redirection disabled.  Win:6.0

## Note

See 'File' for a list of the Application properties. Folder and file names may be case sensitive. Use "as uppercase" or "as lowercase" if you don't know the actual case when making comparisons. For Windows versions of these Inspectors, a properly installed application will register itself with the operating system. It does this by creating a registry sub-key usually named after the executable. The regapp Inspector uses the default value of this sub-key as a string that is the full pathname of the executable.

## Examples

- application "qna.exe" of parent folder of application "bigfix.exe" of the registry
  - ▶ Using the application of folder creation method, this example locates the 'sibling' application qna.exe provided it exists in the same folder in which the registered application 'bigfix.exe' is installed.
- exists application "notepad.exe" of the windows folder
  - ▶ Using the application of folder creation method, this example locates the notepad application provided it exists in the Windows folder.
- name of application "print" of key "HKEY\_CLASSES\_ROOT\.gif" of registry
  - ▶ Returns the name of the application currently responsible for printing gif files.
- name of application "print" of key "HKEY\_CLASSES\_ROOT\giffile" of the registry
  - ▶ The same as above when the default value of the key HKEY\_CLASSES\_ROOT\.gif contains giffile. These two examples demonstrate the method used by Windows to maintain file associations in the registry.
- application of key "HKEY\_CLASSES\_ROOT\mailto" of the registry
  - ▶ This example returns the application responsible for handling mailto requests in your web browser.
- names of regapps
  - ▶ Primarily used in QnA to obtain lists of applications installed under the "app path" key of the registry.
- byte 0 of regapp "bigfix.exe" = 77
  - ▶ TRUE if the first byte in the specified file is ASCII 77.
- regapp "IEXPLORE.EXE"
  - ▶ Returns the application object associated with the named registry key. Checks to see if the executable exists and if so, returns the application object. Case is ignored.
- exists running application whose (name of it as lowercase is "winword.exe")
  - ▶ Returns TRUE if Microsoft Word is currently executing.

## Folder

For every folder that exists in the file system, you can create a folder object. The properties of this object allow you to examine many aspects of the system. Folder objects are derived from filesystem objects. On the Macintosh, there are dozens of specialized folders; access to them depends on the domain. If the domain is not specified, it defaults to the system domain.

### Creation Methods

Key Phrase	Form	Description
<registry key value> as folder	<i>Cast</i>	If the value in the registry is a string, and the string points to an existing folder, a folder object is returned.  Win:1.2
ancestor of <filesystem object>	<i>Plain</i>	Returns all ancestor folders (recursive parent folders) of the given filesystem object (file or folder).  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
application folder <string> of <registry key>	<i>Named</i>	Synonym for pathname of parent folder of regapp <string>.  Win:1.2
application folder <string> of <registry>	<i>Named</i>	Creates a folder object for the name given. Name is used to search through AppPaths of the registry. Application doesn't have to exist. Folder must exist.  Win:1.2
application folder of <registry key>	<i>Plain</i>	Creates a folder object for the name given. If the registry key has a "shell\open\command\" subkey and the unnamed value points to an executable, this will return the parent folder of the executable if the application and folder exist.  Win:1.2
client folder of <site>	<i>Plain</i>	Creates an object corresponding to the folder on the client where site data is gathered.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
folder <string>	<i>NamedGlobal</i>	Creates a folder object for the named folder. This is a global property.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1
folder <string> of <drive>	<i>Named</i>	Creates a folder object for the name provided if it exists on the drive provided.  Win:1.2

Key Phrase	Form	Description
folder <string> of <folder>	<i>Named</i>	Creates a folder object for the named sub-folder. Trailing slashes should be omitted from the name.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
folder of <folder>	<i>Plain</i>	Iterates through the sub-folders of the folder object.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
install folder <integer>	<i>NumberedGlobal</i>	Creates a folder object corresponding to the number provided. The placement of some system folders can be found using numbers that have been associated with those folders. See notes.  Win:1.2
parent folder of <filesystem object>	<i>Plain</i>	The folder containing the specified file or folder.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:4.1
root folder of <drive>	<i>Plain</i>	Creates a folder object for the root of the given drive.  Win:1.2
system wow64 folder	<i>PlainGlobal</i>	Returns a filesystem object corresponding to a "Windows On Windows 64" system folder, which does not exist on 32-bit Windows. You can find out more about the WOW64 system folder at the Microsoft site: <a href="http://msdn.microsoft.com/library/default.asp?url=/library/en-us/sysinfo/base/getsystemwow64directory.asp">http://msdn.microsoft.com/library/default.asp?url=/library/en-us/sysinfo/base/getsystemwow64directory.asp</a> .  Win:6.0
system x32 folder	<i>PlainGlobal</i>	Returns a filesystem object corresponding to a 32-bit system folder. On a 32-bit machine, this is equivalent to the normal system folder.  Win:6.0
system x64 folder	<i>PlainGlobal</i>	Returns a filesystem object corresponding to a 64-bit system folder. This is the same as the system folder, but with file system redirection disabled. For more information about file redirection, see the Microsoft site <a href="http://msdn.microsoft.com/library/default.asp?url=/library/en-us/win64/win64/file_system_redirector.asp">http://msdn.microsoft.com/library/default.asp?url=/library/en-us/win64/win64/file_system_redirector.asp</a> .  Win:6.0
windows folder	<i>PlainGlobal</i>	Creates a folder object of the Windows folder. This is operating system dependent. Under Win98 this is usually c:\Windows.

Key Phrase	Form	Description
		Win:1.2
x32 folder <string>	<i>NamedGlobal</i>	Returns a filesystem object corresponding to a 32-bit folder with the specified pathname.  Win:6.0
x64 folder <string>	<i>NamedGlobal</i>	Returns a filesystem object corresponding to a 64-bit folder with the given pathname. On a 32-bit computer, this is equivalent to a normal folder Inspector, but on a 64-bit machine, this Inspector returns an object that has filesystem redirection disabled. This action is transitive: any resulting filesystem objects will also have redirection disabled. For example, "pathnames of files of x64 folder <path>" will disable redirection when locating the folder, iterating over the files in the folder and calculating pathnames.  Win:6.0

## Properties

Key Phrase	Form	Return Type	Description
application <string> of <folder>	<i>Named</i>	<application>	Returns an application object for the named file located in the folder. See application.  Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
descendant of <folder>	<i>Plain</i>	<file>	Returns a list of all the descendant files of the specified folder.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1
file <string> of <folder>	<i>Named</i>	<file>	Returns a file object for the named file located in the folder.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
file of <folder>	<i>Plain</i>	<file>	Iterates through the files of a folder returning file objects. When combined with a whose clause you can select files with specific properties. See file.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
find file <string> of <folder>	<i>Named</i>	<file>	Iterates through the files of a folder returning file objects whose name matches the search string provided in the name parameter. See example

Key Phrase	Form	Return Type	Description
			below. <small>Win:1.2, Lin:6.0, Sol:6.0, HP-UX:6.0, AIX:6.0</small>
folder <string> of <folder>	<i>Named</i>	<folder>	Returns a folder object for the named sub-folder. Trailing slashes should be omitted from the name. <small>Win:1.2, Lin:3.1, Sol:3.1, HP-UX:4.0, AIX:4.1, Mac:4.1</small>
folder of <folder>	<i>Plain</i>	<folder>	Iterates through the folders of a folder returning folder objects. When combined with a whose clause, you can select folders with specific properties. <small>Win:1.2, Lin:3.1, Sol:3.1, HP-UX:4.0, AIX:4.1, Mac:4.1</small>
security descriptor of <folder>	<i>Plain</i>	<security descriptor>	Specifies the security descriptor associated with the specified folder. <small>Win:4.1</small>

## Note

Folder and file names may be case sensitive. Use “as uppercase” or “as lowercase” if you don’t know the actual case when making comparisons. Be very careful not to iterate through folders that contain lots of files. Counting files in folders that contains lots of files can be slow. Always try to use the most efficient techniques to minimize the client overhead.

## Examples

- exists folder "c:\program files"
- ▶ Checks for the existence of the program files folder.
  
- exists folder "fonts" of the windows folder
- ▶ Returns TRUE if fonts is a subdirectory of the Windows directory.
  
- install folder 11
- ▶ Returns a folder object for system folder identified with this number.
  
- exists file whose (name of it contains ".pdf") of folder "name"
- ▶ Returns TRUE if some file in the folder has a name including the string ".pdf".

---

## Drive

The drive object is available to inspect these aspects of the file system.

## Creation Methods

Key Phrase	Form	Description
drive	<i>PlainGlobal</i>	Iterates through all valid drives on the system. <small>Win:1.2</small>
drive <string>	<i>NamedGlobal</i>	Creates the drive object for the name specified. <small>Win:1.2</small>
drive of <filesystem object>	<i>Plain</i>	Returns the drive associated with the specified file or folder (filesystem object). <small>Win:6.0</small>

## Properties

Key Phrase	Form	Return Type	Description
file system type of <drive>	<i>Plain</i>	<string>	Value as reported by GetVolumeInformation. <small>Win:1.2</small>
file_supports_encryption of <drive>	<i>Plain</i>	<boolean>	TRUE if bit is returned by GetVolumeInformation. <small>Win:1.2</small>
file_supports_object_ids of <drive>	<i>Plain</i>	<boolean>	TRUE if bit is returned by GetVolumeInformation. <small>Win:1.2</small>
file_supports_reparse_points of <drive>	<i>Plain</i>	<boolean>	TRUE if bit is returned by GetVolumeInformation. <small>Win:1.2</small>
file_supports_sparse_files of <drive>	<i>Plain</i>	<boolean>	TRUE if bit is returned by GetVolumeInformation. <small>Win:1.2</small>
file_volume_quotas of <drive>	<i>Plain</i>	<boolean>	TRUE if bit is returned by GetVolumeInformation. <small>Win:1.2</small>
folder <string> of <drive>	<i>Named</i>	<folder>	Returns a folder object corresponding to the name given provided that folder exists on the drive. <small>Win:1.2</small>

Key Phrase	Form	Return Type	Description
free space of <drive>	<i>Plain</i>	<integer>	Returns the number of unused bytes of storage for the drive. (Only available for fixed disks). <small>Win: 1.2</small>
fs_case_is_preserved of <drive>	<i>Plain</i>	<boolean>	TRUE if bit is returned by GetVolumeInformation. <small>Win: 1.2</small>
fs_case_sensitive of <drive>	<i>Plain</i>	<boolean>	TRUE if bit is returned by GetVolumeInformation. <small>Win: 1.2</small>
fs_file_compression of <drive>	<i>Plain</i>	<boolean>	TRUE if bit is returned by GetVolumeInformation. <small>Win: 1.2</small>
fs_persistent_acls of <drive>	<i>Plain</i>	<boolean>	TRUE if bit is returned by GetVolumeInformation. <small>Win: 1.2</small>
fs_unicode_stored_on_disk of <drive>	<i>Plain</i>	<boolean>	TRUE if bit is returned by GetVolumeInformation. <small>Win: 1.2</small>
fs_vol_is_compressed of <drive>	<i>Plain</i>	<boolean>	TRUE if bit is returned by GetVolumeInformation. <small>Win: 1.2</small>
name of <drive>	<i>Plain</i>	<string>	Returns the name of the drive. Names look like 'c:' and 'D:'. <small>Win: 1.2</small>
numeric type of <drive>	<i>Plain</i>	<integer>	Returns the type of drive as an integer. <small>Win: 1.2</small>
root folder of <drive>	<i>Plain</i>	<folder>	Returns the folder corresponding to the root of the drive. <small>Win: 1.2</small>
total space of <drive>	<i>Plain</i>	<integer>	Returns the size in bytes of the drive. (Only available for fixed disks). <small>Win: 1.2</small>
type of <drive>	<i>Plain</i>	<string>	Returns the type of drive as a string.

Key Phrase	Form	Return Type	Description
			Win:1.2

### Note

The drive object does not exist if the file is located on a file server. The expression drive of file "command.com" of folder "\\oak\c\windows" will fail even though the file exists. Drive objects do not exist for shared files and shared folders unless they have been mapped as a drive letter. The name of drives may be upper or lower case. The type of drive can be inspected. The values as string and integer are:

Type of drive	Numeric type
DRIVE_UNKNOWN	0
DRIVE_NO_ROOT_DIR	1
DRIVE_REMOVABLE	2
DRIVE_FIXED	3
DRIVE_REMOTE	4
DRIVE_CDROM	5
DRIVE_RAMDISK	6

### Examples

- free space of drive "c:" < 1000000  
▶ Returns TRUE if there is less than one million bytes of space left on drive C.
- name of drive of regapp "vshield.exe" as lowercase = "e:"  
▶ Returns TRUE if the application exists on drive E.
- numeric type of drive "e:" = 5  
▶ Returns TRUE if drive E is a CD-ROM. (See notes).
- total space of drive "c:" > 2000000000  
▶ Returns TRUE when the drive is capable of holding more than 2 billion bytes.
- type of drive of the system folder = "DRIVE\_FIXED"  
▶ Returns TRUE if the system folder is on a fixed disk drive.

## File Shortcut

Shortcuts to files can be constructed in the file system. The shortcut contains some additional properties that can be inspected.

### Creation Methods

Key Phrase	Form	Description
shortcut of <file>	<i>Plain</i>	Creates a shortcut object for the file. If the file is not a shortcut, this property does not exist.  Win:1.2

### Properties

Key Phrase	Form	Return Type	Description
argument string of <file shortcut>	<i>Plain</i>	<string>	Returns the arguments that are passed to the application to which the shortcut points when the user attempts to open the shortcut.  Win:1.2
icon index of <file shortcut>	<i>Plain</i>	<integer>	The index number of the icon in the file containing the icon associated with the shortcut.  Win:1.2
icon pathname of <file shortcut>	<i>Plain</i>	<string>	The full path name of the file containing the icon associated with the shortcut.  Win:1.2
pathname of <file shortcut>	<i>Plain</i>	<string>	Returns the full path name of the object to which the shortcut points.  Win:1.2
start in pathname of <file shortcut>	<i>Plain</i>	<string>	Returns the full path name the system sets the current directory when the user launches the shortcut.  Win:1.2

### Examples

■ pathname of shortcut of file "BigFix.lnk" of (value "Common Desktop" of key "HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders" of registry as folder)

► Returns the pathname associated with shortcut.

- pathname of parent folder of system folder = pathname of windows folder
- Checks that the system folder is located inside the Windows folder.

## File Section

Many programs and utilities store their settings in 'ini' files. This object is designed to access these settings. An 'ini' file is composed of zero or more named sections, each with zero or more keys. Each key is identified by name and has a string value.

### Creation Methods

Key Phrase	Form	Description
section <string> of <file>	<i>Named</i>	Creates a file section for the name given. A case-insensitive search is performed to locate the named section in the file. Searching through files for configuration data can be a slow process. Particularly for large ini files. In this case you may want to find another method that requires less computation.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1

### Properties

Key Phrase	Form	Return Type	Description
key <string> of <file section>	<i>Named</i>	<string>	Returns a string containing the value for the name provided. A case-insensitive search is performed through the section of the file.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1

### Note

Files with an extension of .ini are common in Windows systems, but rare in unix systems. However they are a handy cross-platform way of maintaining a collection of named variables.

### Examples

- exists Section "General" of file "oeminfo.ini" of system folder
- Returns TRUE if a section named "General" appears in the named "ini" file.
- key "Manufacturer" of section "General" of file "oeminfo.ini" of system folder
- Returns the name of the computer manufacturer, such as "Dell Computer Corporation".

## File Content

Content objects can be constructed from file objects to inspect their contents.

### Creation Methods

Key Phrase	Form	Description
<file content> as lowercase	<i>Cast</i>	Returns the contents of the file as lower case characters. <small>Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<file content> as uppercase	<i>Cast</i>	Returns the contents of the file as upper case characters. <small>Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
content of <file>	<i>Plain</i>	Creates a content object for a file. <small>Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

### Properties

Key Phrase	Form	Return Type	Description
<file content> as lowercase	<i>Cast</i>	<file content>	Returns a lowercase version of the content provided. <small>Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
<file content> as uppercase	<i>Cast</i>	<file content>	Returns an uppercase version of the content provided. <small>Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Operators

Key phrase	Return Type	Description
<file content> contains <string>	<boolean>	Returns TRUE if the string is located in the content provided.  Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Note

See "file section" for a more detailed inspection of .ini files.

## Examples

- content of file "oeminfo.ini" of system folder contains "Manufacturer=Dell"
- ▶ Returns TRUE if the exact character sequence "Dell" is located in the file.
- content of file "oeminfo.ini" of system folder as lowercase contains "emachines"
- ▶ Returns TRUE if either of the strings "emachines" or "eMachines" is found in the file.

---

## Version

This is the numeric method of indicating the file version, which is compact, convenient and fast. It makes use of a short string to define the version number.

## Creation Methods

Key Phrase	Form	Description
<string> as version	<i>Cast</i>	Turns a string into a version object.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1
<version> as version	<i>Cast</i>	Reflexive cast of version.  Win:1.2
file version of <file>	<i>Plain</i>	Creates a version object associated with the FILEVERSION property of the file.  Win:1.2
pad of <version>	<i>Plain</i>	Creates a version object which is padded with zero values.  Win:1.2

Key Phrase	Form	Description
product version of <file>	<i>Plain</i>	Creates a version object associated with the PRODUCTVERSION property of the file.  Win:1.2
raw file version of <file>	<i>Plain</i>	Same as file version, but allows a workaround for anomalous behavior on Windows systems with the Windows language pack installed (the MUI).  Win:4.1
raw product version of <file>	<i>Plain</i>	Same as product version, but allows a workaround for anomalous behavior on Windows systems with the Windows language pack installed (the MUI).  Win:4.1
raw version of <file>	<i>Plain</i>	Same as version, but allows a workaround for anomalous behavior on Windows systems with the Windows language pack installed (the MUI).  Win:4.1
version <string>	<i>NamedGlobal</i>	Creates a version object corresponding to the name provided. Syntax: version "1.2".  Win:1.2, Lin:3.1, Sol:3.1, HP-UX:4.0, AIX:4.1, Mac:4.1
version of <file>	<i>Plain</i>	Shorthand for file version of <file>.  Win:1.2, Mac:4.1

## Properties

Key Phrase	Form	Return Type	Description
<version> as string	<i>Cast</i>	<string>	Turns a version type into a string of the form "1.2.3.4".  Win:1.2, Lin:3.1, Sol:3.1, HP-UX:4.0, AIX:4.1, Mac:4.1
<version> as version	<i>Cast</i>	<version>	Reflexive cast of version.  Win:1.2
pad of <version>	<i>Plain</i>	<version>	Returns a version object which is padded with zero values.  Win:1.2

## Operators

Key phrase	Return Type	Description
<version> {cmp} <string>	<boolean>	Returns a boolean TRUE or FALSE, depending on the comparison operator, where: <ul style="list-style-type: none"> <li>{cmp} is one of: =, !=, &lt;, &lt;=, &gt;, &gt;= .</li> </ul> Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
<version> {cmp} <version>	<boolean>	Returns a boolean TRUE or FALSE, depending on the comparison operator, where: <ul style="list-style-type: none"> <li>{cmp} is one of: =, !=, &lt;, &lt;=, &gt;, &gt;= .</li> </ul> Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

## Note

Using the numeric version data is better than identifying an application based on version block strings. If you know the numeric version information and that the developer has identified each release of his application uniquely, then this is the way to proceed. It requires far less overhead than the other method. Furthermore, if you know that the numeric version data is monotonically increasing then you can compare their values using the special comparison operators.

## Examples

- `"MyApp 1.2" as version = version "1.2"`
- ▶ The version cast looks through the string until it identifies something that can be interpreted as a version. This is convenient for extracting version numbers out of strings that contain both version numbers and textual description.
- `file version of file "Winsock.dll" of windows folder = "4.0.0.1111"`
- ▶ Returns TRUE if the dll has the specified version number.
- `pad of version "1.2" = version "1.2.0.0"`
- ▶ Returns TRUE.
- `product version of regapp "bigfix.exe" > version "1.0.21"`
- ▶ TRUE if the application has a version of 1.0.22 or higher. FALSE if the application has a version of 1.0.21 or less.
- `product version of file "qna.exe" of parent folder of regapp "bigfix.exe" = product version of regapp "bigfix.exe"`
- ▶ Verifies the existence a co-executable located in the same folder with the proper version.

- `product version of regapp "bigfix.exe" > version "1.0.21"`
- ▶ Returns TRUE if the application has a version of 1.0.22 or higher, and FALSE if the application has a version of 1.0.21 or less.
  
- `version of file "Winsock.dll" of windows folder = "4.0.0.1111"`
- ▶ The plain version inspector is a shorthand for file version.
  
- `version of regapp "bigfix.exe" as string = "1.0.45.0"`
- ▶ Returns TRUE if the BigFix application has the specified version.

## File Version Block

You can inspect the version blocks of a file. There may be several language-specific version blocks. Version blocks contain version and name information in a human readable form for the specified language. This is the information that Windows displays in the file properties dialog. This technique uses string values and has a limited array of comparators. For better speed, utility and compactness see the version object.

### Creation Methods

Key Phrase	Form	Description
first raw version block of <file>	<i>Plain</i>	Returns the first version block directly from a PE file. If the first block is sufficient for your purposes, use this version inspector for best speed.  Win:4.1
only raw version block of <file>	<i>Plain</i>	Returns the only version block directly from a PE file.  Win:4.1
only version block of <file>	<i>Plain</i>	Most applications only have 1 version block. This inspector allows language independent access when there is only one version block present.  Win:1.2
raw version block <integer> of <file>	<i>Numbered</i>	Returns the numbered version block directly from a PE file.  Win:4.1
raw version block <string> of <file>	<i>Named</i>	Returns the named version block directly from a PE file.  Win:4.1
raw version block of <file>	<i>Plain</i>	Returns the version block directly from a PE file.  Win:4.1

Key Phrase	Form	Description
version block <integer> of <file>	<i>Numbered</i>	You can identify the particular version block you are looking up by ordinal number. 'Version block 1' is equivalent to 'Only Version block'.  Win:1.2
version block <string> of <file>	<i>Named</i>	You can identify the particular version block you are looking up by name. The name you provide should match the id string of the version block.  Win:1.2
version block of <file>	<i>Plain</i>	Iterates through the version blocks of a file.  Win:1.2

## Properties

Key Phrase	Form	Return Type	Description
codepage of <file version block>	<i>Plain</i>	<string>	A string representation of the codepage portion of the id of this version block. See notes for known codepage strings. For example, 'Unicode'.  Win:1.2
id of <file version block>	<i>Plain</i>	<string>	A string representation containing both the language and codepage of this version block. The format is 8 hex digits, 4 of the codepage concatenated with 4 of the language. For example, '040904b0'. See notes for known values.  Win:1.2
language of <file version block>	<i>Plain</i>	<string>	A string representation of the language portion of the id of this version block. For example, 'English (United States)'. See notes for known values.  Win:1.2
value <string> of <file version block>	<i>Named</i>	<string>	Returns a string corresponding to the name provided. Values have names such as 'CompanyName', 'FileDescription', 'FileVersion'.  Win:1.2

## Note

The value, ID, language and codepage properties of the file version block have the following typical values:

### value:

Each application can define its own set of values. Standard values include:

CompanyName  
FileDescription \*  
FileVersion \*  
InternalName  
LegalCopyright \*  
OriginalFilename  
ProductName  
ProductVersion  
Comments  
LegalTrademarks  
PrivateBuild  
SpecialBuild

\* As displayed on the version property sheet of the properties of a file.

### id:

The version block id is an eight character string. The left 4 characters of the string identify the language while the right 4 characters of the string identify the codepage of a version block. When looking up a version block by its name, you specify the id as a string. The id's of version blocks are case insensitive.

### language:

The language inspector returns the full language name. Language names are found using the left 4 hex characters of the id. Thus if the id of the version block is '040904b0', then the language returned would be 'English (United States)'.

Here are some sample language identifiers (left 4 hex chars):

0000	Language Neutral	0800	Language Neutral	1801	Arabic (Morocco)
0400	Process Default Language	0801	Arabic (Iraq)	1809	English (Ireland)
0401	Arabic (Saudi Arabia)	0804	Chinese (PRC)	180A	Spanish (Panama)
0402	Bulgarian	0807	German (Swiss)	1C01	Arabic (Tunisia)
0403	Catalan	0809	English (British)	1C09	English (South Africa)
0404	Chinese (Taiwan)	080A	Spanish (Mexican)	1C0A	Spanish (Dominican Republic)
0405	Czech	080C	French (Belgian)	2001	Arabic (Oman)
0406	Danish	0810	Italian (Swiss)	2009	English (Jamaica)
0407	German (Standard)	0812	Korean (Johab)	200A	Spanish (Venezuela)
0408	Greek	0813	Dutch (Belgian)	2401	Arabic (Yemen)
0409	English (United States)	0814	Norwegian (Nynorsk)	2409	English (Caribbean)
040A	Spanish (traditional Sort)	0816	Portuguese (Standard)	240A	Spanish (Colombia)
040B	Finnish	081A	Serbian (Latin)	2801	Arabic (Syria)
040C	French (Standard)	081D	Swedish (Finland)	2809	English (Belize)
040E	Hungarian	0C01	Arabic (Egypt)	280A	Spanish (Peru)
040F	Icelandic	0C04	Chinese (Hong Kong)	2C01	Arabic (Jordan)
0410	Italian (Standard)	0C07	German (Austrian)	2C09	English (Trinidad)
0411	Japanese	0C09	English (Australian)	2C0A	Spanish (Argentina)
0412	Korean	0C0A	Spanish (Modern Sort)	3001	Arabic (Lebanon)
0413	Dutch (Standard)	0C0C	French (Canadian)	300A	Spanish (Ecuador)
0414	Norwegian (Bokmal)	0C1A	Serbian (Cyrillic)	3401	Arabic (Kuwait)
0415	Polish	1001	Arabic (Lybia)	340A	Spanish (Chile)
0416	Portuguese (Brazilian)	1004	Chinese (Singapore)	3801	Arabic (U.A.E)
0418	Romanian	1007	German (Luxembourg)	380A	Spanish (Uruguay)
0419	Russian	1009	English (Canadian)	3C01	Arabic (Bahrain)
041A	Croatian	100A	Spanish (Guatemala)	3C0A	Spanish (Paraguay)
041B	Slovak	100C	French (Swiss)	4001	Arabic (Qatar)
041D	Swedish	1401	Arabic (Algeria)	400A	Spanish (Bolivia)
041F	Turkish	1407	German (Liechtenstein)	440A	Spanish (El Salvador)
0423	Belarusian	1409	English (New Zealand)	480A	Spanish (Honduras)
0424	Slovene	140A	Spanish (Costa Rica)	4C0A	Spanish (Nicaragua)
042D	Basque	140C	French (Luxembourg)	500A	Spanish (Puerto Rico)

codepage:

The right 4 characters of the id correspond to the codepage as in these examples:

0000	7-bit ASCII	04B0	Unicode	04E5	Windows, Greek
03A4	Windows, Japan	0400	Windows, Latin-2	04E7	Windows, Hebrew
03B5	Windows, Korean	04E3	Windows, Cyrillic	2710	Macintosh, Roman
03B6	Windows, Taiwan	04E4	Windows, Multilingual	2711	Macintosh, Japanese

The string 'Unknown' is returned for an unidentified language or codepage.

## Examples

- value "CompanyName" of version block 1 of regapp "bigfix.exe" = "BigFix Inc."
- ▶ Returns TRUE if the "CompanyName" value of the given file's version block equals the specified string.
- exists version block "040904B0" of regapp "bigfix.exe"
- ▶ Returns TRUE if the designated version block exists. The case of the name of the version block is ignored.
- codepage of only version block of regapp "bigfix.exe" is "Unicode"
- ▶ Returns TRUE if the version block codepage for the specified file is unicode.
- id of only version block of regapp "bigfix.exe" is "040904b0"
- ▶ Returns TRUE if the given file's version block id is the specified string.
- language of version block 1 of regapp "bigfix.exe" = "English (United States)"
- ▶ Returns TRUE if the given file's version block language is as specified.
- value "FileVersion" of version block 1 of regapp "bigfix.exe" as version
- ▶ When casting a string value to a version, the parser skips through the string until it identifies something that can be interpreted as a version. This is convenient for extracting version numbers from strings containing added text.

## File Line

A file line is a string from a text file.

### Creation Methods

Key Phrase	Form	Description
line <integer> of <file>	<i>Numbered</i>	Returns the nth line in a file. A file line is just a string, except that you can use the additional properties "next line" and "previous line".  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
line containing <string> of <file>	<i>Named</i>	Returns the line with the specified search string in the given file.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
line of <file>	<i>Plain</i>	Returns the lines of a specified file.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
line starting with <string> of <file>	<i>Named</i>	Returns a line from the given file beginning with the specified phrase.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
next line of <file line>	<i>Plain</i>	Returns the line after the specified line in a file (provided that it is not the last line). This Inspector can be chained indefinitely, eg., next line of next line of ....  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
previous line of <file line>	<i>Plain</i>	Returns the line before the nth line in a file, provided n>1. You may repeat this command up to three times.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1

## Properties

Key Phrase	Form	Return Type	Description
line number of <file line>	<i>Plain</i>	<integer>	Returns the line number of a given line. Can be used to locate specific lines in a file.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
next line of <file line>	<i>Plain</i>	<file line>	Returns the line after the specified line in a file (provided that it is not the last line). This Inspector can be chained indefinitely, eg., next line of next line of ....  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1
previous line of <file line>	<i>Plain</i>	<file line>	Returns the line before the nth line in a file, provided n>1. You may repeat this command up to three times.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, Mac:5.1

## Examples

- line 2 of (file "printmon.inf" of system folder)
  - ▶ Returns the second line of the specified file.
- lines containing "Signature]" of file "mmdriver.inf" of system folder
  - ▶ Returns a list corresponding to the top-level sections involving signatures in the specified .inf file.
- lines of file "mmdriver.inf" of system folder
  - ▶ Returns a list of the lines of the specified file.
- lines starting with "[" of file "mmdriver.inf" of system folder
  - ▶ Returns a list corresponding to all the top-level sections (lines that start with "[") in the specified .inf file.
- next line of line containing "[mciavi]" of file "mmdriver.inf" of system folder
  - ▶ Returns the line after the one containing "[mciavi]", which is a string such as "1:MSVFW32.DLL".
- previous line of previous line of previous line of line containing "[mciavi]" of file "mmdriver.inf" of system folder
  - ▶ Returns the line 3 lines previous to the one containing the specified phrase in the given file.

- line number of line containing "[mciavi]" of file "mmdriver.inf" of system folder
  - Returns the line number of the specified line in the given file.
- next line of line containing "[mciavi]" of file "mmdriver.inf" of system folder
  - Returns the line after the one containing "[mciavi]", which is a string such as "1:MSVFW32.DLL".
- previous line of previous line of previous line of line containing "[mciavi]" of file "mmdriver.inf" of system folder
  - Returns the line 3 lines previous to the one containing the specified phrase in the given file.

## Xml Dom Document

These are the Inspectors for the XML Document Object Module (DOM) for specified XML files. XML dom document objects are derived from XML dom node objects.

### Creation Methods

Key Phrase	Form	Description
owner document of <xml dom node>	<i>Plain</i>	Returns the name of the document that contains the specified node.  Win:5.1
xml document of <file>	<i>Plain</i>	Returns the XML Document Object Module (DOM) for the specified file.  Win:5.1
xml document of <string>	<i>Plain</i>	Returns an XML document object from the given <string>, typically a file name.  Win:6.0

### Examples

- xml document of file "c:\test.xml" as xml
  - Returns the test.xml document in a form like: <?xml version="1.0"?>%0d%0a<a:Books xmlns:a="x-schema:bookschema.xml">%0d%0a%09<a:Book>%0d%0a%09%09<title>Presenting XML</title>%0d%0a%09%09<author>Richard Light</author>%0d%0a%09</a:Book>%0d%0a</a:Books>%0d%0a.

## Xml Dom Node

These are the Inspectors for the XML Document Object Module (DOM) nodes.

### Creation Methods

Key Phrase	Form	Description
attribute <integer> of <xml dom node>	<i>Numbered</i>	Returns the numbered attribute of the specified XML DOM node.  Win:5.1
attribute <string> of <xml dom node>	<i>Named</i>	Returns the named attribute of the specified node.  Win:5.1
attribute of <xml dom node>	<i>Plain</i>	Returns the attribute(s) of the specified XML DOM node.  Win:5.1
child node <integer> of <xml dom node>	<i>Numbered</i>	Returns the child node by number.  Win:5.1
child node of <xml dom node>	<i>Plain</i>	By chaining this Inspector, you can find the child nodes of any given node.  Win:5.1
first child of <xml dom node>	<i>Plain</i>	Returns the first child node in the specified node. When applied to an XML DOM file, it returns the first node in the file.  Win:5.1
last child of <xml dom node>	<i>Plain</i>	Returns the last child node in the specified node. When applied to an XML DOM file, it returns the last node in the file.  Win:5.1
next sibling of <xml dom node>	<i>Plain</i>	Returns the next child node after the current one.  Win:5.1
parent node of <xml dom node>	<i>Plain</i>	Returns the parent node of the specified node. The top of the hierarchy is the document itself, so a phrase such as "exists parent node of xml dom document" will return FALSE.  Win:5.1
previous sibling of <xml dom node>	<i>Plain</i>	Returns the child node before the one specified.  Win:5.1

Key Phrase	Form	Description
select <string> of <xml dom node>	<i>Named</i>	Uses an Xpath string to specify an XML DOM node. For instance, to select all elements BBB which are children of the root element AAA, use: <ul style="list-style-type: none"> <li>• selects "/AAA/BBB" of xml dom document &lt;string&gt;.</li> </ul> Win:6.0
xpath <( string, string )> of <xml dom node>	<i>Indexed</i>	The iterated named property xpaths (<namespace>, <query>) provides a way of specifying the namespaces for the query. If the XML document you are querying over uses namespaces, you must use them in the query and use this property. Win:6.0
xpath <string> of <xml dom node>	<i>Named</i>	Returns an iterated list of matching xml dom nodes, given the xpath query specified by <string>. Win:6.0

## Properties

Key Phrase	Form	Return Type	Description
<xml dom node> as text	<i>Cast</i>	<string>	Casts an xml document object module node as text. Win:6.0
<xml dom node> as xml	<i>Cast</i>	<string>	Casts an xml document object module node as xml. Win:6.0
attribute <integer> of <xml dom node>	<i>Numbered</i>	<xml dom node>	Returns the numbered attribute of the specified XML DOM node. Win:5.1
attribute <string> of <xml dom node>	<i>Named</i>	<xml dom node>	Returns the named attribute of the specified node. Win:5.1
attribute of <xml dom node>	<i>Plain</i>	<xml dom node>	Returns the attribute(s) of the specified XML DOM node. Win:5.1
child node <integer> of <xml dom node>	<i>Numbered</i>	<xml dom node>	Returns the child node by number. Win:5.1

Key Phrase	Form	Return Type	Description
child node of <xml dom node>	<i>Plain</i>	<xml dom node>	By chaining this Inspector, you can find the child nodes of any given node.  Win:5.1
first child of <xml dom node>	<i>Plain</i>	<xml dom node>	Returns the first child node in the specified node. When applied to an XML DOM file, it returns the first node in the file.  Win:5.1
last child of <xml dom node>	<i>Plain</i>	<xml dom node>	Returns the last child node in the specified node. When applied to an XML DOM file, it returns the last node in the file.  Win:5.1
next sibling of <xml dom node>	<i>Plain</i>	<xml dom node>	Returns the next child node after the current one.  Win:5.1
node name of <xml dom node>	<i>Plain</i>	<string>	Returns the name of the specified XML DOM node as a string.  Win:5.1
node type of <xml dom node>	<i>Plain</i>	<integer>	Returns the numeric node type of the specified Document Object Module (DOM) node, 1-12 as shown in the creation Inspector.  Win:5.1
node value of <xml dom node>	<i>Plain</i>	<string>	Returns the node value, which varies depending on the node type. If the standard interface produces a null type, the Inspector throws NoSuchObject.  Win:5.1
owner document of <xml dom node>	<i>Plain</i>	<xml dom document>	Returns a document belonging to the owner of the specified node.  Win:5.1
parent node of <xml dom node>	<i>Plain</i>	<xml dom node>	Returns the parent node of the specified node.  Win:5.1
previous sibling of <xml dom node>	<i>Plain</i>	<xml dom node>	Returns the child node before the one specified.  Win:5.1
select <string> of <xml dom node>	<i>Named</i>	<xml dom node>	Uses an Xpath string to specify an XML DOM node. For instance, to select all elements BBB which are children of the root element AAA, use:

Key Phrase	Form	Return Type	Description
			<ul style="list-style-type: none"> <li>selects "/AAA/BBB" of xml dom document &lt;string&gt;.</li> </ul> <p>Win:6.0</p>
xpath <( string, string )> of <xml dom node>	<i>Indexed</i>	<xml dom node>	<p>The iterated named property xpath (&lt;namespace&gt;, &lt;query&gt;) provides a way of specifying the namespaces for the query. If the XML document you are querying over uses namespaces, you must use them in the query and use this property.</p> <p>Win:6.0</p>
xpath <string> of <xml dom node>	<i>Named</i>	<xml dom node>	<p>Returns an iterated list of matching xml dom nodes, given the xpath query specified by &lt;string&gt;.</p> <p>Win:6.0</p>

## Note

Some of the examples in this section refer to test.xml, a structured file like this:

```
<?xml version="1.0"?>
<a:Books xmlns:a="x-schema:bookschema.xml" >
  <a:Book>
    <title>Presenting XML</title>
    <author>Richard Light</author>
  </a:Book>
</a:Books>
```

## Examples

- node names of child nodes of child node 1 of xml document of file "icsxml\cmnicfg.xml" of system folder
  - ▶ Returns a list of the names of the children of the first node in the specified document.
- node names of child nodes of child node 1 of last child of xml document of file "icsxml\cmnicfg.xml" of system folder
  - ▶ Returns the names of the specified chain of child nodes.
- node name of next sibling of first child of xml document of file "icsxml\cmnicfg.xml" of system folder
  - ▶ Returns the name of the second node of the specified file.

■ `xpath ( "xmlns:a='x-schema:bookschema.xml'", "/a:Books/a:Book" )`  
of xml document of file "c:\test.xml" as xml

► Returns an xml dom node such as `<a:Book xmlns:a="x-schema:bookschema.xml">%0d%0a%09<title>Presenting XML</title>%0d%0a%09<author>Richard Light</author>%0d%0a</a:Book>`.

■ node value of attribute 0 of child node 0 of xml document of file "icsxml\cmnicfg.xml" of system folder

► Returns the value of the first attribute of the first node of the specified document.

■ node value of attribute "xmlns" of child node 1 of xml document of (file "icsxml\cmnicfg.xml" of system folder)

► Returns the value of the named attribute (xmlns) of the specified file.

■ node names of attributes of child nodes of xml document of file "icsxml\cmnicfg.xml" of system folder

► Returns the names of the attributes of each node in the specified file.

■ node names of child nodes of child node 1 of xml document of file "icsxml\cmnicfg.xml" of system folder

► Returns a list of the names of the children of the first node in the specified document.

■ node names of child nodes of child node 1 of last child of xml document of file "icsxml\cmnicfg.xml" of system folder

► Returns the names of the specified chain of child nodes.

■ node name of next sibling of first child of xml document of file "icsxml\cmnicfg.xml" of system folder

► Returns the name of the second node of the specified file.

■ node names of child nodes of xml document of file "C:\WINDOWS\system32\icsxml\cmnicfg.xml"

► Returns a list of the names of each node in the specified XML document.

■ node types of child nodes of xml document of file "icsxml\cmnicfg.xml" of system folder

► Returns a list of numeric types for each of the nodes in the specified XML document.

■ node value of first child of xml document of file "icsxml\cmnicfg.xml" of system folder

► Returns the value of the first node in the specified file. If the first statement of the file is `<xml version="1.0">`, for instance, the name would be "xml" and the value would be `version="1.0"`.

## Application Usage Summary

To enable these Inspectors, you first need to create the client setting `_BESClient_UsageManager_EnableAppUsageSummary` and initialize it to 1. You can also limit the summary to a subset of applications by creating `_BESClient_UsageManager_EnableAppUsageSummaryApps` and initializing it to a list of apps to include (or exclude). The value of this setting should look like `+:app1:app2:app3:` to add apps to the scope, and `-:app1:app2:` to exclude apps. The case is ignored. For instance, to enable summaries on the Word application, use the value `+:winword.exe:`.

### Creation Methods

Key Phrase	Form	Description
application usage summary	<i>PlainGlobal</i>	Returns an application usage summary containing information including the start time, duration and other statistics on client applications.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
application usage summary <string>	<i>NamedGlobal</i>	Returns the usage summary for the application specified in <string>.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

### Properties

Key Phrase	Form	Return Type	Description
first start time of <application usage summary>	<i>Plain</i>	<time>	Returns the start time of the specified application since the computer was configured to track it, regardless of reboots.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
last start time of <application usage summary>	<i>Plain</i>	<time>	Returns the last time this specified application was started.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
last time seen of <application usage summary>	<i>Plain</i>	<time>	Returns the last time this specified application was seen running.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
name of <application usage summary>	<i>Plain</i>	<string>	Returns the names of the applications that are currently enabled for usage summaries.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

Key Phrase	Form	Return Type	Description
running of <application usage summary>	<i>Plain</i>	<boolean>	Returns TRUE if the specified application is currently running. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
total duration of <application usage summary>	<i>Plain</i>	<time interval>	Returns the total elapsed time that the specified application has been running. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
total run count of <application usage summary>	<i>Plain</i>	<integer>	Returns the number of times that the specified application has been run since the client was configured to track it. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

## Examples

- last start time of application usage summary "winword.exe"
- Returns the date and time Word was last started.

# System Objects

These are the keywords available for querying various aspects of the system, including the name and version of the operating sytem. This chapter also covers the keywords used to describe the vendors and types of the various processors that coexist in a typical computer system. Some of these Inspectors are system-specific, but are included to provide cross-platform compatibility.

## Bios

On Windows computers, this object returns strings that identify the version of the BIOS. On other computers, all bios expressions will fail gracefully, rather than generating an error.

### Creation Methods

Key Phrase	Form	Description
bios	<i>PlainGlobal</i>	This Windows-only Inspector creates the Bios object. On a non-Windows operating system, it returns FALSE.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1

### Properties

Key Phrase	Form	Return Type	Description
<bios> as string	<i>Cast</i>	<string>	This Windows-only Inspector returns a string that is the concatenation of the BIOS name and date. On a non-Windows operating system, it returns FALSE.  Win:3.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1
date of <bios>	<i>Plain</i>	<string>	This Windows-only Inspector returns the date string stored in the bios. This string is formatted as MM/DD/YY. On a non-Windows operating system, it returns FALSE.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1
version of <bios>	<i>Plain</i>	<string>	This Windows-only Inspector returns the version string stored in the bios. This string may not exist. The format depends upon your BIOS manufacturer. On a non-Windows operating system, it returns FALSE.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1

## Examples

- `date of bios = "09/16/97"`
  - ▶ Returns TRUE if the BIOS date matches the value provided.
- `following text of last "/" of date of bios`
  - ▶ Returns the year of the bios as a string. For "09/07/99" it would return "99".
- `preceding text of first "/" of date of bios`
  - ▶ Returns the month of the bios date as a string.
- `preceding text of first "/" of following text of first "/" of date of bios`
  - ▶ Returns the day of the bios date as a string.
- `version of bios as lowercase contains "phoenix"`
  - ▶ Returns TRUE if the version string contains "phoenix", "PHOENIX" or "Phoenix".

## Operating System

The operating system object provides access to several important properties of the system.

