Windows Mobile Devices
Inspector Library

Compatible with BES 7.2

July 25, 2009

© 2009 BigFix, Inc. All rights reserved.

BigFix®, Fixlet®, Relevance Engine®, Powered by BigFix™ 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, or (2) an endorsement of the company or its products by BigFix, Inc.

Except as set forth in the last sentence of this paragraph: (1) 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., and (2) you may not use this documentation for any purpose except in connection with your properly licensed use or evaluation of BigFix software and any other use, including for reverse engineering such software or creating derivative works thereof, is prohibited. If your license to access and use the software that this documentation accompanies is terminated, you must immediately return this documentation to BigFix, Inc. and destroy all copies you may have. You may treat only those portions of this documentation specifically designated in the "Acknowledgements and Notices" section below as notices applicable to third party software in accordance with the terms of such notices.

All inquiries regarding the foregoing should be addressed to:

BigFix, Inc.

1480 64th Street, Suite 200

Emeryville, California 94608

Contents

Copyright © 2009 BigFix, Inc.

Copyright © 2009 BigFix, Inc.

Preface

The ***Windows Mobile Devices Inspector Library*** is a guide to the ordinary phrases (known as Inspectors) of the **Relevance Language**™ as they apply to Window Mobile (WM) Devices. Using this guide, you can write your own Relevance Expressions and use them to target actions to exactly those mobile devices 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 Relevance Language Reference***. In addition, there are a suite of Core Inspectors that you should refer to in the ***BigFix Core Inspector Library***.

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 customized Fixlet messages to keep their users updated and their support calls to a minimum. 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.

Copyright © 2009 BigFix, Inc.

- **Firewall Objects.** This chapter details the firewall Inspectors that examine the authorized applications, policies, services, settings and more.
- **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.
- **Key phrases (Inspector List).** This chapter provides an alphabetical list of all the Inspector keywords along with the form, context object type, and resulting object type.

Copyright © 2009 BigFix, Inc.

Conventions Used in this manual

This document makes use of the following conventions and nomenclature:

Convention	Use
Bold Sans	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.
Italics	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"

Copyright © 2009 BigFix, Inc.

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.:

WM:7.2, Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1

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

WM: the Windows Mobile devices version of the BigFix Enterprise Suite (BES).

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

Lin: 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.

Copyright © 2009 BigFix, Inc.

Introduction

This manual details the properties and operators of the BigFix Inspector keywords as they apply to Windows Mobile devices. This document should be used along with the ***BigFix Core Inspector Guide*** to provide a complete library reference. 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 Mobile (WM) devices.. Contact your BigFix sales representative for information about Inspector Guides for other operating systems, including Windows, Solaris, Mac, HPUX, 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 several **forms** depending upon their context:

Form	Syntax required
Cast	<object> as keyword
Global	keyword
Index	keyword *index* of <object>
Named	keyword "*name*" of <object>
NamedGlobal	keyword "*name*"
Numbered	keyword *number* of <object>
NumberedGlobal	keyword *number*
Plain	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.

Copyright © 2009 BigFix, Inc.

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.

Integer Range

These Inspectors specify a range between two 64-bit signed integers.

Creation Methods

Key Phrase	*Form*	Description
distance of <selected server>	*Plain*	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, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
lower bound of <integer range>	*Plain*	<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, WM:7.2
upper bound of <integer range>	*Plain*	<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, WM:7.2

Copyright © 2009 BigFix, Inc.

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	*PlainGlobal*	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, WM:7.2
registry	*PlainGlobal*	Creates an object for accessing the registry. Win:1.2, WM:7.2
x32 registry	*PlainGlobal*	Returns a 32-bit registry object. This Inspector is equivalent to the ordinary registry Inspector. Win:6.0, WM:7.2
x64 registry	*PlainGlobal*	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. Win:6.0, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
application <string> of <registry>	*Named*	<application>	Returns an application object matching name provided. See application. Win:1.2, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
application folder <string> of <registry>	*Named*	<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, WM:7.2
application of <registry>	*Plain*	<application>	Iterates through the properly installed applications. See application. Win:1.2, WM:7.2
file extension <string> of <registry>	*Named*	<registry key>	Returns a key associated with the named extension. See registry key. Win:1.2, WM:7.2
file type <string> of <registry>	*Named*	<registry key>	Returns a key associated the named file type. See registry key. Win:1.2, WM:7.2
key <string> of <registry>	*Named*	<registry key>	Returns a key associated with the name provided. See registry key. Win:1.2, WM:7.2

Examples

- `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
file extension <string> of <registry>	*Named*	Creates a key object provided the registry indicates support for the named file extension. Win:1.2, WM:7.2
file type <string> of <registry>	*Named*	Creates a key object provided the registry indicates support for the named file type. Win:1.2, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Description
key <string> of <registry key>	*Named*	Creates an object for the named sub-key of the key. Win:1.2, WM:7.2
key <string> of <registry>	*Named*	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, WM:7.2
key of <registry key>	*Plain*	Iterates through the sub-keys of a key. Win:1.2, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
application <string> of <registry key>	*Named*	<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, WM:7.2
application folder <string> of <registry key>	*Named*	<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, WM:7.2
application folder of <registry key>	*Plain*	<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, WM:7.2
application of <registry key>	*Plain*	<application>	Returns the application 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, WM:7.2
default value of <registry key>	*Plain*	<registry key value>	Returns the unnamed value associated with a key as a string. It does not necessarily exist. Win:1.2, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
key <string> of <registry key>	*Named*	<registry key>	Returns a key for the named sub-key. Win:1.2, WM:7.2
key of <registry key>	*Plain*	<registry key>	Iterates through the sub-keys of the key. Win:1.2, WM:7.2
name of <registry key>	*Plain*	<string>	Returns the name of the key as a string. Win:1.2, WM:7.2
value <string> of <registry key>	*Named*	<registry key value>	Returns the named value stored under the key. See registry key value. Win:1.2, WM:7.2
value of <registry key>	*Plain*	<registry key value>	Iterates through values stored under a key. Win:1.2, WM:7.2

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

Copyright © 2009 BigFix, Inc.

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.

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>	*Plain*	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, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Description
value <string> of <registry key>	*Named*	Creates an object with the value of the key. The name property of the value will match the name provided. Win:1.2, WM:7.2
value of <registry key>	*Plain*	Creates an object with all the values of a key. Win:1.2, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
<registry key value> as application	*Cast*	<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, WM:7.2
<registry key value> as file	*Cast*	<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, WM:7.2
<registry key value> as folder	*Cast*	<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, WM:7.2
<registry key value> as integer	*Cast*	<integer>	Returns the value stored in the registry entry provided it can be fully represented as an integer. Win:1.2, WM:7.2
<registry key value> as string	*Cast*	<string>	Returns a string if the data of the value is of type REG_SZ. Win:1.2, WM:7.2
<registry key value> as system file	*Cast*	<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, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
<registry key value> as time	*Cast*	<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, WM:7.2
name of <registry key value>	*Plain*	<string>	Returns the name of the value as a string. (see escape of <string> for more information). Win:1.2, WM:7.2
size of <registry key value>	*Plain*	<integer>	Returns the size of the data as an integer. Win:1.2, WM:7.2
type of <registry key value>	*Plain*	<registry key value type>	Returns the type of the data of the value. See type of value of key or registry. Win:1.2, WM:7.2

Operators

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

Copyright © 2009 BigFix, Inc.

Examples

- `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.

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>	*Plain*	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, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
<registry key value type> as string	*Cast*	<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, WM:7.2

Copyright © 2009 BigFix, Inc.

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, WM:7.2
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, Mac:6.0, WM:7.2
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, WM:7.2
compressed of <filesystem object>	*Plain*	<boolean>	Returns TRUE if the file or folder (filesystem object) has been compressed. Win:6.0, WM:7.2
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, WM:7.2
drive of <filesystem object>	*Plain*	<drive>	Returns the drive associated with the specified file or folder (filesystem object). Win:6.0, WM:7.2
hidden of <filesystem object>	*Plain*	<boolean>	Returns TRUE if the file or folder (filesystem object) is marked as hidden. Win:6.0, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
location of <filesystem object>	*Plain*	<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, WM:7.2
modification time of <filesystem object>	*Plain*	<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, WM:7.2
name of <filesystem object>	*Plain*	<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, WM:7.2
normal of <filesystem object>	*Plain*	<boolean>	Returns TRUE if the file or folder (filesystem object) is 'normal'. Win:6.0, WM:7.2
offline of <filesystem object>	*Plain*	<boolean>	Returns TRUE if the file or folder (the filesystem object) is marked as 'offline'. Win:6.0, WM:7.2
parent folder of <filesystem object>	*Plain*	<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, WM:7.2
pathname of <filesystem object>	*Plain*	<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, WM:7.2
readonly of <filesystem object>	*Plain*	<boolean>	Returns TRUE if the file or folder (the filesystem object) is marked as read-only. Win:6.0, WM:7.2
system of <filesystem object>	*Plain*	<boolean>	Returns TRUE if the file or folder (the filesystem object) is marked as a system folder. Win:6.0, WM:7.2
temporary of <filesystem object>	*Plain*	<boolean>	Returns TRUE if the file or folder (the filesystem object) is marked as a temporary folder. Win:6.0, WM:7.2

Copyright © 2009 BigFix, Inc.

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.

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.
• 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.

WARNING: Some file content Inspectors can cause contention issues with other applications, regardless of the platform. These Inspectors open up the file for read access with maximal sharing with other applications. However, if other applications try to access the file with exclusive rights, they will fail. The set of Inspectors that hold a handle to the file are: 'lines of file', 'contents of file' and 'sha1 of file'.

Creation Methods

See application objects for additional creation methods

Key Phrase	***Form***	**Description**
<registry key value> as file	*Cast*	If the value contains a string and the string points to an existing file, a file object is returned. Win:1.2, WM:7.2
<registry key value> as system file	*Cast*	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, WM:7.2
descendant of <folder>	*Plain*	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, Mac:6.0, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Description
download file <string>	*NamedGlobal*	This inspector is available in relevance subsitution action processing. It returns a file object with the given name from the named folder or the download folder. This is equivalent to 'file "name" of download folder'. The file should exist or the result will not exist. Win:7.2, Lin:7.2, Sol:7.2, HPUX:7.2, AIX:7.2, Mac:7.2, WM:7.2
file <string>	*NamedGlobal*	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, WM:7.2
file <string> of <folder>	*Named*	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, WM:7.2
file of <folder>	*Plain*	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, WM:7.2
find file <string> of <folder>	*Named*	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, WM:7.2
masthead of <site>	*Plain*	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, WM:7.2
system file <string>	*NamedGlobal*	Creates the file objects corresponding to the named file within the system folder. Win:1.2, WM:7.2
windows file <string>	*NamedGlobal*	Returns a file object corresponding to the relative pathname (within the Windows folder) provided. See file. Win:1.2, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
<file> as string	*Cast*	<string>	Creates a string containing the full pathname of the specified file. See <file>. Win:1.2, , , , , WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
byte <integer> of <file>	*Numbered*	<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, Mac:6.0, WM:7.2
content of <file>	*Plain*	<file content>	Returns an object that can be used to search for a string in the file. See content. WARNING: This Inspector maintains a handle to the specified file, so during its operation it will block any other applications that attempt to open the file. Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:6.0, WM:7.2
executable file format of <file>	*Plain*	<string>	Returns a four-byte string containing the format specifier for the specified file. Win:4.1, WM:7.2
file version of <file>	*Plain*	<version>	Returns the file version extracted from the file's resource block. See version. Win:1.2, WM:7.2
first raw version block of <file>	*Plain*	<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, WM:7.2
line <integer> of <file>	*Numbered*	<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, WM:7.2
line containing <string> of <file>	*Named*	<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, WM:7.2
line of <file>	*Plain*	<file line>	Iterates over all the lines of the specified file. NOTE: lines are truncated to 1023 characters. WARNING: This Inspector maintains a handle to the specified file, so during its operation it will block any other applications that attempt to open the file. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:5.1, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
line starting with <string> of <file>	*Named*	<file line>	Same as line <string> of <file>, returns the lines of the given file that start with the specified string. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:5.1, WM:7.2
only raw version block of <file>	*Plain*	<file version block>	Returns the only version block directly from a PE file. Win:4.1, WM:7.2
only version block of <file>	*Plain*	<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, WM:7.2
pem encoded certificate of <file>	*Plain*	<x509 certificate>	Reads and returns the certificate from a file in the PEM format. This can be used to analyze encryption credentials on decrypting relays or root servers. Win:7.1, WM:7.2
product version of <file>	*Plain*	<version>	Returns the product version extracted from the file's resource block. See version. Win:1.2, WM:7.2
raw file version of <file>	*Plain*	<version>	Returns the file version directly from a PE file. Win:4.1, WM:7.2
raw product version of <file>	*Plain*	<version>	Returns the product version directly from a PE file. Win:4.1, WM:7.2
raw version block <integer> of <file>	*Numbered*	<file version block>	Returns the numbered version block directly from a PE file. Win:4.1, WM:7.2
raw version block <string> of <file>	*Named*	<file version block>	Returns the named version block directly from a PE file. Win:4.1, WM:7.2
raw version block of <file>	*Plain*	<file version block>	Returns the version block directly from a PE file. Win:4.1, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
raw version of <file>	*Plain*	<version>	Returns the version directly from a PE file. Win:4.1, WM:7.2
sha1 of <file>	*Plain*	<string>	Returns the sha1 checksum of the file hex encoded as a 40 character long string. WARNING: This Inspector maintains a handle to the specified file, so during its operation it will block any other applications that attempt to open the file. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
size of <file>	*Plain*	<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, WM:7.2
variable of <file>	*Plain*	<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, WM:7.2
version block <integer> of <file>	*Numbered*	<file version block>	You can identify the particular version block you want to access by ordinal number. Win:1.2, WM:7.2
version block <string> of <file>	*Named*	<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, WM:7.2
version block of <file>	*Plain*	<file version block>	Iterates through the version blocks of a file. Win:1.2, WM:7.2
version of <file>	*Plain*	<version>	Synonym for file version of <file>. Win:1.2, Mac:4.1, WM:7.2
xml document of <file>	*Plain*	<xml dom document>	Returns an XML Document Object Model (DOM) for the specified file. Win:5.1, WM:7.2

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.

Copyright © 2009 BigFix, Inc.

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

- `wait "{pathname of download file "update.exe"}"`

In an Action script, this line causes the BES Client to perform relevance substitution to compute the full path to the downloaded file (previously collected by a download command in the same Action script). After relevance substitution, the Client launches the specified executable and waits for it to complete before moving on to other Action lines.

- `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.

- `file version of application "iexplore.exe" of the registry < "4"`

Test for older version of IE -- returns TRUE is version is less than 4.

- `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.

Copyright © 2009 BigFix, Inc.

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.

• 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.

Creation Methods

Key Phrase	*Form*	Description
<registry key value> as application	*Cast*	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, WM:7.2
application <string>	*NamedGlobal*	Creates an application object for the name provided. Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, WM:7.2
application <string> of <folder>	*Named*	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, WM:7.2
application <string> of <registry key>	*Named*	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, WM:7.2
application <string> of <registry>	*Named*	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, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Description
application of <registry key>	*Plain*	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, WM:7.2
application of <registry>	*Plain*	Iterates through the 'App Paths' registry key creating objects for the applications that exist. See notes. Win:1.2, WM:7.2
recent application	*PlainGlobal*	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, WM:7.2
recent application <string>	*NamedGlobal*	If named application has been executed recently, this inspector creates an application object. Only specify the last component of the filename. Win:1.2, Lin:6.0, WM:7.2
regapp	*PlainGlobal*	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, WM:7.2
regapp <string>	*NamedGlobal*	Returns an application object for the name provided. See application and regapp. Win:1.2, WM:7.2
running application	*PlainGlobal*	Iterates through the list of running applications. Win:1.2, Lin:6.0, WM:7.2
running application <string>	*NamedGlobal*	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, WM:7.2

Copyright © 2009 BigFix, Inc.

Examples

- `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.

Copyright © 2009 BigFix, Inc.

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	*Cast*	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, WM:7.2
ancestor of <filesystem object>	*Plain*	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, Mac:6.0, WM:7.2
application folder <string> of <registry key>	*Named*	Synonym for pathname of parent folder of regapp <string>. Win:1.2, WM:7.2
application folder <string> of <registry>	*Named*	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, WM:7.2
application folder of <registry key>	*Plain*	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, WM:7.2
client folder of <site>	*Plain*	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, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	***Form***	**Description**
csidl folder <integer>	*NumberedGlobal*	Returns the csidl folder corresponding to the specified integer. The windows SHGetSpecialFolderLocation API is used to look up paths to special folders, which are identified by passing the specified integer as the second argument of the API call. These values and their meaning are described in the windows ShlObj.h include file found in the development sdk. • Note that some of these folders do not exist in the Local System context. Win:7.0, WM:7.2
descendant folder of <folder>	*Plain*	Returns the descendant folders, recursively, of the given folder. The folder equivalent of "descendants of <folder>". Win:7.0, Lin:7.0, Sol:7.0, HPUX:7.0, AIX:7.0, Mac:7.1, WM:7.2
folder <string>	*NamedGlobal*	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, WM:7.2
folder <string> of <drive>	*Named*	Creates a folder object for the name provided if it exists on the drive provided. Win:1.2, WM:7.2
folder <string> of <folder>	*Named*	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, WM:7.2
folder of <folder>	*Plain*	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, WM:7.2
install folder <integer>	*NumberedGlobal*	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, WM:7.2
parent folder of <filesystem object>	*Plain*	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, WM:7.2
root folder of <drive>	*Plain*	Creates a folder object for the root of the given drive. Win:1.2, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Description
system wow64 folder	*PlainGlobal*	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: http://msdn.microsoft.com/library/default.asp?url=/library/en-us/sysinfo/base/getsystemwow64directory.asp. Win:6.0, WM:7.2
system x32 folder	*PlainGlobal*	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, WM:7.2
system x64 folder	*PlainGlobal*	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 http://msdn.microsoft.com/library/default.asp?url=/library/en-us/win64/win64/file_system_redirector.asp. Win:6.0, WM:7.2
windows folder	*PlainGlobal*	Creates a folder object of the Windows folder. This is operating system dependent. Under Win98 this is usually c:\Windows. Win:1.2, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
application <string> of <folder>	*Named*	<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, WM:7.2
descendant folder of <folder>	*Plain*	<folder>	Returns the descendant folders, recursively, of the given folder. The folder equivalent of "descendants of <folder>". Win:7.0, Lin:7.0, Sol:7.0, HPUX:7.0, AIX:7.0, Mac:7.1, WM:7.2
descendant of <folder>	*Plain*	<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, Mac:6.0, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
file <string> of <folder>	*Named*	<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, WM:7.2
file of <folder>	*Plain*	<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, WM:7.2
find file <string> of <folder>	*Named*	<file>	Iterates through the files of a folder returning file objects whose name matches the search string provided in the name parameter. See example below. Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, WM:7.2
folder <string> of <folder>	*Named*	<folder>	Returns 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, WM:7.2
folder of <folder>	*Plain*	<folder>	Iterates through the folders of a folder returning folder objects. When combined with a whose clause, you can select folders with specific properties. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2

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 careful not to iterate through folders that contain a large number of files. Counting files in such a folder can be slow. Always try to use the most efficient techniques to minimize the client overhead. Consider using the “find file” Inspector which allows you to filter sets of files by using the wildcard.

Examples

- `install folder 11`
  - Returns a folder object for system folder identified with this number.

Copyright © 2009 BigFix, Inc.

Drive

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

Creation Methods

Key Phrase	*Form*	Description
drive	*PlainGlobal*	Iterates through all valid drives on the system. Win:1.2, WM:7.2
drive <string>	*NamedGlobal*	Creates the drive object for the name specified. Win:1.2, WM:7.2
drive of <filesystem object>	*Plain*	Returns the drive associated with the specified file or folder (filesystem object). Win:6.0, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
file system type of <drive>	*Plain*	<string>	Value as reported by GetVolumeInformation. Win:1.2, WM:7.2
file_supports_encryption of <drive>	*Plain*	<boolean>	TRUE if bit is returned by GetVolumeInformation. Win:1.2, WM:7.2
file_supports_object_ids of <drive>	*Plain*	<boolean>	TRUE if bit is returned by GetVolumeInformation. Win:1.2, WM:7.2
file_supports_reparse_points of <drive>	*Plain*	<boolean>	TRUE if bit is returned by GetVolumeInformation. Win:1.2, WM:7.2
file_supports_sparse_files of <drive>	*Plain*	<boolean>	TRUE if bit is returned by GetVolumeInformation. Win:1.2, WM:7.2
file_volume_quotas of <drive>	*Plain*	<boolean>	TRUE if bit is returned by GetVolumeInformation. Win:1.2, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
folder <string> of <drive>	*Named*	<folder>	Returns a folder object corresponding to the name given provided that folder exists on the drive. Win:1.2, WM:7.2
free space of <drive>	*Plain*	<integer>	Returns the number of unused bytes of storage for the drive. (Only available for fixed disks). Win:1.2, WM:7.2
fs_case_is_preserved of <drive>	*Plain*	<boolean>	TRUE if bit is returned by GetVolumeInformation. Win:1.2, WM:7.2
fs_case_sensitive of <drive>	*Plain*	<boolean>	TRUE if bit is returned by GetVolumeInformation. Win:1.2, WM:7.2
fs_file_compression of <drive>	*Plain*	<boolean>	TRUE if bit is returned by GetVolumeInformation. Win:1.2, WM:7.2
fs_persistent_acls of <drive>	*Plain*	<boolean>	TRUE if bit is returned by GetVolumeInformation. Win:1.2, WM:7.2
fs_unicode_stored_on_disk of <drive>	*Plain*	<boolean>	TRUE if bit is returned by GetVolumeInformation. Win:1.2, WM:7.2
fs_vol_is_compressed of <drive>	*Plain*	<boolean>	TRUE if bit is returned by GetVolumeInformation. Win:1.2, WM:7.2
name of <drive>	*Plain*	<string>	Returns the name of the drive. Names look like 'c:' and 'D:'. Win:1.2, WM:7.2
numeric type of <drive>	*Plain*	<integer>	Returns the type of drive as an integer. Win:1.2, WM:7.2
root folder of <drive>	*Plain*	<folder>	Returns the folder corresponding to the root of the drive. Win:1.2, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
total space of <drive>	*Plain*	<integer>	Returns the size in bytes of the drive. (Only available for fixed disks). Win:1.2, WM:7.2
type of <drive>	*Plain*	<string>	Returns the type of drive as a string. Win:1.2, WM:7.2

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).

Copyright © 2009 BigFix, Inc.

- `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 Content

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

Creation Methods

Key Phrase	*Form*	Description
<file content> as lowercase	*Cast*	Returns the contents of the file as lower case characters. Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:6.0, WM:7.2
<file content> as uppercase	*Cast*	Returns the contents of the file as upper case characters. Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:6.0, WM:7.2
content of <file>	*Plain*	Creates a content object for a file. WARNING: This Inspector maintains a handle to the specified file, so during its operation it will block any other applications that attempt to open the file. Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:6.0, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
<file content> as lowercase	*Cast*	<file content>	Returns a lowercase version of the content provided. Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:6.0, WM:7.2
<file content> as uppercase	*Cast*	<file content>	Returns an uppercase version of the content provided. Win:1.2, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:6.0, WM:7.2

Copyright © 2009 BigFix, Inc.

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, Mac:6.0, WM:7.2

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

Examples

- `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.

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>	*Plain*	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, WM:7.2
only raw version block of <file>	*Plain*	Returns the only version block directly from a PE file. Win:4.1, WM:7.2
only version block of <file>	*Plain*	Most applications only have 1 version block. This inspector allows language independent access when there is only one version block present. Win:1.2, WM:7.2
raw version block <integer> of <file>	*Numbered*	Returns the numbered version block directly from a PE file. Win:4.1, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Description
raw version block <string> of <file>	*Named*	Returns the named version block directly from a PE file. Win:4.1, WM:7.2
raw version block of <file>	*Plain*	Returns the version block directly from a PE file. Win:4.1, WM:7.2
version block <integer> of <file>	*Numbered*	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, WM:7.2
version block <string> of <file>	*Named*	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, WM:7.2
version block of <file>	*Plain*	Iterates through the version blocks of a file. Win:1.2, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
codepage of <file version block>	*Plain*	<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, WM:7.2
id of <file version block>	*Plain*	<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, WM:7.2
language of <file version block>	*Plain*	<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, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
value <string> of <file version block>	*Named*	<string>	Returns a string corresponding to the name provided. Values have names such as 'CompanyName', 'FileDescription', 'FileVersion'. Win:1.2, WM:7.2

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)'.