### Creation Methods

Key Phrase	Form	Description
operating system	<i>PlainGlobal</i>	Creates the global operating system object.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

### Properties

Key Phrase	Form	Return Type	Description
<operating system> as string	<i>Cast</i>	<string>	Returns a string containing the name of the operating system concatenated with the release.  Win:3.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
boot time of <operating system>	<i>Plain</i>	<time>	Returns the time of the last restart.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
build number high of <operating system>	<i>Plain</i>	<integer>	Numeric representation of the most significant 16 bits of the build number.  Win:1.2

Key Phrase	Form	Return Type	Description
build number low of <operating system>	<i>Plain</i>	<integer>	Numeric representation of the least significant 16 bits of the build number.  Win:1.2
build number of <operating system>	<i>Plain</i>	<integer>	Returns the integer build number of the operating system. • Note: On the Macintosh, this returns a string as of BES version 5.1.  Win:1.2, Mac:4.1
csd version of <operating system>	<i>Plain</i>	<string>	Returns the Corrective Service Disk version of the operating system. The szCSDVersion as returned by the GetVersionEx system call. The format varies depending on the installed service packs. For WinNT it contains a string such as "Service Pack 3", for Win95 it can contain a string such as "B".  Win:1.2
major version of <operating system>	<i>Plain</i>	<integer>	Returns integer which is the dwMajorVersion returned by the GetVersionEx system call. Note that while the WinNT major version tracks the release (3 for 3.51, 4 for 4.0, and 5.0 for Windows 2000, 5.1 for Windows XP), the major version for Win95 and Win98 is always 4.  Win:1.2
metric <integer> of <operating system>	<i>Numbered</i>	<integer>	This inspector uses the windows GetSystemMetrics API. The integer constants and their meaning are defined by Microsoft. For example, the integer 87 indicates that the operating system is a media center edition. The integer constants are defined in WinUser.h.  Win:6.0
minor version of <operating system>	<i>Plain</i>	<integer>	Numeric representation of the minor version of the operating system.  Win:1.2
name of <operating system>	<i>Plain</i>	<string>	Returns the name of the operating system as a string. Names might include Win98, WinNT, etc.  Win:1.2, Lin:3.1, Sol:3.1, HP-UX:4.0, AIX:4.1, Mac:4.1

Key Phrase	Form	Return Type	Description
performance counter frequency of <operating system>	<i>Plain</i>	<hertz>	The rate at which the performance counter is being incremented (per second).  Win:1.2
performance counter of <operating system>	<i>Plain</i>	<integer>	Retrieves a 64-bit performance counter value.  Win:1.2
platform id of <operating system>	<i>Plain</i>	<integer>	Returns the dwPlatformId as returned by the GetVersionEx system call. Possible values are 1 (Win95/95) and 2 (WinNT).  Win:1.2
product type of <operating system>	<i>Plain</i>	<operating system product type>	Returns the product type of the operating system, which includes Workstations, Domain Controllers and Servers.  Win:6.0
release of <operating system>	<i>Plain</i>	<string>	Information about the release of the operating system, typically formatted as <Major version>.<Minor version>.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
service pack major version of <operating system>	<i>Plain</i>	<integer>	Returns the major version number of the current service pack of the specified OS.  Win:6.0
service pack minor version of <operating system>	<i>Plain</i>	<integer>	Returns the minor version number of the current service pack of the specified OS.  Win:6.0
suite mask of <operating system>	<i>Plain</i>	<operating system suite mask>	Returns the bit-mapped suite mask for the operating system, which contains further fine-grain information about the version.  Win:6.0
uptime of <operating system>	<i>Plain</i>	<time interval>	Returns a time interval that represents the elapsed time since the operating system was last booted. • Note: Depending on the Laptop, this interval may not include time spent in hibernation.  Win:5.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1
x64 of <operating system>	<i>Plain</i>	<boolean>	Returns TRUE if the current operating system is 64-bits.  Win:6.0

## Examples

- `now - boot time of operating system > week`
  - ▶ Returns TRUE if the computer hasn't been rebooted for over a week.
- `build number high of operating system = 1027`
  - ▶ Returns TRUE if the high word of the build number = 0403 hex.
- `build number low of operating system = 1212`
  - ▶ Returns TRUE if the low word of the build number = 04BC hex.
- `build number of operating system = 67306684`
  - ▶ Returns TRUE if the build number = 040304BC in hexadecimal.
- `csd version of the operating system = "B"`
  - ▶ Returns TRUE on a Win95 System with Corrective Service Disk version = "B".
- `major version of operating system = 4`
  - ▶ Returns TRUE if the major version (before the dot) is 4, such as 4.1, 4.2, etc.
- `Metric 87 of operating system`
  - ▶ Returns TRUE if the OS is a Media Center Edition.
- `minor version of operating system = 0`
  - ▶ Returns TRUE if the minor part of a version number (after the dot) is 0, such as 4.0, 5.0, etc.
- `name of operating system = "WinXP"`
  - ▶ Returns TRUE on a WinXP System.
- `platform id of operating system = 1`
  - ▶ Returns TRUE on a Win95 System.
- `release of operating system = "OSR2.1"`
  - ▶ Returns TRUE if the Win95 computer is running under operating system release 2.5.

## Processor

The processor object is used to identify the number and properties of processors in the system. You can identify the manufacturer of the CPU as well as the speed and other features. Many operating systems provide for multiple processors. You can inspect any one of them by their ordinal number.

### Creation Methods

Key Phrase	Form	Description
main processor	<i>PlainGlobal</i>	Creates the object associated with the 'Primary' processor. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
processor	<i>PlainGlobal</i>	Iterates through the processors in the system. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
processor <integer>	<i>NumberedGlobal</i>	Creates the processor object for the number specified. The first processor is processor number 1. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1</small>

### Properties

Key Phrase	Form	Return Type	Description
brand id of <processor>	<i>Plain</i>	<integer>	This inspector returns the integer known as the brand id, returned from the assembly language cpuid extended instruction. <small>Win:6.0</small>
brand string of <processor>	<i>Plain</i>	<string>	Returns the vendor-defined brand names for newer processors. <small>Win:1.2</small>
extended family of <processor>	<i>Plain</i>	<integer>	Integer representing the extended family of CPU. See the notes for the meaning of these numbers. <small>Win:1.2</small>
extended model of <processor>	<i>Plain</i>	<integer>	Integer representing the extended model of CPU. See the notes for the meaning of these numbers. <small>Win:1.2</small>
family name of <processor>	<i>Plain</i>	<string>	Returns the family name of the CPU, dependent on the type of client computer, for instance Pentium, Sparc, PowerPC G4, etc. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

Key Phrase	Form	Return Type	Description
family of <processor>	<i>Plain</i>	<integer>	Returns an integer representing the family of the CPU. See the notes for the meaning of these numbers. <ul style="list-style-type: none"> <li>• Note: As of BES 6.0, this Inspector returns a string on Unix computers.</li> </ul> Win:1.2, Lin:3.1, Sol:4.1, AIX:4.1
feature mask of <processor>	<i>Plain</i>	<integer>	Returns the feature flags from the CPUID instruction. The feature mask contains bits that identify extra features the processor may provide such as MMX support or if the Processor ID is enabled on the processor. Win:1.2
model of <processor>	<i>Plain</i>	<integer>	Returns the model number of the CPU. <ul style="list-style-type: none"> <li>• Note: On Unix computers, this Inspector returns a &lt;string&gt; as of BES 6.0.</li> </ul> Win:1.2, Lin:3.1, Sol:4.1, HP-UX:4.0, AIX:4.1
speed of <processor>	<i>Plain</i>	<hertz>	Returns the speed of the processor in Hertz. Win:1.2, Lin:3.1, Sol:3.1, HP-UX:4.0, AIX:4.1, Mac:4.1
stepping of <processor>	<i>Plain</i>	<integer>	Returns the stepping number of the processor. This item can be helpful in identifying very specific processor features or limitations. Win:1.2, Lin:3.1
type of <processor>	<i>Plain</i>	<integer>	Numeric type of the CPU. Values include: <ul style="list-style-type: none"> <li>• 0 - standard,</li> <li>• 1 - overdrive,</li> <li>• 2 - dual CPU capable,</li> <li>• 3 - reserved.</li> </ul> <ul style="list-style-type: none"> <li>• Note: this Inspector returns a &lt;string&gt; type as of BES version 6.0 on Unix machines and version 5.1 on the Macintosh.</li> </ul> Win:1.2, Sol:3.1, AIX:4.1, Mac:4.1
vendor name of <processor>	<i>Plain</i>	<string>	The manufacturer of the CPU. Names include: <ul style="list-style-type: none"> <li>• GenuineIntel,</li> <li>• AuthenticAMD,</li> <li>• CyrixInstead,</li> <li>• CentaurHauls,</li> <li>• AmbiguousCPU.</li> </ul> Win:1.2, Lin:3.1

## Examples

- `number of processors > 1`
- ▶ Returns TRUE if the computer is a multi-processor system.
  
- `family name of main processor = "Pentium III"`
- ▶ Returns TRUE for a computer with a Pentium III cpu.
  
- `bit 18 of feature mask of main processor`
- ▶ Returns TRUE if the processor ID feature is enabled on this processor.
  
- `speed of main processor < 2000 * MHz`
- ▶ Returns TRUE is the cpu is slower than 2Ghz.
  
- `vendor name of main processor = "GenuineIntel"`
- ▶ Returns TRUE for an Intel processor chip.
  
- `number of processors whose (vendor name of it = "AuthenticAMD" or vendor name of it = "CyrixInstead" or vendor name of it = "CentaurHauls") = 1`
- ▶ Returns TRUE for a single processor system with the given vendors.

---

## Ram

The ram object is used to inspect properties of the computer's random access memory.

### Creation Methods

Key Phrase	Form	Description
ram	<i>PlainGlobal</i>	Creates the object that can be accessed to inspect the amount of ram on the machine. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
random access memory	<i>PlainGlobal</i>	Same as 'ram'. <small>Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1</small>

### Properties

Key Phrase	Form	Return Type	Description
size of <ram>	<i>Plain</i>	<integer>	Returns the number of bytes of random access memory on the current machine. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

## Examples

- `size of ram / (1024 * 1024)`
- Returns the size of RAM in megabytes.

## Active Device

On Windows NT systems (including XP and 2K), these Inspectors returns a set of objects corresponding to the active devices on the machine, e.g., modems, graphics cards, printers, etc.

## Creation Methods

Key Phrase	Form	Description
active device	<i>PlainGlobal</i>	Creates a list of all active devices found using the Configuration Manager SetupDiGetClassDevs NT API. The locations of active devices are relative to the Windows Directory.  Win:1.2

## Properties

Key Phrase	Form	Return Type	Description
class of <active device>	<i>Plain</i>	<string>	Returns the name of the class of the active device provided.  Win:1.2
description of <active device>	<i>Plain</i>	<string>	Returns the description of the active device provided.  Win:1.2
driver key of <active device>	<i>Plain</i>	<registry key>	The key identified by adding the value of 'driver key value name of active device' to HKLM\System\CurrentControlSet\Control\Class\.  Win:1.2
driver key value name of <active device>	<i>Plain</i>	<string>	Returns the driver key value name of the active device provided.  Win:1.2
friendly name of <active device>	<i>Plain</i>	<string>	Returns the friendly name of the active device.  Win:1.2

Key Phrase	Form	Return Type	Description
hardware id of <active device>	<i>Plain</i>	<string>	Returns the hardware id of the active device provided. <small>Win:1.2</small>
location information of <active device>	<i>Plain</i>	<string>	Returns a string containing information about the bus location of the device. <small>Win:1.2</small>
manufacturer of <active device>	<i>Plain</i>	<string>	Returns the manufactures string of the active device. <small>Win:1.2</small>
problem id of <active device>	<i>Plain</i>	<integer>	Configuration manager defined number describing device installation or use problems. Returned by call to CM_Get_DevInst_Status. <small>Win:1.2</small>
service key value name of <active device>	<i>Plain</i>	<string>	Returns the service key value name. <small>Win:1.2</small>
status of <active device>	<i>Plain</i>	<integer>	Configuration manager defined status bits conveying device driver status. <small>Win:1.2</small>

## Examples

- `number of active devices`
  - Returns the number of active devices as determined by the Configuration Manager.
- `exists active device file "system32\ntoskrnl.exe"`
  - Returns TRUE if ntoskrnl.exe exists in the System32 folder of the Windows folder.
- `exists active device file "C:\WINNT\System32\ntoskrnl.exe"`
  - Returns FALSE because this Inspector path is relative to the Windows directory.
- `exists active device whose (class of it = "Display")`
  - Returns TRUE if there is an active device named "Display".
- `(description of it) of active devices whose ( class of it = "Display" )`
  - Provides a list of the descriptions of the active display devices.

- exists service key value name whose (it = "PGPdisk" ) of active devices
- Returns TRUE if the designated service key value name exists on this system.

## License

These Inspectors are available only through the BigFix Enterprise System. They inspect the properties of the client's BigFix license.

### Creation Methods

Key Phrase	Form	Description
client license	<i>PlainGlobal</i>	Creates the global object containing client licensing information.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

### Properties

Key Phrase	Form	Return Type	Description
common name of <license>	<i>Plain</i>	<string>	Returns the name of the person (such as John Smith) who requested the action site license.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1
email address of <license>	<i>Plain</i>	<string>	Returns the email address of the person (such as John_Smith@bigcorp.com) who requested the action site license.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1
evaluation of <license>	<i>Plain</i>	<boolean>	Returns TRUE if client is running an evaluation license.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
expiration date of <license>	<i>Plain</i>	<time>	Returns date when license will expire.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
expiration state of <license>	<i>Plain</i>	<string>	Returns a string, one of "Unrestricted", "Grace" or "Restricted".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
maximum seat count of <license>	<i>Plain</i>	<integer>	Returns maximum seat count allowed by the license.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

Key Phrase	Form	Return Type	Description
organization of <license>	<i>Plain</i>	<string>	Returns the organization of the person (such as Bigcorp, Inc.) who requested the action site license.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1
registrar number of <license>	<i>Plain</i>	<integer>	A unique number assigned to the issuer of the Action Site certificate.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1
seat count state of <license>	<i>Plain</i>	<string>	Returns one of "Unrestricted", "Grace" or "Restricted".  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
seat of <license>	<i>Plain</i>	<integer>	The license number assigned to the client.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
site number of <license>	<i>Plain</i>	<integer>	A unique number assigned to the Action Site certificate.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1
start date of <license>	<i>Plain</i>	<time>	The starting date specified for the BigFix license.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1

## Local Mssql Database

These Inspectors retrieve the properties of the MS SQL databases on the local machine.

### Creation Methods

Key Phrase	Form	Description
local mssql database	<i>PlainGlobal</i>	Creates the local MSSQL database objects.  Win:1.2
local mssql database <string>	<i>NamedGlobal</i>	Creates the local Microsoft SQL (MSSQL) database object identified by the name provided.  Win:1.2

## Properties

Key Phrase	Form	Return Type	Description
audit level of <local mssql database>	<i>Plain</i>	<integer>	Returns the integer audit level of the MSSQL database.  Win:1.2
has blank sa password of <local mssql database>	<i>Plain</i>	<boolean>	Returns TRUE if the MSSQL database sa account has a blank password.  Win:1.2
instance name of <local mssql database>	<i>Plain</i>	<string>	Returns the name of the MSSQL database.  Win:1.2
login mode of <local mssql database>	<i>Plain</i>	<integer>	Returns the login mode of the MSSQL database.  Win:1.2
running of <local mssql database>	<i>Plain</i>	<boolean>	Returns a boolean indicating if the MSSQL database is running.  Win:1.2

## Service

Provides access to all services configured on Windows NT, 2K and XP systems. On a non-Windows system, expressions using these objects will fail gracefully instead of generating an error.

## Creation Methods

Key Phrase	Form	Description
main gather service	<i>PlainGlobal</i>	Returns FALSE. Included for compatibility with Windows Inspectors.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
relay service	<i>PlainGlobal</i>	Returns a service object for the relay component of BES.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
running service	<i>PlainGlobal</i>	Creates objects corresponding to all the running services.  Win:1.2
running service <string>	<i>NamedGlobal</i>	Creates the running service object for the specified name.  Win:1.2

Key Phrase	Form	Description
service	<i>PlainGlobal</i>	Creates objects for all the services.  Win:1.2
service <string>	<i>NamedGlobal</i>	Creates the service object matching the specified name, regardless of its running state.  Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Properties

Key Phrase	Form	Return Type	Description
<service> as string	<i>Cast</i>	<string>	Returns a string containing the Service name, Display name, and State of the service.  Win:1.2
can interact with desktop of <service>	<i>Plain</i>	<boolean>	Indicates the system is configured to allow the service to interact with the desktop.  Win:1.2
checkpoint of <service>	<i>Plain</i>	<integer>	Service specific value indicating its checkpoint state.  Win:1.2
display name of <service>	<i>Plain</i>	<string>	Returns the display name of the service.  Win:1.2
file of <service>	<i>Plain</i>	<file>	Returns a file object corresponding to the specified <service>.  Win:3.0
image path of <service>	<i>Plain</i>	<string>	Returns the full path to the service executable.  Win:6.0
login account of <service>	<i>Plain</i>	<string>	Returns the login account under which the service is configured to run.  Win:1.2
service name of <service>	<i>Plain</i>	<string>	Returns the name of the service.  Win:1.2
service specific exit code of <service>	<i>Plain</i>	<integer>	Service specific exit code.  Win:1.2

Key Phrase	Form	Return Type	Description
start type of <service>	<i>Plain</i>	<string>	<p>Returns a string that represents the service startup configuration. It describes when the driver is loaded, which can be one of:</p> <ul style="list-style-type: none"> <li>• boot: started by OS loader (usually these are needed to launch the OS).</li> <li>• system: started during OS initialization (used by PnP drivers that do device detection after the loader is done).</li> <li>• auto: started by the Service Control Manager (SCM).</li> <li>• demand: started on demand, either by PnP manager when a device is enumerated or by the SCM in response to user demand.</li> <li>• disabled: can't be started (used to temporarily disable driver services).</li> </ul> <p>Win:3.0</p>
state of <service>	<i>Plain</i>	<string>	<p>Returns one of Continuing, Pausing, Paused, Running, Starting, Stopping, Stopped, Unknown.</p> <p>Win:1.2, Lin:6.0, Sol:6.0, HP-UX:6.0, AIX:6.0</p>
win32 exit code of <service>	<i>Plain</i>	<integer>	<p>Service specific Win32 exit code.</p> <p>Win:1.2</p>

## Examples

### ■ running services

► Returns a list of all the currently running services.

### ■ expand environment string of ( image path of service "AudioSrv" )

► Returns a path name for the Windows Audio service, such as  
C:\WINDOWS\System32\svchost.exe -k netsvcs.

## Language

A language is composed of a primary language (for example, Swiss) and a sub-language (for example, Swiss German).

### Creation Methods

Key Phrase	Form	Description
system locale	<i>PlainGlobal</i>	Determines which bitmap fonts, and OEM, ANSI, and MAC code pages are defaults for the system. This only affects applications that are not fully Unicode.  Win:4.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1
system ui language	<i>PlainGlobal</i>	Determines the default language of menus and dialogs, messages, INF files, and help files.  Win:4.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1
user locale	<i>PlainGlobal</i>	Determines which settings are used for formatting dates, times, currency, and numbers as a default for each user. Also determines the sort order for sorting text.  Win:4.1
user ui language	<i>PlainGlobal</i>	Non-MUI: Same as system UI Language. • MUI: Determines the language of menus and dialogs, messages, and help files.  Win:4.1

### Properties

Key Phrase	Form	Return Type	Description
<language> as string	<i>Cast</i>	<string>	Returns the language of the system locale.  Win:4.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1
primary language of <language>	<i>Plain</i>	<primary language>	Extracts the primary language identifier from a language.  Win:4.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1

### Examples

- `system locale as string`
- Returns English (United States) if the system locale is US English.

## Primary Language

A primary language identifier indicates the written/spoken language that is used by the system. However, to identify the language that is used in a country or region you must combine the primary language with a sub-language identifier to form language identifiers.

### Creation Methods

Key Phrase	Form	Description
primary language of <language>	<i>Plain</i>	Extracts the primary language identifier from a language.  Win:4.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1

### Properties

Key Phrase	Form	Return Type	Description
<primary language> as string	<i>Cast</i>	<string>	Returns the primary language.  Win:4.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1

### Examples

- `primary language of system locale`
- Returns "English" for an English-language system.

## Firewall Objects

These Inspectors retrieve the various firewall settings for the computer.

---

### Firewall

The Firewall Inspectors allow you to view the settings of the Windows Firewall on Windows Clients. By inspecting the firewall properties, you can determine which applications have access to unsolicited traffic and how the firewall is configured for various subsets of your network. The Windows Firewall is supported on Windows XP SP2. For more information, search for 'Windows Firewall API' at the msdn site (<http://msdn.microsoft.com/>).

### Creation Methods

Key Phrase	Form	Description
firewall	<i>PlainGlobal</i>	Returns the global firewall object for this computer.  Win:5.1

### Properties

Key Phrase	Form	Return Type	Description
current profile type of <firewall>	<i>Plain</i>	<firewall profile type>	Returns the current profile type, corresponding to the Microsoft Windows Firewall enumerated type: NET_FW_PROFILE_CURRENT.  Win:5.1
local policy of <firewall>	<i>Plain</i>	<firewall policy>	Returns the local policy of the specified firewall.  Win:5.1

---

## Firewall Authorized Application

These Inspectors apply to applications that are authorized to exchange traffic through the Windows Firewall. These properties correspond to the `INetFwAuthorizedApplication` interface in the Windows Firewall API. The Windows Firewall is supported on Windows XP SP2. For more information, search for 'Windows Firewall API' at the msdn site (<http://msdn.microsoft.com/>).

### Creation Methods

Key Phrase	Form	Description
authorized application of <firewall profile>	<i>Plain</i>	Provides access to the properties of any application that has been authorized to have firewall openings. There are several methods attached to this Inspector, including: <ul style="list-style-type: none"> <li>• Name</li> <li>• ProcessImageFileName</li> <li>• IpVersion</li> <li>• Scope</li> <li>• RemoteAddress</li> <li>• Enabled.</li> </ul> Win:5.1

### Properties

Key Phrase	Form	Return Type	Description
enabled of <firewall authorized application>	<i>Plain</i>	<boolean>	Returns the contents of the Enabled property for the specified application. Returns TRUE if the settings for this application are currently enabled. Win:5.1
ip version of <firewall authorized application>	<i>Plain</i>	<ip version>	Returns the contents of the IpVersion property for the specified application. Win:5.1
name of <firewall authorized application>	<i>Plain</i>	<string>	Returns the contents of the Friendly Name property for the specified application. Win:5.1
process image file name of <firewall authorized application>	<i>Plain</i>	<string>	Returns the contents of the ProcessImageFileName property for the specified application. Win:5.1

Key Phrase	Form	Return Type	Description
remote addresses of <firewall authorized application>	<i>Plain</i>	<string>	Returns the contents of the RemoteAddresses property for the specified application. This property accesses a set of remote addresses that an application can use to listen for traffic.  Win:5.1
scope of <firewall authorized application>	<i>Plain</i>	<firewall scope>	Returns the contents of the Scope property for the specified application. This property controls the network scope that a port can listen to.  Win:5.1

## Firewall Profile

These Inspectors provide access to the firewall profile. These properties correspond to the INetFwProfile interface in the Windows Firewall API. For more information, search for 'Windows Firewall API' at the msdn site (<http://msdn.microsoft.com/>).

## Creation Methods

Key Phrase	Form	Description
current profile of <firewall policy>	<i>Plain</i>	Returns the profile currently in effect for the specified firewall policy.  Win:5.1
domain profile of <firewall policy>	<i>Plain</i>	Returns the domain profile of the specified firewall policy. The domain profile settings are used when a computer is connected to a network that contains the organization's domain controllers.  Win:5.1
standard profile of <firewall policy>	<i>Plain</i>	Returns the standard profile of the specified firewall policy. The standard profile settings are used when a computer is connected to a network that does not contain the organization's domain controllers.  Win:5.1

## Properties

Key Phrase	Form	Return Type	Description
authorized application of <firewall profile>	<i>Plain</i>	<firewall authorized application>	Access to the AuthorizedApplications collection for this profile. This Inspector can be iterated through all the authorized applications in each firewall profile.  Win:5.1
exceptions allowed of <firewall profile>	<i>Plain</i>	<boolean>	Returns the property that indicates whether exceptions should be allowed by the firewall.  Win:6.0
firewall enabled of <firewall profile>	<i>Plain</i>	<boolean>	Gets the value of the FirewallEnabled setting.  Win:5.1
globally open port of <firewall profile>	<i>Plain</i>	<firewall open port>	Provides access to the GloballyOpenPorts collection for this profile. This property can be iterated.  Win:5.1
icmp settings of <firewall profile>	<i>Plain</i>	<firewall icmp settings>	Gets the object governing settings for ICMP packets.  Win:5.1
notifications disabled of <firewall profile>	<i>Plain</i>	<boolean>	Gets the value of the NotificationsDisabled setting, TRUE or FALSE.  Win:5.1
remote admin settings of <firewall profile>	<i>Plain</i>	<firewall remote admin settings>	Gets the object containing the remote administration settings.  Win:5.1
service of <firewall profile>	<i>Plain</i>	<firewall service>	Gets the collection containing the services for this profile. This Inspector can be iterated over all services.  Win:5.1
type of <firewall profile>	<i>Plain</i>	<firewall profile type>	Returns the type of the specified firewall profile: domain, standard or current.  Win:5.1

Key Phrase	Form	Return Type	Description
unicast responses to multicast broadcast disabled of <firewall profile>	<i>Plain</i>	<boolean>	Gets the value of the UnicastResponsesToMulticastBroadcastDisabled setting.  Win:5.1

## Examples

- firewall enabled of current profile of local policy of firewall
- Verify that the firewall is enabled.

## Firewall Profile Type

These Inspectors provide access to the firewall profile type. The Windows Firewall is supported on Windows XP SP2. For more information, search for 'Windows Firewall API' at the msdn site (<http://msdn.microsoft.com/>).

## Creation Methods

Key Phrase	Form	Description
current firewall profile type	<i>PlainGlobal</i>	Retrieves the type of firewall profile that is currently in effect.  Win:5.1
current profile type of <firewall>	<i>Plain</i>	Creates a domain firewall profile type for comparison.  Win:5.1
domain firewall profile type	<i>PlainGlobal</i>	Returns the current profile type, corresponding to the Microsoft Windows Firewall enumerated type: NET_FW_PROFILE_DOMAIN.  Win:5.1
firewall profile type <integer>	<i>NumberedGlobal</i>	Returns the firewall profile type corresponding to the given integer: <ul style="list-style-type: none"> <li>• 0: Domain</li> <li>• 1: Standard</li> <li>• 2: Current.</li> </ul> Win:5.1

Key Phrase	Form	Description
standard firewall profile type	<i>PlainGlobal</i>	Returns the Standard firewall profile type.  Win:5.1
type of <firewall profile>	<i>Plain</i>	Returns the type of the specified firewall profile, corresponding to the Microsoft Windows Firewall enumerated types: <ul style="list-style-type: none"> <li>• NET_FW_PROFILE_DOMAIN</li> <li>• NET_FW_PROFILE_STANDARD</li> <li>• NET_FW_PROFILE_CURRENT.</li> </ul> Win:5.1

## Operators

Key phrase	Return Type	Description
<firewall profile type> = <firewall profile type>	< <i>boolean</i> >	Compares two firewall policies for equality only.  Win:5.1

## Examples

- `current profile type of firewall = domain firewall profile type`
  - Returns TRUE if the current profile type is domain.
- `current profile type of firewall = standard firewall profile type`
  - Returns TRUE if the current profile type is standard.

## Firewall Policy

These Inspectors provide access to the local, current, domain and standard firewall policies. These properties correspond to the INetFwPolicy interface in the Windows Firewall API. The Windows Firewall is supported on Windows XP SP2. For more information, search for 'Windows Firewall API' at the msdn site (<http://msdn.microsoft.com/>).

### Creation Methods

Key Phrase	Form	Description
local policy of <firewall>	<i>Plain</i>	Returns the local policy of the specified firewall.  Win:5.1

### Properties

Key Phrase	Form	Return Type	Description
current profile of <firewall policy>	<i>Plain</i>	<firewall profile>	Returns the profile currently in effect for the specified firewall policy.  Win:5.1
domain profile of <firewall policy>	<i>Plain</i>	<firewall profile>	Returns the domain profile of the specified firewall policy. The domain profile settings are used when a computer is connected to a network that contains the organization's domain controllers.  Win:5.1
standard profile of <firewall policy>	<i>Plain</i>	<firewall profile>	Returns the standard profile of the specified firewall policy. The standard profile settings are used when a computer is connected to a network that does not contain the organization's domain controllers. This Inspector corresponds to the Microsoft Windows Firewall enumerated type NET_FW_PROFILE_STANDARD.  Win:5.1

## Firewall Scope

These Inspectors provide access to the firewall scope. The Windows Firewall is supported on Windows XP SP2. For more information, search for 'Windows Firewall API' at the msdn site (<http://msdn.microsoft.com/>).

### Creation Methods

Key Phrase	Form	Description
all firewall scope	<i>PlainGlobal</i>	Returns the scope of computers that allow ALL traffic through the firewall, corresponding to the Microsoft enumerated type NET_FW_SCOPE_ALL.  Win:5.1
custom firewall scope	<i>PlainGlobal</i>	Returns the custom firewall scope, corresponding to the Microsoft Windows Firewall enumerated type: NET_FW_SCOPE_CUSTOM.  Win:5.1
firewall scope <integer>	<i>NumberedGlobal</i>	Returns the scope of addresses from which a port can listen, corresponding to the Microsoft enumerated types: <ul style="list-style-type: none"> <li>• NET_FW_SCOPE_ALL</li> <li>• NET_FW_SCOPE_LOCAL_SUBNET</li> <li>• NET_FW_SCOPE_CUSTOM.</li> </ul> Win:5.1
local subnet firewall scope	<i>PlainGlobal</i>	Returns the local subnet firewall scope, corresponding to the Microsoft Windows Firewall enumerated type: NET_FW_SCOPE_LOCAL_SUBNET.  Win:5.1
scope of <firewall authorized application>	<i>Plain</i>	Retrieves the contents of the Scope property of the authorized application.  Win:5.1
scope of <firewall open port>	<i>Plain</i>	Retrieves the contents of the Scope property of the open port.  Win:5.1
scope of <firewall service>	<i>Plain</i>	Retrieves the contents of the Scope property of the firewall service.  Win:5.1

## Operators

Key phrase	Return Type	Description
<firewall scope> = <firewall scope>	<boolean>	Compares two firewall scopes for equality only.  Win:5.1

## Firewall Open Port

These Inspectors provide access to the properties of a port that has been opened in the Windows Firewall. These properties correspond to the INetFwOpenPort interface in the Windows Firewall API. The Windows Firewall is supported on Windows XP SP2. For more information, search for 'Windows Firewall API' at the msdn site (<http://msdn.microsoft.com/>).

## Creation Methods

Key Phrase	Form	Description
globally open port of <firewall profile>	<i>Plain</i>	Provides access to the GloballyOpenPorts collection for this profile.  Win:5.1
globally open port of <firewall service>	<i>Plain</i>	Returns the collection of globally open ports associated with the firewall service.  Win:5.1

## Properties

Key Phrase	Form	Return Type	Description
built in of <firewall open port>	<i>Plain</i>	<boolean>	Returns the contents of the BuiltIn property of the firewall open port.  Win:5.1
enabled of <firewall open port>	<i>Plain</i>	<boolean>	Returns the contents of the Enabled property of the firewall open port.  Win:5.1
ip version of <firewall open port>	<i>Plain</i>	<ip version>	Returns the IpVersion property of the firewall open port.  Win:5.1
name of <firewall open port>	<i>Plain</i>	<string>	Returns the Name property of the firewall open port.  Win:5.1

Key Phrase	Form	Return Type	Description
port of <firewall open port>	<i>Plain</i>	<integer>	Returns the Port property of the firewall open port. <small>Win:5.1</small>
protocol of <firewall open port>	<i>Plain</i>	<internet protocol>	Returns the Protocol property of the firewall open port. <small>Win:5.1</small>
remote addresses of <firewall open port>	<i>Plain</i>	<string>	Returns the RemoteAddresses property of the firewall open port. <small>Win:5.1</small>
scope of <firewall open port>	<i>Plain</i>	<firewall scope>	Returns the Scope property of the firewall open port. <small>Win:5.1</small>

## Firewall Service

These Inspectors provide access to the properties of a service that may be authorized to listen through the firewall. These properties correspond to the INetFwService interface in the Windows Firewall API. The Windows Firewall is supported on Windows XP SP2. For more information, search for 'Windows Firewall API' at the msdn site (<http://msdn.microsoft.com/>).

### Creation Methods

Key Phrase	Form	Description
service of <firewall profile>	<i>Plain</i>	Gets the collection containing the services for this profile. <small>Win:5.1</small>

## Properties

Key Phrase	Form	Return Type	Description
customized of <firewall service>	<i>Plain</i>	<boolean>	Returns a flag that indicates whether at least one of the ports associated with the service has been customized. Either TRUE or FALSE.  Win:5.1
enabled of <firewall service>	<i>Plain</i>	<boolean>	Returns the enabled flag for the specified firewall service.  Win:5.1
globally open port of <firewall service>	<i>Plain</i>	<firewall open port>	Returns the collection of globally open ports associated with the firewall service.  Win:5.1
ip version of <firewall service>	<i>Plain</i>	<ip version>	Returns the the IP version for the specified firewall service.  Win:5.1
name of <firewall service>	<i>Plain</i>	<string>	Returns the friendly name of the firewall service.  Win:5.1
remote addresses of <firewall service>	<i>Plain</i>	<string>	Returns the contents of the RemoteAddresses property for the specified firewall service.  Win:5.1
scope of <firewall service>	<i>Plain</i>	<firewall scope>	Retrieves the contents of the Scope property of the firewall service.  Win:5.1
type of <firewall service>	<i>Plain</i>	<firewall service type>	Returns the type of the specified firewall service (file and print, upnp, remote desktop or none).  Win:5.1

## Examples

- exists globally open port whose (port of it = 52311 and protocol of it = udp and enabled of it) of current profile of local policy of firewall
- Returns TRUE if the BES Client can receive pings.

## Firewall Service Type

These Inspectors provide access to the firewall service type. These properties correspond to the `INetFwService.Type` interface in the Windows Firewall API. The Windows Firewall is supported on Windows XP SP2. For more information, search for 'Windows Firewall API' at the msdn site (<http://msdn.microsoft.com/>).

### Creation Methods

Key Phrase	Form	Description
file_and_print firewall service type	<i>PlainGlobal</i>	Returns the global service type for file and print sharing, corresponding to the Microsoft enumerated type: <code>NET_FW_SERVICE_FILE_AND_PRINT</code> .  Win:6.0
firewall service type <integer>	<i>NumberedGlobal</i>	Returns the firewall service type specified by <integer>, corresponding to the Microsoft Windows Firewall enumerated types: <ul style="list-style-type: none"> <li>• <code>NET_FW_SERVICE_FILE_AND_PRINT</code></li> <li>• <code>NET_FW_SERVICE_UPNP</code></li> <li>• <code>NET_FW_SERVICE_REMOTE_DESKTOP</code></li> <li>• <code>NET_FW_SERVICE_NONE</code>.</li> </ul> Win:5.1
none firewall service type	<i>PlainGlobal</i>	Returns the no firewall service type, corresponding to the Microsoft Windows Firewall enumerated type: <code>NET_FW_SERVICE_NONE</code> .  Win:6.0
remote desktop firewall service type	<i>PlainGlobal</i>	Returns the remote desktop firewall service type, corresponding to the Microsoft Windows Firewall enumerated type: <code>NET_FW_SERVICE_REMOTE_DESKTOP</code> .  Win:5.1
type of <firewall service>	<i>Plain</i>	Returns the type of the specified firewall service, corresponding to the Microsoft Windows Firewall enumerated types: <ul style="list-style-type: none"> <li>• <code>NET_FW_SERVICE_FILE_AND_PRINT</code></li> <li>• <code>NET_FW_SERVICE_UPNP</code></li> <li>• <code>NET_FW_SERVICE_REMOTE_DESKTOP</code></li> <li>• <code>NET_FW_SERVICE_NONE</code>.</li> </ul> Win:5.1

Key Phrase	Form	Description
upnp firewall service type	<i>PlainGlobal</i>	Returns the UPnP (Universal Plug and Play) firewall service type, corresponding to the Microsoft Windows Firewall enumerated type: NET_FW_SERVICE_UPNP. <ul style="list-style-type: none"> <li>• Note: UPnP is not the same as PnP. UPnP is used for network connectivity via TCP/IP to various devices (scanners, printers, etc.).</li> </ul> Win:5.1

## Operators

Key phrase	Return Type	Description
<firewall service type> = <firewall service type>	<boolean>	Compares two firewall service types for equality only. Win:5.1

## Examples

- names of services of current profile of local policy of firewall
- Returns the names of the services of the current profile of the firewall's local policy, such as File and Printer Sharing, UPnP Framework & Remote Desktop.

---

## Firewall Icmp Settings

These Inspectors provide access to the settings controlling Internet Control Message Protocol (ICMP) packets. These properties correspond to the INetFwIcmpSettings interface in the Windows Firewall API. The Windows Firewall is supported on Windows XP SP2. For more information, search for 'Windows Firewall API' at the msdn site (<http://msdn.microsoft.com/>).

## Creation Methods

Key Phrase	Form	Description
icmp settings of <firewall profile>	<i>Plain</i>	The Microsoft Windows Firewall INetFwIcmpSettings interface provides access to the settings controlling Internet Control Message Protocol (ICMP) packets. Win:5.1

## Properties

Key Phrase	Form	Return Type	Description
allow inbound echo request of <firewall icmp settings>	<i>Plain</i>	<boolean>	Returns the value of the AllowInboundEchoRequest property. Type common to IPv4 and IPv6.  Win:5.1
allow inbound mask request of <firewall icmp settings>	<i>Plain</i>	<boolean>	Returns the value of the AllowInboundMaskRequest property. Type common to IPv4 only.  Win:5.1
allow inbound router request of <firewall icmp settings>	<i>Plain</i>	<boolean>	Returns the value of the AllowInboundRouterRequest property. Type common to IPv4 only.  Win:5.1
allow inbound timestamp request of <firewall icmp settings>	<i>Plain</i>	<boolean>	Returns the value of the AllowInboundTimestampRequest property. Type common to IPv4 only.  Win:5.1
allow outbound destination unreachable of <firewall icmp settings>	<i>Plain</i>	<boolean>	Returns the value of the AllowOutboundDestinationUnreachable property. Type common to IPv4 and IPv6.  Win:5.1
allow outbound packet too big of <firewall icmp settings>	<i>Plain</i>	<boolean>	Returns the value of the AllowOutboundPacketTooBig property. Type common to IPv6 only.  Win:5.1
allow outbound parameter problem of <firewall icmp settings>	<i>Plain</i>	<boolean>	Returns the value of the AllowOutboundParameterProblem property. Type common to IPv4 and IPv6.  Win:5.1
allow outbound source quench of <firewall icmp settings>	<i>Plain</i>	<boolean>	Returns the value of the AllowOutboundSourceQuench property. Type common to IPv4 only.  Win:5.1

Key Phrase	Form	Return Type	Description
allow outbound time exceeded of <firewall icmp settings>	<i>Plain</i>	<boolean>	Returns the value of the AllowOutboundTimeExceeded property. Type common to IPv4 and IPv6.  Win:5.1
allow redirect of <firewall icmp settings>	<i>Plain</i>	<boolean>	Accesses the AllowRedirect property. Type common to IPv4 and IPv6.  Win:5.1

## Firewall Remote Admin Settings

These Inspectors provide access to the settings that control remote administration. These properties correspond to the INetFwRemoteAdminSettings interface in the Windows Firewall API. The Windows Firewall is supported on Windows XP SP2. For more information, search for 'Windows Firewall API' at the msdn site (<http://msdn.microsoft.com/>).

### Creation Methods

Key Phrase	Form	Description
remote admin settings of <firewall profile>	<i>Plain</i>	Gets the object containing the remote administration settings. These settings include the following properties: <ul style="list-style-type: none"> <li>• Enabled</li> <li>• IpVersion</li> <li>• RemoteAddresses</li> <li>• Scope.</li> </ul> Win:5.1

## Internet Protocol

Returns the firewall internet protocol corresponding to the Microsoft enumerated types, either tcp or udp. The Windows Firewall is supported on Windows XP SP2. For more information, search for 'Windows Firewall API' at the msdn site (<http://msdn.microsoft.com/>).

### Creation Methods

Key Phrase	Form	Description
internet protocol <integer>	<i>NumberedGlobal</i>	Returns the firewall internet protocol specified, either tcp or udp. These correspond to the Microsoft Windows Firewall enumerated types: <ul style="list-style-type: none"> <li>• NET_FW_IP_PROTOCOL_TCP</li> <li>• NET_FW_IP_PROTOCOL_UDP.</li> </ul> Win:5.1
protocol of <firewall open port>	<i>Plain</i>	Returns the Protocol property of the firewall open port. Win:5.1
tcp	<i>PlainGlobal</i>	Returns an internet protocol corresponding to the Microsoft Windows Firewall enumerated type: NET_FW_IP_PROTOCOL_TCP. Win:5.1
udp	<i>PlainGlobal</i>	Returns an internet protocol corresponding to the Microsoft Windows Firewall enumerated type: NET_FW_IP_PROTOCOL_UDP. Win:5.1

### Operators

Key phrase	Return Type	Description
<internet protocol> = <internet protocol>	<i>&lt;boolean&gt;</i>	Compares two firewall internet protocols for equality only. Win:5.1

### Examples

■ exists globally open port whose (port of it = 52311 and protocol of it = udp and enabled of it) of current profile of local policy of firewall

► Returns TRUE if the BES Client can receive pings.

## Ip Version

Returns the firewall ip version information corresponding to the Microsoft enumerated types, either ipv4 or ipv6. The Windows Firewall is supported on Windows XP SP2. For more information, search for 'Windows Firewall API' at the msdn site (<http://msdn.microsoft.com/>).

### Creation Methods

Key Phrase	Form	Description
any ip version	<i>PlainGlobal</i>	Returns a type corresponding to the Microsoft enumerated value NET_FW_IP_VERSION_ANY.  Win:5.1
ip version <integer>	<i>NumberedGlobal</i>	Returns the the IP version for the <integer> port corresponding to the Microsoft enumerated types: <ul style="list-style-type: none"> <li>• NET_FW_IP_VERSION_V4</li> <li>• NET_FW_IP_VERSION_V6</li> <li>• NET_FW_IP_VERSION_ANY</li> <li>• NET_FW_IP_VERSION_MAX.</li> </ul> Win:5.1
ip version of <firewall authorized application>	<i>Plain</i>	Returns the the IP version for the specified firewall authorized application.  Win:5.1
ip version of <firewall open port>	<i>Plain</i>	Returns the the IP version for the specified firewall open port.  Win:5.1
ip version of <firewall service>	<i>Plain</i>	Returns the the IP version for the specified firewall service.  Win:5.1
ipv4	<i>PlainGlobal</i>	Provides a comparison value for a firewall ip version inspector.  Win:5.1
ipv6	<i>PlainGlobal</i>	Provides a comparison value for a firewall ip version inspector.  Win:5.1

## Operators

Key phrase	Return Type	Description
<ip version> = <ip version>	< <i>boolean</i> >	Compares two firewall ip versions for equality only. <small>Win:5.1</small>

## DMI Objects

The Inspectors for the Desktop Management Interface (DMI).