Copyright © 2009 BigFix, Inc.

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	Croation	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)

Copyright © 2009 BigFix, Inc.

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

- `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>	*Numbered*	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, WM:7.2
line containing <string> of <file>	*Named*	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, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Description
line of <file>	*Plain*	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, WM:7.2
line starting with <string> of <file>	*Named*	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, WM:7.2
next line of <file line>	*Plain*	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, WM:7.2
previous line of <file line>	*Plain*	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, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
line number of <file line>	*Plain*	<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, WM:7.2
next line of <file line>	*Plain*	<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, WM:7.2
previous line of <file line>	*Plain*	<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, WM:7.2

Copyright © 2009 BigFix, Inc.

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	*PlainGlobal*	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, Mac:6.0, WM:7.2
application usage summary <string>	*NamedGlobal*	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, Mac:6.0, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
first start time of <application usage summary>	*Plain*	<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, Mac:6.0, WM:7.2
last start time of <application usage summary>	*Plain*	<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, Mac:6.0, WM:7.2
last time seen of <application usage summary>	*Plain*	<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, Mac:6.0, WM:7.2
name of <application usage summary>	*Plain*	<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, Mac:6.0, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
running of <application usage summary>	*Plain*	<boolean>	Returns TRUE if the specified application is currently running. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:6.0, WM:7.2
total duration of <application usage summary>	*Plain*	<time interval>	Returns the total elapsed time that the specified application has been running. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:6.0, WM:7.2
total run count of <application usage summary>	*Plain*	<integer>	Returns the number of times that the specified application has been run since the client was configured to track it. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:6.0, WM:7.2

Copyright © 2009 BigFix, Inc.

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 compatiblity.

Operating System

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

Creation Methods

Key Phrase	*Form*	Description
operating system	*PlainGlobal*	Creates the global operating system object. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
<operating system> as string	*Cast*	<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, WM:7.2
build number high of <operating system>	*Plain*	<integer>	Numeric representation of the most significant 16 bits of the build number. Win:1.2, WM:7.2
build number low of <operating system>	*Plain*	<integer>	Numeric representation of the least significant 16 bits of the build number. Win:1.2, WM:7.2
csd version of <operating system>	*Plain*	<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, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
ia64 of <operating system>	*Plain*	<boolean>	Returns TRUE iff the BES Client is running on Itanium. Win:7.0, WM:7.2
major version of <operating system>	*Plain*	<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, WM:7.2
minor version of <operating system>	*Plain*	<integer>	Numeric representation of the minor version of the operating system. Win:1.2, WM:7.2
name of <operating system>	*Plain*	<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, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
performance counter frequency of <operating system>	*Plain*	<hertz>	The rate at which the performance counter is being incremented (per second). Win:1.2, WM:7.2
performance counter of <operating system>	*Plain*	<integer>	Retrieves a 64-bit performance counter value. Win:1.2, WM:7.2
platform id of <operating system>	*Plain*	<integer>	Returns the dwPlatformId as returned by the GetVersionEx system call. Possible values are 1 (Win95/95) and 2 (WinNT). Win:1.2, WM:7.2
product info numeric of <operating system>	*Plain*	<integer>	This Windows-specific inspector returns the integer from the Windows GetProductInfo API. The inspector only provides meaningful results for Windows Vista and newer versions of the OS. The major/minor version of the OS must be 6.0 or greater for the result to be meaningful. For more information, refer to the Microsoft article at http://msdn2.microsoft.com/en-us/library/ms724358(VS.85).aspx. Win:7.0, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
product info string of <operating system>	*Plain*	<string>	On Windows versions 6.0 and newer (Vista minimum), this inspector returns a string derived from the GetProductInfo API. It will be one of the following values: • Unlicensed • Business • Cluster Server • Server Datacenter • Server Datacenter Core • Enterprise • Server Enterprise • Server Enterprise Core • Server Enterprise Itanium • Home Basic • Home Server • Server for Small Business • Small Business Server • Small Business Server Premium • Server Standard • Server Standard Core • Starter • Storage Server Enterprise • Storage Server Standard • Storage Server Workgroup • Ultimate • Web Server • Unknown. Win:7.0, WM:7.2
product type of <operating system>	*Plain*	<operating system product type>	Returns the product type of the operating system, which includes Workstations, Domain Controllers and Servers. Win:6.0, WM:7.2
release of <operating system>	*Plain*	<string>	Information about the release of the operating system, formatted as a <version> on the Macintosh, but a <string> on Unix and Windows. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
service pack major version of <operating system>	*Plain*	<integer>	Returns the major version number of the current service pack of the specified OS. Win:6.0, WM:7.2
service pack minor version of <operating system>	*Plain*	<integer>	Returns the minor version number of the current service pack of the specified OS. Win:6.0, WM:7.2
suite mask of <operating system>	*Plain*	<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, WM:7.2
x64 of <operating system>	*Plain*	<boolean>	Returns TRUE if the current operating system is 64-bits. Win:6.0, WM:7.2

Examples

- `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.
- `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.
- `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.
- `platform id of operating system = 1`
  - Returns TRUE on a Win95 System.

Copyright © 2009 BigFix, Inc.

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	*PlainGlobal*	Creates the object associated with the 'Primary' processor. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
processor	*PlainGlobal*	Iterates through the processors in the system. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
processor <integer>	*NumberedGlobal*	Creates the processor object for the number specified. The first processor is processor number 1. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
brand id of <processor>	*Plain*	<integer>	This inspector returns the integer known as the brand id, returned from the assembly language cpuid extended instruction. Win:6.0, WM:7.2
brand string of <processor>	*Plain*	<string>	Returns the vendor-defined brand names for newer processors. Win:1.2, WM:7.2
extended family of <processor>	*Plain*	<integer>	Integer representing the extended family of CPU. See the notes for the meaning of these numbers. Win:1.2, WM:7.2
extended model of <processor>	*Plain*	<integer>	Integer representing the extended model of CPU. See the notes for the meaning of these numbers. Win:1.2, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
family name of <processor>	*Plain*	<string>	Returns the family name of the CPU, dependent on the type of client computer, for instance Pentium, Sparc, PowerPC G4, etc. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
feature mask of <processor>	*Plain*	<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, WM:7.2
model of <processor>	*Plain*	<integer>	Returns the model number of the CPU. • Note: On Solaris, HPUX and AIX computers, this Inspector returns a <string> as of BES 6.0. Win:1.2, Lin:3.1, WM:7.2
speed of <processor>	*Plain*	<hertz>	Returns the speed of the processor in Hertz. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
stepping of <processor>	*Plain*	<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, WM:7.2
type of <processor>	*Plain*	<integer>	Numeric type of the CPU. Values include: • 0 - standard • 1 - overdrive • 2 - dual CPU capable • 3 - reserved • Note: this Inspector returns a <string> type as of BES version 6.0 on Unix machines and version 5.1 on the Macintosh. Win:1.2, WM:7.2
vendor name of <processor>	*Plain*	<string>	The manufacturer of the CPU. Names include: • GenuineIntel • AuthenticAMD • CyrixInstead • CentaurHauls • AmbiguousCPU. Win:1.2, Lin:3.1, WM:7.2

Copyright © 2009 BigFix, Inc.

Examples

- `number of processors > 1`
  - Returns TRUE if the computer is a multi-processor system.

- `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	*PlainGlobal*	Creates the object that can be accessed to inspect the amount of ram on the machine. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
random access memory	*PlainGlobal*	Same as 'ram'. Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
available of <ram>	*Plain*	<integer>	Returns the total amount of RAM (in bytes) currently available on the Windows Mobile device. This is the same as for for the Windows client. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
load of <ram>	*Plain*	<integer>	Returns the amount of memory being used on the Windows Mobile device as a percentage. 0 = no memory used, 100 = all memory used. This is the same as for for the Windows client. WM:7.2
size of <ram>	*Plain*	<integer>	Returns the number of bytes of random access memory on the current machine. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2

Examples

- `available of ram`

  Returns the number of bytes of RAM, such as 72351744.

- `load of ram`

  Returns a number from 0 to 100 indicating the percentage of RAM currently being used in the Windows Mobile device.

- `size of ram / (1024 * 1024)`

  Returns the size of RAM in megabytes.

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	*PlainGlobal*	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, WM:7.2
system ui language	*PlainGlobal*	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, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Description
user locale	*PlainGlobal*	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, WM:7.2
user ui language	*PlainGlobal*	Non-MUI: Same as system UI Language. • MUI: Determines the language of menus and dialogs, messages, and help files. Win:4.1, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
<language> as string	*Cast*	<string>	Returns the language of the system locale. Win:4.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, WM:7.2
primary language of <language>	*Plain*	<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, WM:7.2

Examples

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

Copyright © 2009 BigFix, Inc.

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>	*Plain*	Extracts the primary language identifier from a language. Win:4.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
<primary language> as string	*Cast*	<string>	Returns the primary language. Win:4.1, Lin:5.1, Sol:5.1, HPUX:5.1, AIX:5.1, WM:7.2

Examples

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

Copyright © 2009 BigFix, Inc.

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	*PlainGlobal*	Creates the site object corresponding to the site that provided the current Fixlet. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
site	*PlainGlobal*	Iterates through all the sites. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
site <string>	*NamedGlobal*	Creates the site object that corresponds to the name provided. The name is interpreted as a site locator and is therefore a URL. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
client folder of <site>	*Plain*	<folder>	The folder containing the site content on the client machine. Site content is gathered into this location. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
fixlet of <site>	*Plain*	<fixlet>	Iterates through the Fixlet messages of the specified site. Win:5.0, Lin:5.0, Sol:5.0, HPUX:5.0, AIX:5.0, Mac:6.0, WM:7.2
gather schedule authority of <site>	*Plain*	<string>	Returns a string corresponding to the authority of the site schedule, e.g.: Publisher, Custom, Manual or Disabled. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
gather schedule time interval of <site>	*Plain*	<time interval>	Returns the time interval between automatic gathering of site content. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
group <integer> of <site>	*Numbered*	<site group>	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, Mac:6.0, WM:7.2
last gather time of <site>	*Plain*	<time>	Returns the time of last successful gathering from the site. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
masthead of <site>	*Plain*	<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. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
name of <site>	*Plain*	<string>	The name of the site. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
relevant fixlet of <site>	*Plain*	<fixlet>	Iterates through the Relevant Fixlet messages for the specified site. Win:5.0, Lin:5.0, Sol:5.0, HPUX:5.0, AIX:5.0, Mac:6.0, WM:7.2
setting <string> of <site>	*Named*	<setting>	Returns the setting whose name matches the string provided from the Fixlet site settings. Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
setting of <site>	*Plain*	<setting>	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, WM:7.2
site tag of <site>	*Plain*	<string>	Returns the last component of the specified site's url, eg. ‘actionsite’, ‘enteprisesecurity’, etc. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:6.0, WM:7.2
site version list of <site>	*Plain*	<site version list>	Returns the last gathered site version list (manyversion) of the specified site. Win:7.0, Lin:7.1, Sol:7.1, HPUX:7.1, AIX:7.0, Mac:7.1, WM:7.2
subscribe time of <site>	*Plain*	<time>	Returns the time that the current machine began subscribing to the site. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
type of <site>	*Plain*	<string>	Returns one of the following 4 literal strings: • Master Action Site • Operator Site • Custom Site • Fixlet Site. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:6.0, WM:7.2
url of <site>	*Plain*	<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, WM:7.2
version of <site>	*Plain*	<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, WM:7.2

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.

Copyright © 2009 BigFix, Inc.

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>	*Numbered*	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, Mac:6.0, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
id of <site group>	*Plain*	<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, Mac:6.0, WM:7.2
member of <site group>	*Plain*	<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, Mac:6.0, WM:7.2

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>	*Plain*	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, Mac:6.0, WM:7.2
relevant fixlet of <site>	*Plain*	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, Mac:6.0, WM:7.2

Copyright © 2009 BigFix, Inc.

Properties

Key Phrase	*Form*	Return Type	Description
header <string> of <fixlet>	*Named*	<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, Mac:6.0, WM:7.2
header of <fixlet>	*Plain*	<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, Mac:6.0, WM:7.2
id of <fixlet>	*Plain*	<integer>	Returns the numeric ID number of the specified Fixlet message. Win:5.0, Lin:5.0, Sol:5.0, HPUX:5.0, AIX:5.0, Mac:6.0, WM:7.2
relevance of <fixlet>	*Plain*	<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, Mac:6.0, WM:7.2

Fixlet_header

Fixlet headers are name:value pairs that 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
header <string> of <fixlet>	*Named*	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, Mac:6.0, WM:7.2
header of <fixlet>	*Plain*	Iterates over all the headers of the Fixlet message. Win:5.0, Lin:5.0, Sol:5.0, HPUX:5.0, AIX:5.0, Mac:6.0, WM:7.2

Copyright © 2009 BigFix, Inc.

Properties

Key Phrase	*Form*	Return Type	Description
name of <fixlet_header>	*Plain*	<string>	Headers are name:value pairs, separated by a colon. This Inspector returns the name on the left hand side of the pair. Win:5.0, Lin:5.0, Sol:5.0, HPUX:5.0, AIX:5.0, Mac:6.0, WM:7.2
value of <fixlet_header>	*Plain*	<string>	Headers are name:value pairs, separated by a colon. This Inspector returns the value on the right hand side of the pair. Win:5.0, Lin:5.0, Sol:5.0, HPUX:5.0, AIX:5.0, Mac:6.0, WM:7.2

Examples

- `number of relevant fixlets whose (value of header "x-fixlet-source-severity" of it as lowercase = "critical") of site "enterprise security".`
  - Returns the number of critical fixlets in the Enterprise Security site.

Copyright © 2009 BigFix, Inc.

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	*PlainGlobal*	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, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
administrator <string> of <client>	*Named*	<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, Mac:6.0, WM:7.2
administrator of <client>	*Plain*	<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, Mac:6.0, WM:7.2
setting <string> of <client>	*Named*	<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, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
setting of <client>	*Plain*	<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, WM:7.2

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>	*Named*	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, Mac:6.0, WM:7.2
administrator of <client>	*Plain*	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, Mac:6.0, WM:7.2
setting <string> of <client>	*Named*	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, WM:7.2
setting <string> of <site>	*Named*	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, WM:7.2
setting of <client>	*Plain*	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, WM:7.2
setting of <site>	*Plain*	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, WM:7.2

Copyright © 2009 BigFix, Inc.

Properties

Key Phrase	*Form*	Return Type	Description
<setting> as string	*Cast*	<string>	Returns a string formatted as <name>=<value> for the setting. Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
effective date of <setting>	*Plain*	<time>	Returns the date when the setting was last modified. Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
enabled of <setting>	*Plain*	<boolean>	Returns TRUE if the specified setting is enabled. Win:7.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:6.0, Mac:4.1, WM:7.2
name of <setting>	*Plain*	<string>	Returns the name of the setting. Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
value of <setting>	*Plain*	<string>	Returns the value of the setting. Win:2.0, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2

Examples

- `names of settings of site "actionsite"`

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

Selected Server

These Inspectors return information about the BES Server or BES Relay to which the BigFix agent or client reports.

Creation Methods

Key Phrase	*Form*	Description
selected server	*PlainGlobal*	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, WM:7.2

Copyright © 2009 BigFix, Inc.

Properties

Key Phrase	*Form*	Return Type	Description
competition size of <selected server>	*Plain*	<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, WM:7.2
competition weight of <selected server>	*Plain*	<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, WM:7.2
distance of <selected server>	*Plain*	<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, WM:7.2
gateway address <integer> of <selected server>	*Numbered*	<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, WM:7.2
gateway address of <selected server>	*Plain*	<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, WM:7.2
ip address of <selected server>	*Plain*	<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, WM:7.2
name of <selected server>	*Plain*	<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, WM:7.2
port number of <selected server>	*Plain*	<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, WM:7.2
priority of <selected server>	*Plain*	<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, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
weight of <selected server>	*Plain*	<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, WM:7.2

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
operating system product type <integer>	*NumberedGlobal*	Returns an object corresponding to the numbered OS product type. Win:6.0, WM:7.2
product type of <operating system>	*Plain*	Returns the product type of the operating system, which includes Workstations, Domain Controllers and Servers. Win:6.0, WM:7.2

Operators

Key phrase	Return Type	Description
<operating system product type> = <operating system product type>	<*boolean*>	Compare two operating system product types for equality. Win:6.0, WM:7.2

Copyright © 2009 BigFix, Inc.

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>	*Plain*	Returns the bit-mapped suite mask for the operating system, which contains further fine-grain information about the version. Win:6.0, WM:7.2

Current Relay

These Inspectors refer to the BES Server or Relay that the client last registered with.

Creation Methods

Key Phrase	*Form*	Description
current relay	*PlainGlobal*	Returns an object corresponding to the server or relay that the client last registered with. This may be a BES Relay or the BES root server. Win:7.0, Lin:7.0, Sol:7.0, HPUX:7.0, AIX:7.0, Mac:7.1, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
version of <current relay>	*Plain*	<version>	Returns a version object that is the version of the server that the client last registered with. This may be a BES Relay or the BES root server. Win:7.0, Lin:7.0, Sol:7.0, HPUX:7.0, AIX:7.0, Mac:7.1, WM:7.2

Copyright © 2009 BigFix, Inc.

Root Server

These Inspectors refer to the root server that the Bes Client is currently connected to.

Creation Methods

Key Phrase	*Form*	Description
root server	*PlainGlobal*	Returns an object representing the root BES Server to which the client last registered. Win:7.0, Lin:7.0, Sol:7.0, HPUX:7.0, AIX:7.0, Mac:7.1, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
host name of <root server>	*Plain*	<string>	The host (DNS) name of the BES root server that the BES Client last registered with. Win:7.0, Lin:7.0, Sol:7.0, HPUX:7.0, AIX:7.0, Mac:7.1, WM:7.2
id of <root server>	*Plain*	<integer>	The DSA Server ID of the BES root server that the BES Client last registered with. Win:7.0, Lin:7.0, Sol:7.0, HPUX:7.0, AIX:7.0, Mac:7.1, WM:7.2

Copyright © 2009 BigFix, Inc.

Windows Mobile Device Objects

These Inspectors retrieve information about Windows Mobile devices, such as smart phones which are being used as BES Clients.

Phone

These Inspectors return various pieces of information about the Windows mobile phone, including serial numbers, the owner's name, email address and more.

Creation Methods

Key Phrase	*Form*	Description
phone	*PlainGlobal*	Creates an object corresponding to the Client phone. This can be used to query other custome aspects of the device. WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
identifier of <phone>	*Plain*	<string>	Returns a string corresponding to the identifier of the specified phone. WM:7.2
manufacturer of <phone>	*Plain*	<string>	Returns a string corresponding to the manufacturer of the specified phone. WM:7.2
model of <phone>	*Plain*	<string>	Returns a string corresponding to the model of the specified phone. WM:7.2
operator name of <phone>	*Plain*	<string>	Returns a string corresponding to the operator name of the specified phone. WM:7.2
owner address of <phone>	*Plain*	<string>	Returns a string corresponding to the address of the owner of the specified phone. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	***Form***	**Return Type**	**Description**
owner company of <phone>	*Plain*	<string>	Returns a string corresponding to the company name of the owner of the specified phone. WM:7.2
owner email of <phone>	*Plain*	<string>	Returns a string corresponding to the email address of the owner of the specified phone. WM:7.2
owner name of <phone>	*Plain*	<string>	Returns a string corresponding to the name of the owner of the specified phone. WM:7.2
owner notes of <phone>	*Plain*	<string>	Returns a string containing the owner notes of the specified phone. WM:7.2
phone number of <phone>	*Plain*	<string>	Returns a string containing the phone number of the specified phone. WM:7.2
rated speed of <phone>	*Plain*	<hertz>	Returns a string corresponding to the rated speed of the specified phone. WM:7.2
revision of <phone>	*Plain*	<string>	Returns a string identifying the revision of the specified phone. WM:7.2
roaming status of <phone>	*Plain*	<string>	Returns a string identifying the roaming status of the specified phone. WM:7.2
serial number of <phone>	*Plain*	<string>	Returns a string corresponding to the serial number of the specified phone. WM:7.2
signal strength of <phone>	*Plain*	<integer>	Returns a string corresponding to the signal strength of the specified phone as a percentage. WM:7.2
subscriber number of <phone>	*Plain*	<string>	Returns a string corresponding to the subscriber number of the specified phone. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
type of <phone>	*Plain*	<string>	Returns a string identifying the type of the specified phone. WM:7.2

Examples

- `identifier of phone`
  - Returns a string identifying the brand of the phone, such as: 'Samsung Blackjack'.

- `manufacturer of phone`
  - Returns a string containg the name of the phone's manufacturer, such as 'SAMSUNG Electronics'.

- `model of phone`
  - Returns a string containing the model name of the phone, such as 'SAMSUNG MITs'.

- `operator name of phone`
  - Returns a string containing the name of the Windows Mobile device service provider, such as 'ATT'.

- `owner address of phone`
  - Returns a string containing the street address of the phone's owner, such as: '12345 Some Street, Denver, Colorado'.

- `owner company of phone`
  - Returns a string containing the name of the company that owns the phone, such as 'ACME Inc'.

- `owner email of phone`
  - Returns a string containing the email address of the phone's owner, such as: 'john.smith@mail.com'.

- `owner name of phone`
  - Returns a string containing the name of the phone's owner, such as: 'John Smith'.

- `owner notes of phone`
  - Returns a string containing notes written by the phone's owner, such as: 'These are my important notes'.

- `phone number of phone`
  - Returns a string containing the phone number of the Windows Mobile device.

Copyright © 2009 BigFix, Inc.

- `rated speed of phone`
  - Returns a hertz object indicating the phone's rated speed, such as '419430400 hertz'.

- `revision of phone`
  - Returns a string containing revision of the phone, such as 'i607UCGB4'.

- `roaming status of phone`
  - Returns a string identifying the roaming status of the phone, such as 'Unavailable'.

- `serial number of phone`
  - Returns a string containing the serial number of the phone, such as '35546001011618/1 04'.

- `signal strength of phone`
  - Returns an integer between 0 and 100, indicating the strength of the phone connection as a percentage.

- `subscriber number of phone`
  - Returns a string containing the phone number of the phone's subscriber.

- `type of phone`
  - Returns a string identifying the type of the phone, such as: 'SmartPhone'.

Oma Csp

These are Windows Mobile Inspectors for Outlook Mobile Access (OMA) Configuration Service Providers (CSPs). They allow you to Inspect various features and security settings on a Windows Mobile device. Some of these Inspectors return XML strings that can be directly used in Actions to configure or provision Windows Mobile devices.