### Dmi B32\_bit\_memory\_error\_information

#### Creation Methods

Key Phrase	Form
b32_bit_memory_error_information <integer> of <dmi>	<i>Numbered</i>
b32_bit_memory_error_information of <dmi>	<i>Plain</i>

#### Properties

Key Phrase	Form	Return Type
device_error_address of <dmi b32_bit_memory_error_information>	<i>Plain</i>	<integer>
error_granularity of <dmi b32_bit_memory_error_information>	<i>Plain</i>	<integer>
error_operation of <dmi b32_bit_memory_error_information>	<i>Plain</i>	<integer>
error_resolution of <dmi b32_bit_memory_error_information>	<i>Plain</i>	<integer>
error_type of <dmi b32_bit_memory_error_information>	<i>Plain</i>	<integer>
length of <dmi b32_bit_memory_error_information>	<i>Plain</i>	<integer>
memory_array_error_address of <dmi b32_bit_memory_error_information>	<i>Plain</i>	<integer>
vendor_syndrome of <dmi b32_bit_memory_error_information>	<i>Plain</i>	<integer>

## Dmi B64\_bit\_memory\_error\_information

### Creation Methods

Key Phrase	Form
b64_bit_memory_error_information <integer> of <dmi>	<i>Numbered</i>
b64_bit_memory_error_information of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
device_error_address of <dmi b64_bit_memory_error_information>	<i>Plain</i>	<integer>
error_granularity of <dmi b64_bit_memory_error_information>	<i>Plain</i>	<integer>
error_operation of <dmi b64_bit_memory_error_information>	<i>Plain</i>	<integer>
error_resolution of <dmi b64_bit_memory_error_information>	<i>Plain</i>	<integer>
error_type of <dmi b64_bit_memory_error_information>	<i>Plain</i>	<integer>
length of <dmi b64_bit_memory_error_information>	<i>Plain</i>	<integer>
memory_array_error_address of <dmi b64_bit_memory_error_information>	<i>Plain</i>	<integer>
vendor_syndrome of <dmi b64_bit_memory_error_information>	<i>Plain</i>	<integer>

## Dmi Base\_board\_information

### Creation Methods

Key Phrase	Form
base_board_information <integer> of <dmi>	<i>Numbered</i>
base_board_information of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
length of <dmi base_board_information>	<i>Plain</i>	<integer>
manufacturer of <dmi base_board_information>	<i>Plain</i>	<string>
product of <dmi base_board_information>	<i>Plain</i>	<string>
serial_number of <dmi base_board_information>	<i>Plain</i>	<string>
version of <dmi base_board_information>	<i>Plain</i>	<string>

## Dmi Bios\_information

### Creation Methods

Key Phrase	Form
bios_information <integer> of <dmi>	<i>Numbered</i>
bios_information of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
bios_characteristics of <dmi bios_information>	<i>Plain</i>	<integer>
bios_release_date of <dmi bios_information>	<i>Plain</i>	<string>
bios_rom_size of <dmi bios_information>	<i>Plain</i>	<integer>
bios_starting_address_segment of <dmi bios_information>	<i>Plain</i>	<integer>
bios_version of <dmi bios_information>	<i>Plain</i>	<string>
length of <dmi bios_information>	<i>Plain</i>	<integer>

Key Phrase	Form	Return Type
vendor of <dmi bios_information>	<i>Plain</i>	<string>

## Dmi Bios\_language\_information

### Creation Methods

Key Phrase	Form
bios_language_information <integer> of <dmi>	<i>Numbered</i>
bios_language_information of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
current_language of <dmi bios_language_information>	<i>Plain</i>	<string>
flags of <dmi bios_language_information>	<i>Plain</i>	<integer>
installable_languages of <dmi bios_language_information>	<i>Plain</i>	<integer>
length of <dmi bios_language_information>	<i>Plain</i>	<integer>

## Dmi Built\_in\_pointing\_device

### Creation Methods

Key Phrase	Form
built_in_pointing_device <integer> of <dmi>	<i>Numbered</i>
built_in_pointing_device of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
interface of <dmi built_in_pointing_device>	<i>Plain</i>	<integer>
length of <dmi built_in_pointing_device>	<i>Plain</i>	<integer>
number_of_buttons of <dmi built_in_pointing_device>	<i>Plain</i>	<integer>
type of <dmi built_in_pointing_device>	<i>Plain</i>	<integer>

## Dmi Cache\_information

### Creation Methods

Key Phrase	Form
cache_information <integer> of <dmi>	<i>Numbered</i>
cache_information of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
associativity of <dmi cache_information>	<i>Plain</i>	<integer>
cache_configuration of <dmi cache_information>	<i>Plain</i>	<integer>
cache_speed of <dmi cache_information>	<i>Plain</i>	<integer>
current_sram_type of <dmi cache_information>	<i>Plain</i>	<integer>
error_correction_type of <dmi cache_information>	<i>Plain</i>	<integer>
installed_size of <dmi cache_information>	<i>Plain</i>	<integer>
length of <dmi cache_information>	<i>Plain</i>	<integer>
maximum_cache_size of <dmi cache_information>	<i>Plain</i>	<integer>
socket_designation of <dmi cache_information>	<i>Plain</i>	<string>
supported_sram_type of <dmi cache_information>	<i>Plain</i>	<integer>
system_cache_type of <dmi cache_information>	<i>Plain</i>	<integer>

## Dmi Cooling\_device

### Creation Methods

Key Phrase	Form
cooling_device <integer> of <dmi>	<i>Numbered</i>
cooling_device of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
cooling_unit_group of <dmi cooling_device>	<i>Plain</i>	<integer>
device_type_and_status of <dmi cooling_device>	<i>Plain</i>	<integer>
length of <dmi cooling_device>	<i>Plain</i>	<integer>
nominal_speed of <dmi cooling_device>	<i>Plain</i>	<integer>
oem_defined of <dmi cooling_device>	<i>Plain</i>	<integer>
temperature_probe_handle of <dmi cooling_device>	<i>Plain</i>	<integer>

## Dmi Electrical\_current\_probe

### Creation Methods

Key Phrase	Form
electrical_current_probe <integer> of <dmi>	<i>Numbered</i>
electrical_current_probe of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
accuracy of <dmi electrical_current_probe>	<i>Plain</i>	<integer>
description of <dmi electrical_current_probe>	<i>Plain</i>	<string>
length of <dmi electrical_current_probe>	<i>Plain</i>	<integer>
location_and_status of <dmi electrical_current_probe>	<i>Plain</i>	<integer>
maximum_value of <dmi	<i>Plain</i>	<integer>

Key Phrase	Form	Return Type
electrical_current_probe>		
minimum_value of <dmi electrical_current_probe>	<i>Plain</i>	<integer>
nominal_value of <dmi electrical_current_probe>	<i>Plain</i>	<integer>
oem_defined of <dmi electrical_current_probe>	<i>Plain</i>	<integer>
resolution of <dmi electrical_current_probe>	<i>Plain</i>	<integer>
tolerance of <dmi electrical_current_probe>	<i>Plain</i>	<integer>

---

## Dmi End\_of\_table

### Creation Methods

Key Phrase	Form
end_of_table <integer> of <dmi>	<i>Numbered</i>
end_of_table of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
length of <dmi end_of_table>	<i>Plain</i>	<integer>

---

## Dmi Group\_associations

### Creation Methods

Key Phrase	Form
group_associations <integer> of <dmi>	<i>Numbered</i>
group_associations of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
group_name of <dmi group_associations>	<i>Plain</i>	<string>
item_handle of <dmi group_associations>	<i>Plain</i>	<integer>
item_type of <dmi group_associations>	<i>Plain</i>	<integer>

Key Phrase	Form	Return Type
length of <dmi group_associations>	<i>Plain</i>	<integer>

## Dmi Hardware\_security

### Creation Methods

Key Phrase	Form
hardware_security <integer> of <dmi>	<i>Numbered</i>
hardware_security of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
hardware_security_settings of <dmi hardware_security>	<i>Plain</i>	<integer>
length of <dmi hardware_security>	<i>Plain</i>	<integer>

## Dmi Inactive

### Creation Methods

Key Phrase	Form
inactive <integer> of <dmi>	<i>Numbered</i>
inactive of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
length of <dmi inactive>	<i>Plain</i>	<integer>

## Dmi Management\_device

### Creation Methods

Key Phrase	Form
management_device <integer> of <dmi>	<i>Numbered</i>
management_device of <dmi>	<i>Plain</i>

## Properties

Key Phrase	Form	Return Type
address of <dmi management_device>	<i>Plain</i>	<integer>
address_type of <dmi management_device>	<i>Plain</i>	<integer>
description of <dmi management_device>	<i>Plain</i>	<string>
length of <dmi management_device>	<i>Plain</i>	<integer>
type of <dmi management_device>	<i>Plain</i>	<integer>

---

## Dmi Management\_device\_component

### Creation Methods

Key Phrase	Form
management_device_component <integer> of <dmi>	<i>Numbered</i>
management_device_component of <dmi>	<i>Plain</i>

## Properties

Key Phrase	Form	Return Type
component_handle of <dmi management_device_component>	<i>Plain</i>	<integer>
description of <dmi management_device_component>	<i>Plain</i>	<string>
length of <dmi management_device_component>	<i>Plain</i>	<integer>
management_device_handle of <dmi management_device_component>	<i>Plain</i>	<integer>
threshold_handle of <dmi management_device_component>	<i>Plain</i>	<integer>

---

## Dmi Management\_device\_threshold\_data

### Creation Methods

Key Phrase	Form
management_device_threshold_data <integer> of <dmi>	<i>Numbered</i>
management_device_threshold_data of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
length of <dmi management_device_threshold_data>	<i>Plain</i>	<integer>
lower_threshold_critical of <dmi management_device_threshold_data>	<i>Plain</i>	<integer>
lower_threshold_non_critical of <dmi management_device_threshold_data>	<i>Plain</i>	<integer>
lower_threshold_non_recoverable of <dmi management_device_threshold_data>	<i>Plain</i>	<integer>
upper_threshold_critical of <dmi management_device_threshold_data>	<i>Plain</i>	<integer>
upper_threshold_non_critical of <dmi management_device_threshold_data>	<i>Plain</i>	<integer>
upper_threshold_non_recoverable of <dmi management_device_threshold_data>	<i>Plain</i>	<integer>

## Dmi Memory\_array\_mapped\_address

### Creation Methods

Key Phrase	Form
memory_array_mapped_address <integer> of <dmi>	<i>Numbered</i>
memory_array_mapped_address of <dmi>	<i>Plain</i>

## Properties

Key Phrase	Form	Return Type
ending_address of <dmi memory_array_mapped_address>	<i>Plain</i>	<integer>
length of <dmi memory_array_mapped_address>	<i>Plain</i>	<integer>
memory_array_handle of <dmi memory_array_mapped_address>	<i>Plain</i>	<integer>
partition_width of <dmi memory_array_mapped_address>	<i>Plain</i>	<integer>
starting_address of <dmi memory_array_mapped_address>	<i>Plain</i>	<integer>

## Dmi Memory\_controller\_information

### Creation Methods

Key Phrase	Form
memory_controller_information <integer> of <dmi>	<i>Numbered</i>
memory_controller_information of <dmi>	<i>Plain</i>

## Properties

Key Phrase	Form	Return Type
current_interleave of <dmi memory_controller_information>	<i>Plain</i>	<integer>
error_correcting_capability of <dmi memory_controller_information>	<i>Plain</i>	<integer>
error_detecting_method of <dmi memory_controller_information>	<i>Plain</i>	<integer>
length of <dmi memory_controller_information>	<i>Plain</i>	<integer>
maximum_memory_module_size of <dmi memory_controller_information>	<i>Plain</i>	<integer>
memory_module_voltage of <dmi memory_controller_information>	<i>Plain</i>	<integer>
number_of_associated_memory_slots of <dmi memory_controller_information>	<i>Plain</i>	<integer>

Key Phrase	Form	Return Type
supported_interleave of <dmi memory_controller_information>	<i>Plain</i>	<integer>
supported_memory_types of <dmi memory_controller_information>	<i>Plain</i>	<integer>
supported_speeds of <dmi memory_controller_information>	<i>Plain</i>	<integer>

## Dmi Memory\_device

### Creation Methods

Key Phrase	Form
memory_device <integer> of <dmi>	<i>Numbered</i>
memory_device of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
bank_locator of <dmi memory_device>	<i>Plain</i>	<string>
data_width of <dmi memory_device>	<i>Plain</i>	<integer>
device_locator of <dmi memory_device>	<i>Plain</i>	<string>
device_set of <dmi memory_device>	<i>Plain</i>	<integer>
form_factor of <dmi memory_device>	<i>Plain</i>	<integer>
length of <dmi memory_device>	<i>Plain</i>	<integer>
memory_array_handle of <dmi memory_device>	<i>Plain</i>	<integer>
memory_error_information_handle of <dmi memory_device>	<i>Plain</i>	<integer>
memory_type of <dmi memory_device>	<i>Plain</i>	<integer>
size of <dmi memory_device>	<i>Plain</i>	<integer>
speed of <dmi memory_device>	<i>Plain</i>	<integer>
total_width of <dmi memory_device>	<i>Plain</i>	<integer>
type_detail of <dmi memory_device>	<i>Plain</i>	<integer>

## Dmi Memory\_device\_mapped\_address

### Creation Methods

Key Phrase	Form
memory_device_mapped_address <integer> of <dmi>	<i>Numbered</i>
memory_device_mapped_address of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
ending_address of <dmi memory_device_mapped_address>	<i>Plain</i>	<integer>
interleave_position of <dmi memory_device_mapped_address>	<i>Plain</i>	<integer>
interleaved_data_depth of <dmi memory_device_mapped_address>	<i>Plain</i>	<integer>
length of <dmi memory_device_mapped_address>	<i>Plain</i>	<integer>
memory_array_mapped_address_handle of <dmi memory_device_mapped_address>	<i>Plain</i>	<integer>
memory_device_handle of <dmi memory_device_mapped_address>	<i>Plain</i>	<integer>
partition_row_position of <dmi memory_device_mapped_address>	<i>Plain</i>	<integer>
starting_address of <dmi memory_device_mapped_address>	<i>Plain</i>	<integer>

## Dmi Memory\_module\_information

### Creation Methods

Key Phrase	Form
memory_module_information <integer> of <dmi>	<i>Numbered</i>
memory_module_information of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
bank_connections of <dmi memory_module_information>	<i>Plain</i>	<integer>
current_memory_type of <dmi memory_module_information>	<i>Plain</i>	<integer>
current_speed of <dmi memory_module_information>	<i>Plain</i>	<integer>
enabled_size of <dmi memory_module_information>	<i>Plain</i>	<integer>
error_status of <dmi memory_module_information>	<i>Plain</i>	<integer>
installed_size of <dmi memory_module_information>	<i>Plain</i>	<integer>
length of <dmi memory_module_information>	<i>Plain</i>	<integer>
socket_designation of <dmi memory_module_information>	<i>Plain</i>	<string>

## Dmi On\_board\_devices\_information

### Creation Methods

Key Phrase	Form
on_board_devices_information <integer> of <dmi>	<i>Numbered</i>
on_board_devices_information of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
device_description <integer> of <dmi on_board_devices_information>	<i>Numbered</i>	<string>
device_description of <dmi on_board_devices_information>	<i>Plain</i>	<string>
device_type <integer> of <dmi on_board_devices_information>	<i>Numbered</i>	<integer>
device_type of <dmi on_board_devices_information>	<i>Plain</i>	<integer>
length of <dmi on_board_devices_information>	<i>Plain</i>	<integer>

## Dmi Out\_of\_band\_remote\_access

### Creation Methods

Key Phrase	Form
out_of_band_remote_access <integer> of <dmi>	<i>Numbered</i>
out_of_band_remote_access of <dmi>	<i>Plain</i>

## Properties

Key Phrase	Form	Return Type
connections of <dmi out_of_band_remote_access>	<i>Plain</i>	<integer>
length of <dmi out_of_band_remote_access>	<i>Plain</i>	<integer>
manufacturer_name of <dmi out_of_band_remote_access>	<i>Plain</i>	<string>

## Dmi Physical\_memory\_array

### Creation Methods

Key Phrase	Form
physical_memory_array <integer> of <dmi>	<i>Numbered</i>
physical_memory_array of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
length of <dmi physical_memory_array>	<i>Plain</i>	<integer>
location of <dmi physical_memory_array>	<i>Plain</i>	<integer>
maximum_capacity of <dmi physical_memory_array>	<i>Plain</i>	<integer>
memory_error_correction of <dmi physical_memory_array>	<i>Plain</i>	<integer>
memory_error_information_handle of <dmi physical_memory_array>	<i>Plain</i>	<integer>
number_of_memory_devices of <dmi physical_memory_array>	<i>Plain</i>	<integer>
use of <dmi physical_memory_array>	<i>Plain</i>	<integer>

## Dmi Port\_connector\_information

### Creation Methods

Key Phrase	Form
port_connector_information <integer> of <dmi>	<i>Numbered</i>
port_connector_information of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
external_connector_type of <dmi port_connector_information>	<i>Plain</i>	<integer>
external_reference_designator of <dmi port_connector_information>	<i>Plain</i>	<string>
internal_connector_type of <dmi port_connector_information>	<i>Plain</i>	<integer>
internal_reference_designator of <dmi port_connector_information>	<i>Plain</i>	<string>
length of <dmi port_connector_information>	<i>Plain</i>	<integer>
port_type of <dmi port_connector_information>	<i>Plain</i>	<integer>

## Dmi Portable\_battery

### Creation Methods

Key Phrase	Form
portable_battery <integer> of <dmi>	<i>Numbered</i>
portable_battery of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
design_capacity of <dmi portable_battery>	<i>Plain</i>	<integer>
design_capacity_multiplier of <dmi portable_battery>	<i>Plain</i>	<integer>
design_voltage of <dmi portable_battery>	<i>Plain</i>	<integer>

Key Phrase	Form	Return Type
device_chemistry of <dmi portable_battery>	<i>Plain</i>	<integer>
device_name of <dmi portable_battery>	<i>Plain</i>	<string>
length of <dmi portable_battery>	<i>Plain</i>	<integer>
location of <dmi portable_battery>	<i>Plain</i>	<string>
manufacture_date of <dmi portable_battery>	<i>Plain</i>	<string>
manufacturer of <dmi portable_battery>	<i>Plain</i>	<string>
maximum_error_in_battery_data of <dmi portable_battery>	<i>Plain</i>	<integer>
oem_specific of <dmi portable_battery>	<i>Plain</i>	<integer>
sbds_device_chemistry of <dmi portable_battery>	<i>Plain</i>	<string>
sbds_manufacture_date of <dmi portable_battery>	<i>Plain</i>	<integer>
sbds_serial_number of <dmi portable_battery>	<i>Plain</i>	<integer>
sbds_version_number of <dmi portable_battery>	<i>Plain</i>	<string>
serial_number of <dmi portable_battery>	<i>Plain</i>	<string>

## Dmi Processor\_information

### Creation Methods

Key Phrase	Form
processor_information <integer> of <dmi>	<i>Numbered</i>
processor_information of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
current_speed of <dmi processor_information>	<i>Plain</i>	<integer>
external_clock of <dmi processor_information>	<i>Plain</i>	<integer>
l1_cache_handle of <dmi processor_information>	<i>Plain</i>	<integer>
l2_cache_handle of <dmi processor_information>	<i>Plain</i>	<integer>

Key Phrase	Form	Return Type
l3_cache_handle of <dmi processor_information>	<i>Plain</i>	<integer>
length of <dmi processor_information>	<i>Plain</i>	<integer>
max_speed of <dmi processor_information>	<i>Plain</i>	<integer>
processor_family of <dmi processor_information>	<i>Plain</i>	<integer>
processor_id of <dmi processor_information>	<i>Plain</i>	<integer>
processor_manufacturer of <dmi processor_information>	<i>Plain</i>	<string>
processor_type of <dmi processor_information>	<i>Plain</i>	<integer>
processor_upgrade of <dmi processor_information>	<i>Plain</i>	<integer>
processor_version of <dmi processor_information>	<i>Plain</i>	<string>
socket_designation of <dmi processor_information>	<i>Plain</i>	<string>
status of <dmi processor_information>	<i>Plain</i>	<integer>
voltage of <dmi processor_information>	<i>Plain</i>	<integer>

## Dmi System\_boot\_information

### Creation Methods

Key Phrase	Form
system_boot_information <integer> of <dmi>	<i>Numbered</i>
system_boot_information of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
length of <dmi system_boot_information>	<i>Plain</i>	<integer>

## Dmi System\_enclosure\_or\_chassis

### Creation Methods

Key Phrase	Form
system_enclosure_or_chassis <integer> of <dmi>	<i>Numbered</i>
system_enclosure_or_chassis of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
asset_tag_number of <dmi system_enclosure_or_chassis>	<i>Plain</i>	<string>
bootup_state of <dmi system_enclosure_or_chassis>	<i>Plain</i>	<integer>
length of <dmi system_enclosure_or_chassis>	<i>Plain</i>	<integer>
manufacturer of <dmi system_enclosure_or_chassis>	<i>Plain</i>	<string>
oem_defined of <dmi system_enclosure_or_chassis>	<i>Plain</i>	<integer>
power_supply_state of <dmi system_enclosure_or_chassis>	<i>Plain</i>	<integer>
security_status of <dmi system_enclosure_or_chassis>	<i>Plain</i>	<integer>
serial_number of <dmi system_enclosure_or_chassis>	<i>Plain</i>	<string>
thermal_state of <dmi system_enclosure_or_chassis>	<i>Plain</i>	<integer>
type of <dmi system_enclosure_or_chassis>	<i>Plain</i>	<integer>
version of <dmi system_enclosure_or_chassis>	<i>Plain</i>	<string>

## Dmi System\_information

### Creation Methods

Key Phrase	Form
system_information <integer> of <dmi>	<i>Numbered</i>
system_information of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
length of <dmi system_information>	<i>Plain</i>	<integer>
manufacturer of <dmi system_information>	<i>Plain</i>	<string>
product_name of <dmi system_information>	<i>Plain</i>	<string>
serial_number of <dmi system_information>	<i>Plain</i>	<string>
version of <dmi system_information>	<i>Plain</i>	<string>
wake_up_type of <dmi system_information>	<i>Plain</i>	<integer>

## Dmi System\_power\_controls

### Creation Methods

Key Phrase	Form
system_power_controls <integer> of <dmi>	<i>Numbered</i>
system_power_controls of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
length of <dmi system_power_controls>	<i>Plain</i>	<integer>
next_scheduled_power_on_day_of_month of <dmi system_power_controls>	<i>Plain</i>	<integer>
next_scheduled_power_on_hour of <dmi system_power_controls>	<i>Plain</i>	<integer>
next_scheduled_power_on_minute of <dmi system_power_controls>	<i>Plain</i>	<integer>

Key Phrase	Form	Return Type
next_scheduled_power_on_month of <dmi system_power_controls>	<i>Plain</i>	<integer>
next_scheduled_power_on_second of <dmi system_power_controls>	<i>Plain</i>	<integer>

## Dmi System\_reset

### Creation Methods

Key Phrase	Form
system_reset <integer> of <dmi>	<i>Numbered</i>
system_reset of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
capabilities of <dmi system_reset>	<i>Plain</i>	<integer>
length of <dmi system_reset>	<i>Plain</i>	<integer>
reset_count of <dmi system_reset>	<i>Plain</i>	<integer>
reset_limit of <dmi system_reset>	<i>Plain</i>	<integer>
timeout of <dmi system_reset>	<i>Plain</i>	<integer>
timer_interval of <dmi system_reset>	<i>Plain</i>	<integer>

## Dmi System\_slots

### Creation Methods

Key Phrase	Form
system_slots <integer> of <dmi>	<i>Numbered</i>
system_slots of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
current_usage of <dmi system_slots>	<i>Plain</i>	<integer>

Key Phrase	Form	Return Type
length of <dmi system_slots>	<i>Plain</i>	<integer>
slot_characteristics_1 of <dmi system_slots>	<i>Plain</i>	<integer>
slot_characteristics_2 of <dmi system_slots>	<i>Plain</i>	<integer>
slot_data_bus_width of <dmi system_slots>	<i>Plain</i>	<integer>
slot_designation of <dmi system_slots>	<i>Plain</i>	<string>
slot_id of <dmi system_slots>	<i>Plain</i>	<integer>
slot_length of <dmi system_slots>	<i>Plain</i>	<integer>
slot_type of <dmi system_slots>	<i>Plain</i>	<integer>

## Dmi Temperature\_probe

### Creation Methods

Key Phrase	Form
temperature_probe <integer> of <dmi>	<i>Numbered</i>
temperature_probe of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
accuracy of <dmi temperature_probe>	<i>Plain</i>	<integer>
description of <dmi temperature_probe>	<i>Plain</i>	<string>
length of <dmi temperature_probe>	<i>Plain</i>	<integer>
location_and_status of <dmi temperature_probe>	<i>Plain</i>	<integer>
maximum_value of <dmi temperature_probe>	<i>Plain</i>	<integer>
minimum_value of <dmi temperature_probe>	<i>Plain</i>	<integer>
nominal_value of <dmi temperature_probe>	<i>Plain</i>	<integer>
oem_defined of <dmi temperature_probe>	<i>Plain</i>	<integer>
resolution of <dmi temperature_probe>	<i>Plain</i>	<integer>
tolerance of <dmi temperature_probe>	<i>Plain</i>	<integer>

## Dmi Voltage\_probe

### Creation Methods

Key Phrase	Form
voltage_probe <integer> of <dmi>	<i>Numbered</i>
voltage_probe of <dmi>	<i>Plain</i>

### Properties

Key Phrase	Form	Return Type
accuracy of <dmi voltage_probe>	<i>Plain</i>	<integer>
description of <dmi voltage_probe>	<i>Plain</i>	<string>
length of <dmi voltage_probe>	<i>Plain</i>	<integer>
location_and_status of <dmi voltage_probe>	<i>Plain</i>	<integer>
maximum_value of <dmi voltage_probe>	<i>Plain</i>	<integer>
minimum_value of <dmi voltage_probe>	<i>Plain</i>	<integer>
nominal_value of <dmi voltage_probe>	<i>Plain</i>	<integer>
oem_defined of <dmi voltage_probe>	<i>Plain</i>	<integer>
resolution of <dmi voltage_probe>	<i>Plain</i>	<integer>
tolerance of <dmi voltage_probe>	<i>Plain</i>	<integer>

# WMI Objects

The Inspectors for Windows Management Instrumentation (WMI).

## Wmi

A wmi object provides access to the WMI (Windows Management Interface) query facility. This object provides access to an enormous amount of configuration data.

### Creation Methods

Key Phrase	Form	Description
full wmi <string>	<i>NamedGlobal</i>	Returns a wmi object which can retrieve all values, including system values.  Win:2.0
wmi	<i>PlainGlobal</i>	Returns the wmi object corresponding to the "root\cimv2" namespace.  Win:2.0
wmi <string>	<i>NamedGlobal</i>	Returns the wmi object corresponding to the namespace string provided.  Win:2.0

### Properties

Key Phrase	Form	Return Type	Description
select <string> of <wmi>	<i>Named</i>	<wmi select>	Returns the wmi select whose name matches the string provided.  Win:2.0
select object <string> of <wmi>	<i>Named</i>	<wmi object>	Returns the desired property (specified by <string>) from the given wmi object.  Win:6.0

## Wmi Select

A value returned as a result of a WMI select query. Be aware that microsoft moves their web content around from time to time, but right now you can find some good references at: [http://msdn.microsoft.com/library/en-us/wmisdk/wmi/wmi\\_classes.asp](http://msdn.microsoft.com/library/en-us/wmisdk/wmi/wmi_classes.asp). Please read the inspector documentation for accessing other 'namespaces' of the wmi. To get the asset tag from a dell, use: Q: string value of select "SerialNumber from Win32\_systemenclosure" of wmi.

### Creation Methods

Key Phrase	Form	Description
property <string> of <wmi object>	<i>Named</i>	Returns the Inspector properties of the form <string> of <wmi object>.  Win:6.0
property of <wmi object>	<i>Plain</i>	Returns the Inspector properties of the specified wmi object.  Win:6.0
select <string> of <wmi>	<i>Named</i>	Returns the wmi select whose name matches the string provided.  Win:2.0

## Note

Here are a few other examples of using the wmi Inspectors. Each of the examples below hands back dozens of settings:

Q: selects "\*" from Win32\_ComputerSystem" of wmi  
 Q: selects "\*" from win32\_keyboard" of wmi  
 Q: selects "\*" from win32\_CDROMDrive" of wmi  
 Q: selects "\*" from win32\_DiskDrive" of wmi  
 Q: selects "\*" from win32\_BIOS" of wmi  
 Q: selects "\*" from win32\_CacheMemory" of wmi  
 Q: selects "\*" from win32\_DMIOChannel" of wmi  
 Q: selects "\*" from win32\_FloppyController" of wmi  
 Q: selects "\*" from win32\_IDEController" of wmi  
 Q: selects "\*" from win32\_IRQResource" of wmi  
 Q: selects "\*" from win32\_MemoryDevice" of wmi  
 Q: selects "\*" from win32\_MotherboardDevice" of wmi  
 Q: selects "\*" from win32\_ParallelPort" of wmi  
 Q: selects "\*" from Win32\_PNPDevice" of wmi  
 Q: selects "\*" from win32\_Processor" of wmi  
 Q: selects "\*" from win32\_SerialPort" of wmi  
 Q: selects "\*" from win32\_SoundDevice" of wmi  
 Q: selects "\*" from win32\_NetworkAdapter" of wmi  
 Q: selects "\*" from win32\_NetworkAdapterSetting" of wmi  
 Q: selects "\*" from win32\_Battery" of wmi  
 Q: selects "\*" from win32\_PrinterPrinterDriver" of wmi  
 Q: selects "\*" from win32\_PrinterSetting" of wmi

## Properties

Key Phrase	Form	Return Type	Description
<wmi select> as string	<i>Cast</i>	<string>	Returns a string formatted as <name>=<value> for the wmi select.  Win:2.0
boolean value <integer> of <wmi select>	<i>Numbered</i>	<boolean>	Returns the WMI value cast to boolean.  Win:4.1
boolean value of <wmi select>	<i>Plain</i>	<boolean>	Returns the boolean value of a <wmi select> object (exists only for boolean objects).  Win:2.0
integer value <integer> of <wmi select>	<i>Numbered</i>	<integer>	Returns the WMI value cast to an integer.  Win:4.1

Key Phrase	Form	Return Type	Description
integer value of <wmi select>	<i>Plain</i>	<integer>	Returns the integer value of a <wmi select> object whose value is of type integer.  Win:2.0
name of <wmi select>	<i>Plain</i>	<string>	Returns the name of the wmi select.  Win:2.0
string value <integer> of <wmi select>	<i>Numbered</i>	<string>	Returns the numbered string of a WMI array value.  Win:4.1
string value of <wmi select>	<i>Plain</i>	<string>	Returns the string value of the wmi select.  Win:2.0
time value <integer> of <wmi select>	<i>Numbered</i>	<time>	Returns the numbered time of a WMI array value.  Win:4.1
time value of <wmi select>	<i>Plain</i>	<time>	Returns the time value of the wmi select whose value is of type time.  Win:2.0
type of <wmi select>	<i>Plain</i>	<integer>	Returns the type of data stored in the wmi select value. Possible values are: <ul style="list-style-type: none"> <li>• 8 (VT_BSTR)</li> <li>• 11 (VT_BOOL)</li> <li>• 22 (VT_INT)</li> <li>• (See MSDN documentation for the meaning of additional values.).</li> </ul> Win:2.0

## Wmi Object

These Inspectors allow you to analyze the properties of WMI objects.

### Creation Methods

Key Phrase	Form	Description
select object <string> of <wmi>	<i>Named</i>	Returns the desired property (specified by <string>) from the given wmi object.  Win:6.0

### Properties

Key Phrase	Form	Return Type	Description
<wmi object> as string	<i>Cast</i>	<string>	Casts the given wmi object as a string type.  Win:6.0
property <string> of <wmi object>	<i>Named</i>	<wmi select>	Returns the Inspector properties of the form <string> of <wmi object>.  Win:6.0
property of <wmi object>	<i>Plain</i>	<wmi select>	Returns the Inspector properties of the specified wmi object.  Win:6.0

### Examples

- Property "SerialNumber" of select object "\*" from win32\_operatingsystem" of wmi
  - ▶ Returns serial number of the selected wmi object, in a form like SerialNumber=76487-OEM-0012903-00925.
- properties "select" of type "wmi"
  - ▶ Returns Inspector properties of the form "select of <wmi>", in this case,
    - selects <string> of <wmi>: wmi select.

# Site Objects

These keywords query the properties of Fixlet sites to which the client is subscribed.

## Site

A Site object is provided to access properties of Fixlet sites.

### Creation Methods

Key Phrase	Form	Description
current site	<i>PlainGlobal</i>	Creates the site object corresponding to the site that provided the current Fixlet. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
site	<i>PlainGlobal</i>	Iterates through all the sites. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
site <string>	<i>NamedGlobal</i>	Creates the site object that corresponds to the name provided. The name is interpreted as a site locator and is therefore a URL. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

### Properties

Key Phrase	Form	Return Type	Description
client folder of <site>	<i>Plain</i>	<folder>	The folder containing the site content on the client machine. Site content is gathered into this location. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
fixlet of <site>	<i>Plain</i>	<fixlet>	Iterates through the Fixlet messages of the specified site. <small>Win:5.0, Lin:5.0, Sol:5.0, HPUX:5.0, AIX:5.0</small>
gather schedule authority of <site>	<i>Plain</i>	<string>	Returns a string corresponding to the authority of the site schedule, e.g.: Publisher, Custom, Manual or Disabled. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
gather schedule time interval of <site>	<i>Plain</i>	<time interval>	Returns the time interval between automatic gathering of site content. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

Key Phrase	Form	Return Type	Description
group <integer> of <site>	<i>Numbered</i>	<site group>	Returns an object corresponding to the numbered group of the specified site. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
last gather time of <site>	<i>Plain</i>	<time>	Returns the time of last successful gathering from the site. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
masthead of <site>	<i>Plain</i>	<file>	Each site has a masthead, and the masthead is saved into the site data folder upon successful creation. This property returns a file object that corresponds to the copy in the site data folder. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
name of <site>	<i>Plain</i>	<string>	The name of the site. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
relevant fixlet of <site>	<i>Plain</i>	<fixlet>	Iterates through the Relevant Fixlet messages for the specified site. <small>Win:5.0, Lin:5.0, Sol:5.0, HPUX:5.0, AIX:5.0</small>
setting <string> of <site>	<i>Named</i>	<setting>	Returns the setting whose name matches the string provided from the Fixlet site settings. <small>Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
setting of <site>	<i>Plain</i>	<setting>	Returns one or more settings from the site settings. <small>Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
site tag of <site>	<i>Plain</i>	<string>	Returns the last component of the specified site's url, eg. 'actionsite', 'enterprisesecurity', etc. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
subscribe time of <site>	<i>Plain</i>	<time>	Returns the time that the current machine began subscribing to the site. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
type of <site>	<i>Plain</i>	<string>	Returns one of the following 4 literal strings: <ul style="list-style-type: none"> <li>• Master Action Site</li> <li>• Operator Site</li> <li>• Custom Site</li> <li>• Fixlet Site.</li> </ul> <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

Key Phrase	Form	Return Type	Description
url of <site>	<i>Plain</i>	<string>	Returns the Locator found in the masthead. A site locator is used to synchronize with the site. It normally contains the URL of a remote file system folder, or the URL of a cgi-bin program that provides a remote directory listing of the site.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
version of <site>	<i>Plain</i>	<integer>	Returns the version number of the site content.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

## Note

The 'as string' property yields a string formatted with the site name.

## Examples

- exists site "actionsite"
- ▶ TRUE when the action site exists on the target machine.
  
- exists file "siteicon.bmp" of client folder of current site
- ▶ TRUE if the specified file exists in the client folder.
  
- last gather time of current site > now - 30 \* day
- ▶ Return TRUE if it has been over 30 days since last gathering, or synchronizing, with the site.
  
- last gather time of current site < time "4 Aug 1997 01:00 pdt"
- ▶ Returns TRUE if the site was last synchronized before the specified date.
  
- modification time of masthead of current site < time "4 Aug 1997 01:00 pdt"
- ▶ Returns TRUE if the masthead of the current site is older than the specified date.

---

## Site Group

These Inspectors return information on the automatic groups defined for a given site.

### Creation Methods

Key Phrase	Form	Description
group <integer> of <site>	<i>Numbered</i>	Returns an object corresponding to the numbered group of the specified site.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

### Properties

Key Phrase	Form	Return Type	Description
id of <site group>	<i>Plain</i>	<integer>	Returns the numeric ID of the specified site group. This is the number assigned to an automatic group when it is first defined.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
member of <site group>	<i>Plain</i>	<boolean>	Returns TRUE if the current computer is a member of the specified group.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

---

## Fixlet

These Inspectors can provide important information about the Fixlet messages at any site. These Inspectors only work in the context of property evaluation, not Fixlet evaluation.

### Creation Methods

Key Phrase	Form	Description
fixlet of <site>	<i>Plain</i>	This Inspector iterates over all the Fixlet messages in the given site.  Win:5.0, Lin:5.0, Sol:5.0, HPUX:5.0, AIX:5.0
relevant fixlet of <site>	<i>Plain</i>	Iterates over all the relevant Fixlet messages in the specified site.  Win:5.0, Lin:5.0, Sol:5.0, HPUX:5.0, AIX:5.0

## Properties

Key Phrase	Form	Return Type	Description
header <string> of <fixlet>	<i>Named</i>	<fixlet_header>	Returns the named header (case insensitive) of the specified Fixlet message. Fixlet headers are name:value pairs.  Win:5.0, Lin:5.0, Sol:5.0, HPUX:5.0, AIX:5.0
header of <fixlet>	<i>Plain</i>	<fixlet_header>	Iterates over all the headers of the Fixlet message.  Win:5.0, Lin:5.0, Sol:5.0, HPUX:5.0, AIX:5.0
id of <fixlet>	<i>Plain</i>	<integer>	Returns the ID number of the specified Fixlet message.  Win:5.0, Lin:5.0, Sol:5.0, HPUX:5.0, AIX:5.0
relevance of <fixlet>	<i>Plain</i>	<boolean>	Returns a boolean TRUE or False, depending on the Relevance of the specified Fixlet message.  Win:5.0, Lin:5.0, Sol:5.0, HPUX:5.0, AIX:5.0

## Client Objects

These Inspectors retrieve information about the organization of the BES Client computers.

### Client

The client object allows access to properties of the client application hosting the relevance evaluation, typically a BigFix program. In addition, the client maintains a collection of settings with both name and value properties that are inspectable using the client object. Client objects are derived from file objects, so they share all the file properties.

### Creation Methods

Key Phrase	Form	Description
client	<i>PlainGlobal</i>	Returns the client object corresponding to the BigFix application evaluating the current relevance expression.  Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

### Properties

Key Phrase	Form	Return Type	Description
administrator <string> of <client>	<i>Named</i>	<setting>	If the administrator named in the <string> is enabled on the given <client> computer, this property returns a setting with the given name and the value 'allow.' For instance, if the name of the administrator is joe_admin, then the client would return a setting object with the name 'joe_admin' and a value of 'allow'. Casting this as a string would return 'joe_admin=allow'.  Win:3.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1
administrator of <client>	<i>Plain</i>	<setting>	Creates a setting object corresponding to the administrators of the given <client>.  Win:3.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1
setting <string> of <client>	<i>Named</i>	<setting>	Returns a client setting whose name matches the string provided from the client settings.  Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

Key Phrase	Form	Return Type	Description
setting of <client>	<i>Plain</i>	<setting>	Returns one or more settings from the client settings.  Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

## Setting

A setting is a simple object with name and value properties. It is a property of a client, or a property of a site. Settings of a site are assigned by site authors. Settings of the client are assigned by the BES console operator.

### Creation Methods

Key Phrase	Form	Description
administrator <string> of <client>	<i>Named</i>	Creates a setting with the given name on the given <client> computer.  Win:3.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1
administrator of <client>	<i>Plain</i>	Creates a setting object consisting of the administrator for the given <client> computer.  Win:3.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1
setting <string> of <client>	<i>Named</i>	Returns the setting whose name matches the string provided from the client settings.  Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
setting <string> of <site>	<i>Named</i>	Returns the setting whose name matches the string provided from the site settings.  Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
setting of <client>	<i>Plain</i>	Returns one or more settings from the client settings.  Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
setting of <site>	<i>Plain</i>	Returns one or more settings from the site settings.  Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

## Properties

Key Phrase	Form	Return Type	Description
<setting> as string	<i>Cast</i>	<string>	Returns a string formatted as <name>=<value> for the setting. <small>Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
effective date of <setting>	<i>Plain</i>	<time>	Returns the date when the setting was last modified. <small>Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
name of <setting>	<i>Plain</i>	<string>	Returns the name of the setting. <small>Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
value of <setting>	<i>Plain</i>	<string>	Returns the value of the setting. <small>Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

## Examples

- names of settings of site "actionsite"
- Returns the names of all the settings of the site named "actionsite".

---

## Selected Server

The BES Server or BES Relay to which the BigFix agent or client reports.

## Creation Methods

Key Phrase	Form	Description
selected server	<i>PlainGlobal</i>	The BES Server or BES Relay to which the agent reports. Returned as the "selected server" type. <small>Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>

## Properties

Key Phrase	Form	Return Type	Description
competition size of <selected server>	<i>Plain</i>	<integer>	The number of servers in the competition from which this server was selected.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
competition weight of <selected server>	<i>Plain</i>	<integer>	The total of the weights of the servers in the competition from which this server was selected.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
distance of <selected server>	<i>Plain</i>	<integer range>	The distance, in IP gateway hops, to the server. Among servers with the same priority, closer servers are preferred. Returns an integer range, since the exact distance may not be known.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
gateway address <integer> of <selected server>	<i>Numbered</i>	<ipv4 address>	The ip address of a gateway between the agent and the selected server at the given distance from the agent, if known.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
gateway address of <selected server>	<i>Plain</i>	<ipv4 address>	All known ip addresses of gateways between the agent and the selected server.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
ip address of <selected server>	<i>Plain</i>	<ipv4 address>	The ip address to which reports are sent.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
name of <selected server>	<i>Plain</i>	<string>	The DNS name of the server, if known.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
port number of <selected server>	<i>Plain</i>	<integer>	The port number to which reports are sent.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
priority of <selected server>	<i>Plain</i>	<integer>	The priority assigned to the server by the BES console. Servers with low priorities are preferred to servers with high priority.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
weight of <selected server>	<i>Plain</i>	<integer>	The weight assigned to the server by the BES console. Servers with the same priority and approximate distance compete to be chosen; servers with higher weights are more likely to be chosen.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1

## Operating System Product Type

These Inspectors return the product type of the operating system, which includes Workstations, Domain Controllers and Servers.

### Creation Methods

Key Phrase	Form	Description
nt domain controller product type	<i>PlainGlobal</i>	Returns an object corresponding to OS product type of nt domain controller.  Win:6.0
nt server product type	<i>PlainGlobal</i>	Returns an object corresponding to OS product type of nt server.  Win:6.0
nt workstation product type	<i>PlainGlobal</i>	Returns an object corresponding to OS product type of nt workstation.  Win:6.0
operating system product type <integer>	<i>NumberedGlobal</i>	Returns an object corresponding to the numbered OS product type.  Win:6.0
product type of <operating system>	<i>Plain</i>	Returns the product type of the operating system, which includes Workstations, Domain Controllers and Servers.  Win:6.0

### Operators

Key phrase	Return Type	Description
<operating system product type> = <operating system product type>	< <i>boolean</i> >	Compare two operating system product types for equality.  Win:6.0

## Operating System Suite Mask

These Inspectors provide detailed information about the operating system version.

### Creation Methods

Key Phrase	Form	Description
suite mask of <operating system>	<i>Plain</i>	Returns the bit-mapped suite mask for the operating system, which contains further fine-grain information about the version.  Win:6.0

### Examples

- suite mask of operating system
- Returns the suite mask for the operating system.

## Local Group

These Inspectors return information on local groups as defined on the local BES Client computer using the windows NetLocalGroupEnum API, one of Windows Network Management Functions. Local groups have names, comments, members and security IDs.

### Creation Methods

Key Phrase	Form	Description
local group	<i>PlainGlobal</i>	Returns local groups defined on the local computer using the windows NetLocalGroupEnum API. Several local groups are defined simply by a default operating system install, and have names such as Administrators, Backup Operators, Guests, Network Configuration Operators, Power users, Users, etc. Some software applications also define local groups in order to help manage protections.  Win:6.0
local group <string>	<i>NamedGlobal</i>	Returns a local group corresponding to the given name, such as Administrator, Guests, etc.  Win:6.0

## Properties

Key Phrase	Form	Return Type	Description
comment of <local group>	<i>Plain</i>	<string>	Returns a string containing a comment associated with the specified local group (Administrator, Guest, Users).  Win:6.0
member of <local group>	<i>Plain</i>	<local group member>	Returns a list of the members of the specified local group.  Win:6.0
name of <local group>	<i>Plain</i>	<string>	Returns a list if the names of the local group.  Win:6.0

## Examples

- `sids of members of local group "Administrators"`
  - ▶ Returns a list of the member security IDs of the local administrators group.
- `comment of local group "Administrators"`
  - ▶ Returns the string "Administrators have complete and unrestricted access to the computer/domain".
- `names of local groups`
  - ▶ Returns a list of the local groups, such as Administrators, Guests, Users, etc.

---

## Local Group Member

These Inspectors return information (such as security IDs) on members of local groups as defined on the local BES Client computer using the windows NetLocalGroupEnum API, one of Windows Network Management Functions.

## Creation Methods

Key Phrase	Form	Description
member of <local group>	<i>Plain</i>	Returns an object corresponding to a member of the specified local group.  Win:6.0

## Properties

Key Phrase	Form	Return Type	Description
<local group member> as string	<i>Cast</i>	<string>	Casts a local group member as a string. <small>Win:6.0</small>
sid of <local group member>	<i>Plain</i>	<security identifier>	Returns the security ID for the specified local group member. <small>Win:6.0</small>

## Examples

- members of local group "Administrators"
- Returns a list of the members of the local administration group.

---

## Event Log

These Inspectors return information about the specified Windows Event logs, including the System, Security and the Application log.