Creation Methods

Key Phrase	*Form*	Description
oma csp	*PlainGlobal*	Creates a global object corresponding to the OMA CSP on the current Windows Mobile device. WM:7.2
oma csp <(string, string)>	*Index<(string, string)>Global*	Creates a global object corresponding the Open Mobile Alliance (OMA) Configuration Service Provider (CSP) parameter for the two specified oma csps parameter strings. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Description
oma csp <(string, string, string)>	*Index<(string, string, string)>Global*	Returns the value of the Open Mobile Alliance (OMA) Configuration Service Provider (CSP) parameter for the three specified oma csps parameter strings. WM:7.2
oma csp <(string, string, string, string)>	*Index<(string, string, string, string)>Global*	Returns the value of the Open Mobile Alliance (OMA) Configuration Service Provider (CSP) parameter for the four specified oma csps parameter strings. WM:7.2
oma csp <string>	*NamedGlobal*	Returns the value of the Open Mobile Alliance (OMA) Configuration Service Provider (CSP) parameter for the specified oma csps parameter string. This form of the oma csp Inspector takes a string that can contain any number of comma-separated parameters. WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
autorun policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current autorun policy from the SecurityPolicy Configuration Service Provider. • 0 indicates that applications are allowed to run automatically from the Multimedia Card when inserted. • 1 indicates that applications are restricted from autorunning. WM:7.2
block incoming calls of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'block incoming calls' status from the SecurityPolicy Configuration Service Provider. WM:7.2
block outgoing calls of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'block outgoing calls' status from the SecurityPolicy Configuration Service Provider. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	***Form***	**Return Type**	**Description**
bluetooth mode of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current bluetooth mode from the SecurityPolicy Configuration Service Provider. WM:7.2
bluetooth policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current bluetooth policy from the SecurityPolicy Configuration Service Provider. This setting indicates whether a Bluetooth-enabled device will allow other devices to perform a search on the device. Possible values are: • 0 blocks other devices from searching. • 1 allows other devices to search. WM:7.2
boolean <string> of <oma csp>	*Named*	<boolean>	Returns the result of the specified OMA CSP query as a boolean value. WM:7.2
call waiting enabled of <oma csp>	*Plain*	<boolean>	Returns the current 'call waiting enabled' status (TRUE or FALSE) from the SecurityPolicy Configuration Service Provider. WM:7.2
construct xml <string> of <oma csp>	*Named*	<string>	Returns an XML snippet to query an OMA CSP based on the parameters passed in <string>. WM:7.2
desktop quick connect authentication policy of <oma csp>	*Plain*	<integer>	Returns the current 'desktop quick connect authentication' policy from the SecurityPolicy Configuration Service Provider. This setting indicates how device authentication will be handled when connecting to the desktop. Possible values are: • 0 User must authenticate the device upon connection, if the device lock is active. • 1 If user chooses quick connect, the desktop will uniquely identify the device and allow it to connect without requiring the user to manually unlock it. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
drm security policy of <oma csp>	*Plain*	<integer>	Returns a bit-map integer corresponding to the current Digital Rights Management (DRM) security policy from the SecurityPolicy Configuration Service Provider. The given role bit-map indicates which DRM rights messages will be accepted by the DRM engine. WM:7.2
encrypt removable storage policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'encrypt removable storage' policy from the SecurityPolicy Configuration Service Provider. This setting indicates if the user is allowed to change mobile encryption settings for the removable storage media. Possible values are: • 0 indicates that the user is not allowed to change the encryption settings. • 1 indicates that the user can change the encryption settings. This is the default. WM:7.2
fixed dialing enabled of <oma csp>	*Plain*	<boolean>	Returns the current 'fixed dialing enabled' setting (TRUE or FALSE) from the SecurityPolicy Configuration Service Provider. WM:7.2
forward all calls enabled of <oma csp>	*Plain*	<boolean>	Returns the current 'forward all calls enabled' setting (TRUE or FALSE) from the SecurityPolicy Configuration Service Provider. WM:7.2
forward all calls of <oma csp>	*Plain*	<string>	Returns a string corresponding to the current 'forward all calls' setting from the SecurityPolicy Configuration Service Provider. WM:7.2
forward all calls timeout of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'forward all calls timeout' setting from the SecurityPolicy Configuration Service Provider. WM:7.2
forward all calls to of <oma csp>	*Plain*	<string>	Returns a string corresponding to the current 'forward all calls to' string from the SecurityPolicy Configuration Service Provider. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	***Form***	**Return Type**	**Description**
forward calls enabled when busy of <oma csp>	*Plain*	<boolean>	Returns the current 'forward calls enabled when busy' setting (TRUE or FALSE) from the SecurityPolicy Configuration Service Provider. WM:7.2
forward calls enabled when no answer of <oma csp>	*Plain*	<boolean>	Returns the current 'forward calls enabled when no answer' setting (TRUE or FALSE) from the SecurityPolicy Configuration Service Provider. WM:7.2
forward calls enabled when unavailable of <oma csp>	*Plain*	<boolean>	Returns the current 'forward calls enabled when unavailable' setting (TRUE or FALSE) from the SecurityPolicy Configuration Service Provider. WM:7.2
forward calls timeout when busy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'forward calls timeout when busy' setting from the SecurityPolicy Configuration Service Provider. WM:7.2
forward calls timeout when no answer of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'forward calls timeout when no answer' setting from the SecurityPolicy Configuration Service Provider. WM:7.2
forward calls timeout when unavailable of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'forward calls timeout when unavailable' setting from the SecurityPolicy Configuration Service Provider. WM:7.2
forward calls to when busy of <oma csp>	*Plain*	<string>	Returns a string corresponding to the current 'forward calls to when busy' setting from the SecurityPolicy Configuration Service Provider. WM:7.2
forward calls to when no answer of <oma csp>	*Plain*	<string>	Returns a string corresponding to the current 'forward calls to when no answer' setting from the SecurityPolicy Configuration Service Provider. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
forward calls to when unavailable of <oma csp>	*Plain*	<string>	Returns a string corresponding to the current 'forward calls to when unavailable' setting from the SecurityPolicy Configuration Service Provider. WM:7.2
forward calls when busy of <oma csp>	*Plain*	<string>	Returns a string corresponding to the current 'forward calls when busy' setting from the SecurityPolicy Configuration Service Provider. WM:7.2
forward calls when no answer of <oma csp>	*Plain*	<string>	Returns a string corresponding to the current 'forward calls when no answer' setting from the SecurityPolicy Configuration Service Provider. WM:7.2
forward calls when unavailable of <oma csp>	*Plain*	<string>	Returns a string corresponding to the current 'forward calls when unavailable' setting from the SecurityPolicy Configuration Service Provider. WM:7.2
grant manager policy of <oma csp>	*Plain*	<integer>	Returns an integer bit-mask corresponding to the current 'grant manager' policy from the SecurityPolicy Configuration Service Provider. This setting grants the system administrative privileges held by the role manager to other security roles, without modifying metabase role assignments. The bit-mask describes which roles are granted system administrative privileges. WM:7.2
grant user authenticated policy of <oma csp>	*Plain*	<integer>	Returns an integer bit-mask corresponding to the current 'grant user authenticated' policy from the SecurityPolicy Configuration Service Provider. This setting grants privileges held by the User Authenticated role to other security roles without modifying metabase role assignments. The bit-mask describes which roles are granted system administrative privileges. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
html message policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'html message' policy from the SecurityPolicy Configuration Service Provider. This setting specifies whether message transports will allow HTML messages. • 0 indicates that HTML messages are not allowed. • 1 indicates that HTML messages are allowed. WM:7.2
integer <string> of <oma csp>	*Named*	<integer>	Returns the result of the specified OMA CSP query as an integer value. WM:7.2
message authentication retry number policy of <oma csp>	*Plain*	<integer>	Returns a one-byte integer corresponding to the current 'message authentication retry number' policy from the SecurityPolicy Configuration Service Provider. This indicates the maximum number of times the user is allowed to try authenticating a Wireless Application Protocol (WAP) PIN-signed message. The default value is 3 for WM. Possible values are 1 through 256. WM:7.2
message encryption negotiation policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'message encryption negotiation' policy from the SecurityPolicy Configuration Service Provider. This setting indicates whether the Inbox application can negotiate the encryption algorithm in the case that a recipient's certificate doesn't support the specified encryption algorithm. Possible values are: • 0 doesn't allow negotiation. • 1 allows negotiation to a strong algorithm. • 2 allows negotiation to any algorithm. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	***Form***	**Return Type**	**Description**
network pin prompt policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'network personal identification number (PIN) prompt' policy from the SecurityPolicy Configuration Service Provider. This setting is used when an over-the-air (OTA) OMA Client Provisioning message is only signed with a network PIN. This setting indicates whether or not the user will be prompted to accept the device setting changes. Possible values are: • 0 indicates that the device will prompt the user. • 1 indicates that the user is not prompted. This is the default. WM:7.2
network type of <oma csp>	*Plain*	<string>	Returns the current 'network type' policy from the SecurityPolicy Configuration Service Provider. WM:7.2
obex enabled of <oma csp>	*Plain*	<boolean>	Returns the current 'obex enabled' policy from the SecurityPolicy Configuration Service Provider. This indicates whether or not the phone can exchange binary objects, either by infrared or bluetooth. WM:7.2
oma cp network pin policy of <oma csp>	*Plain*	<integer>	Returns the current 'oma cp network personal identification number (PIN)' policy from the SecurityPolicy Configuration Service Provider. This setting indicates whether the OMA network PIN-signed message will be accepted. The message's role bit-mask and the policy's role mask are ANDed together. If the result is non-zero, then the message will be accepted. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
oma cp user network pin policy of <oma csp>	*Plain*	<integer>	Returns the current 'oma cp user network personal identification number (PIN)' policy from the SecurityPolicy Configuration Service Provider. This setting indicates whether the OMA user network PIN-signed message will be accepted. The message's role bit-mask and the policy's role mask are ANDed together. If the result is non-zero, then the message will be accepted. WM:7.2
oma cp user pin policy of <oma csp>	*Plain*	<integer>	Returns the current 'oma cp user personal identification number (PIN)' policy from the SecurityPolicy Configuration Service Provider. This setting indicates whether the OMA-user PIN or user MAC-signed message will be accepted. The message's role bit-mask and the policy's role mask are ANDed together. If the result is non-zero, then the message will be accepted. WM:7.2
ota provisioning policy of <oma csp>	*Plain*	<integer>	Returns an integer bit-mask corresponding to the current 'ota provisioning' policy from the SecurityPolicy Configuration Service Provider. This setting indicates which provisioning messages are accepted by the configuration host based on the role bit-maps assigned to the messages. This policy restricts the provisioning messages that come from the Push Router. A specified role bit-mask indicates system administrative privileges are provided to the given mask. WM:7.2
password required policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'password required' policy from the SecurityPolicy Configuration Service Provider. Possible values are: • 0 indicates that a password is required. This is the default. • A value other than 0 indicates that a password is not required. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	***Form***	**Return Type**	**Description**
privileged applications policy of <oma csp>	*Plain*	<integer>	Returns the current 'privileged applications' policy from the SecurityPolicy Configuration Service Provider. This setting indicates which security model has been implemented on the WM device. Possible values are: • 0 indicates that a two-tier security model is enabled. • 1 indicates that a one-tier security model is enabled. • Any value other than 1 is treated as 0. WM:7.2
process xml query <string> of <oma csp>	*Named*	<string>	This Inspector will take the value passed in <string> and then ask the system to process it. In order to use it, the value provided must be a valid OMA CSP XML query that is not trying to set a value (only queries are allowed). A typical use is to take the results of the 'construct xml query' Inspector and pass it in as the query string. WM:7.2
rapi policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current RAPI (Remote API) policy from the SecurityPolicy Configuration Service Provider. • 0 indicates that the ActiveSync service is shut down and RAPI calls are rejected. • 1 indicates that full access to ActiveSync is provided and RAPI calls are allowed without restrictions. • 2 indicates that access to ActiveSync is restricted to the User-Authenticated role. RAPI calls are then checked against this role mask before being granted. WM:7.2
security policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'security policy' policy from the SecurityPolicy Configuration Service Provider. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	***Form***	**Return Type**	**Description**
send caller id of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'send caller id' policy from the SecurityPolicy Configuration Service Provider. WM:7.2
service indication message policy of <oma csp>	*Plain*	<integer>	Returns an integer bit-mask corresponding to the current 'service indication message' policy from the SecurityPolicy Configuration Service Provider. An SI message is sent to WM 6 Standard to notify users of new services and service updates. This setting indicates whether SI messages are accepted in the form of a role bit-mask. WM:7.2
service loading message policy of <oma csp>	*Plain*	<integer>	Returns an integer bit-mask corresponding to the current 'service loading message' policy from the SecurityPolicy Configuration Service Provider. An SL message downloads new services to the WM device. This setting indicates whether SL messages are accepted in the form of a role bit-mask. WM:7.2
sharepoint access policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'sharepoint access' policy from the SecurityPolicy Configuration Service Provider. This setting indicates whether Outlook Mobile SharePoint or UNC access is enabled through ActiveSync protocol to fetch documents. Possible values are: • 0 doesn't allow SharePoint or UNC file access. • 1 allows Outlook Mobile to fetch documents on a corporate SharePoint site or UNC. WM:7.2
sl security policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'sl security' policy from the SecurityPolicy Configuration Service Provider. This setting indicates that the operator can override https to use http, or wsps to use wsp. Possible values are: • 0 use https or wsps. • 1 use http or wsp. This is the default value. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
smime encryption algorithm policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'smime encryption algorithm' policy from the SecurityPolicy Configuration Service Provider. This setting indicates which algorithm is used to encrypt a message. Possible values are: • 0 specifies the default algorithm. • 1 is an invalid value. • 2 specifies the triple DES algorithm. • 3 specifies the DES algorithm. • 4 specifies the RC2 128-bit algorithm. • 5 specifies the RC2 64-bit algorithm. • 6 specifies the RC2 40-bit algorithm. WM:7.2
smime encryption policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'smime encryption' policy from the SecurityPolicy Configuration Service Provider. This setting indicates whether the Inbox application will send all messages encrypted. • 0 all messages must be encrypted. • 1 encrypting messages is optional. WM:7.2
smime signing algorithm policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'smime signing algorithm' policy from the SecurityPolicy Configuration Service Provider. This setting indicates which algorithm is used to sign a message. Possible values are: • 0 specifies the default algorithm. • 1 is an invalid value. • 2 specifies the SHA algorithm. • 3 specifies the MD5 algorithm. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	***Form***	**Return Type**	**Description**
smime signing policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'smime signing' policy from the SecurityPolicy Configuration Service Provider. This setting indicates whether the Inbox application will send all messaged signed. • 0 all messages must be signed. • 1 signing messages is optional. WM:7.2
software certificates policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'software certificates' policy from the SecurityPolicy Configuration Service Provider. This setting indicates whether software certificates can be used to sign outgoing messages. Possible values are: • 0 indicates that software certificates cannot be used to sign messages. • 1 indicates that software certificates can be used to sign messages. This is the default. WM:7.2
storage card encryption of <oma csp>	*Plain*	<boolean>	Returns the current 'storage card encryption' policy from the SecurityPolicy Configuration Service Provider. WM:7.2
string <string> of <oma csp>	*Named*	<string>	Returns the result of the specified OMA CSP query as a string value. WM:7.2
timezone of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current timezone policy from the SecurityPolicy Configuration Service Provider. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
trusted provisioning server policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'trusted provisioning server' policy from the SecurityPolicy Configuration Service Provider. Possible values are: • 0 indicates that assigning TPS role assignment is disabled. • 1 indicates TPS role assignment is enabled and the TPS role can be assigned to mobile operators. This is the WM default. WM:7.2
trusted wap proxy policy of <oma csp>	*Plain*	<integer>	Returns an integer bit-map corresponding to the current 'trusted wap proxy' policy from the SecurityPolicy Configuration Service Provider. This setting indicates the level of permissions required to create, modify or delete a trusted proxy. The security roles that can have Trusted WAP Proxy level permissions are returned as a bit-mask. WM:7.2
unauthenticated message policy of <oma csp>	*Plain*	<integer>	Returns an integer bit-mask corresponding to the current 'unauthenticated message' policy from the SecurityPolicy Configuration Service Provider. This setting indicates whether to accept unsigned WAP messages processed by the default security provider in the Push Router, based on their origin. The message source must match one of the security roles specified by this policy. This setting indicates whether unauthenticated messages are accepted in the form of a role bit-mask. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	***Form***	**Return Type**	**Description**
unsigned applications policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'unsigned applications' policy from the SecurityPolicy Configuration Service Provider. The possible values are: • 0 indicates that unsigned apps are not allowed to run on the device. • 1 indicates that unsigned apps are allowed to run on the device. This is the default for WM. • Any value other than 1 is treated as 0. WM:7.2
unsigned cabs policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'unsigned CABS' policy from the SecurityPolicy Configuration Service Provider. This indicates whether unsigned .cab files can be installed on the device. Possible values are: • 0 is equivalent to having none of the role mask bits set and indicates that no unsigned .cab files can be installed. • A specified role bit-mask indicates accepted unsigned .cab files are installed with the given role mask. WM:7.2
unsigned prompt policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'unsigned prompt' policy from the SecurityPolicy Configuration Service Provider. This setting indicates whether a user must be prompted to accept or reject unsigned .exe, theme, .dll or .cab files. Possible values are: • 0 indicates that the user will be prompted. This is the WM default. • 1 indicates that the user will not be prompted. • Any value other than 1 is treated as 0. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
unsigned themes policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'unsigned themes' policy from the SecurityPolicy Configuration Service Provider. Possible values are: • 0 is equivalent to having none of the role-mask bits set, and indicates that no unsigned Theme files can be installed. • A specified role bit-mask indicates accepted unsigned Theme files are installed with the given role mask. WM:7.2
value <string> of <oma csp>	*Named*	<string>	Returns the result of the specified OMA CSP query as a string value. WM:7.2
wsp push policy of <oma csp>	*Plain*	<integer>	Returns an integer corresponding to the current 'Wireless Session Protocol (WSP) push' policy from the SecurityPolicy Configuration Service Provider. This setting indicates whether WSP notifications from the WAP stack are routed. Possible values are: • 0 indicates that routing of WSP notifications is not allowed. • 1 indicates that Routing is allowed. This is the WM default. WM:7.2

Examples

- `value "URL" of oma csp ("BrowseFavorite", "Southridge Video Store")`

  Returns a URL corresponding to the specified browser favorite site, such as: 'http://www.southridgevideo.com'.

- `value "TAPI_FORWARD_ADDRESS" of oma csp ("Tapi","Busy","Voice")`

  Returns a string containing the forwarding telephone number for the specified parameters, such as '5551212'.

- `integer "TAPI_BARRING_OUT" of oma csp "Tapi"`

  Returns an integer corresponding to the value of the chosen parameter.

Copyright © 2009 BigFix, Inc.

- `process xml query (construct xml "TAPI_FORWARD_ENABLED" of oma csp "Tapi,Busy,Voice") of oma csp`
  - Processes the given xml query and returns the resulting xml, in this case: <wap-provisioningdoc> <characteristic type="Tapi"> <characteristic type="Busy"> <characteristic type="Voice"> <parm-query name="TAPI_FORWARD_ENABLED"/> </characteristic> </characteristic> </characteristic> </wap-provisioningdoc>.

- `value "TAPI_FORWARD_ADDRESS" of oma csp "Tapi,Busy,Voice"`
  - Returns a string containing the forwarding telephone number for the specified parameters.

- `autorun policy of oma csp`
  - Returns 0 or 1, depending on whether apps are allowed to run automatically or not. This is the same as: integer "2" of oma csp "SecurityPolicy".

- `block incoming calls of oma csp`
  - Returns an integer corresponding to the current status of the given Security Policy. This is the same as: integer "TAPI_BARRING_IN" of oma csp "Tapi".

- `block outgoing calls of oma csp`
  - Returns an integer corresponding to the current status of the given Security Policy. This is the same as: integer "TAPI_BARRING_OUT" of oma csp "Tapi".

- `bluetooth mode of oma csp`
  - Returns an integer corresponding to the current status of the given Security Policy. This is the same as: integer "BtMode" of oma csp "Bluetooth".

- `bluetooth policy of oma csp`
  - Returns an integer corresponding to the current status of the given Security Policy. This is the same as: integer "4135" of oma csp "SecurityPolicy".

- `boolean "TAPI_FORWARD_ENABLED" of oma csp "Tapi,Busy,Voice"`
  - Returns the boolean value of the specified OMA CSP query.

- `call waiting enabled of oma csp`
  - Returns TRUE if call waiting is enabled.

- `construct xml "TAPI_FORWARD_ENABLED" of oma csp "Tapi,Busy,Voice"`
  - Returns a snippet of XML like the following: <wap-provisioningdoc> <characteristic type="Tapi"> <characteristic type="Busy"> <characteristic type="Voice"> <parm-query name="TAPI_FORWARD_ENABLED"/> </characteristic> </characteristic> </characteristic> </wap-provisioningdoc>.

- `construct xml "TAPI_FORWARD_ADDRESS" of oma csp "Tapi,Busy,Voice"`
  - Returns a snippet of XML like the following: <wap-provisioningdoc> <characteristic type="Tapi"> <characteristic type="Busy"> <characteristic type="Voice"> <parm-query

Copyright © 2009 BigFix, Inc.

name="TAPI_FORWARD_ADDRESS" /> </characteristic> </characteristic> </characteristic> </wap-provisioningdoc>.

- `desktop quick connect authentication policy of oma csp`

  Returns an integer corresponding to the current status of the given Security Policy. This is the same as: integer "4146" of oma csp "SecurityPolicy".

- `drm security policy of oma csp`

  Returns an integer corresponding to the current status of the given Security Policy. This is the same as: integer "4129" of oma csp "SecurityPolicy".

- `encrypt removable storage policy of oma csp`

  Returns an integer corresponding to the current status of the given Security Policy. This is the same as: integer "4134" of oma csp "SecurityPolicy".

- `fixed dialing enabled of oma csp`

  Returns TRUE if fixed dialing is enabled. This is the same as: boolean "TAPI_FIXEDDIAL_ENABLED" of oma csp "Tapi".

- `forward all calls enabled of oma csp`

  Returns TRUE if call forwarding is enabled. This is the same as: boolean "TAPI_FORWARD_ENABLED" of oma csp ("Tapi","Unconditional","Voice").

- `forward all calls of oma csp`

  Returns a string of the form: 'Forward calls to 5551212 when Unconditional after 20 seconds'.

- `forward all calls timeout of oma csp`

  Returns an integer such as 20. This is the same as: integer "TAPI_FORWARD_TIMEOUT" of oma csp ("Tapi","Unconditional","Voice").

- `forward all calls to of oma csp`

  Returns a telephone number as a string. This is the same as: string "TAPI_FORWARD_ADDRESS" of oma csp ("Tapi","Unconditional","Voice").

- `forward calls enabled when busy of oma csp`

  Returns TRUE if the call forwarding is enabled when busy.

- `forward calls enabled when no answer of <oma csp`

  Returns TRUE if the call forwarding is enabled when there is no answer.

- `forward calls enabled when unavailable of oma csp`

  Returns TRUE if the call forwarding is enabled when the user is unreachable.

Copyright © 2009 BigFix, Inc.

- `forward calls timeout when busy of oma csp`
  - Returns an integer corresponding to the specified timeout.

- `forward calls timeout when no answer of oma csp`
  - Returns an integer corresponding to the specified timeout.

- `forward calls timeout when unavailable of oma csp`
  - Returns an integer corresponding to the specified timeout.

- `forward calls to when busy of oma csp`
  - Returns a string corresponding to the forwarding phone number when busy.

- `forward calls to when no answer of oma csp`
  - Returns a string corresponding to the forwarding phone number when there is no answer.

- `forward calls to when unavailable of oma csp`
  - Returns a string corresponding to the forwarding phone number when the user is unreachable.

- `forward calls when busy of oma csp`
  - Returns a string of the form: 'Forward calls to 5551212 when Busy after 5 seconds'.

- `forward calls when no answer of oma csp`
  - Returns a string of the form: 'Forward calls to 5551212 when No-Reply after 25 seconds'.

- `forward calls when unavailable of oma csp`
  - Returns a string of the form: 'Forward calls to 5551212 when Not-Reachable after 5 seconds'.

- `grant manager policy of oma csp`
  - Returns an integer corresponding to the current status of the given Security Policy. This is the same as: integer "4119" of oma csp "SecurityPolicy".

- `grant user authenticated policy of oma csp`
  - Returns an integer corresponding to the current status of the given Security Policy. This is the same as: integer "4120" of oma csp "SecurityPolicy".

- `html message policy of oma csp`
  - Returns an integer corresponding to the current status of the given Security Policy. This is the same as: integer "4136" of oma csp "SecurityPolicy".

- `integer "TAPI_BARRING_OUT" of oma csp "Tapi"`
  - Returns an integer corresponding to the current status of the specified Security Policy string constant.

Copyright © 2009 BigFix, Inc.

- `message authentication retry number policy of oma csp`

  Returns an integer corresponding to the current status of the given Security Policy. This is the same as: integer "4105" of oma csp "SecurityPolicy".

- `message encryption negotiation policy of oma csp`

  Returns an integer corresponding to the current status of the given Security Policy. This is the same as: integer "4144" of oma csp "SecurityPolicy".

- `network pin prompt policy of oma csp`

  Returns an integer corresponding to the current status of the given Security Policy. This is the same as: integer "4132" of oma csp "SecurityPolicy".

- `network type of oma csp`

  Returns a network type as a string, such as 'ATT'. This is the same as: string "TAPI_FORWARD_ADDRESS" of oma csp ("Tapi","Unconditional","Voice").

- `obex enabled of oma csp`

  Returns TRUE if object exchange protocol is enabled. This is the same as: boolean "IsEnabled" of oma csp ("Obex","HKLM\Software\Microsoft\Obex").

- `oma cp network pin policy of oma csp`

  Returns an integer bit-mask corresponding to the current status of the given Security Policy, such as '3200'. This is the same as: integer "4141" of oma csp "SecurityPolicy".

- `oma cp user network pin policy of oma csp`

  Returns an integer bit-mask corresponding to the current status of the given Security Policy, such as '3200'. This is the same as: integer "4143" of oma csp "SecurityPolicy".