## Creation Methods

Key Phrase	Form	Description
application event log	<i>PlainGlobal</i>	Returns an object corresponding to an application event log, one of the event logs created by most Windows systems. <small>Win:6.0</small>
event log <string>	<i>NamedGlobal</i>	Returns the event log object with the specified name. <small>Win:6.0</small>
security event log	<i>PlainGlobal</i>	Returns an event log object for the security event log. <small>Win:6.0</small>
system event log	<i>PlainGlobal</i>	Returns a system event log, which records OS or component events, such as the failure of a bootup service. <small>Win:6.0</small>

## Properties

Key Phrase	Form	Return Type	Description
oldest record number of <event log>	<i>Plain</i>	<integer>	Returns an integer corresponding to the oldest record number on the Client computer's event log.  Win:6.0
record <integer> of <event log>	<i>Numbered</i>	<event log record>	Returns the nth record corresponding to the specified event log.  Win:6.0
record count of <event log>	<i>Plain</i>	<integer>	Returns the record count for the specified event log.  Win:6.0
record of <event log>	<i>Plain</i>	<event log record>	Returns the record corresponding to the specified event log, for instance the application or system event log.  Win:6.0

## Examples

- exists application event log
  - ▶ Returns TRUE if the application event log exists on this computer.
- exists event log "Application"
  - ▶ Returns TRUE if the application event log exists on this computer.
- oldest record number of application event log
  - ▶ Returns the number of the oldest record in the application event log. This is not the same as the record count.
- exists record (oldest record number of it) of application event log
  - ▶ Returns TRUE if there is an oldest record in the application event log.
- record count of application event log
  - ▶ Returns the current record count of the application event log.

## Event Log Record

These Inspectors return individual records from the Windows Event logs, which record information about operating system events.

### Creation Methods

Key Phrase	Form	Description
record <integer> of <event log>	<i>Numbered</i>	Returns the nth record corresponding to the specified event log.  Win:6.0
record of <event log>	<i>Plain</i>	Retrieves the record from the event log.  Win:6.0

### Properties

Key Phrase	Form	Return Type	Description
category of <event log record>	<i>Plain</i>	<integer>	Returns the category of the specified event log, which is sometimes used to further describe the related action.  Win:6.0
computer of <event log record>	<i>Plain</i>	<string>	Returns the name of the computer that has entered a record in the specified log.  Win:6.0
description of <event log record>	<i>Plain</i>	<string>	Returns a human-readable description of the specified event log record.  Win:6.0
event id of <event log record>	<i>Plain</i>	<integer>	Returns an integer corresponding to the ID of the specified record in the Client computer's event log.  Win:6.0
event type of <event log record>	<i>Plain</i>	<event log event type>	Returns the type of the specified event log record, such as error, warning, information, etc.  Win:6.0
length of <event log record>	<i>Plain</i>	<integer>	Returns the length of the specified record. This is not the same as the length of the description.  Win:6.0

Key Phrase	Form	Return Type	Description
record number of <event log record>	<i>Plain</i>	<integer>	Returns the integer value of the record number corresponding to the specified event log record.  Win:6.0
source of <event log record>	<i>Plain</i>	<string>	Returns the source name (from the application, service, or component that logged the event) of the specified event log record.  Win:6.0
time generated of <event log record>	<i>Plain</i>	<time>	Returns the time that the specified event log record was generated.  Win:6.0
time written of <event log record>	<i>Plain</i>	<time>	Returns the time that the specified event record was written to the log.  Win:6.0
user sid of <event log record>	<i>Plain</i>	<security identifier>	Returns the user security ID for the specified record in the event log.  Win:6.0

## Examples

- exists record (oldest record number of it) of application event log
  - ▶ Returns TRUE if there is an oldest record in the application event log.
- exists records of application event log
  - ▶ Returns TRUE if the application event log contains any records.
- category of record (oldest record number of it) of application event log
  - ▶ Returns the category of the oldest record of the application event log.
- computer of record (oldest record number of it) of application event log
  - ▶ Returns the name of the computer that logged the last entry in the application event log.
- description of record (oldest record number of it) of application event log
  - ▶ Returns a description of the oldest record in the application event log. Typically includes a description of the programs and what happened for each entry in the event log.

■ event id of record (oldest record number of it) of application event log

► Returns the id number of the oldest record in the application event log.

■ length of record (oldest record number of it) of application event log

► Returns the length of the specified record in the application event log.

■ record number of record (oldest record number of it) of application event log

► Returns the integer record number corresponding to the oldest record in the application event log.

■ source of record (oldest record number of it) of application event log

► Returns name of the source of the oldest record in the application event log.

■ time generated of record (oldest record number of it) of application event log

► Returns the time (in day, month, year, time, zone format) that the oldest record in the application event log was generated.

■ time written of record (oldest record number of it) of application event log

► Returns the time (in day, month, year, time, zone format) that the oldest record in the application event log was written.

■ user sid of record (oldest record number of it) of application event log

► Returns the user security ID for the oldest record in the application event log, for instance NT AUTHORITY\SYSTEM.

## Event Log Event Type

These Inspectors return information about the types of Windows Event log entries, which record various operating system events including errors, warnings and general information.

### Creation Methods

Key Phrase	Form	Description
audit failure event log event type	<i>PlainGlobal</i>	Returns an object corresponding to an audit failure -- an event related to the failed execution of an action.  Win:6.0
audit success event log event type	<i>PlainGlobal</i>	Returns an object corresponding to an audit success in an event log.  Win:6.0
error event log event type	<i>PlainGlobal</i>	Returns an object corresponding to an error event in the log, such as the failure of a service to start.  Win:6.0
event log event type <integer>	<i>NumberedGlobal</i>	Returns an event type object corresponding to the specified number. The enumerated types include: <ul style="list-style-type: none"> <li>• 1: error event</li> <li>• 2: warning event</li> <li>• 4: information event</li> <li>• 8: audit success event</li> <li>• 16: audit failure event.</li> </ul> Win:6.0
event type of <event log record>	<i>Plain</i>	Returns the event type of the specified record from the event log.  Win:6.0
information event log event type	<i>PlainGlobal</i>	Returns an object corresponding to an information event -- An informational event which is generally related to a successful action.  Win:6.0
warning event log event type	<i>PlainGlobal</i>	Returns an object corresponding to a warning in the event log. Warnings can be used to prevent future system problems.  Win:6.0

## Operators

Key phrase	Return Type	Description
<code>&lt;event log event type&gt; = &lt;event log event type&gt;</code>	<code>&lt;boolean&gt;</code>	Compare two event log event types for equality.  Win:6.0

## Examples

- `audit failure event log event type= event type of record (oldest record number of it) of application event log`
  - Returns TRUE if the oldest record of the application event log contains an audit failure.
- `error event log event type= event type of record (oldest record number of it) of application event log`
  - Returns TRUE if the oldest record of the application event log contains an error.

# Environment Objects

The environment objects are provided to access environment variables. Note that you are inspecting the environment of the application executing the relevance clause (typically the BigFix Agent/Client), which may or may not match the environment of other applications on the computer.

## Environment

Environment variables define a particular set of paths and variables for a computer or an application. These Inspectors let you examine this set.

### Creation Methods

Key Phrase	Form	Description
environment	<i>PlainGlobal</i>	Creates the one and only environment object. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1</small>

### Properties

Key Phrase	Form	Return Type	Description
variable <string> of <environment>	<i>Named</i>	<environment variable>	Returns an environment variable that matches the given name. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1</small>
variable of <environment>	<i>Plain</i>	<environment variable>	Iterates through all the environment variables defined. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1</small>

### Examples

- `exists environment`
- ▶ TRUE if the computer has an environment object.
- `value of variable "path" of environment contains "\extras\"`
- ▶ TRUE if there is an environment variable named "path" and its value contains "\extras\".
- `number of variables of environment`
- ▶ Returns the total number of variables in the environment space.

## Environment Variable

Every variable defined by the environment has both a name and a value. Both names and values are treated as strings.

### Creation Methods

Key Phrase	Form	Description
variable <string> of <environment>	<i>Named</i>	Creates the variable of the environment matching the name provided. The capitalization of the name is ignored.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1
variable of <environment>	<i>Plain</i>	Iterates through all the environment variables defined.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1

### Properties

Key Phrase	Form	Return Type	Description
<environment variable> as string	<i>Cast</i>	<string>	Casting the variable as a string yields a string containing the variable name and the value of the variable separated by ' = '.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1
name of <environment variable>	<i>Plain</i>	<string>	Returns the name of the variable.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1
value of <environment variable>	<i>Plain</i>	<string>	Returns the value of the variable.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:5.1

### Examples

- exists variable "PATH" of environment
- ▶ TRUE if a path variable has been defined in this environment.
- number of variables of environment
- ▶ Returns the total number of variables in this environment.

# Authorization Objects

These inspectors retrieve security and access settings.

## Access Control Entry

An Access Control Entity, or ACE, is an entry in an access control list (ACL). An ACE contains a set of access rights and a security identifier (SID) that identifies a trustee for whom the rights are allowed, denied, or audited.

### Creation Methods

Key Phrase	Form	Description
entry of <access control list>	<i>Plain</i>	Iterates the ACEs of an ACL.  Win:4.1

### Properties

Key Phrase	Form	Return Type	Description
access mode of <access control entry>	<i>Plain</i>	<integer>	For a discretionary ACL (DACL), this flag indicates whether the ACL allows (1) or denies (3) the specified access rights.  Win:4.1
append permission of <access control entry>	<i>Plain</i>	<boolean>	For a file ACE, returns TRUE if the ACE grants or denies append permissions.  Win:4.1
change notification permission of <access control entry>	<i>Plain</i>	<boolean>	For a registry key ACE, returns TRUE if the ACE grants or denies change notification permissions.  Win:4.1
create file permission of <access control entry>	<i>Plain</i>	<boolean>	For a folder ACE, returns TRUE if the ACE grants or denies create file permissions.  Win:4.1
create folder permission of <access control entry>	<i>Plain</i>	<boolean>	For a folder ACE, returns TRUE if the ACE grants or denies create folder permissions.  Win:4.1

Key Phrase	Form	Return Type	Description
create link permission of <access control entry>	<i>Plain</i>	<boolean>	For a registry key ACE, returns TRUE if the ACE grants or denies create key link permissions. <small>Win:4.1</small>
create subkey permission of <access control entry>	<i>Plain</i>	<boolean>	For a registry key ACE, returns TRUE if the ACE grants or denies creation of subkey permissions. <small>Win:4.1</small>
delete child permission of <access control entry>	<i>Plain</i>	<boolean>	For a folder ACE, returns TRUE if the ACE grants or denies child deletion permissions. <small>Win:4.1</small>
delete permission of <access control entry>	<i>Plain</i>	<boolean>	For any ACE, returns TRUE if the ACE grants or generic delete permissions. <small>Win:4.1</small>
enumerate subkeys permission of <access control entry>	<i>Plain</i>	<boolean>	For a registry key ACE, returns TRUE if the ACE grants or enumerate subkey permissions. <small>Win:4.1</small>
execute permission of <access control entry>	<i>Plain</i>	<boolean>	For a file ACE, returns TRUE if the ACE grants or denies execute permissions. <small>Win:4.1</small>
generic all permission of <access control entry>	<i>Plain</i>	<boolean>	For any ACE, returns TRUE if the ACE grants or denies all generic permissions. <small>Win:4.1</small>
generic execute permission of <access control entry>	<i>Plain</i>	<boolean>	For any ACE, returns TRUE if the ACE grants or denies generic execute permissions. <small>Win:4.1</small>
generic read permission of <access control entry>	<i>Plain</i>	<boolean>	For any ACE, returns TRUE if the ACE grants or denies generic read permissions. <small>Win:4.1</small>
generic write permission of <access control entry>	<i>Plain</i>	<boolean>	For any ACE, returns TRUE if the ACE grants or denies generic write permissions. <small>Win:4.1</small>

Key Phrase	Form	Return Type	Description
inheritance of <access control entry>	<i>Plain</i>	<integer>	<p>A set of bit flags that determines whether other containers or objects can inherit the ACE from the primary object to which the ACL is attached. The actual values of the constants are:</p> <ul style="list-style-type: none"> <li>• NO_INHERITANCE = 0</li> <li>• SUB_OBJECTS_ONLY_INHERIT = 1</li> <li>• SUB_CONTAINERS_ONLY_INHERIT = 2</li> <li>• SUB_CONTAINERS_AND_OBJECTS_INHERIT = 3</li> <li>• OBJECT_INHERIT_ACE = 1</li> <li>• CONTAINER_INHERIT_ACE = 2</li> <li>• NO_PROPAGATE_INHERIT_ACE = 4</li> <li>• INHERIT_ONLY_ACE = 8.</li> </ul> <p>Win:4.1</p>
list permission of <access control entry>	<i>Plain</i>	<boolean>	<p>For a folder ACE, returns TRUE if the ACE grants or denies list permissions.</p> <p>Win:4.1</p>
maximum allowed permission of <access control entry>	<i>Plain</i>	<boolean>	<p>For any ACE, returns TRUE if the ACE grants or denies maximum allowed permissions.</p> <p>Win:4.1</p>
query value permission of <access control entry>	<i>Plain</i>	<boolean>	<p>For a registry key ACE, returns TRUE if the ACE grants or denies query value permissions.</p> <p>Win:4.1</p>
read attributes permission of <access control entry>	<i>Plain</i>	<boolean>	<p>For a file or folder ACE, returns TRUE if the ACE grants or denies read attributes permissions.</p> <p>Win:4.1</p>
read control permission of <access control entry>	<i>Plain</i>	<boolean>	<p>For any ACE, returns TRUE if the ACE grants or denies reading access control permissions.</p> <p>Win:4.1</p>
read extended attributes permission of <access control entry>	<i>Plain</i>	<boolean>	<p>For a file or folder ACE, returns TRUE if the ACE grants or denies read extended attributes permissions.</p> <p>Win:4.1</p>
read permission of <access control entry>	<i>Plain</i>	<boolean>	<p>For a file ACE, returns TRUE if the ACE grants or denies read permissions.</p> <p>Win:4.1</p>
set value permission of <access control entry>	<i>Plain</i>	<boolean>	<p>For a registry key ACE, returns TRUE if the ACE grants or denies set value permissions.</p> <p>Win:4.1</p>

Key Phrase	Form	Return Type	Description
synchronize permission of <access control entry>	<i>Plain</i>	<boolean>	For any ACE, returns TRUE if the ACE grants or denies synchronize permissions. <small>Win:4.1</small>
traverse permission of <access control entry>	<i>Plain</i>	<boolean>	For the specified folder ACE, returns TRUE if it grants or denies traverse • folder permission. <small>Win:4.1</small>
trustee of <access control entry>	<i>Plain</i>	<security identifier>	Returns the trustee to whom the specified ACE applies. <small>Win:4.1</small>
trustee type of <access control entry>	<i>Plain</i>	<integer>	Returns the type of trustee to whom the specified ACE applies. <small>Win:4.1</small>
write attributes permission of <access control entry>	<i>Plain</i>	<boolean>	For a file or folder ACE, returns TRUE if the ACE grants or denies write attribute permissions. <small>Win:4.1</small>
write dac permission of <access control entry>	<i>Plain</i>	<boolean>	For any ACE, returns TRUE if the ACE grants or denies write DAC permissions. <small>Win:4.1</small>
write extended attributes permission of <access control entry>	<i>Plain</i>	<boolean>	For a file or folder ACE, returns TRUE if the ACE grants or denies write extended attribute permissions. <small>Win:4.1</small>
write owner permission of <access control entry>	<i>Plain</i>	<boolean>	For any ACE, returns TRUE if the ACE grants or denies write owner permissions. <small>Win:4.1</small>
write permission of <access control entry>	<i>Plain</i>	<boolean>	For a file ACE, returns TRUE if the ACE grants or denies write permissions. <small>Win:4.1</small>

## Access Control List

An Access Control List, or ACL, is a list of security protections that applies to an object. An object can be a file, process, event, or anything else having a security descriptor. An entry in an access control list (ACL) is an access control entry (ACE). Access rights run up through the object hierarchy, so rights granted at a low level can be vetoed by rights higher up. Some of the following Inspectors determine the effective permissions for a given trustee by traversing the hierarchy. They work by exposing the `GetEffectiveRightsFromAcl` method, as explained here:  
<http://tinyurl.com/dtmje>.

### Creation Methods

Key Phrase	Form	Description
dacl of <security descriptor>	<i>Plain</i>	Returns the discretionary access control list associated with the given security descriptor.  Win:4.1

### Note

Requires Windows XP, Windows 2000 Professional, or Windows NT Workstation 3.1 and later.

### Properties

Key Phrase	Form	Return Type	Description
effective access mode for <string> of <access control list>	<i>Named</i>	<integer>	Returns an integer corresponding to the access mode for the trustee specified by <string> of the given access control list.  Win:6.0
effective access system security permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has access system security permissions on the given access control list.  Win:6.0
effective append permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has append permissions on the given access control list.  Win:6.0
effective change notification permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has change notification permissions on the given access control list.  Win:6.0

Key Phrase	Form	Return Type	Description
effective create file permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has file creation permissions on the given access control list.  Win:6.0
effective create folder permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has folder creation permissions on the given access control list.  Win:6.0
effective create link permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has link creation permissions on the given access control list.  Win:6.0
effective create subkey permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has subkey creation permissions on the given access control list.  Win:6.0
effective delete child permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has child deletion permissions on the given access control list.  Win:6.0
effective delete permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has delete permissions on the given access control list.  Win:6.0
effective enumerate subkeys permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has subkey enumeration permissions on the given access control list.  Win:6.0
effective execute permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has execution permissions on the given access control list.  Win:6.0
effective generic all permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has all generic permissions on the given access control list.  Win:6.0

Key Phrase	Form	Return Type	Description
effective generic execute permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has generic execution permissions on the given access control list.  Win:6.0
effective generic read permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has generic read permissions on the given access control list.  Win:6.0
effective generic write permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has generic write permissions on the given access control list.  Win:6.0
effective list permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has list permissions on the given access control list.  Win:6.0
effective maximum allowed permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has maximum allowed permissions on the given access control list.  Win:6.0
effective query value permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has query value permissions on the given access control list.  Win:6.0
effective read attributes permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has read attribute permissions on the given access control list.  Win:6.0
effective read control permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has read control permissions on the given access control list.  Win:6.0
effective read extended attributes permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has extended read attribute permissions on the given access control list.  Win:6.0

Key Phrase	Form	Return Type	Description
effective read permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has read permissions on the given access control list.  Win:6.0
effective set value permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has value setting permissions on the given access control list.  Win:6.0
effective synchronize permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has synchronization permissions on the given access control list.  Win:6.0
effective traverse permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has traverse permissions on the given access control list.  Win:6.0
effective write attributes permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has attribute writing permissions on the given access control list.  Win:6.0
effective write dac permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has dac writing permissions on the given access control list.  Win:6.0
effective write extended attributes permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has extended attribute writing permissions on the given access control list.  Win:6.0
effective write owner permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has write owner permissions on the given access control list.  Win:6.0
effective write permission for <string> of <access control list>	<i>Named</i>	<boolean>	Returns TRUE if the trustee specified by <string> has write permissions on the given access control list.  Win:6.0

Key Phrase	Form	Return Type	Description
entry of <access control list>	<i>Plain</i>	<access control entry>	Iterates the ACEs of a ACL.  Win:4.1

## Note

The ACCESS\_MASK is returned from the effective access mode as a double word defining standard, specific, and generic rights. These rights are used in access control entries (ACEs) and are the primary means of determining access to an object.

Bits	Meaning
0 through 15	Specific rights. Contains the access mask specific to the object type associated with the mask.
16 through 23	Contains the object's standard access rights.
24	The Access system security bit is used to indicate access to a system access control list (SACL). If this flag is set in the access mask of an audit access ACE (successful or unsuccessful access), the SACL access will be audited.
25	Maximum allowed.
26 through 27	Reserved.
28	Generic all.
29	Generic execute.
30	Generic write.
31	Generic read.

The standard rights bits from 16 to 23 contain the object's standard access rights and can be a combination of the following predefined flags:

Bit	Flag	Meaning
16	DELETE	Delete access.
17	READ_CONTROL	Read access to the owner, group, and discretionary access control list (DACL) of the security descriptor.
18	WRITE_DAC	Write access to the DACL.
19	WRITE_OWNER	Write access to owner.
20	SYNCHRONIZE	Synchronize access.

## Examples

- number of dacls of (security descriptor of file "mshtml.dll" of system folder) = 1
    - ▶ Returns TRUE if there is exactly one discretionary access control list associated with the security descriptor of the specified file.
  
  - effective access mode for "Administrators" of dacls of security descriptors of system folder as hexadecimal
    - ▶ Returns a hex value corresponding to the access mode of the system folder for users logged in as Administrators.
  
  - effective append permission for "Power Users" of dacls of security descriptors of windows folder
    - ▶ Returns TRUE if Power Users have append permissions on the system folder.
  
  - effective create folder permissions for "Administrators" of dacls of security descriptors of folders of folder "c:\\"ul style="list-style-type: none;">  - ▶ Returns a list of TRUE/FALSE values corresponding to the ability of the Administrator to create new folders in each of the existing folders of the c: drive.
- 
- effective synchronize permission for "Administrators" of dacls of security descriptors of system folder
  - ▶ Returns TRUE if the Administrator has permission to synchronize with the system folder.

---

## Security Descriptor

A structure and associated data that contains the security information for a securable object. A security descriptor identifies the object's owner and primary group. It can also contain a DACL that controls access to the object, and a SACL that controls the logging of attempts to access the object.

## Creation Methods

Key Phrase	Form	Description
security descriptor of <file>	<i>Plain</i>	Specifies the security descriptor associated with the specified file.  Win:4.1
security descriptor of <folder>	<i>Plain</i>	Specifies the security descriptor associated with the specified folder.  Win:4.1

Key Phrase	Form	Description
security descriptor of <network share>	<i>Plain</i>	Specifies the security descriptor associated with the specified network share.  Win:4.1
security descriptor of <registry key>	<i>Plain</i>	Specifies the security descriptor associated with the specified registry key.  Win:4.1

## Properties

Key Phrase	Form	Return Type	Description
<security descriptor> as string	<i>Cast</i>	<string>	Returns the security descriptor in string format.  Win:4.1
control of <security descriptor>	<i>Plain</i>	<integer>	Returns the integer property obtained by using the Microsoft Windows GetSecurityDescriptorControl API. This integer contains bits that indicate DACL behaviors as well as default behaviors. See the MSDN documentation of SECURITY_DESCRIPTOR_CONTROL for more information.  Win:6.0
dacl of <security descriptor>	<i>Plain</i>	<access control list>	Returns the discretionary access control list associated with the given security descriptor.  Win:4.1
group of <security descriptor>	<i>Plain</i>	<security identifier>	Returns the security identifier of the group of the specified security descriptor.  Win:4.1
owner of <security descriptor>	<i>Plain</i>	<security identifier>	Returns the security identifier of the owner of the specified security descriptor.  Win:4.1

## Security Identifier

A Security Identifier, or SID, is a data structure that identifies user, group, and computer accounts. Every account on a network is issued a unique SID when the account is first created. Internal processes in Windows refer to an account's SID rather than the account's user or group name.

### Creation Methods

Key Phrase	Form	Description
group of <security descriptor>	<i>Plain</i>	Returns the SID of the group of the specified security descriptor.  Win:4.1
owner of <security descriptor>	<i>Plain</i>	Returns the security identifier of the owner of the specified security descriptor.  Win:4.1
sid of <local group member>	<i>Plain</i>	Returns the security ID for the specified local group member.  Win:6.0
trustee of <access control entry>	<i>Plain</i>	Returns the trustee to whom the specified ACE applies.  Win:4.1
user sid of <event log record>	<i>Plain</i>	Returns the user security ID for the specified record in the event log.  Win:6.0

### Properties

Key Phrase	Form	Return Type	Description
<security identifier> as string	<i>Cast</i>	<string>	Returns the security identifier in string format.  Win:4.1
account name of <security identifier>	<i>Plain</i>	<string>	Retrieves the name of the account for this SID and the name of the first domain on which this SID is found.  Win:4.1
domain name of <security identifier>	<i>Plain</i>	<string>	Returns the domain name of the first domain on which the specified SID is found.  Win:4.1

# User Objects

## Local User

A Local User object is provided to access the user data of the local machine. Note that domain users are not available through this Inspector.

### Creation Methods

Key Phrase	Form	Description
domain user	<i>PlainGlobal</i>	Returns all of the users that are members of the domain for which the machine is a user.  Win:4.1
domain user <string>	<i>NamedGlobal</i>	Returns the local user object corresponding to the specified name.  Win:4.1
local user	<i>PlainGlobal</i>	Creates an object with all the local user accounts.  Win:1.2
local user <string>	<i>NamedGlobal</i>	Creates an object with a named user account.  Win:1.2

### Properties

Key Phrase	Form	Return Type	Description
account disabled flag of <local user>	<i>Plain</i>	<boolean>	Indicates that this account is disabled.  Win:1.2
account expiration of <local user>	<i>Plain</i>	<time>	Returns the time when this account is set to expire.  Win:1.2
accounts operator flag of <local user>	<i>Plain</i>	<boolean>	This user has the accounts operator privilege.  Win:1.2
admin privilege of <local user>	<i>Plain</i>	<boolean>	Indicates that the user has a privilege level of 'admin'.  Win:1.2

Key Phrase	Form	Return Type	Description
allowed workstations string of <local user>	<i>Plain</i>	<string>	Returns a list of workstations this user is allowed to login to. If this string is empty, no restrictions apply.  Win:1.2
application parameter string of <local user>	<i>Plain</i>	<string>	Returns a string used by Microsoft products to store user configuration information.  Win:1.2
bad password count of <local user>	<i>Plain</i>	<integer>	Returns the number of attempts to logon to this account with a bad password.  Win:1.2
code page of <local user>	<i>Plain</i>	<integer>	Returns the code page for the user's preferred language.  Win:1.2
comment of <local user>	<i>Plain</i>	<string>	Returns the comment associated with this user's account.  Win:1.2
communications operator flag of <local user>	<i>Plain</i>	<boolean>	This user has the communications operator privilege.  Win:1.2
country code of <local user>	<i>Plain</i>	<integer>	Returns the country code of the user's preferred language.  Win:1.2
full name of <local user>	<i>Plain</i>	<string>	Returns the full name of the user.  Win:1.2
guest privilege of <local user>	<i>Plain</i>	<boolean>	Indicates that the user has a privilege level of 'guest'.  Win:1.2
home directory drive of <local user>	<i>Plain</i>	<string>	Returns the name of the drive assigned to the user's home directory.  Win:1.2
home directory of <local user>	<i>Plain</i>	<string>	Returns the directory where the user files are stored for the particular user.  Win:1.2

Key Phrase	Form	Return Type	Description
home directory required flag of <local user>	<i>Plain</i>	<boolean>	Indicates that a home directory is required for the user.  Win:1.2
interdomain trust account flag of <local user>	<i>Plain</i>	<boolean>	This is an account which specifies that a domain should trust other domains.  Win:1.2
last logoff of <local user>	<i>Plain</i>	<time>	Returns the time when the user last logged off.  Win:1.2
last logon of <local user>	<i>Plain</i>	<time>	Returns the time when the user last logged on.  Win:1.2
locked out flag of <local user>	<i>Plain</i>	<boolean>	Indicates that this user is currently locked out.  Win:1.2
logon count of <local user>	<i>Plain</i>	<integer>	Returns the number of times which this account has successfully logged on to the local machine.  Win:1.2
logon script of <local user>	<i>Plain</i>	<string>	Returns the pathname of this user's logon script.  Win:1.2
logon server of <local user>	<i>Plain</i>	<string>	Returns the name of the server to which logon requests are sent for this account.  Win:1.2
maximum storage of <local user>	<i>Plain</i>	<integer>	Returns the user's disk quota. Will return FALSE if the user has no disk quota.  Win:1.2
name of <local user>	<i>Plain</i>	<string>	Returns the name of the user.  Win:1.2
no password required flag of <local user>	<i>Plain</i>	<boolean>	Indicates that no password is required for this user.  Win:1.2
normal account flag of <local user>	<i>Plain</i>	<boolean>	Indicates that this account has a default account type that represents a typical user.  Win:1.2

Key Phrase	Form	Return Type	Description
password age of <local user>	Plain	<time interval>	Gives the time since the user's password was last changes.  Win:1.2
password change disabled flag of <local user>	Plain	<boolean>	Indicates that this user is not allowed to change his password.  Win:1.2
password expiration disabled flag of <local user>	Plain	<boolean>	Indicates that the password on this account will never expire.  Win:1.2
password expired of <local user>	Plain	<boolean>	Indicates that the user's password has expired.  Win:1.2
primary group id of <local user>	Plain	<integer>	Returns the RID of the user's primary group.  Win:1.2
print operator flag of <local user>	Plain	<boolean>	This user has the print operator privilege.  Win:1.2
profile folder of <local user>	Plain	<string>	Returns the pathname of the folder which contains the user's profile.  Win:1.2
script flag of <local user>	Plain	<boolean>	Indicates that the logon script executed.  Win:1.2
server operator flag of <local user>	Plain	<boolean>	This user has the server operator privilege.  Win:1.2
server trust account flag of <local user>	Plain	<boolean>	This is an account for a backup domain controller.  Win:1.2
temporary duplicate account flag of <local user>	Plain	<boolean>	Indicates that this is a temporary duplicate account.  Win:1.2
user comment of <local user>	Plain	<string>	Returns the user comment of this user.  Win:1.2
user id of <local user>	Plain	<integer>	Returns the user's RID number.  Win:1.2

Key Phrase	Form	Return Type	Description
user privilege of <local user>	<i>Plain</i>	<boolean>	Indicates that the user has a privilege level of 'user'.  Win:1.2
workstation trust account flag of <local user>	<i>Plain</i>	<boolean>	This account is for a workstation or server.  Win:1.2

## Examples

- exists local user "Administrator"
  - ▶ TRUE if there exists a local user named Administrator.
- exists local user whose (bad password count of it > 5)
  - ▶ TRUE if there have been more than 5 bad password attempts on this account.
- password age of local user "Administrator" > 30 \* day
  - ▶ TRUE if the Administrator's password is older than 30 days.

---

## Current User

This object exists if the user is logged in to the current machine.

### Creation Methods

Key Phrase	Form	Description
current user	<i>PlainGlobal</i>	Creates the current user object if one is logged in to the desktop.  Win:1.2, Mac:4.1

### Properties

Key Phrase	Form	Return Type	Description
name of <current user>	<i>Plain</i>	<string>	Returns the name of the current user.  Win:1.2, Mac:4.1

### Examples

- `local user (name of current user)`
- Provides access to all the local user properties of the currently logged-in user.

## Action Objects

These are the keywords associated with properties that can be inspected while BigFix Actions are being executed.

### Action

These are the keywords associated with properties available for inspection during the execution of BigFix Actions.

### Creation Methods

Key Phrase	Form	Description
action	<i>PlainGlobal</i>	Creates an action object corresponding to the BigFix Action currently being parsed. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:5.1</small>
action <integer>	<i>NumberedGlobal</i>	Creates an action object matching the <integer> id. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
active action	<i>PlainGlobal</i>	Creates an action object corresponding to the currently executing action. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

### Properties

Key Phrase	Form	Return Type	Description
active of <action>	<i>Plain</i>	<boolean>	Returns TRUE if the action is currently running (active). <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
active start time of <action>	<i>Plain</i>	<time>	Returns the time the action started. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
complete time of <action>	<i>Plain</i>	<time>	Returns the time the action completed. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>
constrained of <action>	<i>Plain</i>	<boolean>	Returns TRUE if action is unable to run yet. <small>Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1</small>

Key Phrase	Form	Return Type	Description
group leader of <action>	<i>Plain</i>	<boolean>	Returns TRUE if the action is a group action and the action component is the group leader. When you deploy a mult-action from the BES Console, it constructs a group action with a group leader to control the overall behavior of the action. This inspector is used internally to manage the progress of the group action.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
id of <action>	<i>Plain</i>	<integer>	Returns the action id of the action.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
last change time of <action>	<i>Plain</i>	<time>	Returns the time when the action state last changed.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
origin fixlet id of <action>	<i>Plain</i>	<integer>	Returns the Fixlet id that contained the action.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
parameter <string> of <action>	<i>Named</i>	<string>	Returns the value of parameter <string> for the active action. Parameters only live as long as the action is active.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
pending login of <action>	<i>Plain</i>	<boolean>	Returns TRUE if the specified action included an 'action requires login' command, and a login has not yet occurred since the action has run.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
pending of <action>	<i>Plain</i>	<boolean>	Returns TRUE if action is available to run.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
pending restart of <action>	<i>Plain</i>	<boolean>	Returns TRUE if the specified action included an 'action requires restart' command and a restart has not occurred since the action has run.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
pending time of <action>	<i>Plain</i>	<time>	Returns the time the action became pending.  Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

Key Phrase	Form	Return Type	Description
status of <action>	<i>Plain</i>	<string>	Returns one of the following strings: <ul style="list-style-type: none"><li>• Running = when the action is currently active.</li><li>• Executed = no longer relevant and action has completed.</li><li>• Not Relevant = action was not relevant.</li><li>• Waiting = action is relevant, but waiting to run.</li><li>• Not Executed = action is relevant, unconstrained, but has not yet started.</li><li>• Failed = action is relevant, unconstrained, has completed, but is still relevant.</li></ul> Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1
waiting for download of <action>	<i>Plain</i>	<boolean>	Returns TRUE if client is waiting for mirroring server to have downloads required by the action. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1

# Networking Objects

This chapter includes the various networking Inspectors.

## Network

These are the keywords used to query the local network configuration.

### Creation Methods

Key Phrase	Form	Description
network	<i>PlainGlobal</i>	Creates an object containing properties of the network. <small>Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1</small>

### Properties

Key Phrase	Form	Return Type	Description
adapter of <network>	<i>Plain</i>	<network adapter>	Returns the one or more network adapter objects of the network. <small>Win:2.0</small>
connection of <network>	<i>Plain</i>	<connection>	Returns the connection of the specified network. <small>Win:5.0</small>
dns server of <network>	<i>Plain</i>	<network address list>	Returns a list of DNS servers used by the local computer. <small>Win:4.1</small>
interface <integer> of <network>	<i>Numbered</i>	<network interface>	Returns the particular interface of the network. <small>Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1</small>
interface of <network>	<i>Plain</i>	<network interface>	Returns all the interfaces of the network. <small>Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1</small>
ip interface <integer> of <network>	<i>Numbered</i>	<network ip interface>	Returns the particular ip interface of the network. <small>Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1</small>
ip interface of <network>	<i>Plain</i>	<network ip interface>	Returns all the ip interfaces of the network. <small>Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1</small>

Key Phrase	Form	Return Type	Description
winsock2 supported of <network>	<i>Plain</i>	<boolean>	Indicates that winsock2 is supported by the network. If this returns FALSE, many of the other properties of the interface are not available for inspection.  Win:1.2

---

## Network Interface

The network interface object describes a generic network interface, and has information about the name and family of that interface. On the Mac these are commonly of type AF\_INET, AF\_LINK and AF\_INET6.

### Creation Methods

Key Phrase	Form	Description
interface <integer> of <network>	<i>Numbered</i>	Creates an object with the specified network interface.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
interface of <network>	<i>Plain</i>	Creates an object with all the interfaces of the network.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1

### Properties

Key Phrase	Form	Return Type	Description
family of <network interface>	<i>Plain</i>	<integer>	Returns an family designator of the address family (i.e., 2=AFI_NET).  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1

### Examples

- names of interfaces of network
- Returns a list of the network interface names, e.g., lo0, gif0, stf0, en0.

## Network Ip Interface

In general, the network ip interface object holds locally determined properties of logical network devices configured on the computer. On the Mac, these correspond to interfaces of type AF\_INET. The properties that are available depend on the socket support installed on the computer. For Windows computers with winsock 2 support installed, for instance, the information is obtained by an ioctl call and includes Interface address, Interface broadcast address, Interface network mask, Broadcast support flag, Multicast support flag, Loopback interface flag and Point to point interface flag.

### Creation Methods

Key Phrase	Form	Description
ip interface <integer> of <network>	<i>Numbered</i>	Creates an object with the specified ip interface of the network.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
ip interface of <network>	<i>Plain</i>	Creates an object or an object list (using the plural keyword) with all the ip interfaces of the network.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1

### Properties

Key Phrase	Form	Return Type	Description
address of <network ip interface>	<i>Plain</i>	<ipv4 address>	Returns the ip address of the ip interface.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
broadcast address of <network ip interface>	<i>Plain</i>	<ipv4 address>	Returns the broadcast address of the interface.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
broadcast support of <network ip interface>	<i>Plain</i>	<boolean>	Indicates that broadcast messages are supported by the ip interface.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
loopback of <network ip interface>	<i>Plain</i>	<boolean>	Indicates that the particular network ip interface is a loopback interface.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
multicast support of <network ip interface>	<i>Plain</i>	<boolean>	Indicates that multicast messages are supported by the ip interface.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1

Key Phrase	Form	Return Type	Description
point to point of <network ip interface>	<i>Plain</i>	<boolean>	Indicates that the interface is a point-to-point interface. Usually TRUE for dialup connections.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
subnet address of <network ip interface>	<i>Plain</i>	<ipv4 address>	The subnet to which the interface belongs.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
subnet mask of <network ip interface>	<i>Plain</i>	<ipv4 address>	The subnet mask of the interface.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1

## Examples

- names of ip interfaces of network
  - ▶ Returns a list of the names of the network IP interfaces, e.g., lo0, en0.
- addresses of ip interfaces of network
  - ▶ Returns a list of the IP addresses of the network IP interfaces, e.g., 127.0.0.1, 192.168.1.100, etc.
- address of ip interface whose (loopback of it = false) of network = "192.168.127.127"
  - ▶ Returns TRUE if the given IP address doesn't have loopback.

## Ipv4 Address

This is an Internet Protocol address, version 4. IP addresses are composed of four single-byte integers separated by periods, like "192.5.0.7".

## Creation Methods

Key Phrase	Form	Description
address of <network adapter>	<i>Plain</i>	Returns the ip address of the network adapter.  Win:2.0
address of <network address list>	<i>Plain</i>	Returns the ip address of the network adapter list.  Win:2.0
address of <network ip interface>	<i>Plain</i>	Creates an object with the ip address of the interface.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
broadcast address of <network ip interface>	<i>Plain</i>	Creates an object with the broadcast address of the interface.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1

Key Phrase	Form	Description
dhcp server of <network adapter>	<i>Plain</i>	Returns the ip address of the dhcp server of the network adapter.  Win:2.0
gateway address <integer> of <selected server>	<i>Numbered</i>	The ip address of a gateway between the agent and the selected server at the given distance from the agent, if known.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
gateway address of <selected server>	<i>Plain</i>	All known ip addresses of gateways between the agent and the selected server.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
gateway of <network adapter>	<i>Plain</i>	Returns the ip address of the gateway of the network adapter.  Win:2.0
ip address of <selected server>	<i>Plain</i>	The ip address to which reports are sent.  Win:4.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
ipv4 address <string>	<i>NamedGlobal</i>	Creates an object with an ip address for the string provided.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
primary wins server of <network adapter>	<i>Plain</i>	Returns the ip address of the primary wins server of the network adapter.  Win:2.0
secondary wins server of <network adapter>	<i>Plain</i>	Returns the ip address of the secondary wins server of the network adapter.  Win:2.0
subnet address of <network adapter>	<i>Plain</i>	Returns the subnet address of the network adapter.  Win:2.0
subnet address of <network address list>	<i>Plain</i>	Returns the subnet address of the network address list.  Win:2.0
subnet address of <network ip interface>	<i>Plain</i>	Creates an object with the subnet address of the network interface.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
subnet mask of <network adapter>	<i>Plain</i>	Returns the subnet mask of the network adapter.  Win:2.0

Key Phrase	Form	Description
subnet mask of <network address list>	<i>Plain</i>	Returns the subnet mask of the network adapter list.  Win:2.0
subnet mask of <network ip interface>	<i>Plain</i>	Creates an object with the address bitwise ANDed with the subnet mask.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
target ip address of <port mapping>	<i>Plain</i>	Returns the target IP address of the specified port mapping.  Win:4.1

## Properties

Key Phrase	Form	Return Type	Description
<ipv4 address> as string	<i>Cast</i>	<string>	Converts the ipv4 address to a string.  Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1

## Operators

Key phrase	Return Type	Description
<ipv4 address> {cmp} <ipv4 address>	<boolean>	Returns a boolean TRUE/FALSE depending on the result of the comparison, where: <ul style="list-style-type: none"> <li>{cmp} is one of: =, !=, &lt;, &lt;=, &gt;, &gt;= .</li> </ul> Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1
<ipv4 address> {cmp} <string>	<boolean>	Returns a boolean TRUE/FALSE depending on the result of the comparison, where: <ul style="list-style-type: none"> <li>{cmp} is one of: =, !=, &lt;, &lt;=, &gt;, &gt;= .</li> </ul> Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1

## Examples

■ exists ip interface whose (address of it = "127.0.0.1" and loopback of it) of network

► Returns TRUE if the specified ip interface (with loopback) exists on this computer.

■ addresses of ip interfaces of network

► Returns a list of IP addresses configured on the machine.

■ ipv4 address "192.168.100.1"

► Returns the four-byte ip address 192.168.100.1.

## Network Adapter

One or more network adapters may be inspected using this property of the network object. Each network adapter has a number of interesting properties such as the MAC address.

### Creation Methods

Key Phrase	Form	Description
adapter of <network>	<i>Plain</i>	Returns one or more adapters of the network.  Win:2.0

### Properties

Key Phrase	Form	Return Type	Description
address list of <network adapter>	<i>Plain</i>	<network address list>	Returns the address list of the network adapter.  Win:2.0
address of <network adapter>	<i>Plain</i>	<ipv4 address>	Returns the ip address of the network adapter (returns the first address if it is a list).  Win:2.0
description of <network adapter>	<i>Plain</i>	<string>	Returns the description of the network adapter.  Win:2.0
dhcp enabled of <network adapter>	<i>Plain</i>	<boolean>	Returns TRUE if dhcp is enabled on the network adapter.  Win:2.0
dhcp server of <network adapter>	<i>Plain</i>	<ipv4 address>	Returns the ip address of the dhcp server of the network adapter.  Win:2.0
dns server of <network adapter>	<i>Plain</i>	<network address list>	Returns a list of DNS servers used by the specified adapter.  Win:4.1
gateway list of <network adapter>	<i>Plain</i>	<network address list>	Returns the gateway network address list of the network adapter.  Win:2.0
gateway of <network adapter>	<i>Plain</i>	<ipv4 address>	Returns the ip address of the gateway of the network adapter.  Win:2.0

Key Phrase	Form	Return Type	Description
internet connection firewall of <network adapter>	<i>Plain</i>	<internet connection firewall>	Creates a Windows XP firewall object. <small>Win:4.1</small>
lease expires of <network adapter>	<i>Plain</i>	<time>	Returns the time that the dhcp lease will expire of the network adapter. <small>Win:2.0</small>
lease obtained of <network adapter>	<i>Plain</i>	<time>	Returns the time that the dhcp lease was obtained of the network adapter. <small>Win:2.0</small>
link speed of <network adapter>	<i>Plain</i>	<integer>	This is a property of a network adapter. It returns the maximum speed of the NIC card in bits per second. <small>Win:6.0</small>
mac address of <network adapter>	<i>Plain</i>	<string>	Returns the mac address of the network adapter. <small>Win:2.0</small>
name of <network adapter>	<i>Plain</i>	<string>	Returns the name of the network adapter. <small>Win:2.0</small>
primary wins server of <network adapter>	<i>Plain</i>	<ipv4 address>	Returns the ip address of the primary wins server of the network adapter. <small>Win:2.0</small>
secondary wins server of <network adapter>	<i>Plain</i>	<ipv4 address>	Returns the ip address of the secondary wins server of the network adapter. <small>Win:2.0</small>
subnet address of <network adapter>	<i>Plain</i>	<ipv4 address>	Returns the subnet address of the network adapter. <small>Win:2.0</small>
subnet mask of <network adapter>	<i>Plain</i>	<ipv4 address>	Returns the subnet mask of the network adapter. <small>Win:2.0</small>

Key Phrase	Form	Return Type	Description
wakeonlan enabled of <network adapter>	<i>Plain</i>	<boolean>	Returns true if the specified network adapter is configured to react to Wake-On-Lan requests. Wake-On-Lan is a mechanism used to trigger a boot of a machine in standby mode by sending a special packet. • Note: Wake-On-Lan is only supported for Windows 2000 and XP machines.  Win:5.1
wins enabled of <network adapter>	<i>Plain</i>	<boolean>	Returns TRUE if WINS is enabled on the network adapter.  Win:2.0

## Network Address List

A network adapter may be configured to respond to a list of network addresses. This object type provides access to such a list.