- `oma cp user pin policy of oma csp`

  Returns an integer bit-mask corresponding to the current status of the given Security Policy, such as '3200'. This is the same as: integer "4142" of oma csp "SecurityPolicy".

- `ota provisioning policy of oma csp`

  Returns an integer bit-mask corresponding to the current status of the given Security Policy, such as '3728'. This is the same as: integer "4111" of oma csp "SecurityPolicy".

- `password required policy of oma csp`

  Returns a 0 if a password is required. This is the same as: integer "4131" of oma csp "SecurityPolicy".

- `privileged applications policy of oma csp`

  Returns a 1 if a one-tier security model is enabled, otherwise, a two-tier model is used. This is the same as: integer "4123" of oma csp "SecurityPolicy".

Copyright © 2009 BigFix, Inc.

- `process xml query (construct xml "TAPI_FORWARD_ADDRESS" of oma csp "Tapi,Busy,Voice") of oma csp`

  Returns a string containing an XML snippet such as: '<wap-provisioningdoc> <characteristic type="Tapi"> <characteristic type="Busy"> <characteristic type="Voice"> <parm-query name="TAPI_FORWARD_ADDRESS" value="5551212" /> </characteristic> </characteristic> </characteristic> </wap-provisioningdoc>'.

- `rapi policy of oma csp`

  Returns an integer (0-2) corresponding to the current status of the given Security Policy, such as '3200'. This is the same as: integer "4097" of oma csp "SecurityPolicy".

- `security policy of oma csp`

  Returns an integer corresponding to the current status of the given Security Policy. This is the same as: integer "4124" of oma csp "SecurityPolicy".

- `send caller id of oma csp`

  Returns the 'send caller id' policy as an integer. This is the same as: integer "TAPI_SEND_CALLID" of oma csp "Tapi".

- `service indication message policy of oma csp`

  Returns an integer bit-mask corresponding to the current status of the given Security Policy. This is the same as: integer "4109" of oma csp "SecurityPolicy".

- `service loading message policy of oma csp`

  Returns an integer bit-mask corresponding to the current status of the given Security Policy. This is the same as: integer "4108" of oma csp "SecurityPolicy".

- `sharepoint access policy of oma csp`

  Returns an integer (0 or 1) corresponding to the current status of the given Security Policy. This is the same as: integer "4145" of oma csp "SecurityPolicy".3073.

- `sl security policy of oma csp`

  Returns an integer (0 or 1) corresponding to the current status of the given Security Policy. This is the same as: integer "4124" of oma csp "SecurityPolicy".

- `smime encryption algorithm policy of oma csp`

  Returns an integer (0-6) corresponding to the current status of the given Security Policy. This is the same as: integer "4140" of oma csp "SecurityPolicy".

- `smime encryption policy of oma csp`

  Returns an integer (0 or 1) corresponding to the current status of the given Security Policy. This is the same as: integer "4138" of oma csp "SecurityPolicy".

Copyright © 2009 BigFix, Inc.

- `smime signing algorithm policy of oma csp`

  Returns an integer (0-3) corresponding to the current status of the given Security Policy. This is the same as: integer "4139" of oma csp "SecurityPolicy".

- `smime signing policy of oma csp`

  Returns an integer (0 or 1) corresponding to the current status of the given Security Policy. This is the same as: integer "4137" of oma csp "SecurityPolicy".

- `software certificates policy of oma csp`

  Returns an integer (0 or 1) corresponding to the current status of the given Security Policy. This is the same as: integer "4127" of oma csp "SecurityPolicy".

- `storage card encryption of oma csp`

  Returns a boolean TRUE if the storage card is encrypted.

- `string "TAPI_FORWARD_ADDRESS" of oma csp ("Tapi","Unconditional","Voice")`

  Returns a string containing the forwarding phone number currently set up for the specified parameters.

- `timezone of oma csp`

  Returns an integer corresponding to the time zone set for the phone. This is equivalent to: integer "TimeZone" of oma csp "Clock".

- `trusted provisioning server policy of oma csp`

  Returns an integer (0 or 1) corresponding to the current status of the given Security Policy. This is the same as: integer "4104" of oma csp "SecurityPolicy".

- `trusted wap proxy policy of oma csp`

  Returns an integer bit-map descripting the current trusted wap proxy policy.

- `unauthenticated message policy of oma csp`

  Returns an integer, such as '64', corresponding to the current unauthenticated message policy.

- `unsigned applications policy of oma csp`

  Returns 1 if unsigned apps are allowed to run.

- `unsigned cabs policy of oma csp`

  Returns an integer bit-mask defining the roles for accepting unsigned cab files.

- `unsigned prompt policy of oma csp`

  Returns a 1 if the user is to be prompted before accepting certain unsigned files.

Copyright © 2009 BigFix, Inc.

- `unsigned themes policy of oma csp`
- Returns an integer bit-mask defining the roles for accepting unsigned theme files.

- `value "TAPI_FORWARD_ADDRESS" of oma csp ("Tapi","Unconditional","Voice")`
- This phrase returns the forwarding number, such as '5551212', for the given parameters.

- `wsp push policy of oma csp`
- Returns 1 if routing of WSP notifications is allowed.

Wince Network Connection Detail

These Inspectors return detailed information about the Windows Embedded CE network connections on the Windows Mobile device. For more information about these Inspectors, refer to the MSDN article titled CONNMGR_CONNECTION_DETAILED_STATUS. These Inspectors require that the Mobile device be running WinCE .NET 4.2 or later.

Creation Methods

Key Phrase	*Form*	Description
network connection	*PlainGlobal*	Creates a global object corresponding to the WinCE network connection. WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
adapter name of <wince network connection detail>	*Plain*	<string>	Returns a string corresponding to the null-terminated name of the adapter for the given WinCE network connection. If no adapter name is available, the Inspector returns NULL. WM:7.2
description of <wince network connection detail>	*Plain*	<string>	Returns a string corresponding to the null-terminated description of the given WinCE network connection. If no adapter name is available, the Inspector returns NULL. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	***Form***	**Return Type**	**Description**
destination network of <wince network connection detail>	*Plain*	<string>	Returns a string containing the GUID of the destination network for the specified WinCE connection. WM:7.2
flags of <wince network connection detail>	*Plain*	<string>	Returns a string containing one or more connection options for the specified WinCE network connection. These flags include: billed by time, always on, or suspend and resume. The constants for these flags are explained in greater detail in the MSDN article on the Connection Manager Connection Options Constants. WM:7.2
ip addresses of <wince network connection detail>	*Plain*	<string>	Returns a string containing the available IP addresses for the specified WinCE network connection. If no addresses are available, this Inspector returns NULL. WM:7.2
last connected of <wince network connection detail>	*Plain*	<string>	Returns a string containing the last time that the connection was established for the specified WinCE network connection. WM:7.2
secure of <wince network connection detail>	*Plain*	<boolean>	Returns a boolean describing the security level of the current connection for the specified WinCE network. If TRUE, the connection is secure. WM:7.2
signal quality of <wince network connection detail>	*Plain*	<integer>	Returns the signal quality of the specified WinCE network connection. This is an integer between 0 and 255, with 255 indicating the best signal quality. WM:7.2
source network of <wince network connection detail>	*Plain*	<string>	Returns a string containing the GUID of the source network for the specified WinCE connection. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
status of <wince network connection detail>	*Plain*	<string>	Returns the status of the specified WinCE network connection. This is a string indicating whether the connection is established, suspended, disconnected, waiting, failed or more. These are explained in greater detail in the MSDN article on the Connection Manager Status Constants. WM:7.2
type of <wince network connection detail>	*Plain*	<string>	Returns the type of the specified WinCE network connection. This is a string indicating a cellular, NIC, Bluetooth, Unimodem, VPN, Proxy or PC connection. These are explained in greater detail in the MSDN article on the Connection Manager Connection Type Constants. WM:7.2

Examples

- `adapter name of network connection "My Wifi"`
  - Returns a string such as "TNETW12511".
- `description of network connection "My Wifi"`
  - Returns a string such as: My Wifi.
- `destination network of network connection "My Wifi Network"`
  - Returns a string such as: IID_DestNetInternet.
- `flags of network connection "My Wifi"`
  - Returns a string such as 'Always On'.
- `ip addresses of network connection "My Wifi"`
  - Returns a string containing the available IP address(es) for the given network, such as 'localhost (192.168.1.104)'.
- `last connected of network connection "My Wifi"`
  - Returns a string containing the last connetion time, such as '7/13/2009 12:35:00 AM'.
- `secure of network connection "My Wifi"`
  - Returns TRUE if the WinCE network connection is secure.

Copyright © 2009 BigFix, Inc.

- `signal quality of network connection "My Wifi"`

Returns an integer between 0 and 255, indicating the strength of the specified network connection.

- `source network of network connection "My Wifi"`

Returns a string containing the GUID of the source network for the specified connection, such as: 'Unknown'.

- `status of network connection "My Wifi"`

Returns a string containing the current network connection status, such as: 'Connected'.

- `type of network connection "My Wifi"`

Returns a string, such as 'Wifi', indicating the network type.

Wince_web_browser

These Inspectors return information about the Windows Embedded CE browser on the specified device. Typically, this is a version of Internet Explorer that has been optimized for operation on the small display of a Windows Mobile device.

Creation Methods

Key Phrase	*Form*	Description
default web browser	*PlainGlobal*	Creates a global object corresponding to the current default web browser on the Windows CE device. Windows Embedded CE uses IE, which has been optimized for WinCE devices. WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
version of <wince_web_browser>	*Plain*	<version>	Returns the version of the current web browser on the Windows CE device. WM:7.2

Copyright © 2009 BigFix, Inc.

Examples

- `default web browser`
- Returns a value such as: "iexplore.exe" "" "" "" "".

- `version of default web browser`
- Returns a version, such as 'iexplore.exe 4.1', corresponding to the current web browser.

Base_battery

This is an abstract type from which <battery> and <backup battery> are derived.

Properties

Key Phrase	*Form*	Return Type	Description
full life of <base_battery>	*Plain*	<time interval>	For the specified Windows Mobile battery, this Inspector returns a time interval corresponding to the number of seconds of battery life when at full charge. Base battery is an abstract type that can refer to either the main "battery" or the "backup battery". WM:7.2
life of <base_battery>	*Plain*	<time interval>	For the specified Windows Mobile battery, this Inspector returns a time interval corresponding to the number of seconds of battery life remaining. Base battery is an abstract type that can refer to either the main "battery" or the "backup battery". WM:7.2
life percent of <base_battery>	*Plain*	<integer>	For the specified Windows Mobile battery, this Inspector returns an integer corresponding to the percentage of full battery charge remaining. This is a value in the range 0 to 100. Base battery is an abstract type that can refer to either the main "battery" or the "backup battery". WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
millivolts of <base_battery>	*Plain*	<integer>	For the specified Windows Mobile battery, this Inspector returns an integer corresponding to the amount of battery voltage in millivolts (mV). This is a value in the range of 0 to 65,535. Base battery is an abstract type that can refer to either the main "battery" or the "backup battery". WM:7.2
status of <base_battery>	*Plain*	<string>	Returns a string corresponding to the current status of the battery. This is one of the following: Charging, High, Low, Critical, No battery or Unknown. Base battery is an abstract type that can refer to either the main "battery" or the "backup battery". WM:7.2

Examples

- `full life of battery`
  - Returns a time interval for the battery life, such as 3:45:00.

- `life of backup battery`
  - Returns a time interval denoting the remaining backup battery life, such as '0:04:03'. This is the same return type used for the main battery life.

- `life percent of battery`
  - Returns a number from 0 to 100 indicating the percentage of life left in the battery.

- `millivolts of backup battery`
  - Returns an integer corresponding to the backup battery voltage (in mV).

- `status of battery`
  - Returns a string indicating the current battery status, such as: 'High'. A similar string applies to the backup battery as well.

Copyright © 2009 BigFix, Inc.

Battery

These Inspectors return information about the battery in the Windows Mobile device, including items such as the type, charge and lifetime.