### Creation Methods

Key Phrase	Form	Description
address list of <network adapter>	<i>Plain</i>	Returns the address list of the network adapter.  Win:2.0
dns server of <network adapter>	<i>Plain</i>	Returns a list of DNS servers used by the specified adapter.  Win:4.1
dns server of <network>	<i>Plain</i>	Returns a list of DNS servers used by the local computer.  Win:4.1
gateway list of <network adapter>	<i>Plain</i>	Returns the gateway network address list of the network adapter.  Win:2.0

## Properties

Key Phrase	Form	Return Type	Description
address of <network address list>	<i>Plain</i>	<ipv4 address>	Returns the address of the address list. <small>Win:2.0</small>
subnet address of <network address list>	<i>Plain</i>	<ipv4 address>	Returns the subnet address of the network address list. <small>Win:2.0</small>
subnet mask of <network address list>	<i>Plain</i>	<ipv4 address>	Returns the subnet mask of the network address list. <small>Win:2.0</small>

## Internet Connection Firewall

Provides access to the settings of the Internet Connection Firewall introduced in Windows XP. The Internet Connection Firewall helps to protect a computer that is directly connected to the Internet, or a home network, from network attacks.

## Creation Methods

Key Phrase	Form	Description
firewall of <connection>	<i>Plain</i>	Returns the internet connection firewall object corresponding to the specified connection. <small>Win:5.0</small>
internet connection firewall of <network adapter>	<i>Plain</i>	Creates a Windows XP firewall object. <small>Win:4.1</small>

## Properties

Key Phrase	Form	Return Type	Description
enabled of <internet connection firewall>	<i>Plain</i>	<boolean>	Returns TRUE if the local computer has the Windows XP built-in firewall enabled. <small>Win:4.1</small>
port mapping of <internet connection firewall>	<i>Plain</i>	<port mapping>	Creates a port mapping object for the built-in firewall. <small>Win:4.1</small>

## Port Mapping

Refers to a port mapping object for the built-in firewall.

### Creation Methods

Key Phrase	Form	Description
port mapping of <internet connection firewall>	<i>Plain</i>	Creates a port mapping object for the built-in firewall.  Win:4.1

### Properties

Key Phrase	Form	Return Type	Description
enabled of <port mapping>	<i>Plain</i>	<boolean>	A boolean indicating whether or not the port mapping is enabled.  Win:4.1
external port of <port mapping>	<i>Plain</i>	<integer>	Returns the external port number of the specified ICF port mapping.  Win:4.1
internal port of <port mapping>	<i>Plain</i>	<integer>	Returns the internal port number of the specified ICF port mapping.  Win:4.1
name of <port mapping>	<i>Plain</i>	<string>	Returns the name of the specified port mapping.  Win:4.1
options of <port mapping>	<i>Plain</i>	<integer>	See port mapping at MSDN.  Win:4.1
protocol of <port mapping>	<i>Plain</i>	<string>	Returns a string like "tcp" or "udp", corresponding to the protocol of the specified port mapping.  Win:4.1
target ip address of <port mapping>	<i>Plain</i>	<ipv4 address>	Returns the target IP address of the specified port mapping.  Win:4.1
target name of <port mapping>	<i>Plain</i>	<string>	Returns the target name of the specified port mapping.  Win:4.1

## Network Share

The network share Inspector does not work on Windows 95/98/Me. The password and permission properties are relevant only for shares using share-level security. User-level security is given by the security descriptor. The use limit property will throw NoSuchObject if use is unlimited.

### Creation Methods

Key Phrase	Form	Description
network share	<i>PlainGlobal</i>	Creates a network shared object.  Win:4.1
network share <string>	<i>NamedGlobal</i>	Creates a named network shared object.  Win:4.1

### Properties

Key Phrase	Form	Return Type	Description
attribute permission of <network share>	<i>Plain</i>	<boolean>	Returns TRUE if permission is granted to modify the attributes of the shared resource (such as the date and time when a file was last modified).  Win:4.1
comment of <network share>	<i>Plain</i>	<string>	Returns a string specifying an optional comment about the shared resource.  Win:4.1
create permission of <network share>	<i>Plain</i>	<boolean>	Returns TRUE if permission is granted to create an instance of a shared resource (such as a file).  Win:4.1
delete permission of <network share>	<i>Plain</i>	<boolean>	Returns TRUE if permission is granted to delete the resource.  Win:4.1
execute permission of <network share>	<i>Plain</i>	<boolean>	Returns TRUE if permission is granted to execute the resource.  Win:4.1
name of <network share>	<i>Plain</i>	<string>	Returns a string specifying the name of the specified shared resource.  Win:4.1

Key Phrase	Form	Return Type	Description
password of <network share>	<i>Plain</i>	<string>	A string that specifies the share's password (when the server is running with share-level security). Note that the Windows Server 2003 family, Windows XP, Windows 2000, and Windows NT do not support share-level security.  Win:4.1
path of <network share>	<i>Plain</i>	<string>	A string containing the local path for the shared resource.  Win:4.1
permission permission of <network share>	<i>Plain</i>	<boolean>	Returns TRUE if permission is granted to modify the permissions (read, write, create, execute, and delete) for the specified network share.  Win:4.1
read permission of <network share>	<i>Plain</i>	<boolean>	Returns TRUE if permission is granted to read data from a resource and, by default, to execute the specified network share.  Win:4.1
security descriptor of <network share>	<i>Plain</i>	<security descriptor>	Specifies the security descriptor associated with the specified network share.  Win:4.1
type of <network share>	<i>Plain</i>	<integer>	Specifies an integer value that indicates the type of share. (See the Microsoft document on SHARE_INFO_502).  Win:4.1
use count of <network share>	<i>Plain</i>	<integer>	Specifies an integer value that indicates the number of current connections to the specified network share.  Win:4.1
use limit of <network share>	<i>Plain</i>	<integer>	Specifies an integer value indicating the maximum number of concurrent connections that the shared resource can accommodate.  Win:4.1
write permission of <network share>	<i>Plain</i>	<boolean>	Returns TRUE if permission is granted to write data to the specified network share.  Win:4.1

## Connection

This object is used to query your connections. These are all properties of the Internet Connection Firewall, as returned in the NETCON\_PROPERTIES structure.

### Creation Methods

Key Phrase	Form	Description
connection of <network>	<i>Plain</i>	Returns a connection to the specified network.  Win:5.0

### Properties

Key Phrase	Form	Return Type	Description
device name of <connection>	<i>Plain</i>	<string>	Returns the name of the device associated with the specified connection.  Win:5.0
firewall of <connection>	<i>Plain</i>	<internet connection firewall>	Returns the internet connection firewall object corresponding to the specified connection.  Win:5.0
guid of <connection>	<i>Plain</i>	<string>	Returns the globally-unique identifier (GUID) for the specified connection.  Win:5.0
media type of <connection>	<i>Plain</i>	<media type>	Returns the media type of for the specified connection.  Win:5.0
name of <connection>	<i>Plain</i>	<string>	Returns the name of the specified connection.  Win:5.0
status of <connection>	<i>Plain</i>	<connection status>	Returns the status of the specified connection.  Win:5.0

## Connection Status

This object returns information about the status of your connections.

### Creation Methods

Key Phrase	Form	Description
connection status <integer>	<i>NumberedGlobal</i>	Returns the connection status based on its integer value. This Inspector is included to take advantage of new (or undocumented) additions to the status values.  Win:5.0
connection status authenticating	<i>PlainGlobal</i>	Returns the value NCS_AUTHENTICATING: The connection is waiting for authentication to occur.  Win:5.0
connection status authentication failed	<i>PlainGlobal</i>	Returns the value NCS_AUTHENTICATION_FAILED: Authentication has failed on this connection.  Win:5.0
connection status authentication succeeded	<i>PlainGlobal</i>	Returns the value NCS_AUTHENTICATION_SUCCEEDED: Authentication has succeeded on this connection.  Win:5.0
connection status connected	<i>PlainGlobal</i>	Returns the value NCS_CONNECTED: The connection is in a connected state.  Win:5.0
connection status connecting	<i>PlainGlobal</i>	Returns the value NCS_CONNECTING: The connection is in the process of connecting.  Win:5.0
connection status disconnected	<i>PlainGlobal</i>	Returns the value NCS_DISCONNECTED: The connection is disconnected.  Win:5.0
connection status disconnecting	<i>PlainGlobal</i>	Returns the value NCS_DISCONNECTING: The connection is in the process of disconnecting.  Win:5.0

Key Phrase	Form	Description
connection status hardware disabled	<i>PlainGlobal</i>	Returns the value NCS_HARDWARE_DISABLED: The hardware for the connection is present, but is not enabled.  Win:5.0
connection status hardware malfunction	<i>PlainGlobal</i>	Returns the value NCS_HARDWARE_MALFUNCTION: A malfunction has occurred in the hardware for the connection.  Win:5.0
connection status media disconnected	<i>PlainGlobal</i>	Returns the value NCS_MEDIA_DISCONNECTED: The media, for example the network cable, is disconnected.  Win:5.0
connection status no hardware present	<i>PlainGlobal</i>	Returns the value NCS_NO_HARDWARE_PRESENT: The hardware for the connection, for example network interface card (NIC), is not present.  Win:6.0
status of <connection>	<i>Plain</i>	Returns the current status of the connection. You can compare this status to the 'connection status' global objects, which act as constants. Some of the possible values include: <ul style="list-style-type: none"> <li>• NCS_DISCONNECTED: The connection is disconnected.</li> <li>• NCS_CONNECTING: The connection is in the process of connecting.</li> <li>• NCS_CONNECTED: The connection is in a connected state.</li> <li>• NCS_DISCONNECTING: The connection is in the process of disconnecting.</li> <li>• NCS_HARDWARE_NOT_PRESENT: The hardware for the connection, for example network interface card (NIC), is not present.</li> <li>• NCS_HARDWARE_DISABLED: The hardware for the connection is present, but is not enabled.</li> <li>• NCS_HARDWARE_MALFUNCTION: A malfunction has occurred in the hardware for the connection.</li> <li>• NCS_MEDIA_DISCONNECTED: The media, for example the network cable, is disconnected.</li> <li>• NCS_AUTHENTICATING: The connection is waiting for authentication to occur.</li> <li>• NCS_AUTHENTICATION_SUCCEEDED: Authentication has succeeded on this connection.</li> <li>• NCS_AUTHENTICATION_FAILED: Authentication has failed on this connection.</li> <li>• NCS_INVALID_ADDRESS: The address is invalid.</li> <li>• NCS_CREDENTIALS_REQUIRED: Security credentials are required.</li> </ul> Win:5.0

## Operators

Key phrase	Return Type	Description
<connection status> = <connection status>	<boolean>	Compare the statuses of two connections.  Win:5.0

## Media Type

This object allows you to inspect the media type of your Internet Connection Firewall type connection, as returned in the NETCON\_PROPERTIES structure.

## Creation Methods

Key Phrase	Form	Description
media type <integer>	<i>NumberedGlobal</i>	Returns the media type based on its integer value. This Inspector is included to take advantage of new (or undocumented) additions to the media types.  Win:5.0
media type bridge	<i>PlainGlobal</i>	Returns the value NCM_BRIDGE: Bridged connection.  Win:5.0
media type direct	<i>PlainGlobal</i>	Returns the value NCM_DIRECT: Direct serial connection through a serial port.  Win:5.0
media type isdn	<i>PlainGlobal</i>	Returns the value NCM_ISDN: Connection is through an integrated services digital network (ISDN) line.  Win:5.0
media type lan	<i>PlainGlobal</i>	Returns the value NCM_LAN: Connection is to a local area network (LAN).  Win:5.0
media type of <connection>	<i>Plain</i>	Returns the media type of the specified connection: <ul style="list-style-type: none"> <li>• NCM_NONE: No media is present.</li> <li>• NCM_DIRECT: Direct serial connection through a serial port.</li> <li>• NCM_ISDN: Connection is through an integrated services digital network (ISDN) line.</li> <li>• NCM_LAN: Connection is to a local area network (LAN).</li> <li>• NCM_PHONE: Dial-up connection over a conventional phone line.</li> <li>• NCM_TUNNEL: Virtual private network (VPN) connection.</li> <li>• NCM_PPPOE: Point-to-Point protocol (PPP) over Ethernet.</li> <li>• NCM_BRIDGE: Bridged connection.</li> <li>• NCM_SHAREDACCESSHOST_LAN: Shared connection to a LAN.</li> <li>• NCM_SHAREDACCESSHOST_RAS: Shared connection to a remote or wide area network (WAN).</li> </ul> Win:5.0

Key Phrase	Form	Description
media type phone	<i>PlainGlobal</i>	Returns the value NCM_PHONE: Dial-up connection over a conventional phone line.  Win:5.0
media type pppoe	<i>PlainGlobal</i>	Returns the value NCM_PPPOE: Point-to-Point protocol (PPP) over Ethernet.  Win:5.0
media type shared access host lan	<i>PlainGlobal</i>	Returns the value NCM_SHAREDACCESSHOST_LAN: Shared connection to a LAN.  Win:5.0
media type shared access host ras	<i>PlainGlobal</i>	Returns the value NCM_SHAREDACCESSHOST_RAS: Shared connection to a remote or wide area network (WAN).  Win:5.0
media type tunnel	<i>PlainGlobal</i>	Returns the value NCM_TUNNEL: Virtual private network (VPN) connection.  Win:5.0

## Operators

Key phrase	Return Type	Description
<media type> = <media type>	<boolean>	Compares two media types.  Win:5.0

## Active Directory Local Computer

These are the Active Directory Inspectors for the local computer.

### Creation Methods

Key Phrase	Form	Description
local computer of <active directory server>	<i>Plain</i>	Represents your computer within the Active Directory. <small>Win:4.1, Mac:5.1</small>

### Properties

Key Phrase	Form	Return Type	Description
distinguished name error message of <active directory local computer>	<i>Plain</i>	<string>	Active Directory error if unable to get the distinguished name (this is for debugging purposes). <small>Win:4.1, Mac:5.1</small>
distinguished name of <active directory local computer>	<i>Plain</i>	<string>	Returns the computer's fully qualified active directory name in the distinguished name format, for instance, 'CN=ALBATROSS, CN=Computers, DC=bigfix, DC=com'. <small>Win:4.1, Mac:5.1</small>

### Examples

- distinguished name of local computer of active directory
- Returns CN=mymachinename,CN=Computers,DC=bigfix,DC=com.

## Active Directory Server

These are the Active Directory Server Inspectors.

### Creation Methods

Key Phrase	Form	Description
active directory	<i>PlainGlobal</i>	Returns an object containing the properties of the Active Directory to which your machine is attached.  Win:4.1, Mac:5.1

### Properties

Key Phrase	Form	Return Type	Description
local computer of <active directory server>	<i>Plain</i>	<active directory local computer>	Represents your computer within the Active Directory.  Win:4.1, Mac:5.1

## Microsoft IIS Metabase Objects

The Microsoft IIS Metabase is a repository for most IIS configuration values. The following inspectors retrieve information about the Microsoft IIS Metabase.

---

### Metabase

The IIS metabase is similar in structure to the Windows Registry, providing hierarchal storage of IIS configuration properties for Web sites, virtual directories, FTP, etc.

#### Creation Methods

Key Phrase	Form	Description
metabase	<i>PlainGlobal</i>	Returns the IIS metabase object.  Win:4.1

#### Properties

Key Phrase	Form	Return Type	Description
key <string> of <metabase>	<i>Named</i>	<metabase key>	Returns the named key of the metabase.  Win:4.1
key of <metabase>	<i>Plain</i>	<metabase key>	Returns the root key of the IIS metabase.  Win:4.1

---

### Metabase Identifier

A metabase identifier is a unique numerical identifier which IIS recognizes internally.

#### Creation Methods

Key Phrase	Form	Description
identifier of <metabase value>	<i>Plain</i>	Returns the metabase identifier of the specified value.  Win:4.1

## Properties

Key Phrase	Form	Return Type	Description
<metabase identifier> as integer	<i>Cast</i>	<integer>	The integer value of the specified identifier. Use the Microsoft MetaEdit utility to find the integer value of an identifier.  Win:4.1

## Operators

Key phrase	Return Type	Description
<metabase identifier> = <metabase identifier>	<boolean>	Compares two metabase identifiers for equality.  Win:4.1

## Metabase Key

Like a registry key, a metabase key contains named properties about the IIS metabase.

## Creation Methods

Key Phrase	Form	Description
key <string> of <metabase key>	<i>Named</i>	Returns the named subkey of the specified metabase key.  Win:4.1
key <string> of <metabase>	<i>Named</i>	Returns the named key of the metabase.  Win:4.1
key of <metabase key>	<i>Plain</i>	Iterates the subkeys of the specified metabase key.  Win:4.1
key of <metabase>	<i>Plain</i>	Returns the root key of the IIS metabase.  Win:4.1

## Properties

Key Phrase	Form	Return Type	Description
key <string> of <metabase key>	<i>Named</i>	<metabase key>	Returns the named subkey of the specified metabase key.  Win:4.1

Key Phrase	Form	Return Type	Description
key of <metabase key>	<i>Plain</i>	<metabase key>	Iterates the subkeys of the specified metabase key.  Win:4.1
name of <metabase key>	<i>Plain</i>	<string>	Returns the name of the specified metabase key.  Win:4.1
value of <metabase key>	<i>Plain</i>	<metabase value>	Returns the value of the specified metabase key.  Win:4.1

## Examples

■ names of keys of metabase

► Depending on the metabase configuration, returns the names of the metabase keys, such as:

- LM
- Schema.

■ names of keys of key "/LM" of metabase

► Depending on the metabase configuration, returns the key names in the metabase "/LM" key, such as IISADMIN, W3SVC or MimeMap.

---

## Metabase Type

The type identifier of the data associated with a metabase key value.

## Creation Methods

Key Phrase	Form	Description
type of <metabase value>	<i>Plain</i>	Returns the type of the specified metabase value.  Win:4.1

## Properties

Key Phrase	Form	Return Type	Description
<metabase type> as integer	<i>Cast</i>	<integer>	Meanings same as registry types: <ul style="list-style-type: none"> <li>• 1 = DWORD</li> <li>• 2 = STRING</li> <li>• 3 = BINARY</li> <li>• 4 = EXPANDSZ</li> <li>• 5 = MULTISZ.</li> </ul> Win:4.1

Key Phrase	Form	Return Type	Description
<metabase type> as string	<i>Cast</i>	<string>	Returns metabase types as strings: <ul style="list-style-type: none"> <li>• "DWord"</li> <li>• "String"</li> <li>• "Binary"</li> <li>• "ExpandSz"</li> <li>• "MultiSz".</li> </ul> Win:4.1

## Operators

Key phrase	Return Type	Description
<metabase type> = <metabase type>	<boolean>	Compares two metabase types for equality.  Win:4.1

## Metabase User Type

The user type is a DWORD that specifies how the property value is used. User types enable IIS to classify properties by application.

## Creation Methods

Key Phrase	Form	Description
user type of <metabase value>	<i>Plain</i>	Returns the user type of the specified metabase value.  Win:4.1

## Properties

Key Phrase	Form	Return Type	Description
<metabase user type> as integer	<i>Cast</i>	<integer>	Returns the metabase user type as an integer: <ul style="list-style-type: none"> <li>• 2 = Server</li> <li>• 2 = File</li> <li>• 100 = WAM</li> <li>• 200 = ASP App.</li> </ul> Win:4.1

Key Phrase	Form	Return Type	Description
<metabase user type> as string	<i>Cast</i>	<string>	Returns the metabase user type as a string: <ul style="list-style-type: none"> <li>• "Server"</li> <li>• "File"</li> <li>• "WAM"</li> <li>• "ASP App"</li> <li>• "(Other)".</li> </ul> Win:4.1

## Operators

Key phrase	Return Type	Description
<metabase user type> = <metabase user type>	<boolean>	Compares two metabase user types for equality. Win:4.1

## Metabase Value

This Inspector is used to access values stored in an IIS metabase key. The type of the data stored in the value determines what casting operations are allowed. There are several casting Inspectors that you can use to extract values from the registry.

## Creation Methods

Key Phrase	Form	Description
value of <metabase key>	<i>Plain</i>	Returns the value of the specified metabase key. Win:4.1

## Properties

Key Phrase	Form	Return Type	Description
<metabase value> as integer	<i>Cast</i>	<integer>	Returns the integer value of the metabase value. Win:4.1
<metabase value> as string	<i>Cast</i>	<string>	Returns the string value of the metabase value. Win:4.1
identifier of <metabase value>	<i>Plain</i>	<metabase identifier>	Returns the metabase identifier of the specified value. Win:4.1

Key Phrase	Form	Return Type	Description
inherit attribute of <metabase value>	<i>Plain</i>	<boolean>	Returns TRUE if the specified metabase value has the inherit attribute set.  Win:4.1
insert path attribute of <metabase value>	<i>Plain</i>	<boolean>	Returns TRUE if the specified metabase value has the insert path attribute set.  Win:4.1
reference attribute of <metabase value>	<i>Plain</i>	<boolean>	Returns TRUE if the specified metabase value has the reference attribute set.  Win:4.1
secure attribute of <metabase value>	<i>Plain</i>	<boolean>	Returns TRUE if the specified metabase value has the secure attribute set.  Win:4.1
type of <metabase value>	<i>Plain</i>	<metabase type>	Returns the type of the specified metabase value.  Win:4.1
user type of <metabase value>	<i>Plain</i>	<metabase user type>	Returns the user type of the specified metabase value.  Win:4.1
volatile attribute of <metabase value>	<i>Plain</i>	<boolean>	Returns TRUE if the specified metabase value has the volatile attribute set.  Win:4.1

## Examples

- inherit attributes of values of key "/Schema" of metabase
  - Returns a boolean True or False depending on the inherit attributes of each sub-key in the specified key of the metabase.
- volatile attributes of values of key "/LM" of metabase
  - Returns a boolean True or False depending on the volatile attributes of each sub-key in the specified key of the metabase.

# Introspectors

These Inspectors look into the currently installed relevance engine to retrieve information about specific Inspectors.

## Type

Some Inspectors look at the Relevance language itself, inspecting the Inspectors, so to speak. There are several aspects to view, including the types, properties, casts and operators. This group of Inspectors looks at the various type options available from the Relevance language.

## Creation Methods

Key Phrase	Form	Description
direct object type of <property>	<i>Plain</i>	The type (if any) required after the keyword "of" in an expression using the property. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
index type of <property>	<i>Plain</i>	The type (if any) required before or without the keyword "of" in an expression using the property. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
left operand type of <binary operator>	<i>Plain</i>	The type required before the operator in an expression. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
operand type of <cast>	<i>Plain</i>	The type required before the keyword "as" in an expression using the cast. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
operand type of <unary operator>	<i>Plain</i>	The type required in an expression using the operator. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
parent of <type>	<i>Plain</i>	The types (if any) whose properties are inherited by this type. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
result type of <binary operator>	<i>Plain</i>	The type that the operator produces. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
result type of <property>	<i>Plain</i>	The type that the property produces. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
result type of <unary operator>	<i>Plain</i>	The type that the operator produces. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
right operand type of	<i>Plain</i>	The type required after the operator in an expression.

Key Phrase	Form	Description
<binary operator>		Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
type	<i>PlainGlobal</i>	The inspector types. Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
type <string>	<i>NamedGlobal</i>	The type with the given name. Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1

## Properties

Key Phrase	Form	Return Type	Description
<type> as string	<i>Cast</i>	<string>	A string indicating the type. Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
cast from of <type>	<i>Plain</i>	<cast>	Returns the casts that can be created from the specified <type>. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
name of <type>	<i>Plain</i>	<string>	A string naming the type. Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
parent of <type>	<i>Plain</i>	<type>	The types (if any) whose properties are inherited by this type. Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
property <string> of <type>	<i>Named</i>	<property>	Returns the Inspector property of the specified string and type. Typically there is more than one property, so this is often used in the plural. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
property of <type>	<i>Plain</i>	<property>	Returns the Inspector property of the specified type. Typically there is more than one property, so this is often used in the plural. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
property returning <type> of <type>	<i>Indexed</i>	<property>	Returns Inspectors of the form <type> of <type>. Typically there is more than one property, so this is often used in the plural. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
size of <type>	<i>Plain</i>	<integer>	The number of bytes used in the internal representation of an object of the given type. Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1

## Operators

Key phrase	Return Type	Description
<type> = <type>	<boolean>	Returns TRUE if both expressions denote the same type.  Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1

## Examples

- properties "lines" of type "file"
- ▶ Returns the various line properties of the "file" type.
  
- properties returning (type "file line") of type "file"
- ▶ Returns the list of properties that return the <type> specified given an object of type <type>.

---

## Property

Some Inspectors look at the Relevance language itself, inspecting the Inspectors, so to speak. There are several aspects to view, including the types, properties, casts and operators. This group of Inspectors looks at the properties available from the Relevance language.

## Creation Methods

Key Phrase	Form	Description
property	<i>PlainGlobal</i>	The inspectors invoked with phrases, but without the keyword "as".  Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
property <string>	<i>NamedGlobal</i>	Returns the first inspector property whose name matches the given string. Note that there may be more than one property with a given name.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
property <string> of <type>	<i>Named</i>	Returns the Inspector property of the specified string and type. Typically there is more than one property, so this is often used in the plural.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
property of <type>	<i>Plain</i>	Returns the Inspector property of the specified type. Typically there is more than one property, so this is typically used in the plural.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

Key Phrase	Form	Description
property returning <type>	<i>IndexedGlobal</i>	Produces a list of the Inspector properties that return the "file" type.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
property returning <type> of <type>	<i>Indexed</i>	Returns Inspectors of the form <type> of <type>. Typically there is more than one property, so this is often used in the plural.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Properties

Key Phrase	Form	Return Type	Description
<property> as string	<i>Cast</i>	<string>	A short description of the use of the property.  Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
direct object type of <property>	<i>Plain</i>	<type>	The type (if any) required after the keyword "of" in an expression using the property.  Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
index type of <property>	<i>Plain</i>	<type>	The type (if any) required before or without the keyword "of" in an expression using the property.  Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
multivalued of <property>	<i>Plain</i>	<boolean>	Can the property have more than one value for a single input?.  Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
plural name of <property>	<i>Plain</i>	<string>	The name of the property, in the plural.  Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
result type of <property>	<i>Plain</i>	<type>	The type that the property produces.  Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
singular name of <property>	<i>Plain</i>	<string>	The name of the property, in the singular.  Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
usual name of <property>	<i>Plain</i>	<string>	Returns the usual name of the specified property.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Examples

- `property "first matches"`
- Returns the list of properties accessed by the string provided.

- usual name of property "booleans"
- Returns "boolean".

---

## Binary Operator

Some Inspectors look at the Relevance language itself, inspecting the Inspectors, so to speak. There are several aspects to view, including the types, properties, casts and operators. This group of Inspectors looks at the various binary operators available from the Relevance language.

### Creation Methods

Key Phrase	Form	Description
binary operator	<i>PlainGlobal</i>	The inspectors that have two parameters, and are invoked with punctuation marks or reserved phrases. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
binary operator <string>	<i>NamedGlobal</i>	Typically used in the plural, returns the various possible binary inspectors that use the specified operators. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
binary operator returning <type>	<i>IndexedGlobal</i>	Returns a list of binary operators that return the specified type. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

### Properties

Key Phrase	Form	Return Type	Description
<binary operator> as string	<i>Cast</i>	<string>	A short description of the use of the operator. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
left operand type of <binary operator>	<i>Plain</i>	<type>	The type required before the operator in an expression. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
name of <binary operator>	<i>Plain</i>	<string>	A phrase naming the operator. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
result type of <binary operator>	<i>Plain</i>	<type>	The type that the operator produces. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
right operand type of <binary operator>	<i>Plain</i>	<type>	The type required after the operator in an expression. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>

Key Phrase	Form	Return Type	Description
symbol of <binary operator>	<i>Plain</i>	<string>	A phrase or punctuation mark used to invoke the operator.  Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1

## Examples

- binary operators "&"
- Returns a list of all the concatenation inspectors available.

---

## Unary Operator

Some Inspectors look at the Relevance language itself, inspecting the Inspectors, so to speak. There are several aspects to view, including the types, properties, casts and operators. This group of Inspectors looks at the various unary operators available from the Relevance language.

## Creation Methods

Key Phrase	Form	Description
unary operator	<i>PlainGlobal</i>	The inspectors that have one parameter, and are invoked with punctuation marks or reserved phrases.  Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
unary operator <string>	<i>NamedGlobal</i>	Typically used in the plural, this inspector returns a list of objects that use the specified operator.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0
unary operator returning <type>	<i>IndexedGlobal</i>	Returns a list of the unary operator inspectors (such as negative) that return the specified type.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Properties

Key Phrase	Form	Return Type	Description
<unary operator> as string	<i>Cast</i>	<string>	A short description of the use of the operator.  Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
name of <unary operator>	<i>Plain</i>	<string>	A phrase naming the operator.  Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1

Key Phrase	Form	Return Type	Description
operand type of <unary operator>	<i>Plain</i>	<type>	The type required in an expression using the operator. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
result type of <unary operator>	<i>Plain</i>	<type>	The type that the operator produces. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
symbol of <unary operator>	<i>Plain</i>	<string>	A phrase or punctuation mark used to invoke the operator. <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>

## Examples

■ unary operators "-"

► Returns a list of the objects that can be made negative, such as integers, floating point numbers, etc.

■ unary operators returning (type "hertz")

► Returns - <hertz>: hertz, the only unary operator that returns a hertz object.

---

## Cast

Some Inspectors look at the Relevance language itself, inspecting the Inspectors, so to speak. There are several aspects to view, including the types, properties, casts and operators. This group of Inspectors looks at the various casting operations available from the Relevance language.

## Creation Methods

Key Phrase	Form	Description
cast	<i>PlainGlobal</i>	The inspectors invoked using the keyword "as". <small>Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1</small>
cast <string>	<i>NamedGlobal</i>	Returns a list of the objects that can be cast into the type specified by <string>. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>
cast from of <type>	<i>Plain</i>	Returns the casts that can be created from the specified <type>. <small>Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0</small>

Key Phrase	Form	Description
cast returning <type>	<i>IndexedGlobal</i>	Returns a list of the objects that can be cast into the specified type.  Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0

## Properties

Key Phrase	Form	Return Type	Description
<cast> as string	<i>Cast</i>	<string>	A short description of the use of the cast.  Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
name of <cast>	<i>Plain</i>	<string>	The phrase used after the keyword "as" in an expression using the cast.  Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1
operand type of <cast>	<i>Plain</i>	<type>	The type required before the keyword "as" in an expression using the cast.  Win:5.1, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:4.1

## Examples

- `casts "integer"`
  - Returns a list of the objects that can be cast as integers, eg., <string> as integer, <integer> as integer, etc.
- `casts returning (type "integer")`
  - Returns.

## Key Phrases (Inspectors)

This chapter provides an alphabetical list of the Inspector keywords and their casting operators. Both lists include the context object type (From an object), and the resulting object type (Creates an object). These lists are not all-inclusive; they only include those Inspectors that are relevant to the context of the current Guide.

### Key phrases

This is a list of the key phrases relevant to this document, sorted alphabetically.

Key Phrase	Plural	Creates a	From a	Form
absolute value of <hertz>	absolute values	<hertz>	<hertz>	<i>Plain</i>
absolute value of <integer>	absolute values	<integer>	<integer>	<i>Plain</i>
absolute value of <time interval>	absolute values	<time interval>	<time interval>	<i>Plain</i>
action <integer> of <bes fixlet>	actions	<bes fixlet action>	<bes fixlet>	<i>Numbered</i>
action <string> of <bes fixlet>	actions	<bes fixlet action>	<bes fixlet>	<i>Named</i>
action dependency of <bes action>	action dependencies	<bes action>	<bes action>	<i>Plain</i>
action of <bes action result>	actions	<bes action>	<bes action result>	<i>Plain</i>
action of <bes fixlet>	actions	<bes fixlet action>	<bes fixlet>	<i>Plain</i>
action result of <bes computer>	action results	<bes action result>	<bes computer>	<i>Plain</i>
action script of <bes action>	action scripts	<string>	<bes action>	<i>Plain</i>
action script type of <bes action>	action script types	<string>	<bes action>	<i>Plain</i>
activation of <bes fixlet>	activations	<bes activation>	<bes fixlet>	<i>Plain</i>
active flag of <bes activation>	active flags	<boolean>	<bes activation>	<i>Plain</i>
all computer count	all computer counts	<historical computer count>	<world>	<i>PlainGlobal</i>
all fixlet count	all fixlet counts	<historical fixlet count>	<world>	<i>PlainGlobal</i>
analysis flag of <bes fixlet>	analysis flags	<boolean>	<bes fixlet>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
analysis flag of <bes property>	analysis flags	<boolean>	<bes property>	<i>Plain</i>
analysis of <bes activation>	analyses	<bes fixlet>	<bes activation>	<i>Plain</i>
applicability relevance of <bes action>	applicability relevances	<string>	<bes action>	<i>Plain</i>
applicable computer count of <bes fixlet>	applicable computer counts	<integer>	<bes fixlet>	<i>Plain</i>
applicable computer of <bes fixlet>	applicable computers	<bes computer>	<bes fixlet>	<i>Plain</i>
apply count of <bes action result>	apply counts	<integer>	<bes action result>	<i>Plain</i>
april	aprils	<month>	<world>	<i>PlainGlobal</i>
april <integer>	aprils	<day of year>	<world>	<i>NumberedGlobal</i>
april <integer> of <integer>	aprils	<date>	<integer>	<i>Numbered</i>
april of <integer>	aprils	<month and year>	<integer>	<i>Plain</i>
attribute <integer> of <xml dom node>	attributes	<xml dom node>	<xml dom node>	<i>Numbered</i>
attribute <string> of <xml dom node>	attributes	<xml dom node>	<xml dom node>	<i>Named</i>
attribute of <xml dom node>	attributes	<xml dom node>	<xml dom node>	<i>Plain</i>
august	augusts	<month>	<world>	<i>PlainGlobal</i>
august <integer>	augusts	<day of year>	<world>	<i>NumberedGlobal</i>
august <integer> of <integer>	augusts	<date>	<integer>	<i>Numbered</i>
august of <integer>	augusts	<month and year>	<integer>	<i>Plain</i>
baseline flag of <bes fixlet>	baseline flags	<boolean>	<bes fixlet>	<i>Plain</i>
bes action	bes actions	<bes action>	<world>	<i>PlainGlobal</i>
bes action status constrained	bes action statuses constrained	<bes action status>	<world>	<i>PlainGlobal</i>
bes action status download failed	bes action statuses download failed	<bes action status>	<world>	<i>PlainGlobal</i>
bes action status error	bes action statuses error	<bes action status>	<world>	<i>PlainGlobal</i>
bes action status evaluating	bes action statuses evaluating	<bes action status>	<world>	<i>PlainGlobal</i>

Key Phrase	Plural	Creates a	From a	Form
bes action status expired	bes action statuses expired	<bes action status>	<world>	<i>PlainGlobal</i>
bes action status failed	bes action statuses failed	<bes action status>	<world>	<i>PlainGlobal</i>
bes action status fixed	bes action statuses fixed	<bes action status>	<world>	<i>PlainGlobal</i>
bes action status invalid signature	bes action statuses invalid signature	<bes action status>	<world>	<i>PlainGlobal</i>
bes action status irrelevant	bes action statuses irrelevant	<bes action status>	<world>	<i>PlainGlobal</i>
bes action status locked	bes action statuses locked	<bes action status>	<world>	<i>PlainGlobal</i>
bes action status pending downloads	bes action statuses pending downloads	<bes action status>	<world>	<i>PlainGlobal</i>
bes action status pending login	bes action statuses pending login	<bes action status>	<world>	<i>PlainGlobal</i>
bes action status pending message	bes action statuses pending message	<bes action status>	<world>	<i>PlainGlobal</i>
bes action status pending restart	bes action statuses pending restart	<bes action status>	<world>	<i>PlainGlobal</i>
bes action status postponed	bes action statuses postponed	<bes action status>	<world>	<i>PlainGlobal</i>
bes action status running	bes action statuses running	<bes action status>	<world>	<i>PlainGlobal</i>
bes action status unreported	bes action statuses unreported	<bes action status>	<world>	<i>PlainGlobal</i>
bes action status user cancelled	bes action statuses user cancelled	<bes action status>	<world>	<i>PlainGlobal</i>
bes action status waiting	bes action statuses waiting	<bes action status>	<world>	<i>PlainGlobal</i>
bes computer	bes computers	<bes computer>	<world>	<i>PlainGlobal</i>
bes custom site	bes custom sites	<bes custom site>	<world>	<i>PlainGlobal</i>
bes fixlet	bes fixlets	<bes fixlet>	<world>	<i>PlainGlobal</i>
bes property	bes properties	<bes property>	<world>	<i>PlainGlobal</i>
bes property <string>	bes properties	<bes property>	<world>	<i>NamedGlobal</i>
bes site	bes sites	<bes site>	<world>	<i>PlainGlobal</i>

Key Phrase	Plural	Creates a	From a	Form
bes user	bes users	<bes user>	<world>	<i>PlainGlobal</i>
bes wizard	bes wizards	<bes wizard>	<world>	<i>PlainGlobal</i>
best activation of <bes fixlet>	best activations	<bes activation>	<bes fixlet>	<i>Plain</i>
bin at <time> of <statistic range>	bins at	<statistical bin>	<statistic range>	<i>Indexed</i>
bin of <statistic range>	bins	<statistical bin>	<statistic range>	<i>Plain</i>
binary operator <string>	binary operators	<binary operator>	<world>	<i>NamedGlobal</i>
binary operator returning <type>	binary operators returning	<binary operator>	<world>	<i>IndexedGlobal</i>
bit <integer>	bits	<bit set>	<world>	<i>NumberedGlobal</i>
bit <integer> of <bit set>	bits	<boolean>	<bit set>	<i>Numbered</i>
bit <integer> of <integer>	bits	<boolean>	<integer>	<i>Numbered</i>
bit set <string>	bit sets	<bit set>	<world>	<i>NamedGlobal</i>
body of <bes fixlet>	bodies	<html>	<bes fixlet>	<i>Plain</i>
boolean <string>	booleans	<boolean>	<world>	<i>NamedGlobal</i>
case insensitive regex <string>	case insensitive regexes	<regular expression>	<world>	<i>NamedGlobal</i>
case insensitive regular expression <string>	case insensitive regular expressions	<regular expression>	<world>	<i>NamedGlobal</i>
cast <string>	casts	<cast>	<world>	<i>NamedGlobal</i>
cast from of <type>	casts from	<cast>	<type>	<i>Plain</i>
cast returning <type>	casts returning	<cast>	<world>	<i>IndexedGlobal</i>
category of <bes fixlet>	categories	<string>	<bes fixlet>	<i>Plain</i>
character <integer>	characters	<string>	<world>	<i>NumberedGlobal</i>
character <integer> of <string>	characters	<substring>	<string>	<i>Numbered</i>
character of <string>	characters	<substring>	<string>	<i>Plain</i>
charset of <bes fixlet>	charsets	<string>	<bes fixlet>	<i>Plain</i>
charset of <bes wizard>	charsets	<string>	<bes wizard>	<i>Plain</i>
child node <integer> of <xml dom node>	child nodes	<xml dom node>	<xml dom node>	<i>Numbered</i>
child node of <xml dom node>	child nodes	<xml dom node>	<xml dom node>	<i>Plain</i>
components xml of <bes fixlet>	components xmls	<string>	<bes fixlet>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
computer group flag of <bes action>	computer group flags	<boolean>	<bes action>	<i>Plain</i>
computer of <bes action result>	computers	<bes computer>	<bes action result>	<i>Plain</i>
computer of <bes fixlet result>	computers	<bes computer>	<bes fixlet result>	<i>Plain</i>
computer of <bes property result>	computers	<bes computer>	<bes property result>	<i>Plain</i>
concatenation <string> of <string>	concatenations	<string>	<string>	<i>Named</i>
concatenation of <string>	concatenations	<string>	<string>	<i>Plain</i>
conjunction of <boolean>	conjunctions	<boolean>	<boolean>	<i>Plain</i>
constrain by property name of <bes action>	constrain by property names	<string>	<bes action>	<i>Plain</i>
constrain by property relation of <bes action>	constrain by property relations	<string>	<bes action>	<i>Plain</i>
constrain by property value of <bes action>	constrain by property values	<string>	<bes action>	<i>Plain</i>
content id of <bes fixlet action>	content ids	<string>	<bes fixlet action>	<i>Plain</i>
correlation coefficient of <exponential projection>	correlation coefficients	<floating point>	<exponential projection>	<i>Plain</i>
correlation coefficient of <linear projection>	correlation coefficients	<floating point>	<linear projection>	<i>Plain</i>
count map of <historical fixlet count>	count maps	<fixlet count pair>	<historical fixlet count>	<i>Plain</i>
count of <fixlet count pair>	counts	<integer>	<fixlet count pair>	<i>Plain</i>
count of <historical computer count>	counts	<integer>	<historical computer count>	<i>Plain</i>
creation date of <bes custom site>	creation dates	<time>	<bes custom site>	<i>Plain</i>
creation time of <bes user>	creation times	<time>	<bes user>	<i>Plain</i>
creator of <bes custom site>	creators	<bes user>	<bes custom site>	<i>Plain</i>
current analysis	current analyses	<bes fixlet>	<world>	<i>PlainGlobal</i>
current computer	current computers	<bes computer>	<world>	<i>PlainGlobal</i>
current console user	current console users	<bes user>	<world>	<i>PlainGlobal</i>
current date	current dates	<date>	<world>	<i>PlainGlobal</i>

Key Phrase	Plural	Creates a	From a	Form
current day_of_month	current days_of_month	<day of month>	<world>	<i>PlainGlobal</i>
current day_of_week	current days_of_week	<day of week>	<world>	<i>PlainGlobal</i>
current day_of_year	current days_of_year	<day of year>	<world>	<i>PlainGlobal</i>
current fixlet	current fixlets	<bes fixlet>	<world>	<i>PlainGlobal</i>
current month	current months	<month>	<world>	<i>PlainGlobal</i>
current month_and_year	current months_and_years	<month and year>	<world>	<i>PlainGlobal</i>
current task	current tasks	<bes fixlet>	<world>	<i>PlainGlobal</i>
current wizard	current wizards	<bes wizard>	<world>	<i>PlainGlobal</i>
current year	current years	<year>	<world>	<i>PlainGlobal</i>
custom content flag of <bes user>	custom content flags	<boolean>	<bes user>	<i>Plain</i>
custom flag of <bes fixlet>	custom flags	<boolean>	<bes fixlet>	<i>Plain</i>
custom flag of <bes property>	custom flags	<boolean>	<bes property>	<i>Plain</i>
custom site flag of <bes fixlet>	custom site flags	<boolean>	<bes fixlet>	<i>Plain</i>
custom site of <bes fixlet>	custom sites	<bes custom site>	<bes fixlet>	<i>Plain</i>
custom success relevance of <bes action>	custom success relevances	<string>	<bes action>	<i>Plain</i>
cve id list of <bes fixlet>	cve id lists	<string>	<bes fixlet>	<i>Plain</i>
database id of <bes action>	database ids	<integer>	<bes action>	<i>Plain</i>
database id of <bes activation>	database ids	<integer>	<bes activation>	<i>Plain</i>
database id of <bes computer>	database ids	<integer>	<bes computer>	<i>Plain</i>
database id of <bes property>	database ids	<integer>	<bes property>	<i>Plain</i>
database id of <bes wizard>	database ids	<integer>	<bes wizard>	<i>Plain</i>
database id of <historical computer count>	database ids	<integer>	<historical computer count>	<i>Plain</i>
database id of <historical fixlet count>	database ids	<integer>	<historical fixlet count>	<i>Plain</i>
database name of <bes action>	database names	<string>	<bes action>	<i>Plain</i>
database name of <bes computer>	database names	<string>	<bes computer>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
database name of <bes wizard>	database names	<string>	<bes wizard>	<i>Plain</i>
date <string>	dates	<date>	<world>	<i>NamedGlobal</i>
date <time zone> of <time>	dates	<date>	<time>	<i>Indexed</i>
date range end of <bes action>	date range ends	<date>	<bes action>	<i>Plain</i>
date range start of <bes action>	date range starts	<date>	<bes action>	<i>Plain</i>
day	days	<time interval>	<world>	<i>PlainGlobal</i>
day of <day of year>	days	<day of month>	<day of year>	<i>Plain</i>
day_of_month <integer>	days_of_month	<day of month>	<world>	<i>NumberedGlobal</i>
day_of_month <string>	days_of_month	<day of month>	<world>	<i>NamedGlobal</i>
day_of_month of <date>	days_of_month	<day of month>	<date>	<i>Plain</i>
day_of_week <string>	days_of_week	<day of week>	<world>	<i>NamedGlobal</i>
day_of_week of <date>	days_of_week	<day of week>	<date>	<i>Plain</i>
day_of_year of <date>	days_of_year	<day of year>	<date>	<i>Plain</i>
december	decembers	<month>	<world>	<i>PlainGlobal</i>
december <integer>	decembers	<day of year>	<world>	<i>NumberedGlobal</i>
december <integer> of <integer>	decembers	<date>	<integer>	<i>Numbered</i>
december of <integer>	decembers	<month and year>	<integer>	<i>Plain</i>
default action of <bes fixlet>	default actions	<bes fixlet action>	<bes fixlet>	<i>Plain</i>
default flag of <bes property>	default flags	<boolean>	<bes property>	<i>Plain</i>
default page name of <bes wizard>	default page names	<string>	<bes wizard>	<i>Plain</i>
definition of <bes property>	definitions	<string>	<bes property>	<i>Plain</i>
description of <bes custom site>	descriptions	<string>	<bes custom site>	<i>Plain</i>
detailed status of <bes action result>	detailed statuses	<string>	<bes action result>	<i>Plain</i>
dialog flag of <bes wizard>	dialog flags	<boolean>	<bes wizard>	<i>Plain</i>
digest file name of <bes fixlet>	digest file names	<string>	<bes fixlet>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
direct object type of <property>	direct object types	<type>	<property>	<i>Plain</i>
disjunction of <boolean>	disjunctions	<boolean>	<boolean>	<i>Plain</i>
divided by zero of <floating point>	divided by zeroes	<boolean>	<floating point>	<i>Plain</i>
document flag of <bes wizard>	document flags	<boolean>	<bes wizard>	<i>Plain</i>
download size of <bes fixlet>	download sizes	<integer>	<bes fixlet>	<i>Plain</i>
end date of <bes action>	end dates	<date>	<bes action>	<i>Plain</i>
end flag of <bes action>	end flags	<boolean>	<bes action>	<i>Plain</i>
end of <statistic range>	ends	<time>	<statistic range>	<i>Plain</i>
end of <statistical bin>	ends	<time>	<statistical bin>	<i>Plain</i>
end of <substring>	ends	<string position>	<substring>	<i>Plain</i>
end of <time range>	ends	<time>	<time range>	<i>Plain</i>
end time_of_day of <bes action>	end times_of_day	<time of day>	<bes action>	<i>Plain</i>
error <string>	errors	<undefined>	<world>	<i>NamedGlobal</i>
error flag of <bes property result>	error flags	<boolean>	<bes property result>	<i>Plain</i>
error message of <bes property result>	error messages	<string>	<bes property result>	<i>Plain</i>
evaluation period of <bes property>	evaluation periods	<time interval>	<bes property>	<i>Plain</i>
expiration flag of <bes action>	expiration flags	<boolean>	<bes action>	<i>Plain</i>
expiration time of <bes action>	expiration times	<time>	<bes action>	<i>Plain</i>
exponential fit of <statistical bin>	exponential fits	<exponential projection>	<statistical bin>	<i>Plain</i>
extrapolation <time> of <exponential projection>	extrapolations	<floating point>	<exponential projection>	<i>Indexed</i>
extrapolation <time> of <linear projection>	extrapolations	<floating point>	<linear projection>	<i>Indexed</i>
failure rate of <statistical bin>	failure rates	<floating point>	<statistical bin>	<i>Plain</i>
false	falses	<boolean>	<world>	<i>PlainGlobal</i>
february	februarys	<month>	<world>	<i>PlainGlobal</i>
february <integer>	februarys	<day of year>	<world>	<i>NumberedGlobal</i>
february <integer> of <integer>	februarys	<date>	<integer>	<i>Numbered</i>

Key Phrase	Plural	Creates a	From a	Form
february of <integer>	februarys	<month and year>	<integer>	<i>Plain</i>
final part <time interval> of <time range>	final parts	<time range>	<time range>	<i>Indexed</i>
finite of <floating point>	finites	<boolean>	<floating point>	<i>Plain</i>
first <day of week> of <month and year>	firsts	<date>	<month and year>	<i>Indexed</i>
first <integer> of <string>	firsts	<substring>	<string>	<i>Numbered</i>
first <string> of <string>	firsts	<substring>	<string>	<i>Named</i>
first became relevant of <bes fixlet result>	first became relevants	<time>	<bes fixlet result>	<i>Plain</i>
first child of <xml dom node>	first children	<xml dom node>	<xml dom node>	<i>Plain</i>
first friday of <month and year>	first fridays	<date>	<month and year>	<i>Plain</i>
first match <regular expression> of <string>	first matches	<regular expression match>	<string>	<i>Indexed</i>
first monday of <month and year>	first mondays	<date>	<month and year>	<i>Plain</i>
first saturday of <month and year>	first saturdays	<date>	<month and year>	<i>Plain</i>
first sunday of <month and year>	first sundays	<date>	<month and year>	<i>Plain</i>
first thursday of <month and year>	first thursdays	<date>	<month and year>	<i>Plain</i>
first tuesday of <month and year>	first tuesdays	<date>	<month and year>	<i>Plain</i>
first wednesday of <month and year>	first wednesdays	<date>	<month and year>	<i>Plain</i>
fixlet <integer> of <bes site>	fixlets	<bes fixlet>	<bes site>	<i>Numbered</i>
fixlet flag of <bes fixlet>	fixlet flags	<boolean>	<bes fixlet>	<i>Plain</i>
fixlet of <bes fixlet result>	fixlets	<bes fixlet>	<bes fixlet result>	<i>Plain</i>
fixlet of <bes site>	fixlets	<bes fixlet>	<bes site>	<i>Plain</i>
floating point <string>	floating points	<floating point>	<world>	<i>NamedGlobal</i>
following text of <string position>	following texts	<substring>	<string position>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
following text of <substring>	following texts	<substring>	<substring>	<i>Plain</i>
friday	fridays	<day of week>	<world>	<i>PlainGlobal</i>
geometric mean of <statistical bin>	geometric means	<floating point>	<statistical bin>	<i>Plain</i>
ghz	ghzs	<hertz>	<world>	<i>PlainGlobal</i>
globally visible flag of <bes fixlet>	globally visible flags	<boolean>	<bes fixlet>	<i>Plain</i>
greatest hz	greatest hzs	<hertz>	<world>	<i>PlainGlobal</i>
greatest integer	greatest integers	<integer>	<world>	<i>PlainGlobal</i>
greatest time interval	greatest time intervals	<time interval>	<world>	<i>PlainGlobal</i>
group flag of <bes fixlet>	group flags	<boolean>	<bes fixlet>	<i>Plain</i>
group member flag of <bes action>	group member flags	<boolean>	<bes action>	<i>Plain</i>
hexadecimal integer <string>	hexadecimal integers	<integer>	<world>	<i>NamedGlobal</i>
hexadecimal string <string>	hexadecimal strings	<string>	<world>	<i>NamedGlobal</i>
hidden bes action	hidden bes actions	<bes action>	<world>	<i>PlainGlobal</i>
hidden flag of <bes action>	hidden flags	<boolean>	<bes action>	<i>Plain</i>
hour	hours	<time interval>	<world>	<i>PlainGlobal</i>
hour_of_day of <time of day with time zone>	hours_of_day	<integer>	<time of day with time zone>	<i>Plain</i>
hour_of_day of <time of day>	hours_of_day	<integer>	<time of day>	<i>Plain</i>
hz	hzs	<hertz>	<world>	<i>PlainGlobal</i>
id of <bes action>	ids	<integer>	<bes action>	<i>Plain</i>
id of <bes activation>	ids	<integer>	<bes activation>	<i>Plain</i>
id of <bes computer>	ids	<integer>	<bes computer>	<i>Plain</i>
id of <bes fixlet>	ids	<integer>	<bes fixlet>	<i>Plain</i>
id of <bes property>	ids	<integer>	<bes property>	<i>Plain</i>
id of <bes site>	ids	<integer>	<bes site>	<i>Plain</i>
in console context	in console contexts	<boolean>	<world>	<i>PlainGlobal</i>
in web reports context	in web reports contexts	<boolean>	<world>	<i>PlainGlobal</i>
index type of <property>	index types	<type>	<property>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
inexact of <floating point>	inexacts	<boolean>	<floating point>	<i>Plain</i>
infinite of <floating point>	infinities	<boolean>	<floating point>	<i>Plain</i>
initial part <time interval> of <time range>	initial parts	<time range>	<time range>	<i>Indexed</i>
integer <integer>	integers	<integer>	<world>	<i>NumberedGlobal</i>
integer <string>	integers	<integer>	<world>	<i>NamedGlobal</i>
integer ceiling of <floating point>	integer ceilings	<integer>	<floating point>	<i>Plain</i>
integer floor of <floating point>	integer floors	<integer>	<floating point>	<i>Plain</i>
invalid of <floating point>	invalids	<boolean>	<floating point>	<i>Plain</i>
issuer of <bes action>	issuers	<bes user>	<bes action>	<i>Plain</i>
issuer of <bes activation>	issuers	<bes user>	<bes activation>	<i>Plain</i>
issuer of <bes fixlet>	issuers	<bes user>	<bes fixlet>	<i>Plain</i>
january	januaries	<month>	<world>	<i>PlainGlobal</i>
january <integer>	januaries	<day of year>	<world>	<i>NumberedGlobal</i>
january <integer> of <integer>	januaries	<date>	<integer>	<i>Numbered</i>
january of <integer>	januaries	<month and year>	<integer>	<i>Plain</i>
javascript array <string> of <statistical bin>	javascript arrays	<html>	<statistical bin>	<i>Named</i>
july	julys	<month>	<world>	<i>PlainGlobal</i>
july <integer>	julys	<day of year>	<world>	<i>NumberedGlobal</i>
july <integer> of <integer>	julys	<date>	<integer>	<i>Numbered</i>
july of <integer>	julys	<month and year>	<integer>	<i>Plain</i>
june	junes	<month>	<world>	<i>PlainGlobal</i>
june <integer>	junes	<day of year>	<world>	<i>NumberedGlobal</i>
june <integer> of <integer>	junes	<date>	<integer>	<i>Numbered</i>
june of <integer>	junes	<month and year>	<integer>	<i>Plain</i>
khz	khzs	<hertz>	<world>	<i>PlainGlobal</i>

Key Phrase	Plural	Creates a	From a	Form
kurtosis of <statistical bin>	kurtoses	<floating point>	<statistical bin>	<i>Plain</i>
last <integer> of <string>	lasts	<substring>	<string>	<i>Numbered</i>
last <string> of <string>	lasts	<substring>	<string>	<i>Named</i>
last became nonrelevant of <bes fixlet result>	last became nonrelevants	<time>	<bes fixlet result>	<i>Plain</i>
last became relevant of <bes fixlet result>	last became relevants	<time>	<bes fixlet result>	<i>Plain</i>
last child of <xml dom node>	last children	<xml dom node>	<xml dom node>	<i>Plain</i>
last login time of <bes user>	last login times	<time>	<bes user>	<i>Plain</i>
last report time of <bes computer>	last report times	<time>	<bes computer>	<i>Plain</i>
leap of <year>	leaps	<boolean>	<year>	<i>Plain</i>
least hz	least hzs	<hertz>	<world>	<i>PlainGlobal</i>
least integer	least integers	<integer>	<world>	<i>PlainGlobal</i>
least significant one bit of <bit set>	least significant one bits	<integer>	<bit set>	<i>Plain</i>
least time interval	least time intervals	<time interval>	<world>	<i>PlainGlobal</i>
left operand type of <binary operator>	left operand types	<type>	<binary operator>	<i>Plain</i>
left shift <integer> of <bit set>	left shifts	<bit set>	<bit set>	<i>Numbered</i>
length of <month and year>	lengths	<time interval>	<month and year>	<i>Plain</i>
length of <rope>	lengths	<integer>	<rope>	<i>Plain</i>
length of <statistical bin>	lengths	<time interval>	<statistical bin>	<i>Plain</i>
length of <string>	lengths	<integer>	<string>	<i>Plain</i>
length of <time range>	lengths	<time interval>	<time range>	<i>Plain</i>
length of <year>	lengths	<time interval>	<year>	<i>Plain</i>
less significance <integer> of <floating point>	less significances	<floating point>	<floating point>	<i>Numbered</i>
line number of <bes action result>	line numbers	<integer>	<bes action result>	<i>Plain</i>
linear fit of <statistical bin>	linear fits	<linear projection>	<statistical bin>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
link <html> of <bes action>	links	<html>	<bes action>	<i>Indexed</i>
link <html> of <bes computer>	links	<html>	<bes computer>	<i>Indexed</i>
link <html> of <bes fixlet>	links	<html>	<bes fixlet>	<i>Indexed</i>
link <html> of <bes user>	links	<html>	<bes user>	<i>Indexed</i>
link <html> of <bes wizard>	links	<html>	<bes wizard>	<i>Indexed</i>
link <string> of <bes action>	links	<html>	<bes action>	<i>Named</i>
link <string> of <bes computer>	links	<html>	<bes computer>	<i>Named</i>
link <string> of <bes fixlet>	links	<html>	<bes fixlet>	<i>Named</i>
link <string> of <bes user>	links	<html>	<bes user>	<i>Named</i>
link <string> of <bes wizard>	links	<html>	<bes wizard>	<i>Named</i>
link href of <bes action>	link hrefs	<string>	<bes action>	<i>Plain</i>
link href of <bes computer>	link hrefs	<string>	<bes computer>	<i>Plain</i>
link href of <bes fixlet>	link hrefs	<string>	<bes fixlet>	<i>Plain</i>
link href of <bes user>	link hrefs	<string>	<bes user>	<i>Plain</i>
link href of <bes wizard>	link hrefs	<string>	<bes wizard>	<i>Plain</i>
link of <bes action>	links	<html>	<bes action>	<i>Plain</i>
link of <bes computer>	links	<html>	<bes computer>	<i>Plain</i>
link of <bes fixlet>	links	<html>	<bes fixlet>	<i>Plain</i>
link of <bes user>	links	<html>	<bes user>	<i>Plain</i>
link of <bes wizard>	links	<html>	<bes wizard>	<i>Plain</i>
local time <string>	local times	<time>	<world>	<i>NamedGlobal</i>
local time zone	local time zones	<time zone>	<world>	<i>PlainGlobal</i>
locally visible flag of <bes fixlet>	locally visible flags	<boolean>	<bes fixlet>	<i>Plain</i>
logarithm kurtosis of <statistical bin>	logarithm kurtoses	<floating point>	<statistical bin>	<i>Plain</i>
logarithm skewness of <statistical bin>	logarithm skewnesses	<floating point>	<statistical bin>	<i>Plain</i>
logarithm standard deviation of <statistical bin>	logarithm standard deviations	<floating point>	<statistical bin>	<i>Plain</i>
logarithm variance of <statistical bin>	logarithm variances	<floating point>	<statistical bin>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
management rights flag of <bes action>	management rights flags	<boolean>	<bes action>	<i>Plain</i>
march	marches	<month>	<world>	<i>PlainGlobal</i>
march <integer>	marches	<day of year>	<world>	<i>NumberedGlobal</i>
march <integer> of <integer>	marches	<date>	<integer>	<i>Numbered</i>
march of <integer>	marches	<month and year>	<integer>	<i>Plain</i>
master flag of <bes user>	master flags	<boolean>	<bes user>	<i>Plain</i>
master site flag of <bes fixlet>	master site flags	<boolean>	<bes fixlet>	<i>Plain</i>
match <regular expression> of <string>	matches	<regular expression match>	<string>	<i>Indexed</i>
maximum of <integer>	maxima	<integer>	<integer>	<i>Plain</i>
maximum of <time interval>	maxima	<time interval>	<time interval>	<i>Plain</i>
maximum of <time>	maxima	<time>	<time>	<i>Plain</i>
maximum single computer total of <statistical bin>	maximum single computer totals	<floating point>	<statistical bin>	<i>Plain</i>
maximum value of <statistical bin>	maximum values	<floating point>	<statistical bin>	<i>Plain</i>
may	mays	<month>	<world>	<i>PlainGlobal</i>
may <integer>	mays	<day of year>	<world>	<i>NumberedGlobal</i>
may <integer> of <integer>	mays	<date>	<integer>	<i>Numbered</i>
may of <integer>	mays	<month and year>	<integer>	<i>Plain</i>
mean computer count of <statistical bin>	mean computer counts	<floating point>	<statistical bin>	<i>Plain</i>
mean failing computer count of <statistical bin>	mean failing computer counts	<floating point>	<statistical bin>	<i>Plain</i>
mean logarithm of <statistical bin>	mean logarithms	<floating point>	<statistical bin>	<i>Plain</i>
mean nonzero value count of <statistical bin>	mean nonzero value counts	<floating point>	<statistical bin>	<i>Plain</i>
mean of <floating point>	means	<floating point>	<floating point>	<i>Plain</i>
mean of <integer>	means	<floating point>	<integer>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
mean of <statistical bin>	means	<floating point>	<statistical bin>	<i>Plain</i>
mean sample interval of <statistical bin>	mean sample intervals	<time interval>	<statistical bin>	<i>Plain</i>
mean sample rate of <statistical bin>	mean sample rates	<rate>	<statistical bin>	<i>Plain</i>
mean successful computer count of <statistical bin>	mean successful computer counts	<floating point>	<statistical bin>	<i>Plain</i>
mean total of <statistical bin>	mean totals	<floating point>	<statistical bin>	<i>Plain</i>
mean value count of <statistical bin>	mean value counts	<floating point>	<statistical bin>	<i>Plain</i>
mean zero value count of <statistical bin>	mean zero value counts	<floating point>	<statistical bin>	<i>Plain</i>
menu path of <bes wizard>	menu paths	<string>	<bes wizard>	<i>Plain</i>
message action button flag of <bes action>	message action button flags	<boolean>	<bes action>	<i>Plain</i>
message allow cancel flag of <bes action>	message allow cancel flags	<boolean>	<bes action>	<i>Plain</i>
message of <bes fixlet>	messages	<html>	<bes fixlet>	<i>Plain</i>
message postpone delay of <bes action>	message postpone delays	<time interval>	<bes action>	<i>Plain</i>
message text of <bes action>	message texts	<string>	<bes action>	<i>Plain</i>
message timeout delay of <bes action>	message timeout delays	<time interval>	<bes action>	<i>Plain</i>
message title of <bes action>	message titles	<string>	<bes action>	<i>Plain</i>
mhz	mhzs	<hertz>	<world>	<i>PlainGlobal</i>
microsecond	microseconds	<time interval>	<world>	<i>PlainGlobal</i>
middle action of <bes action>	middle actions	<bes action>	<bes action>	<i>Plain</i>
midnight	midnights	<time of day>	<world>	<i>PlainGlobal</i>
millisecond	milliseconds	<time interval>	<world>	<i>PlainGlobal</i>
minimum of <integer>	minima	<integer>	<integer>	<i>Plain</i>
minimum of <time interval>	minima	<time interval>	<time interval>	<i>Plain</i>
minimum of <time>	minima	<time>	<time>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
minimum single computer total of <statistical bin>	minimum single computer totals	<floating point>	<statistical bin>	<i>Plain</i>
minimum value of <statistical bin>	minimum values	<floating point>	<statistical bin>	<i>Plain</i>
minute	minutes	<time interval>	<world>	<i>PlainGlobal</i>
minute_of_hour of <time of day with time zone>	minutes_of_hour	<integer>	<time of day with time zone>	<i>Plain</i>
minute_of_hour of <time of day>	minutes_of_hour	<integer>	<time of day>	<i>Plain</i>
module <string>	modules	<module>	<world>	<i>NamedGlobal</i>
monday	mondays	<day of week>	<world>	<i>PlainGlobal</i>
month	months	<number of months>	<world>	<i>PlainGlobal</i>
month <integer>	months	<month>	<world>	<i>NumberedGlobal</i>
month <string>	months	<month>	<world>	<i>NamedGlobal</i>
month of <date>	months	<month>	<date>	<i>Plain</i>
month of <day of year>	months	<month>	<day of year>	<i>Plain</i>
month of <month and year>	months	<month>	<month and year>	<i>Plain</i>
month_and_year of <date>	months_and_years	<month and year>	<date>	<i>Plain</i>
more significance <integer> of <floating point>	more significances	<floating point>	<floating point>	<i>Numbered</i>
most significant one bit of <bit set>	most significant one bits	<integer>	<bit set>	<i>Plain</i>
multiple flag of <bes action>	multiple flags	<boolean>	<bes action>	<i>Plain</i>
multiplicity of <integer with multiplicity>	multiplicities	<integer>	<integer with multiplicity>	<i>Plain</i>
multiplicity of <string with multiplicity>	multiplicities	<integer>	<string with multiplicity>	<i>Plain</i>
multivalued of <property>	multivalueds	<boolean>	<property>	<i>Plain</i>
name of <bes action>	names	<string>	<bes action>	<i>Plain</i>
name of <bes activation>	names	<string>	<bes activation>	<i>Plain</i>
name of <bes computer>	names	<string>	<bes computer>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
name of <bes custom site>	names	<string>	<bes custom site>	Plain
name of <bes fixlet>	names	<string>	<bes fixlet>	Plain
name of <bes property>	names	<string>	<bes property>	Plain
name of <bes site>	names	<string>	<bes site>	Plain
name of <bes user>	names	<string>	<bes user>	Plain
name of <bes wizard>	names	<string>	<bes wizard>	Plain
name of <binary operator>	names	<string>	<binary operator>	Plain
name of <cast>	names	<string>	<cast>	Plain
name of <type>	names	<string>	<type>	Plain
name of <unary operator>	names	<string>	<unary operator>	Plain
nan of <floating point>	nans	<boolean>	<floating point>	Plain
navbar name of <bes wizard>	navbar names	<string>	<bes wizard>	Plain
next sibling of <xml dom node>	next siblings	<xml dom node>	<xml dom node>	Plain
node name of <xml dom node>	node names	<string>	<xml dom node>	Plain
node type of <xml dom node>	node types	<integer>	<xml dom node>	Plain
node value of <xml dom node>	node values	<string>	<xml dom node>	Plain
noon	noons	<time of day>	<world>	PlainGlobal
normal of <floating point>	normals	<boolean>	<floating point>	Plain
november	novembers	<month>	<world>	PlainGlobal
november <integer>	novembers	<day of year>	<world>	NumberedGlobal
november <integer> of <integer>	novembers	<date>	<integer>	Numbered
november of <integer>	novembers	<month and year>	<integer>	Plain
now	nows	<time>	<world>	PlainGlobal
numeric value of <string>	numeric values	<integer>	<string>	Plain
october	octobers	<month>	<world>	PlainGlobal
october <integer>	octobers	<day of year>	<world>	NumberedGlobal

Key Phrase	Plural	Creates a	From a	Form
october <integer> of <integer>	octobers	<date>	<integer>	<i>Numbered</i>
october of <integer>	octobers	<month and year>	<integer>	<i>Plain</i>
one bit of <bit set>	one bits	<integer>	<bit set>	<i>Plain</i>
open action count of <bes fixlet>	open action counts	<integer>	<bes fixlet>	<i>Plain</i>
operand type of <cast>	operand types	<type>	<cast>	<i>Plain</i>
operand type of <unary operator>	operand types	<type>	<unary operator>	<i>Plain</i>
operator site flag of <bes action>	operator site flags	<boolean>	<bes action>	<i>Plain</i>
operator site flag of <bes fixlet>	operator site flags	<boolean>	<bes fixlet>	<i>Plain</i>
overflow of <floating point>	overflows	<boolean>	<floating point>	<i>Plain</i>
owner document of <xml dom node>	owner documents	<xml dom document>	<xml dom node>	<i>Plain</i>
owner flag <bes user> of <bes custom site>	owner flags	<boolean>	<bes custom site>	<i>Indexed</i>
owner of <bes custom site>	owners	<bes user>	<bes custom site>	<i>Plain</i>
parent node of <xml dom node>	parent nodes	<xml dom node>	<xml dom node>	<i>Plain</i>
parent of <type>	parents	<type>	<type>	<i>Plain</i>
parenthesized part <integer> of <regular expression match>	parenthesized parts	<substring>	<regular expression match>	<i>Numbered</i>
parenthesized part of <regular expression match>	parenthesized parts	<substring>	<regular expression match>	<i>Plain</i>
plural flag of <bes property result>	plural flags	<boolean>	<bes property result>	<i>Plain</i>
plural name of <property>	plural names	<string>	<property>	<i>Plain</i>
position <integer> of <string>	positions	<string position>	<string>	<i>Numbered</i>
position of <string>	positions	<string position>	<string>	<i>Plain</i>
postaction allow cancel flag of <bes action>	postaction allow cancel flags	<boolean>	<bes action>	<i>Plain</i>
postaction force delay of <bes action>	postaction force delays	<time interval>	<bes action>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
postaction message text of <bes action>	postaction message texts	<string>	<bes action>	<i>Plain</i>
postaction message title of <bes action>	postaction message titles	<string>	<bes action>	<i>Plain</i>
postaction postpone delay of <bes action>	postaction postpone delays	<time interval>	<bes action>	<i>Plain</i>
pre60 flag of <bes wizard>	pre60 flags	<boolean>	<bes wizard>	<i>Plain</i>
preceding text of <string position>	preceding texts	<substring>	<string position>	<i>Plain</i>
preceding text of <substring>	preceding texts	<substring>	<substring>	<i>Plain</i>
previous sibling of <xml dom node>	previous siblings	<xml dom node>	<xml dom node>	<i>Plain</i>
product of <integer>	products	<integer>	<integer>	<i>Plain</i>
property <integer> of <bes fixlet>	properties	<bes property>	<bes fixlet>	<i>Numbered</i>
property <string>	properties	<property>	<world>	<i>NamedGlobal</i>
property <string> of <type>	properties	<property>	<type>	<i>Named</i>
property of <bes fixlet>	properties	<bes property>	<bes fixlet>	<i>Plain</i>
property of <bes property result>	properties	<bes property>	<bes property result>	<i>Plain</i>
property of <type>	properties	<property>	<type>	<i>Plain</i>
property result of <bes computer>	property results	<bes property result>	<bes computer>	<i>Plain</i>
property returning <type>	properties returning	<property>	<world>	<i>IndexedGlobal</i>
property returning <type> of <type>	properties returning	<property>	<type>	<i>Indexed</i>
range <time range> of <statistic range>	ranges	<statistic range>	<statistic range>	<i>Indexed</i>
range after <time> of <time range>	ranges after	<time range>	<time range>	<i>Indexed</i>
range before <time> of <time range>	ranges before	<time range>	<time range>	<i>Indexed</i>
rate <time interval> of <exponential projection>	rates	<floating point>	<exponential projection>	<i>Indexed</i>
rate of <linear projection>	rates	<rate>	<linear projection>	<i>Plain</i>
reader of <bes custom site>	readers	<bes user>	<bes custom site>	<i>Plain</i>
reapplication limit of <bes action>	reapplication limits	<integer>	<bes action>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
regex <string>	regexes	<regular expression>	<world>	<i>NamedGlobal</i>
regular expression <string>	regular expressions	<regular expression>	<world>	<i>NamedGlobal</i>
relative significance place <integer> of <floating point>	relative significance places	<floating point>	<floating point>	<i>Numbered</i>
relative significance place of <floating point>	relative significance places	<floating point>	<floating point>	<i>Plain</i>
relevance of <bes fixlet>	relevances	<string>	<bes fixlet>	<i>Plain</i>
relevant <( bes computer, bes fixlet )>	relevants	<boolean>	<world>	<i>IndexedGlobal</i>
relevant <( bes fixlet, bes computer )>	relevants	<boolean>	<world>	<i>IndexedGlobal</i>
relevant <bes computer> of <bes fixlet>	relevants	<boolean>	<bes fixlet>	<i>Indexed</i>
relevant <bes fixlet> of <bes computer>	relevants	<boolean>	<bes computer>	<i>Indexed</i>
relevant fixlet of <bes computer>	relevant fixlets	<bes fixlet>	<bes computer>	<i>Plain</i>
relevant flag of <bes fixlet result>	relevant flags	<boolean>	<bes fixlet result>	<i>Plain</i>
require user absence of <bes action>	require user absences	<boolean>	<bes action>	<i>Plain</i>
require user presence of <bes action>	require user presences	<boolean>	<bes action>	<i>Plain</i>
requires authoring flag of <bes wizard>	requires authoring flags	<boolean>	<bes wizard>	<i>Plain</i>
reserved flag of <bes property>	reserved flags	<boolean>	<bes property>	<i>Plain</i>
restart flag of <bes action>	restart flags	<boolean>	<bes action>	<i>Plain</i>
result <( bes action, bes computer )>	results	<bes action result>	<world>	<i>IndexedGlobal</i>
result <( bes computer, bes action )>	results	<bes action result>	<world>	<i>IndexedGlobal</i>
result <( bes computer, bes property )>	results	<bes property result>	<world>	<i>IndexedGlobal</i>
result <( bes property, bes computer )>	results	<bes property result>	<world>	<i>IndexedGlobal</i>
result from <bes action> of <bes computer>	results from	<bes action result>	<bes computer>	<i>Indexed</i>
result from <bes computer> of <bes action>	results from	<bes action result>	<bes action>	<i>Indexed</i>
result from <bes computer> of <bes property>	results from	<bes property result>	<bes property>	<i>Indexed</i>

Key Phrase	Plural	Creates a	From a	Form
result from <bes property> of <bes computer>	results from	<bes property result>	<bes computer>	<i>Indexed</i>
result of <bes action>	results	<bes action result>	<bes action>	<i>Plain</i>
result of <bes fixlet>	results	<bes fixlet result>	<bes fixlet>	<i>Plain</i>
result of <bes property>	results	<bes property result>	<bes property>	<i>Plain</i>
result type of <binary operator>	result types	<type>	<binary operator>	<i>Plain</i>
result type of <property>	result types	<type>	<property>	<i>Plain</i>
result type of <unary operator>	result types	<type>	<unary operator>	<i>Plain</i>
retry count of <bes action result>	retry counts	<integer>	<bes action result>	<i>Plain</i>
retry delay of <bes action>	retry delays	<time interval>	<bes action>	<i>Plain</i>
retry limit of <bes action>	retry limits	<integer>	<bes action>	<i>Plain</i>
right operand type of <binary operator>	right operand types	<type>	<binary operator>	<i>Plain</i>
right shift <integer> of <bit set>	right shifts	<bit set>	<bit set>	<i>Numbered</i>
rope <string>	ropes	<rope>	<world>	<i>NamedGlobal</i>
running message text of <bes action>	running message texts	<string>	<bes action>	<i>Plain</i>
running message title of <bes action>	running message titles	<string>	<bes action>	<i>Plain</i>
sans id list of <bes fixlet>	sans id lists	<string>	<bes fixlet>	<i>Plain</i>
saturday	saturdays	<day of week>	<world>	<i>PlainGlobal</i>
script of <bes fixlet action>	scripts	<string>	<bes fixlet action>	<i>Plain</i>
script type of <bes fixlet action>	script types	<string>	<bes fixlet action>	<i>Plain</i>
second	seconds	<time interval>	<world>	<i>PlainGlobal</i>
second_of_minute of <time of day with time zone>	seconds_of_minute	<integer>	<time of day with time zone>	<i>Plain</i>
second_of_minute of <time of day>	seconds_of_minute	<integer>	<time of day>	<i>Plain</i>
select <string> of <xml dom node>	selects	<xml dom node>	<xml dom node>	<i>Named</i>

Key Phrase	Plural	Creates a	From a	Form
selected groups string of <bes action>	selected groups strings	<string>	<bes action>	<i>Plain</i>
september	septembers	<month>	<world>	<i>PlainGlobal</i>
september <integer>	septembers	<day of year>	<world>	<i>NumberedGlobal</i>
september <integer> of <integer>	septembers	<date>	<integer>	<i>Numbered</i>
september of <integer>	septembers	<month and year>	<integer>	<i>Plain</i>
settings flag of <bes action>	settings flags	<boolean>	<bes action>	<i>Plain</i>
show message flag of <bes action>	show message flags	<boolean>	<bes action>	<i>Plain</i>
show running message flag of <bes action>	show running message flags	<boolean>	<bes action>	<i>Plain</i>
shutdown flag of <bes action>	shutdown flags	<boolean>	<bes action>	<i>Plain</i>
significance place <integer> of <floating point>	significance places	<floating point>	<floating point>	<i>Numbered</i>
significance place of <floating point>	significance places	<floating point>	<floating point>	<i>Plain</i>
significance threshold of <floating point>	significance thresholds	<floating point>	<floating point>	<i>Plain</i>
significant digits <integer> of <hertz>	significant digitss	<hertz>	<hertz>	<i>Numbered</i>
significant digits <integer> of <integer>	significant digitss	<integer>	<integer>	<i>Numbered</i>
single flag of <bes action>	single flags	<boolean>	<bes action>	<i>Plain</i>
singular name of <property>	singular names	<string>	<property>	<i>Plain</i>
site of <bes fixlet>	sites	<bes site>	<bes fixlet>	<i>Plain</i>
size of <type>	sizes	<integer>	<type>	<i>Plain</i>
skewness of <statistical bin>	skewnesses	<floating point>	<statistical bin>	<i>Plain</i>
source analysis of <bes property>	source analyses	<bes fixlet>	<bes property>	<i>Plain</i>
source evaluation period of <bes property>	source evaluation periods	<time interval>	<bes property>	<i>Plain</i>
source fixlet of <bes action>	source fixlets	<bes fixlet>	<bes action>	<i>Plain</i>
source id of <bes fixlet>	source ids	<string>	<bes fixlet>	<i>Plain</i>
source id of <bes property>	source ids	<integer>	<bes property>	<i>Plain</i>
source name of <bes property>	source names	<string>	<bes property>	<i>Plain</i>
source of <bes fixlet>	sources	<string>	<bes fixlet>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
source release date of <bes fixlet>	source release dates	<date>	<bes fixlet>	<i>Plain</i>
source relevance of <bes action>	source relevances	<string>	<bes action>	<i>Plain</i>
source severity of <bes fixlet>	source severities	<string>	<bes fixlet>	<i>Plain</i>
source severity of <fixlet count pair>	source severitys	<string>	<fixlet count pair>	<i>Plain</i>
standard deviation of <floating point>	standard deviations	<floating point>	<floating point>	<i>Plain</i>
standard deviation of <integer>	standard deviations	<floating point>	<integer>	<i>Plain</i>
standard deviation of <statistical bin>	standard deviations	<floating point>	<statistical bin>	<i>Plain</i>
start date of <bes action>	start dates	<date>	<bes action>	<i>Plain</i>
start flag of <bes action>	start flags	<boolean>	<bes action>	<i>Plain</i>
start of <statistic range>	starts	<time>	<statistic range>	<i>Plain</i>
start of <statistical bin>	starts	<time>	<statistical bin>	<i>Plain</i>
start of <substring>	starts	<string position>	<substring>	<i>Plain</i>
start of <time range>	starts	<time>	<time range>	<i>Plain</i>
start time_of_day of <bes action>	start times_of_day	<time of day>	<bes action>	<i>Plain</i>
state of <bes action>	states	<string>	<bes action>	<i>Plain</i>
statistic range of <bes property>	statistic ranges	<statistic range>	<bes property>	<i>Plain</i>
status of <bes action result>	statuses	<bes action status>	<bes action result>	<i>Plain</i>
string <string>	strings	<string>	<world>	<i>NamedGlobal</i>
subscription flag of <bes action>	subscription flags	<boolean>	<bes action>	<i>Plain</i>
substring <string> of <string>	substrings	<substring>	<string>	<i>Named</i>
substring after <string> of <string>	substrings after	<substring>	<string>	<i>Named</i>
substring before <string> of <string>	substrings before	<substring>	<string>	<i>Named</i>
substring between <string> of <string>	substrings between	<substring>	<string>	<i>Named</i>
substring separated by <string> of <string>	substrings separated by	<substring>	<string>	<i>Named</i>
success on custom relevance of <bes action>	success on custom relevances	<boolean>	<bes action>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
success on original relevance of <bes action>	success on original relevances	<boolean>	<bes action>	<i>Plain</i>
success on run to completion of <bes action>	success on run to completions	<boolean>	<bes action>	<i>Plain</i>
success rate of <statistical bin>	success rates	<floating point>	<statistical bin>	<i>Plain</i>
sum of <integer>	sums	<integer>	<integer>	<i>Plain</i>
sunday	sundays	<day of week>	<world>	<i>PlainGlobal</i>
symbol of <binary operator>	symbols	<string>	<binary operator>	<i>Plain</i>
symbol of <unary operator>	symbols	<string>	<unary operator>	<i>Plain</i>
targeted by id flag of <bes action>	targeted by id flags	<boolean>	<bes action>	<i>Plain</i>
targeted by list flag of <bes action>	targeted by list flags	<boolean>	<bes action>	<i>Plain</i>
targeted by property flag of <bes action>	targeted by property flags	<boolean>	<bes action>	<i>Plain</i>
targeted computer of <bes action>	targeted computers	<bes computer>	<bes action>	<i>Plain</i>
targeted list of <bes action>	targeted lists	<string>	<bes action>	<i>Plain</i>
targeted name of <bes action>	targeted names	<string>	<bes action>	<i>Plain</i>
targeting method of <bes action>	targeting methods	<string>	<bes action>	<i>Plain</i>
targeting relevance of <bes action>	targeting relevances	<string>	<bes action>	<i>Plain</i>
task flag of <bes fixlet>	task flags	<boolean>	<bes fixlet>	<i>Plain</i>
temporal distribution of <bes action>	temporal distributions	<time interval>	<bes action>	<i>Plain</i>
thursday	thursdays	<day of week>	<world>	<i>PlainGlobal</i>
time <string>	times	<time>	<world>	<i>NamedGlobal</i>
time <time zone> of <time>	times	<time of day with time zone>	<time>	<i>Indexed</i>
time interval <string>	time intervals	<time interval>	<world>	<i>NamedGlobal</i>
time issued of <bes action>	times issued	<time>	<bes action>	<i>Plain</i>
time of <historical computer count>	times	<time>	<historical computer count>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
time of <historical fixlet count>	times	<time>	<historical fixlet count>	<i>Plain</i>
time of <time of day with time zone>	times	<time of day>	<time of day with time zone>	<i>Plain</i>
time range end of <bes action>	time range ends	<time of day>	<bes action>	<i>Plain</i>
time range start of <bes action>	time range starts	<time of day>	<bes action>	<i>Plain</i>
time zone <string>	time zones	<time zone>	<world>	<i>NamedGlobal</i>
time_of_day <string>	times_of_day	<time of day>	<world>	<i>NamedGlobal</i>
top level bes action	top level bes actions	<bes action>	<world>	<i>PlainGlobal</i>
total <time interval> of <statistic range>	totals	<statistical bin>	<statistic range>	<i>Indexed</i>
total lower bound of <statistical bin>	total lower bounds	<floating point>	<statistical bin>	<i>Plain</i>
total of <statistic range>	totals	<statistical bin>	<statistic range>	<i>Plain</i>
total upper bound of <statistical bin>	total upper bounds	<floating point>	<statistical bin>	<i>Plain</i>
true	true	<boolean>	<world>	<i>PlainGlobal</i>
tuesday	tuesdays	<day of week>	<world>	<i>PlainGlobal</i>
tuple string item <integer> of <string>	tuple string items	<string>	<string>	<i>Numbered</i>
tuple string item of <string>	tuple string items	<string>	<string>	<i>Plain</i>
two digit hour of <time of day>	two digit hours	<string>	<time of day>	<i>Plain</i>
two digit minute of <time of day>	two digit minutes	<string>	<time of day>	<i>Plain</i>
two digit second of <time of day>	two digit seconds	<string>	<time of day>	<i>Plain</i>
type of <bes fixlet>	types	<string>	<bes fixlet>	<i>Plain</i>
unary operator <string>	unary operators	<unary operator>	<world>	<i>NamedGlobal</i>
unary operator returning <type>	unary operators returning	<unary operator>	<world>	<i>IndexedGlobal</i>
underflow of <floating point>	underflows	<boolean>	<floating point>	<i>Plain</i>
unique value of <integer>	unique values	<integer with multiplicity>	<integer>	<i>Plain</i>
unique value of <integer>	unique values	<integer>	<integer>	<i>Plain</i>
unique value of <string>	unique values	<string with multiplicity>	<string>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
unique value of <string>	unique values	<string>	<string>	<i>Plain</i>
universal time <string>	universal times	<time>	<world>	<i>NamedGlobal</i>
universal time zone	universal time zones	<time zone>	<world>	<i>PlainGlobal</i>
unlocked computer count of <bes fixlet>	unlocked computer counts	<integer>	<bes fixlet>	<i>Plain</i>
untargeted flag of <bes action>	untargeted flags	<boolean>	<bes action>	<i>Plain</i>
urgent flag of <bes action>	urgent flags	<boolean>	<bes action>	<i>Plain</i>
url of <bes wizard>	urls	<string>	<bes wizard>	<i>Plain</i>
usual name of <property>	usual names	<string>	<property>	<i>Plain</i>
value count of <bes property result>	value counts	<integer>	<bes property result>	<i>Plain</i>
value of <bes property result>	values	<string>	<bes property result>	<i>Plain</i>
variance of <statistical bin>	variances	<floating point>	<statistical bin>	<i>Plain</i>
version string <string> of <module>	version strings	<string>	<module>	<i>Named</i>
wednesday	wednesdays	<day of week>	<world>	<i>PlainGlobal</i>
week	weeks	<time interval>	<world>	<i>PlainGlobal</i>
windows display time <string>	windows display times	<time>	<world>	<i>NamedGlobal</i>
wizard data of <bes fixlet>	wizard datas	<html>	<bes fixlet>	<i>Plain</i>
wizard link of <bes fixlet>	wizard links	<string>	<bes fixlet>	<i>Plain</i>
wizard name of <bes fixlet>	wizard names	<string>	<bes fixlet>	<i>Plain</i>
writer of <bes custom site>	writers	<bes user>	<bes custom site>	<i>Plain</i>
xml document of <string>	xml documents	<xml dom document>	<string>	<i>Plain</i>
xpath <( string, string )> of <xml dom node>	xpaths	<xml dom node>	<xml dom node>	<i>Indexed</i>
xpath <string> of <xml dom node>	xpaths	<xml dom node>	<xml dom node>	<i>Named</i>
year	years	<number of months>	<world>	<i>PlainGlobal</i>
year <integer>	years	<year>	<world>	<i>NumberedGlobal</i>
year <string>	years	<year>	<world>	<i>NamedGlobal</i>
year of <date>	years	<year>	<date>	<i>Plain</i>

Key Phrase	Plural	Creates a	From a	Form
year of <month and year>	years	<year>	<month and year>	<i>Plain</i>
zone of <time of day with time zone>	zones	<time zone>	<time of day with time zone>	<i>Plain</i>
zoned time_of_day <string>	zoned times_of_day	<time of day with time zone>	<world>	<i>NamedGlobal</i>

## Casting Operators

The casting operators allow you to convert one type to another. This is a list of the casting operators sorted by key phrase.

Key Phrase	Creates a	From a
<action lock state> as string	<string>	<action lock state>
<bes action status> as string	<string>	<bes action status>
<binary operator> as string	<string>	<binary operator>
<bios> as string	<string>	<bios>
<bit set> as integer	<integer>	<bit set>
<bit set> as string	<string>	<bit set>
<boolean> as boolean	<boolean>	<boolean>
<boolean> as string	<string>	<boolean>
<cast> as string	<string>	<cast>
<date> as string	<string>	<date>
<day of month> as integer	<integer>	<day of month>
<day of month> as string	<string>	<day of month>
<day of month> as two digits	<string>	<day of month>
<day of week> as string	<string>	<day of week>
<day of week> as three letters	<string>	<day of week>

Key Phrase	Creates a	From a
<day of year> as string	<string>	<day of year>
<environment variable> as string	<string>	<environment variable>
<file content> as lowercase	<file content>	<file content>
<file content> as uppercase	<file content>	<file content>
<file> as string	<string>	<file>
<floating point> as integer	<integer>	<floating point>
<floating point> as scientific notation	<string>	<floating point>
<floating point> as standard notation	<string>	<floating point>
<floating point> as string	<string>	<floating point>
<hertz> as string	<string>	<hertz>
<html> as html	<html>	<html>
<html> as string	<string>	<html>
<integer> as bit set	<bit set>	<integer>
<integer> as bits	<bit set>	<integer>
<integer> as day_of_month	<day of month>	<integer>
<integer> as floating point	<floating point>	<integer>
<integer> as hexadecimal	<string>	<integer>
<integer> as integer	<integer>	<integer>
<integer> as month	<month>	<integer>
<integer> as string	<string>	<integer>
<integer> as year	<year>	<integer>
<ipv4 address> as string	<string>	<ipv4 address>
<language> as string	<string>	<language>
<local group member> as string	<string>	<local group member>
<metabase identifier> as integer	<integer>	<metabase identifier>
<metabase identifier> as string	<string>	<metabase identifier>

Key Phrase	Creates a	From a
<metabase type> as integer	<integer>	<metabase type>
<metabase type> as string	<string>	<metabase type>
<metabase user type> as integer	<integer>	<metabase user type>
<metabase user type> as string	<string>	<metabase user type>
<metabase value> as integer	<integer>	<metabase value>
<metabase value> as string	<string>	<metabase value>
<month and year> as string	<string>	<month and year>
<month> as integer	<integer>	<month>
<month> as string	<string>	<month>
<month> as three letters	<string>	<month>
<month> as two digits	<string>	<month>
<number of months> as string	<string>	<number of months>
<operating system> as string	<string>	<operating system>
<primary language> as string	<string>	<primary language>
<property> as string	<string>	<property>
<rate> as string	<string>	<rate>
<registry key value type> as string	<string>	<registry key value type>
<registry key value> as application	<application>	<registry key value>
<registry key value> as file	<file>	<registry key value>
<registry key value> as folder	<folder>	<registry key value>
<registry key value> as integer	<integer>	<registry key value>
<registry key value> as string	<string>	<registry key value>

Key Phrase	Creates a	From a
<registry key value> as system file	<file>	<registry key value>
<registry key value> as time	<time>	<registry key value>
<rope> as string	<string>	<rope>
<security descriptor> as string	<string>	<security descriptor>
<security identifier> as string	<string>	<security identifier>
<service> as string	<string>	<service>
<setting> as string	<string>	<setting>
<string> as boolean	<boolean>	<string>
<string> as date	<date>	<string>
<string> as day_of_month	<day of month>	<string>
<string> as day_of_week	<day of week>	<string>
<string> as floating point	<floating point>	<string>
<string> as hexadecimal	<string>	<string>
<string> as html	<html>	<string>
<string> as integer	<integer>	<string>
<string> as left trimmed string	<string>	<string>
<string> as local time	<time>	<string>
<string> as local zoned time_of_day	<time of day with time zone>	<string>
<string> as lowercase	<string>	<string>
<string> as month	<month>	<string>
<string> as right trimmed string	<string>	<string>
<string> as string	<string>	<string>
<string> as time	<time>	<string>
<string> as time interval	<time interval>	<string>
<string> as time zone	<time zone>	<string>
<string> as time_of_day	<time of day>	<string>

Key Phrase	Creates a	From a
<string> as trimmed string	<string>	<string>
<string> as universal time	<time>	<string>
<string> as universal zoned time_of_day	<time of day with time zone>	<string>
<string> as uppercase	<string>	<string>
<string> as version	<version>	<string>
<string> as windows display time	<time>	<string>
<string> as year	<year>	<string>
<string> as zoned time_of_day	<time of day with time zone>	<string>
<time interval> as string	<string>	<time interval>
<time of day with time zone> as string	<string>	<time of day with time zone>
<time of day> as string	<string>	<time of day>
<time range> as string	<string>	<time range>
<time zone> as string	<string>	<time zone>
<time> as local string	<string>	<time>
<time> as string	<string>	<time>
<time> as universal string	<string>	<time>
<type> as string	<string>	<type>
<unary operator> as string	<string>	<unary operator>
<version> as string	<string>	<version>
<version> as version	<version>	<version>
<wmi object> as string	<string>	<wmi object>
<wmi select> as string	<string>	<wmi select>
<xml dom node> as text	<string>	<xml dom node>
<xml dom node> as xml	<string>	<xml dom node>
<year> as integer	<integer>	<year>
<year> as string	<string>	<year>

# Appendix

## Folders on Windows Machines

On Windows machines, numeric identifiers can be used to locate many system folders. INF files are used to install system software components and device drives. INF files contain sections entitled DestinationDirs. This section is used with the corresponding CopyFiles section to specify destination locations for files placed on the system during the install. The Number identifies the directory. The numbers are sometimes called LDIDs and sometimes called DIRIDs. We call them install folders. Below is a table of install folders and the method BigFix uses to calculate the location.

Install folder#	Name	Calculated using
10	LDID_WIN	GetWindowsDirectory()
11	LDID_SYS	GetSystemDirectory()
12	LDID_IOS	GetSystemDirectory() + "\IOSUBSYS"
13	LDID_CMD	GetWindowsDirectory() + "\COMMAND"
14	LDID_CPL	GetPathFromCSIDL(CSIDL_CONTROLS)
15	LDID_PRINT	GetPathFromCSIDL(CSIDL_PRINTERS)
17	LDID_INF	GetWindowsDirectory() + "\INF"
18	LDID_HELP	GetWindowsDirectory() + "\HELP"
19	LDID_WINADMIN	*Registered Setup folder "WinAdminDir"
20	LDID_FONTS	GetPathFromCSIDL(CSIDL_CSIDL_FONTS)
21	LDID_VIEWERS	GetSystemDirectory() + "\VIEWERS"
22	LDID_VMM32	GetSystemDirectory() + "\VMM32"
23	LDID_COLOR	*Registered Setup folder "ICMPPath"
24	LDID_APPS	*Registered Setup folder "AppsDir"
25	LDID_SHARED	*Registered Setup folder "SharedDir"
26	LDID_WINBOOT	*Registered Setup folder "WinBootDir"
27	LDID_MACHINE	*Registered Setup folder "MachineDir"
28	LDID_HOST_WINBOOT	*Registered Setup folder "HostWinBootDir"
29	LDID_BOOT	*Registered Setup folder "BootDir"
30	LDID_BOOT_HOST	*Registered Setup folder "BootHost"
31	LDID_OLD_WINBOOT	*Registered Setup folder "OldWinBootDir"
32	LDID_OLD_WIN	*Registered Setup folder "OldWinDir"
33	LDID_OLD_DOS	*Registered Setup folder "OldDosDir"

\*Registered Setup folders are stored in the Windows registry under the key:

**HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\SETUP**

An adjustable set of target locations has been added to the Windows Registry under the key:

**HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\SETUP\VarLDID**

Each value stored under this key is a string whose name is the VarLDID and whose value contains a path to a folder. For example, if the value named 28701 contains C:\Program Files, then

**install folder "28701"**

would return a folder corresponding to that location.

---

## Processors

On Windows machines, the Vendor Name, Family, Type, Model, Extended Family, Extended Model and stepping are calculated using the CUID instruction. The results depend upon the processor and the vendor of the processor. The inspectors return values based upon the Intel specification for the CUID instruction. Other vendors or older processors may behave differently. An attempt is made to identify the Family and Family name for processors that do not support the CUID instruction. You can depend upon the vendor name to distinguish the different vendors except that early versions of the 80486 from AMD are completely indistinguishable from an Intel processor. In this case "AmbiguousCPU" is returned for the vendor name. A complete list of bit values returned by the feature masks property is available in the Intel documentation. These can be found online at the Intel web site.

The speed is measured using a timed sequence of instructions. The speed returned may differ from the expected amount by a couple of MHz.

The CUID instruction is executed with 1 in the EAX register to compute:

Stepping	Bits 0-3
Model	Bits 4-7
Family	Bits 8-11
Type	Bits 12-13
Extended Model	Bits 16-19
Extended Family	Bits 20-23

## Processors (continued)

Numeric values returned for family of processor and string values returned by family name of processor are computed using the table below. For an unidentified family name, the “brand string” is returned, if available.

Vendor Name	Family Name	Family	Model	Extended Family
GenuineIntel	8086	0		
	80286	2		
	80386	3		
	80486	4		
	Pentium	5		
	Pentium Pro	6	0-2	
	Pentium II	6	38418	
	Pentium III	6	7 or greater	
	Pentium 4	15	0	0
AuthenticAMD	486	4		
	K5	5	0-5	
	K6	5	6 or greater	
	Athlon	6	1,2,4	
	Duron	6	3	
CyrixInstead	MediaGX	4		
	6x86	5	2	
	GXm	5	Not 2	
	6x86MX	6		
CentaurHauls	C6	5		

## Index

---

### A

- absolute value of <hertz> · 44, 45, 307
- absolute value of <integer> · 3, 10, 12, 307
- absolute value of <time interval> · 59, 60, 307
- access control entry · 255, 256, 260
- access control list · 252, 256, 257, 258, 259, 260, 261, 262
- access mode of <access control entry> · 252
- accessed time of <filesystem object> · 126
- account disabled flag of <local user> · 264
- account expiration of <local user> · 264
- account name of <security identifier> · 263
- accounts operator flag of <local user> · 264
- accuracy of <dmi electrical\_current\_probe> · 207
- accuracy of <dmi temperature\_probe> · 224
- accuracy of <dmi voltage\_probe> · 225
- action · 5, 26, 29, 85, 86, 95, 102, 111, 141, 177, 178, 233, 245, 248, 270, 271, 272, 307, 313, 321, 324, 333
- action <integer> · 85, 270, 307
- action lock state · 85, 333
- Action Objects · v, 2, 270
- active action · 85, 270, 271
- active device · 85, 115, 116, 119, 128, 175, 176, 177
- active device file · 85, 128, 176
- active device file <string> · 85, 128
- active directory · 291, 292
- active directory local computer · 291, 292
- active of <action> · 270
- active start time of <action> · 270
- adapter of <network> · 273, 279
- address list of <network adapter> · 279, 281
- address of <dmi management\_device> · 210
- address of <network adapter> · 276, 279
- address of <network address list> · 276, 282
- address of <network ip interface> · 275, 276
- address\_type of <dmi management\_device> · 210
- admin privilege of <local user> · 264
- administrator <string> of <client> · 236, 237
- administrator of <client> · 236, 237
- all firewall scope · 85, 191
- allow inbound echo request of <firewall icmp settings> · 197
- allow inbound mask request of <firewall icmp settings> · 197
- allow inbound router request of <firewall icmp settings> · 197
- allow inbound timestamp request of <firewall icmp settings> · 197
- allow outbound destination unreachable of <firewall icmp settings> · 197
- allow outbound packet too big of <firewall icmp settings> · 197
- allow outbound parameter problem of <firewall icmp settings> · 197
- allow outbound source quench of <firewall icmp settings> · 197
- allow outbound time exceeded of <firewall icmp settings> · 198
- allow redirect of <firewall icmp settings> · 198
- allowed workstations string of <local user> · 265
- analysis · 307, 308, 311, 328
- ancestor of <filesystem object> · 126, 139
- ansi code page · 85, 112
- any ip version · 86, 200
- apparent registration server time · 86
- append permission of <access control entry> · 252
- application · 2, 86, 88, 92, 93, 103, 104, 105, 107, 109, 110, 111, 112, 113, 115, 116, 117, 118, 119, 120, 121, 128, 135, 136, 137, 138, 139, 141, 145, 146, 151, 152, 154, 165, 166, 185, 186, 191, 236, 243, 244, 246, 247, 249, 250, 265, 296, 335
- application <string> · 86, 115, 117, 136, 141
- application <string> of <folder> · 136, 141
- application <string> of <registry key> · 117, 136
- application <string> of <registry> · 115, 136
- application event log · 86, 112, 243, 244, 246, 247, 249
- application folder <string> of <registry key> · 117, 139
- application folder <string> of <registry> · 115, 139
- application folder of <registry key> · 117, 139
- application of <registry key> · 118, 136
- application of <registry> · 115, 136
- application parameter string of <local user> · 265
- application usage summary · 86, 165, 166
- application usage summary <string> · 86, 165
- april · 62, 65, 70, 73, 77, 80, 86, 308

april <integer> · 62, 70, 86, 308  
april <integer> of <integer> · 62, 308  
april of <integer> · 77, 308  
archive of <filesystem object> · 126  
argument string of <file shortcut> · 146  
asset\_tag\_number of <dmi  
  system\_enclosure\_or\_chassis> · 221  
associativity of <dmi cache\_information> · 206  
attribute <integer> of <xml dom node> · 160,  
  161, 308  
attribute <string> of <xml dom node> · 160, 161,  
  308  
attribute of <xml dom node> · 160, 161, 308  
attribute permission of <network share> · 284  
audit failure event log event type · 86, 248, 249  
audit level of <local mssql database> · 179  
audit success event log event type · 86, 248  
august · 62, 70, 73, 77, 87, 308  
august <integer> · 62, 70, 87, 308  
august <integer> of <integer> · 62, 308  
august of <integer> · 77, 308  
Authorization Objects · v, 2, 252  
authorized application of <firewall profile> · 185,  
  187

---

## **B**

b32\_bit\_memory\_error\_information <integer> of  
  <dmi> · 202  
b32\_bit\_memory\_error\_information of <dmi> ·  
  202  
b64\_bit\_memory\_error\_information <integer> of  
  <dmi> · 203  
b64\_bit\_memory\_error\_information of <dmi> ·  
  203  
backoffice bit <operating system suite mask> · 87  
bad password count of <local user> · 265  
bank\_connections of <dmi  
  memory\_module\_information> · 215  
bank\_locator of <dmi memory\_device> · 213  
base\_board\_information <integer> of <dmi> ·  
  204  
base\_board\_information of <dmi> · 204  
bes action · 307, 308, 309, 311, 312, 313, 314,  
  316, 317, 318, 319, 320, 321, 322, 324, 325,  
  326, 327, 328, 329, 330, 331, 332, 333  
bes action result · 307, 308, 311, 313, 318, 326,  
  327, 329  
bes action status · 308, 309, 329, 333  
bes activation · 307, 308, 310, 312, 316, 317, 322

bes computer · 307, 308, 309, 311, 312, 316, 318,  
  319, 322, 325, 326, 327, 330  
bes custom site · 309, 311, 312, 313, 323, 324,  
  325, 332  
bes fixlet · 307, 308, 309, 310, 311, 312, 313,  
  314, 315, 316, 317, 318, 319, 320, 321, 323,  
  324, 325, 326, 327, 328, 329, 330, 331, 332  
bes fixlet action · 307, 311, 313, 327  
bes fixlet result · 311, 315, 318, 326, 327  
bes property · 308, 309, 311, 312, 313, 314, 316,  
  323, 324, 325, 326, 327, 328, 329, 332  
bes property result · 311, 314, 324, 325, 326, 327,  
  332  
bes site · 309, 315, 316, 323, 328  
bes user · 310, 311, 312, 317, 318, 319, 320, 323,  
  324, 325, 332  
bes wizard · 310, 312, 313, 314, 319, 321, 323,  
  325, 326, 332  
binary operator · 87, 112, 299, 303, 304, 310, 318,  
  323, 327, 330, 333  
binary operator <string> · 87, 303, 310  
binary operator returning <type> · 87, 303, 310  
bios · 2, 167, 168, 204, 205, 333  
bios\_characteristics of <dmi bios\_information> ·  
  204  
bios\_information <integer> of <dmi> · 204  
bios\_information of <dmi> · 204  
bios\_language\_information <integer> of <dmi> ·  
  205  
bios\_language\_information of <dmi> · 205  
bios\_release\_date of <dmi bios\_information> ·  
  204  
bios\_rom\_size of <dmi bios\_information> · 204  
bios\_starting\_address\_segment of <dmi  
  bios\_information> · 204  
bios\_version of <dmi bios\_information> · 204  
bit <integer> · 6, 12, 39, 40, 87, 310  
bit <integer> of <bit set> · 40, 310  
bit <integer> of <integer> · 6, 12, 310  
bit set · 12, 39, 40, 41, 87, 113, 310, 318, 322,  
  324, 327, 333, 334  
bit set <string> · 39, 87, 310  
blade bit <operating system suite mask> · 87

boolean · 5, 6, 7, 8, 9, 12, 14, 20, 21, 22, 24, 26, 31, 39, 40, 41, 42, 45, 49, 52, 56, 67, 69, 72, 75, 80, 81, 83, 84, 87, 88, 91, 92, 93, 95, 97, 102, 103, 105, 106, 108, 122, 123, 125, 126, 127, 143, 144, 149, 151, 166, 170, 177, 179, 180, 185, 187, 188, 189, 192, 194, 196, 197, 198, 199, 201, 228, 234, 235, 240, 249, 252, 253, 254, 255, 256, 257, 258, 259, 264, 265, 266, 267, 268, 270, 271, 272, 274, 275, 276, 278, 279, 281, 282, 283, 284, 285, 289, 290, 294, 296, 297, 298, 301, 302, 303, 307, 308, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 328, 329, 330, 331, 332, 333, 336  
boolean <string> · 6, 87, 310  
boolean value <integer> of <wmi select> · 228  
boolean value of <wmi select> · 228  
boot time of <operating system> · 168  
bootup\_state of <dmi>  
    system\_enclosure\_or\_chassis · 221  
brand id of <processor> · 172  
brand string of <processor> · 172  
broadcast address of <network ip interface> · 275, 276  
broadcast support of <network ip interface> · 275  
build number high of <operating system> · 168  
build number low of <operating system> · 169  
build number of <operating system> · 169  
built in of <firewall open port> · 192  
built\_in\_pointing\_device <integer> of <dmi> · 205  
built\_in\_pointing\_device of <dmi> · 205  
byte <integer> of <file> · 130

---

## C

cache\_configuration of <dmi cache\_information> · 206  
cache\_information <integer> of <dmi> · 206  
cache\_information of <dmi> · 206  
cache\_speed of <dmi cache\_information> · 206  
can interact with desktop of <service> · 180  
capabilities of <dmi system\_reset> · 223  
case insensitive regex <string> · 41, 87, 310  
case insensitive regular expression <string> · 41, 87, 310  
cast · 12, 25, 88, 113, 149, 150, 151, 228, 299, 300, 305, 306, 310, 323, 324, 333  
cast <string> · 88, 305, 310  
cast from of <type> · 300, 305, 310

cast returning <type> · 88, 306, 310  
casts · 2, 10, 19, 113, 299, 300, 301, 303, 304, 305, 306, 310  
category of <event log record> · 245  
change notification permission of <access control entry> · 252  
character <integer> · 26, 29, 34, 88, 310  
character <integer> of <string> · 29, 34, 310  
character of <string> · 29, 34, 310  
checkpoint of <service> · 180  
child node <integer> of <xml dom node> · 160, 161, 310  
child node of <xml dom node> · 160, 162, 310  
class of <active device> · 175  
client · 2, 31, 85, 86, 88, 134, 135, 139, 142, 165, 166, 172, 177, 178, 231, 233, 236, 237, 238, 272  
client folder of <site> · 139, 231  
client license · 88, 177  
Client Objects · v, 2, 236  
code page of <local user> · 265  
codepage of <file version block> · 153  
comment of <local group> · 242  
comment of <local user> · 265  
comment of <network share> · 284  
common name of <license> · 177  
communications bit <operating system suite mask> · 88  
communications operator flag of <local user> · 265  
competition size of <selected server> · 239  
competition weight of <selected server> · 239  
complete time of <action> · 270  
component\_handle of <dmi>  
    management\_device\_component · 210  
compressed of <filesystem object> · 126  
computer id · 88  
computer name · 88  
computer of <event log record> · 245  
concatenation <string> of <string> · 26, 29, 311  
concatenation of <string> · 26, 29, 311  
conjunction of <boolean> · 6, 8, 311  
connection · 88, 89, 90, 99, 100, 113, 273, 282, 286, 287, 288, 289, 290  
connection of <network> · 273, 286  
connection status · 88, 89, 90, 113, 286, 287, 288, 289  
connection status <integer> · 88, 287  
connection status authenticating · 88, 287  
connection status authentication failed · 89, 287

connection status authentication succeeded · 89, 287  
connection status connected · 89, 113, 287  
connection status connecting · 89, 287  
connection status disconnected · 89, 287  
connection status disconnecting · 89, 287  
connection status hardware disabled · 89, 288  
connection status hardware malfunction · 89, 288  
connection status media disconnected · 89, 288  
connection status no hardware present · 90, 288  
connections of <dmi  
    out\_of\_band\_remote\_access> · 217  
constrained of <action> · 270  
content of <file> · 130, 148  
control of <security descriptor> · 262  
Conventions Used in this manual · 3  
cooling\_device <integer> of <dmi> · 207  
cooling\_device of <dmi> · 207  
cooling\_unit\_group of <dmi cooling\_device> · 207  
country code of <local user> · 265  
create file permission of <access control entry> · 252  
create folder permission of <access control entry> · 252  
create link permission of <access control entry> · 253  
create permission of <network share> · 284  
create subkey permission of <access control entry> · 253  
creation time of <filesystem object> · 126  
csd version of <operating system> · 169  
current date · 9, 16, 62, 65, 70, 76, 77, 80, 81, 90, 311  
current day\_of\_month · 68, 70, 90, 312  
current day\_of\_week · 9, 66, 90, 312  
current day\_of\_year · 70, 90, 312  
current firewall profile type · 90, 113, 188  
current month · 9, 16, 73, 76, 77, 81, 90, 312  
current month\_and\_year · 77, 81, 90, 312  
current profile of <firewall policy> · 186, 190  
current profile type of <firewall> · 184, 188  
current site · 90, 134, 231, 233  
current user · 2, 90, 269  
current year · 80, 82, 84, 90, 312  
current\_interleave of <dmi  
    memory\_controller\_information> · 212  
current\_language of <dmi  
    bios\_language\_information> · 205  
current\_memory\_type of <dmi  
    memory\_module\_information> · 215

current\_speed of <dmi  
    memory\_module\_information> · 215  
current\_speed of <dmi processor\_information> · 219  
current\_sram\_type of <dmi cache\_information> · 206  
current\_usage of <dmi system\_slots> · 223  
custom firewall scope · 91, 191  
custom site subscription effective date <string> · 91  
customized of <firewall service> · 194

---

## D

dacl of <security descriptor> · 256, 262  
data\_width of <dmi memory\_device> · 213  
datacenter bit <operating system suite mask> · 91  
date · 1, 9, 15, 24, 26, 32, 48, 49, 55, 56, 57, 58, 62, 63, 64, 65, 66, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 83, 84, 90, 91, 122, 126, 127, 134, 166, 167, 168, 177, 178, 233, 238, 284, 308, 311, 313, 314, 315, 317, 320, 322, 323, 324, 328, 329, 332, 333, 336  
date <string> · 62, 91, 313  
date <time zone> of <time> · 48, 62, 313  
date of <bios> · 167  
day · 9, 12, 27, 28, 29, 32, 50, 51, 52, 54, 55, 56, 58, 59, 60, 61, 62, 63, 64, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 78, 80, 83, 86, 87, 90, 91, 93, 94, 96, 97, 99, 100, 101, 102, 105, 106, 108, 110, 168, 233, 247, 268, 308, 312, 313, 314, 316, 317, 320, 322, 323, 327, 328, 329, 330, 331, 332, 333, 334, 336, 337  
day of <day of year> · 68, 71, 313  
day of month · 12, 27, 64, 68, 69, 71, 75, 80, 90, 91, 312, 313, 333, 334, 336  
day of week · 27, 62, 63, 64, 66, 67, 90, 91, 94, 100, 105, 106, 108, 110, 312, 313, 316, 322, 327, 330, 331, 332, 333, 336  
day of year · 64, 69, 70, 71, 72, 73, 75, 80, 83, 86, 87, 90, 91, 93, 96, 97, 99, 101, 102, 105, 308, 312, 313, 314, 317, 320, 322, 323, 328, 334  
day\_of\_month <integer> · 68, 91, 313  
day\_of\_month <string> · 68, 91, 313  
day\_of\_month of <date> · 64, 68, 313  
day\_of\_week <string> · 66, 91, 313  
day\_of\_week of <date> · 64, 66, 313  
day\_of\_year of <date> · 64, 70, 313  
december · 63, 70, 73, 76, 77, 91, 313  
december <integer> · 63, 70, 91, 313

december <integer> of <integer> · 63, 313  
december of <integer> · 77, 313  
default value of <registry key> · 118, 121  
default web browser · 92, 137  
delete child permission of <access control entry> · 253  
delete permission of <access control entry> · 253  
delete permission of <network share> · 284  
descendant of <folder> · 128, 141  
description of <active device> · 175  
description of <dmi electrical\_current\_probe> · 207  
description of <dmi management\_device\_component> · 210  
description of <dmi management\_device> · 210  
description of <dmi temperature\_probe> · 224  
description of <dmi voltage\_probe> · 225  
description of <event log record> · 245  
description of <network adapter> · 279  
design\_capacity of <dmi portable\_battery> · 218  
design\_capacity\_multiplier of <dmi portable\_battery> · 218  
design\_voltage of <dmi portable\_battery> · 218  
device key <string> of <registry> · 115, 116  
device key of <registry> · 115, 116  
device name of <connection> · 286  
device\_chemistry of <dmi portable\_battery> · 219  
device\_description <integer> of <dmi on\_board\_devices\_information> · 216  
device\_description of <dmi on\_board\_devices\_information> · 216  
device\_error\_address of <dmi b32\_bit\_memory\_error\_information> · 202  
device\_error\_address of <dmi b64\_bit\_memory\_error\_information> · 203  
device\_locator of <dmi memory\_device> · 213  
device\_name of <dmi portable\_battery> · 219  
device\_set of <dmi memory\_device> · 213  
device\_type <integer> of <dmi on\_board\_devices\_information> · 216  
device\_type of <dmi on\_board\_devices\_information> · 216  
device\_type\_and\_status of <dmi cooling\_device> · 207  
dhcp enabled of <network adapter> · 279  
dhcp server of <network adapter> · 277, 279  
direct object type of <property> · 299, 302, 314  
disjunction of <boolean> · 6, 8, 314  
display name of <service> · 180  
distance of <selected server> · 16, 239

distinguished name error message of <active directory local computer> · 291  
distinguished name of <active directory local computer> · 291  
divided by zero of <floating point> · 20, 314  
dmi · 2, 92  
DMI Objects · iv, 2, 202  
dns name · 92  
dns server of <network adapter> · 279, 281  
dns server of <network> · 273, 281  
domain firewall profile type · 92, 188, 189  
domain name of <security identifier> · 263  
domain profile of <firewall policy> · 186, 190  
domain user · 2, 92, 264  
domain user <string> · 92, 264  
drive · 32, 37, 92, 94, 126, 140, 143, 144, 145, 146, 261, 265  
drive <string> · 92, 143  
drive of <filesystem object> · 126, 143  
driver key of <active device> · 116, 175  
driver key of <registry key> · 117, 118  
driver key value name of <active device> · 175

---

## *E*

effective access mode for <string> of <access control list> · 256  
effective access system security permission for <string> of <access control list> · 256  
effective append permission for <string> of <access control list> · 256  
effective change notification permission for <string> of <access control list> · 256  
effective create file permission for <string> of <access control list> · 257  
effective create folder permission for <string> of <access control list> · 257  
effective create link permission for <string> of <access control list> · 257  
effective create subkey permission for <string> of <access control list> · 257  
effective date of <setting> · 238  
effective delete child permission for <string> of <access control list> · 257  
effective delete permission for <string> of <access control list> · 257  
effective enumerate subkeys permission for <string> of <access control list> · 257  
effective execute permission for <string> of <access control list> · 257

- effective generic all permission for <string> of  
  <access control list> · 257
- effective generic execute permission for <string>  
  of <access control list> · 258
- effective generic read permission for <string> of  
  <access control list> · 258
- effective generic write permission for <string> of  
  <access control list> · 258
- effective list permission for <string> of <access  
  control list> · 258
- effective maximum allowed permission for  
  <string> of <access control list> · 258
- effective query value permission for <string> of  
  <access control list> · 258
- effective read attributes permission for <string>  
  of <access control list> · 258
- effective read control permission for <string> of  
  <access control list> · 258
- effective read extended attributes permission for  
  <string> of <access control list> · 258
- effective read permission for <string> of <access  
  control list> · 259
- effective set value permission for <string> of  
  <access control list> · 259
- effective synchronize permission for <string> of  
  <access control list> · 259
- effective traverse permission for <string> of  
  <access control list> · 259
- effective write attributes permission for <string>  
  of <access control list> · 259
- effective write dac permission for <string> of  
  <access control list> · 259
- effective write extended attributes permission for  
  <string> of <access control list> · 259
- effective write owner permission for <string> of  
  <access control list> · 259
- effective write permission for <string> of <access  
  control list> · 259
- electrical\_current\_probe <integer> of <dmi> ·  
  207
- electrical\_current\_probe of <dmi> · 207
- email address of <license> · 177
- embedded nt bit <operating system suite mask> ·  
  92
- embedded restricted bit <operating system suite  
  mask> · 92
- enabled of <firewall authorized application> · 185
- enabled of <firewall open port> · 192
- enabled of <firewall service> · 194
- enabled of <internet connection firewall> · 282
- enabled of <port mapping> · 283
- enabled\_size of <dmi  
  memory\_module\_information> · 215
- end of <substring> · 33, 36, 314
- end of <time range> · 57, 314
- end\_of\_table <integer> of <dmi> · 208
- end\_of\_table of <dmi> · 208
- ending\_address of <dmi  
  memory\_array\_mapped\_address> · 212
- ending\_address of <dmi  
  memory\_device\_mapped\_address> · 214
- enterprise bit <operating system suite mask> · 93
- entry of <access control list> · 252, 260
- enumerate subkeys permission of <access control  
  entry> · 253
- environment · 2, 29, 93, 181, 250, 251, 334
- Environment Objects · v, 2, 250
- environment variable · 2, 29, 93, 250, 251, 334
- error <string> · 44, 93, 314
- error event log event type · 93, 248, 249
- error\_correcting\_capability of <dmi  
  memory\_controller\_information> · 212
- error\_correction\_type of <dmi  
  cache\_information> · 206
- error\_detecting\_method of <dmi  
  memory\_controller\_information> · 212
- error\_granularity of <dmi  
  b32\_bit\_memory\_error\_information> · 202
- error\_granularity of <dmi  
  b64\_bit\_memory\_error\_information> · 203
- error\_operation of <dmi  
  b32\_bit\_memory\_error\_information> · 202
- error\_operation of <dmi  
  b64\_bit\_memory\_error\_information> · 203
- error\_resolution of <dmi  
  b32\_bit\_memory\_error\_information> · 202
- error\_resolution of <dmi  
  b64\_bit\_memory\_error\_information> · 203
- error\_status of <dmi  
  memory\_module\_information> · 215
- error\_type of <dmi  
  b32\_bit\_memory\_error\_information> · 202
- error\_type of <dmi  
  b64\_bit\_memory\_error\_information> · 203
- escape of <string> · 29, 122
- evaluation of <license> · 177
- event id of <event log record> · 245
- event log · 86, 93, 95, 105, 106, 110, 112, 113,  
  243, 244, 245, 246, 247, 248, 249, 263
- event log <string> · 93, 243
- event log event type · 86, 93, 95, 110, 245, 248,  
  249

event log event type <integer> · 93, 248  
 event log record · 244, 245, 246  
 event type of <event log record> · 245, 248  
 exceptions allowed of <firewall profile> · 187  
 executable file format of <file> · 130  
 execute permission of <access control entry> · 253  
 execute permission of <network share> · 284  
 execution · 2, 86, 248, 257, 258, 270  
 expand environment string of <string> · 29  
 expiration date of <license> · 177  
 expiration state of <license> · 177  
 exponential projection · 311, 314, 325  
 extended family of <processor> · 172  
 extended model of <processor> · 172  
 external port of <port mapping> · 283  
 external\_clock of <dmi processor\_information> · 219  
 external\_connector\_type of <dmi port\_connector\_information> · 218  
 external\_reference\_designator of <dmi port\_connector\_information> · 218

---

## **F**

false · 6, 8, 41, 93, 276, 314  
 family name of <processor> · 172  
 family of <network interface> · 274  
 family of <processor> · 173  
 feature mask of <processor> · 173  
 february · 63, 70, 73, 77, 93, 314, 315  
 february <integer> · 63, 70, 93, 314  
 february <integer> of <integer> · 63, 314  
 february of <integer> · 77, 315  
 file · 1, 3, 15, 16, 32, 42, 46, 47, 85, 94, 104, 105, 106, 107, 108, 110, 111, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 156, 157, 158, 159, 160, 162, 163, 164, 180, 194, 195, 232, 233, 236, 252, 253, 254, 255, 256, 257, 261, 284, 301, 302, 313, 334, 335, 336  
 file <string> · 94, 107, 129, 141  
 file <string> of <folder> · 129, 141  
 file content · 130, 148, 149, 334  
 file extension <string> of <registry> · 115, 117  
 file line · 131, 132, 157, 158, 301  
 file of <folder> · 129, 141  
 file of <service> · 129, 180  
 file section · 133, 147, 149  
 file shortcut · 133  
 File System Objects · iii, 1  
 file system type of <drive> · 143  
 file type <string> of <registry> · 115, 117  
 file version block · 131, 132, 133, 154  
 file version of <file> · 130, 134, 149, 150  
 file\_and\_print firewall service type · 94, 195  
 file\_supports\_encryption of <drive> · 143  
 file\_supports\_object\_ids of <drive> · 143  
 file\_supports\_reparse\_points of <drive> · 143  
 file\_supports\_sparse\_files of <drive> · 143  
 file\_volume\_quotas of <drive> · 143  
 filesystem object · 94, 107, 108, 111, 126, 127, 128, 129, 139, 140, 141, 143  
 Filesystem Objects · 126  
 final part <time interval> of <time range> · 56, 57, 315  
 find file <string> of <folder> · 129, 142  
 finite of <floating point> · 20, 315  
 firewall · 2, 85, 90, 91, 92, 94, 96, 98, 101, 104, 106, 109, 113, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 199, 200, 201, 280, 282, 283, 286  
 firewall authorized application · 187, 200  
 firewall enabled of <firewall profile> · 187  
 firewall icmp settings · 187  
 Firewall Objects · iii, 2, 184  
 firewall of <connection> · 282, 286  
 firewall open port · 187, 192, 193, 194, 199, 200  
 firewall policy · 184, 186, 190  
 firewall profile · 90, 92, 94, 106, 113, 184, 186, 187, 188, 189, 190  
 firewall profile type · 90, 92, 94, 106, 113, 184, 187, 188, 189  
 firewall profile type <integer> · 94, 188  
 firewall remote admin settings · 187  
 firewall scope · 85, 91, 94, 98, 186, 191, 192, 193, 194  
 firewall scope <integer> · 94, 191  
 firewall service · 94, 101, 104, 109, 187, 191, 192, 194, 195, 196, 200  
 firewall service type · 94, 101, 104, 109, 194, 195, 196  
 firewall service type <integer> · 94, 195  
 first <day of week> of <month and year> · 78, 315  
 first <integer> of <string> · 29, 35, 315  
 first <string> of <string> · 29, 35, 315  
 first child of <xml dom node> · 160, 162, 315  
 first friday of <month and year> · 78, 315

first match <regular expression> of <string> · 43, 315  
first monday of <month and year> · 79, 315  
first raw version block of <file> · 131, 152  
first saturday of <month and year> · 79, 315  
first start time of <application usage summary> · 165  
first sunday of <month and year> · 79, 315  
first thursday of <month and year> · 79, 315  
first tuesday of <month and year> · 79, 315  
first wednesday of <month and year> · 79, 315  
fixlet · 231, 232, 234, 235, 307, 311, 312, 315, 326, 328, 329  
fixlet count pair · 311, 329  
fixlet of <site> · 231, 234  
fixlet\_header · 235  
flags of <dmi bios\_language\_information> · 205  
floating point · 7, 8, 9, 10, 11, 12, 13, 14, 15, 18, 19, 20, 21, 22, 23, 24, 25, 27, 31, 32, 46, 94, 305, 311, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 328, 329, 330, 331, 332, 334, 336  
floating point <string> · 18, 94, 315  
folder · 15, 16, 32, 51, 94, 96, 107, 108, 110, 111, 115, 117, 121, 122, 126, 127, 128, 129, 134, 135, 136, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 151, 158, 159, 163, 164, 176, 231, 232, 233, 252, 253, 254, 255, 257, 261, 267, 335, 338  
folder <string> · 94, 139, 140, 142, 144  
folder <string> of <drive> · 140, 144  
folder <string> of <folder> · 140, 142  
folder of <folder> · 140, 142  
following text of <string position> · 34, 35, 315  
following text of <substring> · 35, 36, 316  
form\_factor of <dmi memory\_device> · 213  
free space of <drive> · 144  
friday · 66, 94, 316  
friendly name of <active device> · 175  
fs\_case\_is\_preserved of <drive> · 144  
fs\_case\_sensitive of <drive> · 144  
fs\_file\_compression of <drive> · 144  
fs\_persistent\_acls of <drive> · 144  
fs\_unicode\_stored\_on\_disk of <drive> · 144  
fs\_vol\_is\_compressed of <drive> · 144  
full name of <local user> · 265  
full wmi <string> · 94, 226

---

## G

gateway address <integer> of <selected server> · 239, 277  
gateway address of <selected server> · 239, 277  
gateway list of <network adapter> · 279, 281  
gateway of <network adapter> · 277, 279  
gather schedule authority of <site> · 231  
gather schedule time interval of <site> · 231  
generic all permission of <access control entry> · 253  
generic execute permission of <access control entry> · 253  
generic read permission of <access control entry> · 253  
generic write permission of <access control entry> · 253  
ghz · 44, 46, 94, 316  
globally open port of <firewall profile> · 187, 192  
globally open port of <firewall service> · 192, 194  
greatest hz · 44, 46, 95, 316  
greatest integer · 10, 95, 316  
greatest time interval · 59, 95, 316  
group <integer> of <site> · 232, 234  
group leader of <action> · 271  
group of <security descriptor> · 262, 263  
group\_associations <integer> of <dmi> · 208  
group\_associations of <dmi> · 208  
group\_name of <dmi group\_associations> · 208  
guest privilege of <local user> · 265  
guid of <connection> · 286

---

## H

hardware id of <active device> · 176  
hardware\_security <integer> of <dmi> · 209  
hardware\_security of <dmi> · 209  
hardware\_security\_settings of <dmi hardware\_security> · 209  
has blank sa password of <local mssql database> · 179  
header <string> of <fixlet> · 235  
header of <fixlet> · 235  
hertz · 14, 25, 44, 45, 46, 94, 95, 97, 100, 170, 173, 305, 307, 316, 317, 318, 321, 328, 334  
hexadecimal integer <string> · 10, 95, 316  
hexadecimal string <string> · 26, 95, 316  
hidden of <filesystem object> · 126  
historical computer count · 307, 311, 312, 330

historical fixlet count · 307, 311, 312, 331  
home directory drive of <local user> · 265  
home directory of <local user> · 265  
home directory required flag of <local user> · 266  
hostname · 95  
hour · 50, 52, 54, 55, 56, 59, 62, 95, 316, 322  
hour\_of\_day of <time of day with time zone> ·  
55, 316  
hour\_of\_day of <time of day> · 52, 316  
html · 41, 43, 120, 310, 317, 319, 321, 332, 334,  
336  
hyperthreading capable · 95  
hyperthreading enabled · 95  
hz · 25, 44, 46, 95, 316

---

## *I*

icmp settings of <firewall profile> · 187, 196  
icon index of <file shortcut> · 146  
icon pathname of <file shortcut> · 146  
id of <action> · 271  
id of <file version block> · 153  
id of <fixlet> · 235  
id of <site group> · 234  
identifier of <metabase value> · 293, 297  
image path of <service> · 180  
inactive <integer> of <dmi> · 209  
inactive of <dmi> · 209  
index type of <property> · 299, 302, 316  
inexact of <floating point> · 7, 20, 317  
infinite of <floating point> · 7, 20, 317  
information event log event type · 95, 248  
inherit attribute of <metabase value> · 298  
inheritance of <access control entry> · 254  
initial part <time interval> of <time range> · 56,  
57, 317  
insert path attribute of <metabase value> · 298  
Inspector List · 2  
install folder <integer> · 96, 140  
installable\_languages of <dmi  
bios\_language\_information> · 205  
installed\_size of <dmi cache\_information> · 206  
installed\_size of <dmi  
memory\_module\_information> · 215  
instance name of <local mssql database> · 179

integer · 3, 5, 6, 8, 9, 10, 11, 12, 13, 14, 15, 16,  
17, 18, 19, 20, 21, 22, 23, 24, 25, 27, 29, 30, 33,  
37, 38, 39, 40, 43, 44, 45, 46, 52, 55, 61, 68, 70,  
73, 74, 75, 77, 78, 82, 83, 85, 87, 88, 91, 94, 95,  
96, 97, 98, 99, 101, 102, 103, 108, 112, 113,  
122, 124, 125, 130, 131, 133, 144, 145, 146,  
158, 162, 166, 168, 169, 170, 172, 173, 174,  
176, 177, 178, 179, 180, 181, 188, 193, 195,  
200, 202, 203, 204, 205, 206, 207, 208, 209,  
210, 211, 212, 213, 214, 215, 216, 217, 218,  
219, 220, 221, 222, 223, 224, 225, 228, 229,  
233, 234, 235, 239, 244, 245, 246, 247, 252,  
254, 255, 256, 262, 265, 266, 267, 270, 271,  
274, 280, 283, 285, 287, 289, 294, 295, 296,  
297, 300, 306, 307, 308, 310, 311, 312, 313,  
314, 315, 316, 317, 318, 320, 321, 322, 323,  
324, 325, 327, 328, 329, 330, 331, 332, 333,  
334, 335, 336, 337  
integer <integer> · 10, 96, 317  
integer <string> · 10, 96, 317  
integer ceiling of <floating point> · 10, 20, 317  
integer floor of <floating point> · 11, 20, 317  
integer range · 11, 16, 239  
integer value <integer> of <wmi select> · 228  
integer value of <wmi select> · 229  
integer with multiplicity · 13, 322, 331  
interdomain trust account flag of <local user> ·  
266  
interface <integer> of <network> · 273, 274  
interface of <dmi built\_in\_pointing\_device> · 205  
interface of <network> · 273, 274  
interleave\_position of <dmi  
memory\_device\_mapped\_address> · 214  
interleaved\_data\_depth of <dmi  
memory\_device\_mapped\_address> · 214  
internal port of <port mapping> · 283  
internal\_connector\_type of <dmi  
port\_connector\_information> · 218  
internal\_reference\_designator of <dmi  
port\_connector\_information> · 218  
internet connection firewall · 280, 282, 286  
internet connection firewall of <network adapter>  
· 280, 282  
internet protocol · 96, 108, 109, 193, 199  
internet protocol <integer> · 96, 199  
Introspectors · vi, 2, 299  
invalid of <floating point> · 7, 20, 317  
ip address of <selected server> · 239, 277  
ip interface <integer> of <network> · 273, 275  
ip interface of <network> · 273, 275  
ip version · 86, 96, 185, 192, 194, 200, 201

ip version <integer> · 96, 200  
ip version of <firewall authorized application> · 185, 200  
ip version of <firewall open port> · 192, 200  
ip version of <firewall service> · 194, 200  
ipv4 · 96, 200, 239, 275, 276, 277, 278, 279, 280, 282, 283, 334  
ipv4 address · 96, 239, 275, 276, 277, 278, 279, 280, 282, 283, 334  
ipv4 address <string> · 96, 277  
ipv6 · 96, 200  
item\_handle of <dmi group\_associations> · 208  
item\_type of <dmi group\_associations> · 208

---

## *J*

january · 63, 71, 73, 76, 77, 81, 96, 317  
january <integer> · 63, 71, 96, 317  
january <integer> of <integer> · 63, 317  
january of <integer> · 77, 317  
july · 63, 71, 73, 77, 96, 317  
july <integer> · 63, 71, 96, 317  
july <integer> of <integer> · 63, 317  
july of <integer> · 77, 317  
june · 63, 70, 71, 74, 78, 97, 317  
june <integer> · 63, 71, 97, 317  
june <integer> of <integer> · 63, 317  
june of <integer> · 78, 317

---

## *K*

key <string> of <file section> · 147  
key <string> of <file> · 131  
key <string> of <metabase key> · 294  
key <string> of <metabase> · 293, 294  
key <string> of <registry key> · 117, 118  
key <string> of <registry> · 115, 117  
key of <metabase key> · 294, 295  
key of <metabase> · 293, 294  
key of <registry key> · 117, 118  
Key Phrases (Inspectors) · vi, 307  
keywords · 1, 2, 3, 5, 114, 119, 126, 167, 231, 270, 273, 307  
khz · 44, 45, 97, 317

---

## *L*

l1\_cache\_handle of <dmi processor\_information> · 219

l2\_cache\_handle of <dmi processor\_information> · 219  
l3\_cache\_handle of <dmi processor\_information> · 220  
language · iv, 1, 5, 6, 38, 85, 107, 109, 110, 132, 150, 152, 153, 154, 156, 172, 182, 183, 205, 265, 299, 301, 303, 304, 305, 334  
language of <file version block> · 153  
last <integer> of <string> · 30, 35, 318  
last <string> of <string> · 30, 35, 318  
last change time of <action> · 271  
last child of <xml dom node> · 160, 162, 318  
last gather time of <site> · 232  
last logoff of <local user> · 266  
last logon of <local user> · 266  
last start time of <application usage summary> · 165  
last time seen of <application usage summary> · 165  
leap of <year> · 83, 318  
lease expires of <network adapter> · 280  
lease obtained of <network adapter> · 280  
least hz · 45, 46, 97, 318  
least integer · 9, 11, 97, 318  
least significant one bit of <bit set> · 40, 318  
least time interval · 59, 97, 318  
left operand type of <binary operator> · 299, 303, 318  
left shift <integer> of <bit set> · 39, 40, 318  
length of <dmi  
    b32\_bit\_memory\_error\_information> · 202  
length of <dmi  
    b64\_bit\_memory\_error\_information> · 203  
length of <dmi base\_board\_information> · 204  
length of <dmi bios\_information> · 204  
length of <dmi bios\_language\_information> · 205  
length of <dmi built\_in\_pointing\_device> · 205  
length of <dmi cache\_information> · 206  
length of <dmi cooling\_device> · 207  
length of <dmi electrical\_current\_probe> · 207  
length of <dmi end\_of\_table> · 208  
length of <dmi group\_associations> · 209  
length of <dmi hardware\_security> · 209  
length of <dmi inactive> · 209  
length of <dmi management\_device\_component> · 210  
length of <dmi  
    management\_device\_threshold\_data> · 211  
length of <dmi management\_device> · 210  
length of <dmi memory\_array\_mapped\_address> · 212

length of <dmi memory\_controller\_information> · 212  
length of <dmi  
    memory\_device\_mapped\_address> · 214  
length of <dmi memory\_device> · 213  
length of <dmi memory\_module\_information> · 215  
length of <dmi on\_board\_devices\_information> · 216  
length of <dmi out\_of\_band\_remote\_access> · 217  
length of <dmi physical\_memory\_array> · 217  
length of <dmi port\_connector\_information> · 218  
length of <dmi portable\_battery> · 219  
length of <dmi processor\_information> · 220  
length of <dmi system\_boot\_information> · 220  
length of <dmi system\_enclosure\_or\_chassis> · 221  
length of <dmi system\_information> · 222  
length of <dmi system\_power\_controls> · 222  
length of <dmi system\_reset> · 223  
length of <dmi system\_slots> · 224  
length of <dmi temperature\_probe> · 224  
length of <dmi voltage\_probe> · 225  
length of <event log record> · 245  
length of <month and year> · 79, 318  
length of <rope> · 11, 38, 318  
length of <string> · 11, 30, 318  
length of <time range> · 57, 318  
length of <year> · 83, 318  
less significance <integer> of <floating point> · 18, 20, 318  
license · 88, 177, 178  
line <integer> of <file> · 131, 157  
line containing <string> of <file> · 131, 157  
line number of <file line> · 158  
line of <file> · 131, 157  
line starting with <string> of <file> · 132, 157  
linear projection · 311, 314, 318, 325  
link speed of <network adapter> · 280  
list permission of <access control entry> · 254  
local administrator · 97, 242  
local computer of <active directory server> · 291, 292  
local group · 97, 105, 241, 242, 243, 263, 334  
local group <string> · 97, 241  
local group member · 242, 243, 263, 334  
local mssql database · 97, 98, 178  
local mssql database <string> · 98, 178  
local policy of <firewall> · 184, 190

local subnet firewall scope · 98, 191  
local time <string> · 98, 319  
local time zone · 27, 46, 47, 53, 54, 56, 98, 319  
local user · 92, 98, 264, 268, 269  
local user <string> · 98, 264  
location information of <active device> · 176  
location of <dmi physical\_memory\_array> · 217  
location of <dmi portable\_battery> · 219  
location of <filesystem object> · 127  
location\_and\_status of <dmi  
    electrical\_current\_probe> · 207  
location\_and\_status of <dmi temperature\_probe> · 224  
location\_and\_status of <dmi voltage\_probe> · 225  
locked out flag of <local user> · 266  
logical processor count · 98, 113  
login account of <service> · 180  
login mode of <local mssql database> · 179  
logon count of <local user> · 266  
logon script of <local user> · 266  
logon server of <local user> · 266  
loopback of <network ip interface> · 275  
lower bound of <integer range> · 11, 16  
lower\_threshold\_critical of <dmi  
    management\_device\_threshold\_data> · 211  
lower\_threshold\_non\_critical of <dmi  
    management\_device\_threshold\_data> · 211  
lower\_threshold\_non\_recoverable of <dmi  
    management\_device\_threshold\_data> · 211

---

## *M*

mac address of <network adapter> · 280  
main gather service · 98, 179  
main processor · 99, 172, 174  
major version of <operating system> · 169  
management\_device <integer> of <dmi> · 209  
management\_device of <dmi> · 209  
management\_device\_component <integer> of <dmi> · 210  
management\_device\_component of <dmi> · 210  
management\_device\_handle of <dmi  
    management\_device\_component> · 210  
management\_device\_threshold\_data <integer> of <dmi> · 211  
management\_device\_threshold\_data of <dmi> · 211  
manufacture\_date of <dmi portable\_battery> · 219  
manufacturer of <active device> · 176

- manufacturer of <dmi base\_board\_information> · 204
- manufacturer of <dmi portable\_battery> · 219
- manufacturer of <dmi system\_enclosure\_or\_chassis> · 221
- manufacturer of <dmi system\_information> · 222
- manufacturer\_name of <dmi out\_of\_band\_remote\_access> · 217
- march · 63, 71, 74, 78, 99, 320
- march <integer> · 63, 71, 99, 320
- march <integer> of <integer> · 63, 320
- march of <integer> · 78, 320
- masthead of <site> · 129, 232
- match <regular expression> of <string> · 43, 320
- max\_speed of <dmi processor\_information> · 220
- maximum allowed permission of <access control entry> · 254
- maximum of <integer> · 11, 12, 320
- maximum of <time interval> · 59, 60, 320
- maximum of <time> · 47, 48, 320
- maximum seat count of <license> · 177
- maximum storage of <local user> · 266
- maximum\_cache\_size of <dmi cache\_information> · 206
- maximum\_capacity of <dmi physical\_memory\_array> · 217
- maximum\_error\_in\_battery\_data of <dmi portable\_battery> · 219
- maximum\_memory\_module\_size of <dmi memory\_controller\_information> · 212
- maximum\_value of <dmi electrical\_current\_probe> · 207
- maximum\_value of <dmi temperature\_probe> · 224
- maximum\_value of <dmi voltage\_probe> · 225
- may · 2, 16, 46, 47, 63, 71, 74, 78, 99, 102, 110, 117, 120, 121, 123, 134, 138, 142, 145, 147, 152, 157, 158, 167, 170, 173, 193, 239, 250, 279, 281, 301, 320, 339
- may <integer> · 63, 71, 99, 320
- may <integer> of <integer> · 63, 320
- may of <integer> · 78, 320
- mean of <floating point> · 18, 20, 320
- mean of <integer> · 13, 18, 320
- media type · 99, 100, 286, 289, 290
- media type <integer> · 99, 289
- media type bridge · 99, 289
- media type direct · 99, 289
- media type isdn · 99, 289
- media type lan · 99, 289
- media type of <connection> · 286, 289
- media type phone · 99, 290
- media type pppoe · 100, 290
- media type shared access host lan · 100, 290
- media type shared access host ras · 100, 290
- media type tunnel · 100, 290
- member of <local group> · 242
- member of <site group> · 234
- memory\_array\_error\_address of <dmi b32\_bit\_memory\_error\_information> · 202
- memory\_array\_error\_address of <dmi b64\_bit\_memory\_error\_information> · 203
- memory\_array\_handle of <dmi memory\_array\_mapped\_address> · 212
- memory\_array\_handle of <dmi memory\_device> · 213
- memory\_array\_mapped\_address <integer> of <dmi> · 211
- memory\_array\_mapped\_address of <dmi> · 211
- memory\_array\_mapped\_address\_handle of <dmi memory\_device\_mapped\_address> · 214
- memory\_controller\_information <integer> of <dmi> · 212
- memory\_controller\_information of <dmi> · 212
- memory\_device <integer> of <dmi> · 213
- memory\_device of <dmi> · 213
- memory\_device\_handle of <dmi memory\_device\_mapped\_address> · 214
- memory\_device\_mapped\_address <integer> of <dmi> · 214
- memory\_device\_mapped\_address of <dmi> · 214
- memory\_error\_correction of <dmi physical\_memory\_array> · 217
- memory\_error\_information\_handle of <dmi memory\_device> · 213
- memory\_error\_information\_handle of <dmi physical\_memory\_array> · 217
- memory\_module\_information <integer> of <dmi> · 215
- memory\_module\_information of <dmi> · 215
- memory\_module\_voltage of <dmi memory\_controller\_information> · 212
- memory\_type of <dmi memory\_device> · 213
- metabase · 100, 293, 294, 295, 296, 297, 298, 334, 335
- metabase identifier · 293, 294, 297, 334
- metabase key · 293, 294, 295, 297
- metabase type · 295, 296, 298, 335
- metabase user type · 296, 297, 298, 335
- metabase value · 295, 296, 297, 298, 335
- metric <integer> of <operating system> · 169
- mhz · 44, 45, 46, 100, 321

microsecond · 59, 60, 100, 321  
Microsoft IIS Metabase Objects · vi, 2, 293  
midnight · 50, 51, 100, 321  
millisecond · 25, 59, 60, 100, 321  
minimum of <integer> · 11, 13, 321  
minimum of <time interval> · 60, 321  
minimum of <time> · 47, 48, 321  
minimum\_value of <dmi  
    electrical\_current\_probe> · 208  
minimum\_value of <dmi temperature\_probe> ·  
    224  
minimum\_value of <dmi voltage\_probe> · 225  
minor version of <operating system> · 169  
minute · 50, 52, 53, 55, 56, 60, 100, 322, 327  
minute\_of\_hour of <time of day with time zone> ·  
    55, 322  
minute\_of\_hour of <time of day> · 52, 322  
model of <processor> · 173  
modification time of <filesystem object> · 127  
module · iv, 100, 161, 215, 322, 332  
module <string> · 100, 322  
monday · 66, 80, 100, 322  
month · 12, 16, 27, 50, 62, 63, 64, 65, 68, 69, 70,  
    71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83,  
    84, 86, 87, 90, 91, 93, 96, 97, 99, 101, 102, 105,  
    168, 247, 308, 312, 313, 314, 315, 317, 318,  
    320, 322, 323, 324, 328, 333, 334, 335, 336  
month <integer> · 74, 101, 322  
month <string> · 74, 101, 322  
month and year · 64, 69, 72, 75, 76, 78, 79, 80,  
    83, 84, 90, 308, 312, 313, 315, 317, 318, 320,  
    322, 323, 324, 328, 333, 335  
month of <date> · 64, 74, 322  
month of <day of year> · 72, 74, 322  
month of <month and year> · 74, 79, 322  
month\_and\_year of <date> · 64, 78, 322  
more significance <integer> of <floating point> ·  
    18, 21, 322  
most significant one bit of <bit set> · 40, 322  
multicast support of <network ip interface> · 275  
multiplicity of <integer with multiplicity> · 17,  
    322  
multiplicity of <string with multiplicity> · 37, 322  
multivalued of <property> · 302, 322

---

## N

name of <application usage summary> · 165  
name of <binary operator> · 303, 323  
name of <cast> · 306, 323

name of <connection> · 286  
name of <current user> · 269  
name of <drive> · 144  
name of <environment variable> · 251  
name of <filesystem object> · 127  
name of <firewall authorized application> · 185  
name of <firewall open port> · 192  
name of <firewall service> · 194  
name of <local group> · 242  
name of <local user> · 266  
name of <metabase key> · 295  
name of <network adapter> · 280  
name of <network share> · 284  
name of <operating system> · 169  
name of <port mapping> · 283  
name of <registry key value> · 122  
name of <registry key> · 118  
name of <selected server> · 239  
name of <setting> · 238  
name of <site> · 232  
name of <type> · 300, 323  
name of <unary operator> · 304, 323  
name of <wmi select> · 229  
nan of <floating point> · 7, 21, 323  
native registry · 101, 112, 114  
network · 1, 2, 88, 89, 90, 95, 99, 100, 101, 109,  
    113, 184, 186, 190, 196, 262, 263, 273, 274,  
    275, 276, 277, 278, 279, 280, 281, 282, 284,  
    285, 286, 288, 289, 290  
network adapter · 273, 276, 277, 278, 279, 280,  
    281  
network address list · 273, 277, 279, 281, 282  
network interface · 90, 273, 274, 277, 288  
network ip interface · 273, 275  
network share · 101, 113, 262, 284, 285  
network share <string> · 101, 284  
Networking Objects · v, 273  
next line of <file line> · 157, 158  
next sibling of <xml dom node> · 160, 162, 323  
next\_scheduled\_power\_on\_day\_of\_month of  
    <dmi system\_power\_controls> · 222  
next\_scheduled\_power\_on\_hour of <dmi  
    system\_power\_controls> · 222  
next\_scheduled\_power\_on\_minute of <dmi  
    system\_power\_controls> · 222  
next\_scheduled\_power\_on\_month of <dmi  
    system\_power\_controls> · 223  
next\_scheduled\_power\_on\_second of <dmi  
    system\_power\_controls> · 223  
no password required flag of <local user> · 266  
node name of <xml dom node> · 162, 323

node type of <xml dom node> · 162, 323  
node value of <xml dom node> · 162, 323  
nominal\_speed of <dmi cooling\_device> · 207  
nominal\_value of <dmi electrical\_current\_probe> · 208  
nominal\_value of <dmi temperature\_probe> · 224  
nominal\_value of <dmi voltage\_probe> · 225  
none firewall service type · 101, 195  
noon · 51, 101, 323  
normal account flag of <local user> · 266  
normal of <filesystem object> · 127  
normal of <floating point> · 7, 21, 323  
notifications disabled of <firewall profile> · 187  
november · 63, 71, 74, 78, 101, 323  
november <integer> · 63, 71, 101, 323  
november <integer> of <integer> · 63, 323  
november of <integer> · 78, 323  
now · 47, 51, 56, 58, 62, 65, 77, 86, 101, 171, 227, 233, 323  
nt domain controller product type · 102, 240  
nt server product type · 102, 240  
nt workstation product type · 102, 240  
number of months · 14, 15, 64, 65, 72, 75, 76, 80, 81, 82, 83, 101, 112, 322, 332, 335  
number\_of\_associated\_memory\_slots of <dmi memory\_controller\_information> · 212  
number\_of\_buttons of <dmi built\_in\_pointing\_device> · 205  
number\_of\_memory\_devices of <dmi physical\_memory\_array> · 217  
numeric type of <drive> · 144  
numeric value of <string> · 11, 30, 323

---

## O

october · 63, 71, 74, 78, 102, 323, 324  
october <integer> · 63, 71, 102, 323, 324  
october <integer> of <integer> · 63, 324  
october of <integer> · 78, 324  
oem code page · 102  
oem\_defined of <dmi cooling\_device> · 207  
oem\_defined of <dmi electrical\_current\_probe> · 208  
oem\_defined of <dmi system\_enclosure\_or\_chassis> · 221  
oem\_defined of <dmi temperature\_probe> · 224  
oem\_defined of <dmi voltage\_probe> · 225  
oem\_specific of <dmi portable\_battery> · 219  
offline of <filesystem object> · 127  
oldest record number of <event log> · 244

on\_board\_devices\_information <integer> of <dmi> · 216  
on\_board\_devices\_information of <dmi> · 216  
one bit of <bit set> · 40, 324  
only raw version block of <file> · 132, 152  
only version block of <file> · 132, 152  
operand type of <cast> · 299, 306, 324  
operand type of <unary operator> · 299, 305, 324  
operating system · 3, 4, 5, 31, 87, 92, 95, 97, 102, 110, 119, 129, 138, 141, 167, 168, 169, 170, 171, 172, 240, 241, 245, 248, 335  
operating system product type · 102, 170, 240  
operating system product type <integer> · 102, 240  
operating system suite mask · 87, 170  
options of <port mapping> · 283  
organization of <license> · 178  
origin fixlet id of <action> · 271  
out\_of\_band\_remote\_access <integer> of <dmi> · 216  
out\_of\_band\_remote\_access of <dmi> · 216  
overflow of <floating point> · 7, 21, 324  
owner document of <xml dom node> · 159, 162, 324  
owner of <security descriptor> · 262, 263

---

## P

pad of <version> · 149, 150  
parameter <string> · 26, 102, 271  
parameter <string> of <action> · 26, 102, 271  
parent folder of <filesystem object> · 127, 140  
parent node of <xml dom node> · 160, 162, 324  
parent of <type> · 299, 300, 324  
parenthesized part <integer> of <regular expression match> · 43, 324  
parenthesized part of <regular expression match> · 43, 324  
partition\_row\_position of <dmi memory\_device\_mapped\_address> · 214  
partition\_width of <dmi memory\_array\_mapped\_address> · 212  
password age of <local user> · 267  
password change disabled flag of <local user> · 267  
password expiration disabled flag of <local user> · 267  
password expired of <local user> · 267  
password of <network share> · 285  
path of <network share> · 285

pathname of <file shortcut> · 146  
 pathname of <filesystem object> · 127  
 pending login · 102, 271, 309  
 pending login of <action> · 271  
 pending of <action> · 271  
 pending restart · 102, 103, 271, 309  
 pending restart <string> · 103  
 pending restart of <action> · 271  
 pending time of <action> · 271  
 performance counter frequency of <operating system> · 170  
 performance counter of <operating system> · 170  
 permission permission of <network share> · 285  
 personal bit <operating system suite mask> · 103  
 physical processor count · 98, 103, 113  
 physical\_memory\_array <integer> of <dmi> · 217  
 physical\_memory\_array of <dmi> · 217  
 platform id of <operating system> · 170  
 plural name of <property> · 302, 324  
 point to point of <network ip interface> · 276  
 port mapping · 278, 282, 283  
 port mapping of <internet connection firewall> · 282, 283  
 port number of <selected server> · 239  
 port of <firewall open port> · 193  
 port\_connector\_information <integer> of <dmi> · 218  
 port\_connector\_information of <dmi> · 218  
 port\_type of <dmi port\_connector\_information> · 218  
 portable\_battery <integer> of <dmi> · 218  
 portable\_battery of <dmi> · 218  
 position <integer> of <string> · 30, 33, 324  
 position of <string> · 30, 33, 324  
 power\_supply\_state of <dmi system\_enclosure\_or\_chassis> · 221  
 preceding text of <string position> · 34, 35, 325  
 preceding text of <substring> · 35, 36, 325  
 previous line of <file line> · 157, 158  
 previous sibling of <xml dom node> · 160, 162, 325  
 primary group id of <local user> · 267  
 primary language · 182, 183, 335  
 primary language of <language> · 182, 183  
 primary wins server of <network adapter> · 277, 280  
 Primitive Objects · ii, 1, 6  
 print operator flag of <local user> · 267  
 priority of <selected server> · 239  
 problem id of <active device> · 176

process image file name of <firewall authorized application> · 185  
 processor · 44, 46, 95, 98, 99, 103, 108, 172, 173, 174, 219, 220, 339, 340  
 processor <integer> · 103, 172  
 processor\_family of <dmi processor\_information> · 220  
 processor\_id of <dmi processor\_information> · 220  
 processor\_information <integer> of <dmi> · 219  
 processor\_information of <dmi> · 219  
 processor\_manufacturer of <dmi processor\_information> · 220  
 processor\_type of <dmi processor\_information> · 220  
 processor\_upgrade of <dmi processor\_information> · 220  
 processor\_version of <dmi processor\_information> · 220  
 product of <dmi base\_board\_information> · 204  
 product of <integer> · 11, 13, 325  
 product type of <operating system> · 170, 240  
 product version of <file> · 132, 150  
 product\_name of <dmi system\_information> · 222  
 profile folder of <local user> · 267  
 property · 103, 106, 118, 121, 122, 124, 126, 132, 139, 146, 149, 150, 154, 161, 163, 185, 186, 187, 191, 192, 193, 194, 197, 198, 199, 226, 227, 230, 232, 233, 234, 236, 237, 262, 279, 280, 284, 296, 299, 300, 301, 302, 303, 311, 314, 316, 322, 324, 325, 327, 328, 330, 332, 335, 339  
 property <string> · 103, 227, 230, 300, 301, 325  
 property <string> of <type> · 300, 301, 325  
 property <string> of <wmi object> · 227, 230  
 property of <type> · 300, 301, 325  
 property of <wmi object> · 227, 230  
 property returning <type> · 103, 300, 302, 325  
 property returning <type> of <type> · 300, 302, 325  
 protocol of <firewall open port> · 193, 199  
 protocol of <port mapping> · 283

---

## Q

query value permission of <access control entry> · 254

---

**R**

ram · 103, 174, 175  
random access memory · 103, 174  
range after <time> of <time range> · 57, 325  
range before <time> of <time range> · 57, 58, 325  
rate · 170, 314, 321, 325, 330, 335  
raw file version of <file> · 132, 150  
raw product version of <file> · 132, 150  
raw version block <integer> of <file> · 132, 152  
raw version block <string> of <file> · 132, 152  
raw version block of <file> · 132, 152  
raw version of <file> · 132, 150  
read attributes permission of <access control entry> · 254  
read control permission of <access control entry> · 254  
read extended attributes permission of <access control entry> · 254  
read permission of <access control entry> · 254  
read permission of <network share> · 285  
readonly of <filesystem object> · 127  
recent application · 103, 104, 137  
recent application <string> · 104, 137  
record <integer> of <event log> · 244, 245  
record count of <event log> · 244  
record number of <event log record> · 246  
record of <event log> · 244, 245  
reference attribute of <metabase value> · 298  
regapp · 32, 37, 104, 135, 137, 138, 139, 145, 151, 152, 156  
regapp <string> · 104, 137, 139  
regex <string> · 42, 104, 326  
registrar number of <license> · 178  
registration server · 86  
registry · 1, 14, 29, 101, 102, 104, 111, 112, 114, 115, 116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 128, 135, 136, 137, 138, 139, 147, 175, 252, 253, 254, 262, 294, 295, 297, 335, 336, 338  
registry key · 14, 29, 115, 116, 118, 119, 121, 122, 123, 124, 125, 128, 136, 137, 138, 139, 175, 252, 253, 254, 262, 294, 335, 336  
registry key value · 14, 29, 118, 121, 122, 123, 124, 125, 128, 136, 139, 335, 336  
registry key value type · 14, 122, 124, 125, 335  
Registry Objects · iii, 1, 114  
regular expression · 41, 42, 43, 87, 104, 310, 315, 320, 324, 326  
regular expression <string> · 42, 104, 326  
regular expression match · 43, 315, 320, 324

relative significance place <integer> of <floating point> · 18, 21, 326  
relative significance place of <floating point> · 18, 21, 326  
relay service · 104, 179  
release of <operating system> · 170  
Relevance Language · 3  
relevance of <fixlet> · 235  
relevant fixlet of <site> · 232, 234  
remote addresses of <firewall authorized application> · 186  
remote addresses of <firewall open port> · 193  
remote addresses of <firewall service> · 194  
remote admin settings of <firewall profile> · 187, 198  
remote desktop firewall service type · 104, 195  
reset\_count of <dmi system\_reset> · 223  
reset\_limit of <dmi system\_reset> · 223  
resolution of <dmi electrical\_current\_probe> · 208  
resolution of <dmi temperature\_probe> · 224  
resolution of <dmi voltage\_probe> · 225  
result type of <binary operator> · 299, 303, 327  
result type of <property> · 299, 302, 327  
result type of <unary operator> · 299, 305, 327  
right operand type of <binary operator> · 299, 303, 327  
right shift <integer> of <bit set> · 40, 327  
root folder of <drive> · 140, 144  
rope · 11, 31, 38, 39, 104, 318, 327, 336  
rope <string> · 38, 104, 327  
running application · 104, 105, 137, 139  
running application <string> · 105, 137  
running of <application usage summary> · 166  
running of <local mssql database> · 179  
running service · 105, 179, 181  
running service <string> · 105, 179

---

**S**

saturday · 66, 105, 327  
sbds\_device\_chemistry of <dmi portable\_battery> · 219  
sbds\_manufacture\_date of <dmi portable\_battery> · 219  
sbds\_serial\_number of <dmi portable\_battery> · 219  
sbds\_version\_number of <dmi portable\_battery> · 219

- scope of <firewall authorized application> · 186, 191
- scope of <firewall open port> · 191, 193
- scope of <firewall service> · 191, 194
- script flag of <local user> · 267
- seat count state of <license> · 178
- seat of <license> · 178
- second · 26, 30, 36, 52, 55, 59, 60, 75, 77, 105, 158, 163, 164, 170, 280, 327
- second\_of\_minute of <time of day with time zone> · 55, 327
- second\_of\_minute of <time of day> · 52, 327
- secondary wins server of <network adapter> · 277, 280
- section <string> of <file> · 133, 147
- secure attribute of <metabase value> · 298
- security descriptor · 118, 133, 142, 256, 260, 261, 262, 263, 284, 285, 336
- security descriptor of <file> · 133, 261
- security descriptor of <folder> · 142, 261
- security descriptor of <network share> · 262, 285
- security descriptor of <registry key> · 118, 262
- security event log · 105, 243
- security identifier · 243, 246, 252, 255, 262, 263, 336
- security\_status of <dmi system\_enclosure\_or\_chassis> · 221
- select <string> of <wmi> · 226, 227
- select <string> of <xml dom node> · 161, 162, 327
- select object <string> of <wmi> · 226, 230
- selected server · 105, 238, 239, 277
- september · 63, 71, 74, 78, 105, 328
- september <integer> · 63, 71, 105, 328
- september <integer> of <integer> · 63, 328
- september of <integer> · 78, 328
- serial\_number of <dmi base\_board\_information> · 204
- serial\_number of <dmi portable\_battery> · 219
- serial\_number of <dmi system\_enclosure\_or\_chassis> · 221
- serial\_number of <dmi system\_information> · 222
- server operator flag of <local user> · 267
- server trust account flag of <local user> · 267
- service · 93, 94, 98, 104, 105, 106, 109, 129, 169, 170, 176, 177, 179, 180, 181, 187, 193, 194, 195, 196, 200, 243, 246, 248, 336
- service <string> · 105, 180
- service key value name of <active device> · 176
- service name of <service> · 180
- service of <firewall profile> · 187, 193
- service pack major version of <operating system> · 170
- service pack minor version of <operating system> · 170
- service specific exit code of <service> · 180
- set value permission of <access control entry> · 254
- setting · 29, 165, 187, 188, 232, 236, 237, 238, 259, 336
- setting <string> of <client> · 236, 237
- setting <string> of <site> · 232, 237
- setting of <client> · 237
- setting of <site> · 232, 237
- sha1 of <file> · 133
- shortcut of <file> · 133, 146
- sid of <local group member> · 243, 263
- significance place <integer> of <floating point> · 19, 21, 328
- significance place of <floating point> · 19, 21, 328
- significance threshold of <floating point> · 19, 21, 328
- significant digits <integer> of <hertz> · 45, 328
- significant digits <integer> of <integer> · 11, 13, 328
- single user ts bit <operating system suite mask> · 105
- singular name of <property> · 302, 328
- site · 90, 91, 106, 107, 108, 129, 139, 140, 177, 178, 184, 185, 186, 188, 190, 191, 192, 193, 195, 196, 198, 199, 200, 231, 232, 233, 234, 237, 238, 312, 320, 324, 328, 339
- site <string> · 106, 231
- site group · 232, 234
- site number of <license> · 178
- Site Objects · iv, 2, 231
- site tag of <site> · 232
- size of <dmi memory\_device> · 213
- size of <file> · 133
- size of <ram> · 174
- size of <registry key value> · 122
- size of <type> · 300, 328
- slot\_characteristics\_1 of <dmi system\_slots> · 224
- slot\_characteristics\_2 of <dmi system\_slots> · 224
- slot\_data\_bus\_width of <dmi system\_slots> · 224
- slot\_designation of <dmi system\_slots> · 224
- slot\_id of <dmi system\_slots> · 224
- slot\_length of <dmi system\_slots> · 224

- slot\_type of <dmi system\_slots> · 224
- small business bit <operating system suite mask> · 106
- small business restricted bit <operating system suite mask> · 106
- socket\_designation of <dmi cache\_information> · 206
- socket\_designation of <dmi memory\_module\_information> · 215
- socket\_designation of <dmi processor\_information> · 220
- source of <event log record> · 246
- speed of <dmi memory\_device> · 213
- speed of <processor> · 173
- standard deviation of <floating point> · 19, 22, 329
- standard deviation of <integer> · 13, 19, 329
- standard firewall profile type · 106, 189
- standard profile of <firewall policy> · 186, 190
- start date of <license> · 178
- start in pathname of <file shortcut> · 146
- start of <substring> · 33, 36, 329
- start of <time range> · 58, 329
- start type of <service> · 181
- starting\_address of <dmi memory\_array\_mapped\_address> · 212
- starting\_address of <dmi memory\_device\_mapped\_address> · 214
- state of <service> · 181
- statistic range · 310, 314, 325, 329, 331
- statistical bin · 310, 314, 316, 317, 318, 319, 320, 321, 322, 328, 329, 330, 331, 332
- status of <action> · 272
- status of <active device> · 176
- status of <connection> · 286, 288
- status of <dmi processor\_information> · 220
- stepping of <processor> · 173
- string · 6, 7, 10, 11, 12, 15, 16, 18, 19, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 50, 51, 52, 53, 54, 55, 57, 59, 60, 62, 64, 66, 67, 68, 69, 71, 73, 74, 75, 78, 81, 82, 83, 86, 87, 88, 91, 92, 93, 94, 95, 96, 101, 102, 104, 106, 107, 108, 109, 110, 111, 112, 113, 116, 118, 120, 121, 122, 123, 124, 125, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138, 139, 142, 143, 144, 145, 146, 147, 149, 150, 151, 152, 153, 154, 156, 157, 158, 159, 161, 162, 163, 165, 167, 168, 169, 170, 172, 173, 175, 176, 177, 178, 179, 180, 181, 182, 183, 185, 186, 192, 193, 194, 204, 205, 206, 207, 208, 210, 213, 215, 216, 217, 218, 219, 220, 221, 222, 224, 225, 226, 227, 228, 229, 230, 231, 232, 233, 236, 237, 238, 239, 242, 243, 245, 246, 251, 256, 257, 258, 259, 262, 263, 265, 266, 267, 269, 271, 272, 277, 278, 279, 280, 283, 284, 285, 286, 291, 295, 296, 297, 300, 301, 302, 303, 304, 305, 306, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319, 320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 336, 337, 338, 340
- string <string> · 26, 106, 329
- string position · 24, 30, 33, 36, 314, 315, 324, 325, 329
- string value <integer> of <wmi select> · 229
- string value of <wmi select> · 229
- string with multiplicity · 31, 322, 331
- subnet address of <network adapter> · 277, 280
- subnet address of <network address list> · 277, 282
- subnet address of <network ip interface> · 276, 277
- subnet mask of <network adapter> · 277, 280
- subnet mask of <network address list> · 278, 282
- subnet mask of <network ip interface> · 276, 278
- subscribe time of <site> · 232
- substring · 24, 29, 30, 33, 34, 35, 36, 43, 310, 314, 315, 316, 318, 324, 325, 329
- substring <string> of <string> · 30, 35, 329
- substring after <string> of <string> · 30, 36, 329
- substring before <string> of <string> · 30, 36, 329
- substring between <string> of <string> · 30, 36, 329
- substring separated by <string> of <string> · 30, 36, 329
- suite mask of <operating system> · 170, 241
- sum of <integer> · 11, 13, 330
- sunday · 66, 106, 330

supported\_interleave of <dmi  
memory\_controller\_information> · 213  
supported\_memory\_types of <dmi  
memory\_controller\_information> · 213  
supported\_speeds of <dmi  
memory\_controller\_information> · 213  
supported\_sram\_type of <dmi  
cache\_information> · 206  
symbol of <binary operator> · 304, 330  
symbol of <unary operator> · 305, 330  
synchronize permission of <access control entry>  
· 255  
system event log · 106, 243, 244  
system file <string> · 106, 129  
system folder · 338  
system ini device file · 106, 107, 129  
system ini device file <string> · 107, 129  
system language · 107  
system locale · 107, 182, 183  
System Objects · iii, 1, 2, 167  
system of <filesystem object> · 127  
system ui language · 107, 182  
system wow64 folder · 107, 140  
system x32 folder · 107, 140  
system x64 folder · 108, 140  
system\_boot\_information <integer> of <dmi> ·  
220  
system\_boot\_information of <dmi> · 220  
system\_cache\_type of <dmi cache\_information> ·  
206  
system\_enclosure\_or\_chassis <integer> of <dmi>  
· 221  
system\_enclosure\_or\_chassis of <dmi> · 221  
system\_information <integer> of <dmi> · 222  
system\_information of <dmi> · 222  
system\_power\_controls <integer> of <dmi> · 222  
system\_power\_controls of <dmi> · 222  
system\_reset <integer> of <dmi> · 223  
system\_reset of <dmi> · 223  
system\_slots <integer> of <dmi> · 223  
system\_slots of <dmi> · 223

---

## T

target ip address of <port mapping> · 278, 283  
target name of <port mapping> · 283  
tcp · 108, 199, 283  
temperature\_probe <integer> of <dmi> · 224  
temperature\_probe of <dmi> · 224

temperature\_probe\_handle of <dmi  
cooling\_device> · 207  
temporary duplicate account flag of <local user> ·  
267  
temporary of <filesystem object> · 127  
terminal bit <operating system suite mask> · 108  
thermal\_state of <dmi  
system\_enclosure\_or\_chassis> · 221  
threshold\_handle of <dmi  
management\_device\_component> · 210  
thursday · 66, 108, 330  
time · 8, 9, 14, 15, 25, 26, 27, 28, 29, 46, 47, 48,  
49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61,  
62, 64, 65, 67, 69, 72, 79, 81, 83, 86, 91, 95, 97,  
98, 100, 101, 105, 108, 109, 110, 112, 122, 126,  
127, 130, 134, 136, 165, 166, 168, 170, 171,  
177, 178, 227, 229, 231, 232, 233, 238, 246,  
247, 264, 266, 267, 270, 271, 280, 284, 307,  
310, 311, 313, 314, 315, 316, 317, 318, 319,  
320, 321, 322, 323, 324, 325, 327, 328, 329,  
330, 331, 332, 333, 336, 337  
time <string> · 47, 108, 330  
time <time zone> of <time> · 48, 54, 330  
time generated of <event log record> · 246  
time interval · 8, 9, 15, 25, 28, 46, 48, 49, 52, 54,  
55, 57, 59, 60, 61, 62, 64, 65, 67, 69, 72, 79, 81,  
83, 91, 95, 97, 100, 105, 108, 110, 166, 170,  
231, 267, 307, 313, 314, 316, 318, 320, 321,  
322, 324, 325, 327, 328, 330, 331, 332, 336,  
337  
time interval <string> · 60, 108, 330  
time of <time of day with time zone> · 51, 55,  
331  
time of day · 27, 28, 29, 48, 51, 52, 53, 54, 55, 56,  
61, 64, 65, 100, 101, 108, 112, 314, 316, 321,  
322, 323, 327, 329, 330, 331, 333, 336, 337  
time of day with time zone · 27, 28, 29, 48, 52,  
53, 54, 55, 56, 61, 64, 65, 112, 316, 322, 327,  
330, 331, 333, 336, 337  
time range · 8, 9, 14, 15, 48, 49, 51, 56, 57, 58,  
61, 62, 314, 315, 317, 318, 325, 329, 331, 337  
time value <integer> of <wmi select> · 229  
time value of <wmi select> · 229  
time written of <event log record> · 246  
time zone · 25, 28, 46, 47, 48, 50, 51, 52, 53, 54,  
55, 56, 61, 62, 64, 98, 108, 109, 319, 331, 332,  
333, 336, 337  
time zone <string> · 53, 108, 331  
time\_of\_day <string> · 51, 108, 331  
timeout of <dmi system\_reset> · 223  
timer\_interval of <dmi system\_reset> · 223

tolerance of <dmi electrical\_current\_probe> · 208  
tolerance of <dmi temperature\_probe> · 224  
tolerance of <dmi voltage\_probe> · 225  
total duration of <application usage summary> · 166  
total processor core count · 108  
total run count of <application usage summary> · 166  
total space of <drive> · 145  
total\_width of <dmi memory\_device> · 213  
traverse permission of <access control entry> · 255  
true · 6, 7, 8, 32, 37, 40, 41, 77, 103, 108, 281, 331  
trustee of <access control entry> · 255, 263  
trustee type of <access control entry> · 255  
tuesday · 66, 108, 331  
two digit hour of <time of day> · 52, 331  
two digit minute of <time of day> · 52, 331  
two digit second of <time of day> · 52, 331  
type · 2, 3, 5, 6, 10, 12, 14, 18, 24, 26, 28, 44, 51, 52, 53, 54, 62, 64, 66, 68, 69, 70, 71, 72, 73, 74, 75, 80, 81, 82, 83, 85, 86, 87, 88, 90, 91, 93, 94, 96, 97, 98, 99, 101, 102, 103, 104, 105, 108, 109, 112, 115, 116, 117, 119, 120, 121, 122, 123, 124, 125, 136, 144, 145, 146, 150, 162, 170, 172, 173, 184, 187, 188, 189, 190, 191, 194, 195, 196, 199, 200, 205, 210, 213, 221, 229, 230, 232, 238, 240, 245, 248, 249, 255, 260, 266, 274, 275, 281, 285, 289, 290, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 306, 307, 310, 314, 316, 318, 323, 324, 325, 327, 328, 331, 333, 337  
type <string> · 300  
type of <dmi built\_in\_pointing\_device> · 205  
type of <dmi management\_device> · 210  
type of <dmi system\_enclosure\_or\_chassis> · 221  
type of <drive> · 145  
type of <firewall profile> · 187, 189  
type of <firewall service> · 194, 195  
type of <metabase value> · 295, 298  
type of <network share> · 285  
type of <processor> · 173  
type of <registry key value> · 122, 124  
type of <site> · 232  
type of <wmi select> · 229  
type\_detail of <dmi memory\_device> · 213

---

## U

udp · 109, 194, 199, 283  
unary operator · 109, 304, 305, 323, 324, 327, 330, 331, 337  
unary operator <string> · 109, 304, 331  
unary operator returning <type> · 109, 304, 331  
undefined · 44, 93, 314  
underflow of <floating point> · 7, 22, 331  
unicast responses to multicast broadcast disabled of <firewall profile> · 188  
unique value of <integer> · 13, 17, 331  
unique value of <string> · 31, 37, 331, 332  
universal time <string> · 47, 109, 332  
universal time zone · 47, 48, 53, 56, 109, 332  
upnp firewall service type · 109, 196  
upper bound of <integer range> · 11, 16  
upper\_threshold\_critical of <dmi management\_device\_threshold\_data> · 211  
upper\_threshold\_non\_critical of <dmi management\_device\_threshold\_data> · 211  
upper\_threshold\_non\_recoverable of <dmi management\_device\_threshold\_data> · 211  
uptime of <operating system> · 170  
url of <site> · 233  
use count of <network share> · 285  
use limit of <network share> · 285  
use of <dmi physical\_memory\_array> · 217  
user comment of <local user> · 267  
user id of <local user> · 267  
user language · 109  
user locale · 109, 182  
User Objects · v, 2, 264  
user privilege of <local user> · 268  
user sid of <event log record> · 246, 263  
user type of <metabase value> · 296, 298  
user ui language · 110, 182  
usual name of <property> · 302, 332

---

## V

value <string> of <file version block> · 153  
value <string> of <registry key> · 118, 121  
value of <environment variable> · 251  
value of <metabase key> · 295, 297  
value of <registry key> · 118, 121  
value of <setting> · 238  
variable <string> of <environment> · 250, 251  
variable of <environment> · 250, 251  
variable of <file> · 133

vendor name of <processor> · 173  
 vendor of <dmi bios\_information> · 205  
 vendor\_syndrome of <dmi  
   b32\_bit\_memory\_error\_information> · 202  
 vendor\_syndrome of <dmi  
   b64\_bit\_memory\_error\_information> · 203  
 version · 1, 2, 3, 4, 13, 25, 27, 28, 31, 87, 88, 90,  
   91, 92, 93, 96, 103, 105, 106, 108, 110, 112,  
   114, 126, 128, 130, 131, 132, 133, 134, 135,  
   148, 149, 150, 151, 152, 153, 154, 156, 159,  
   163, 164, 167, 168, 169, 170, 171, 173, 194,  
   200, 201, 204, 221, 222, 233, 241, 276, 332,  
   337  
 version <string> · 110, 150  
 version block <integer> of <file> · 133, 153  
 version block <string> of <file> · 133, 153  
 version block of <file> · 133, 153  
 version of <bios> · 167  
 version of <dmi base\_board\_information> · 204  
 version of <dmi system\_enclosure\_or\_chassis> ·  
   221  
 version of <dmi system\_information> · 222  
 version of <file> · 134, 150  
 version of <site> · 233  
 version string <string> of <module> · 332  
 volatile attribute of <metabase value> · 298  
 voltage of <dmi processor\_information> · 220  
 voltage\_probe <integer> of <dmi> · 225  
 voltage\_probe of <dmi> · 225

---

## W

waiting for download of <action> · 272  
 wake\_up\_type of <dmi system\_information> ·  
   222  
 wakeonlan enabled of <network adapter> · 281  
 warning event log event type · 110, 248  
 wednesday · 66, 110, 332  
 week · 9, 27, 32, 50, 58, 60, 62, 64, 65, 66, 67, 78,  
   90, 91, 110, 171, 312, 313, 332, 336  
 weight of <selected server> · 239  
 win32 exit code of <service> · 181  
 windows display time <string> · 47, 110, 332  
 windows file <string> · 110, 129  
 windows folder · 51, 110, 138, 141, 142, 147,  
   151, 152, 261  
 wins enabled of <network adapter> · 281  
 winsock2 supported of <network> · 274  
 wmi · 94, 110, 226, 227, 228, 229, 230, 337  
 wmi <string> · 110, 226

wmi object · 94, 110, 226, 227, 230, 337  
 WMI Objects · iv, 2, 226  
 wmi select · 226, 227, 228, 229, 230, 337  
 workstation trust account flag of <local user> ·  
   268  
 world · 1, 85, 307, 308, 309, 310, 311, 312, 313,  
   314, 315, 316, 317, 318, 319, 320, 321, 322,  
   323, 325, 326, 327, 328, 329, 330, 331, 332,  
   333  
 World Objects · ii, 1, 85  
 write attributes permission of <access control  
   entry> · 255  
 write dac permission of <access control entry> ·  
   255  
 write extended attributes permission of <access  
   control entry> · 255  
 write owner permission of <access control entry>  
   · 255  
 write permission of <access control entry> · 255  
 write permission of <network share> · 285

---

## X

x32 application <string> · 110, 137  
 x32 file <string> · 111, 130  
 x32 folder <string> · 111, 141  
 x32 registry · 111, 114  
 x64 application <string> · 111, 137  
 x64 file <string> · 111, 130  
 x64 folder <string> · 111, 141  
 x64 of <operating system> · 170  
 x64 registry · 112, 114  
 xml document of <file> · 134, 159  
 xml document of <string> · 159, 332  
 xml dom document · 134, 160, 161, 162, 324, 332  
 xml dom node · 161, 162, 163, 164, 308, 310,  
   315, 318, 323, 324, 325, 327, 332, 337  
 xpath <( string, string )> of <xml dom node> ·  
   161, 163, 332  
 xpath <string> of <xml dom node> · 161, 163,  
   332

---

## Y

year · 3, 9, 12, 28, 32, 50, 62, 63, 64, 70, 71, 72,  
   73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 90,  
   112, 168, 247, 312, 313, 318, 328, 332, 333,  
   334, 337  
 year <integer> · 82, 112, 332  
 year <string> · 82, 112, 332

year of <date> · 64, 83, 332  
year of <month and year> · 79, 83, 333

---

**Z**

zone of <time of day with time zone> · 55, 333  
zoned time\_of\_day <string> · 54, 112, 333