Creation Methods

Key Phrase	*Form*	Description
battery	*PlainGlobal*	Creates an inspectable object corresponding to the main battery of the Windows Mobile device. WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
ac of <battery>	*Plain*	<string>	Returns a string detailing the AC power status of the specified Windows Mobile device battery. This can include offline, online or backup. For more information, see the MSDN article on SYSTEM_POWER_STATUS_EX. WM:7.2
average interval of <battery>	*Plain*	<integer>	Returns an integer corresponding to the time constant in milliseconds (ms) used for integrating the average battery current in milliamps. WM:7.2
average milliamps of <battery>	*Plain*	<integer>	Returns an integer corresponding to the short-term average current drain of the Windows Mobile device (in milliamps). This number is in the range of 0 to 32,767 when charging and 0 to –32,768 when discharging. WM:7.2
chemistry of <battery>	*Plain*	<string>	This Inspector returns a string describing the type of chemistry used by the specified Windows Mobile battery. It can include alkaline, nicad, lithium and more. For details, see the MSDN article on SYSTEM_POWER_STATUS_EX2. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
milliamps of <battery>	*Plain*	<integer>	Returns an integer corresponding to the instantaneous current drain of the Windows Mobile device (in milliamps). This number is in the range of 0 to 32,767 when charging and 0 to –32,768 when discharging. WM:7.2
milliamps per hour of <battery>	*Plain*	<integer>	Returns an integer corresponding to the long-term cumulative average discharge in milliamperes per hour (mA/H). This number can have a value in the range of 0 to –32,768. This value can be reset by charging or changing the batteries. WM:7.2
temperature of <battery>	*Plain*	<floating point>	For this specified Windows Mobile device battery, this Inspector returns a floating point number corresponding to the battery temperature in degrees Celsius. It can be in the range of –3,276.8 to 3,276.7 in increments of 0.1 degrees Celsius. WM:7.2

Examples

- `battery`
  - Returns a value such as: "High" "41" "00:00:00" "00:00:00" "3708" "Offline" "Lithium-ion" "-386" "0" "0" "0" "37".

- `ac of battery`
  - Returns a string such as "Online".

- `ac of backup battery`
  - Returns a string such as "Online".

- `average interval of battery`
  - Returns a number corresponding to the time in milliseconds used to average the battery current.

- `average milliamps of battery`
  - Returns a number corresponding to the current drain of the main battery.

- `average milliamps of backup battery`
  - Returns a number corresponding to the current drain of the backup battery.

Copyright © 2009 BigFix, Inc.

- `chemistry of battery`
  - Returns a sting such as: Lithium-ion. These strings are also used to describe the backup battery.

- `milliamps of battery`
  - Returns a signed integer, such as -376, corresponding to the battery drain.

- `milliamps per hour of battery`
  - Returns a signed integer, such as -53, corresponding to the cumulative average battery drain per hour.

- `temperature of battery`
  - Returns a floating point number corresponding to the temperature of the battery.

Backup_battery

These Inspectors return information about the backup battery in the Windows Mobile device, similar to the information obtainable from the battery Inspectors.

Creation Methods

Key Phrase	*Form*	Description
backup battery	*PlainGlobal*	Creates an inspectable object corresponding to the backup battery of the Windows Mobile device. The backup battery takes over should the main battery run out of charge. WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
full life of <backup_battery>	*Plain*	<time interval>	For the specified Windows Mobile base battery, this Inspector returns a time interval corresponding to the number of seconds of backup battery life when at full charge. WM:7.2
life of <backup_battery>	*Plain*	<time interval>	For the specified Windows Mobile base battery, this Inspector returns a time interval corresponding to the number of seconds of backup battery life remaining. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
life percent of <backup_battery>	*Plain*	<integer>	For the specified Windows Mobile base battery, this Inspector returns an integer corresponding to the percentage of full backup battery charge remaining. This is a value in the range 0 to 100. WM:7.2
millivolts of <backup_battery>	*Plain*	<integer>	For the specified Windows Mobile base battery, this Inspector returns an integer corresponding to the amount of backup battery voltage in millivolts (mV). This is a value in the range of 0 to 65,535. WM:7.2
status of <backup_battery>	*Plain*	<string>	Returns a string corresponding to the current status of the backpup battery. This is one of the following: Charging, High, Low, Critical, No battery or Unknown. WM:7.2

Examples

- `backup battery`

  Returns a value such as: "High" "41" "00:00:00" "00:00:00" "3708" "Offline" "Lithium-ion" "-386" "0" "0" "0" "37".

Copyright © 2009 BigFix, Inc.

GPS

These are inspectors for Mobile Windows to interrogate the Global Positioning Service (GPS) device on the phone. The information available, as well as latitude and longitude, includes heading, altitude and more. The majority of the fields in this structure are translations from the fields defined by the National Marine Electronics Association (NMEA).

Creation Methods

Key Phrase	*Form*	Description
gps	*PlainGlobal*	Creates a global object corresponding to the GPS on the Windows Mobile device. WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
altitude of <gps>	*Plain*	<string>	Returns a string containing the altitude (in meters) of the Windows Mobile device, as determined by the onboard GPS. WM:7.2
enabled of <gps>	*Plain*	<boolean>	Returns TRUE if the Gloabal Positioning Service (GPS) on the Windows Mobile device is enabled. WM:7.2
full status of <gps>	*Plain*	<string>	Returns a string containing the full status of the Windows Mobile device, as determined by the onboard GPS. It is a concatenation of all the inspectable items, with the general form 'feature: {value} units', each separated by a space. The full string looks like 'Name: {name} Status: {ON/OFF} Last sample time: {sample time} Latitude: {latitude} degrees Longitude: {longitude} degrees Heading: {heading} degrees Speed: {speed} knots Altitude: {altitude} m.'. WM:7.2
heading of <gps>	*Plain*	<string>	Returns a string containing the heading in degrees (a heading of zero is true north) of the Windows Mobile device, as determined by the onboard GPS. WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
latitude of <gps>	*Plain*	<string>	Returns a string containing the latitude (in degrees) of the Windows Mobile device, as determined by the onboard GPS. Positive numbers indicate the northern latitudes. WM:7.2
longitude of <gps>	*Plain*	<string>	Returns a string containing the longitude (in degrees) of the Windows Mobile device, as determined by the onboard GPS. Positive numbers indicate east longitudes. WM:7.2
name of <gps>	*Plain*	<string>	Returns a string containing the human-readable name of the embedded GPS of the Windows Mobile device. It might, for example, be something like 'Acme GPS Card, version 3.4.'. WM:7.2
sample time of <gps>	*Plain*	<time>	Returns a time value containing the current sample time used by the onboard GPS of the Windows Mobile device. WM:7.2
speed of <gps>	*Plain*	<string>	Returns a string containing the speed (in knots) of the Windows Mobile device, as determined by the onboard GPS. WM:7.2

Examples

- `altitude of gps`
  - Returns a string such as "150.000".

- `enabled of gps`
  - Returns TRUE if the GPS is enabled.

- `full status of gps`
  - Returns a string of the form: 'Name: QualComm GpsOne Card, version 0.0 Status: ON Last sample time: Mon, 13 Jul 2009 12:50:05 -0800 Latitude: 32.99205 degrees Longitude: -117.05468 degrees Heading: Not available Speed: 0.0000 knots Altitude: 150.000 m'.

- `heading of gps`
  - Returns a string containing the heading of the mobile device in degrees, such as '90' for due east.

Copyright © 2009 BigFix, Inc.

- `latitude of gps`

  Returns a string containing latitude of the Win Mobile device, as determined by the GPS, such as '32.99205'.

- `longitude of gps`

  Returns a string representing the current longitude of the Windows Mobile device, as indicated by the GPS, such as '-117.05468'.

- `name of gps`

  Returns a string containing the model name of the GPS, such as 'QualComm GpsOne Card, version 0.0'.

- `sample time of gps`

  Returns a time object according to the GPS, such as: 'Mon, 13 Jul 2009 12:50:05 -0800'.

- `speed of gps`

  Returns a string indicating the speed of the device in knots, such as: '43.3420'.

Copyright © 2009 BigFix, Inc.

Authorization Objects

These inspectors retrieve security and access settings.

Client_cryptography

These Inspectors expose cryptographic properties exclusive to the client.

Creation Methods

Key Phrase	*Form*	Description
client cryptography	*PlainGlobal*	This Inspector is similar to the global cryptography object except that it returns properties exclusive to the client (whereas <cryptography> is also available in the Console/Web Reports contexts). Win:7.1, Lin:7.1, Sol:7.1, HPUX:7.1, AIX:7.1, Mac:7.1, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
desired encrypt report of <client_cryptography>	*Plain*	<boolean>	Returns TRUE if the client is configured to attempt to encrypt reports. Win:7.1, Lin:7.1, Sol:7.1, HPUX:7.1, AIX:7.1, Mac:7.1, WM:7.2
encrypt report failure message of <client_cryptography>	*Plain*	<string>	If the client is not successfully encrypting reports, this Inspector returns the failure message. Win:7.1, Lin:7.1, Sol:7.1, HPUX:7.1, AIX:7.1, Mac:7.1, WM:7.2
encrypt report of <client_cryptography>	*Plain*	<boolean>	Returns TRUE if the client is successfully encrypting reports. Win:7.1, Lin:7.1, Sol:7.1, HPUX:7.1, AIX:7.1, Mac:7.1, WM:7.2

Copyright © 2009 BigFix, Inc.

User Objects

These Inspectors return information about local and current user accounts, including names, logins, passwords and more.

Logged On User

These Windows and Macintosh Inspectors return information about the currently logged-on user. With the advent of Terminal Services and Fast User Switching, these Inspectors are designed to iterate over all logged on users.

• Windows Note: If Terminal Services are available (NT/2000/2003/XP/Vista) and enabled, these Inspectors iterate over the active and disconnected sessions as returned by WTSEnumerateSessions. Disconnected sessions are those where a user logs on, but is currently inactive. On Vista, the non-interactive session 0 (used for services isolation) is not included. If Terminal Services aren't available, the ACLs on the security descriptor of the "winsta0" window station are examined for user logons. On Windows 9x systems, these Inspectors return the user session associated with the registry value "Current User" of "SYSTEM\CurrentControlSet\Control" if it exists. Otherwise, if a shell process process such as Explorer.exe is running, they return a single session associated with an unnamed user (which occurs when the user cancels the 9x login dialog).

Creation Methods

Key Phrase	***Form***	**Description**
current user	*PlainGlobal*	Returns the active, console (local) user, if logged on. Otherwise does not exist. Win:7.0, Mac:7.1, WM:7.2
logged on user	*PlainGlobal*	Returns the user logged on to this BES Client. This Inspector iterates through all logged-on users, using Fast User Switching, Terminal Services, ACLs, and on Win 9x, the registry. Win:7.0, Mac:7.1, WM:7.2

Properties

Key Phrase	***Form***	**Return Type**	**Description**
active of <logged on user>	*Plain*	<boolean>	Returns TRUE if the specified user session is active (either as a current Fast User or an active terminal services connection). Win:7.0, Mac:7.1, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
name of <logged on user>	*Plain*	<string>	If Terminal Services is available and enabled under NT4/2000/2003/XP/Vista, this Inspector returns the result of WTSQuerySessionInformation with WTSUserName. With Terminal Services disabled, it examines the ACLs on the security descriptor of the "winsta0" window station. Under Windows 9x, returns the "Current User" of "SYSTEM\CurrentControlSet\Control" if it exists. Otherwise returns No Such Object. Win:7.0, WM:7.2
remote of <logged on user>	*Plain*	<boolean>	Returns TRUE if the user session is a remote terminal services connection. Win:7.0, Mac:7.1, WM:7.2

Copyright © 2009 BigFix, Inc.

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	*PlainGlobal*	Creates an action object corresponding to the BigFix Action currently being parsed. Win:6.0, Lin:6.0, Sol:6.0, HPUX:6.0, AIX:6.0, Mac:5.1, WM:7.2
action <integer>	*NumberedGlobal*	Creates an action object matching the <integer> id. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
active action	*PlainGlobal*	Creates an action object corresponding to the currently executing action. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
active of <action>	*Plain*	<boolean>	Returns TRUE if the action is currently running (active). Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
active start time of <action>	*Plain*	<time>	Returns the time the action started. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
complete time of <action>	*Plain*	<time>	Returns the time the action completed. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
constrained of <action>	*Plain*	<boolean>	Returns TRUE if action is unable to run yet. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
group leader of <action>	*Plain*	<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, Mac:7.1, WM:7.2
id of <action>	*Plain*	<integer>	Returns the numeric ID associated with the specified Action. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
last change time of <action>	*Plain*	<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, WM:7.2
offer accepted of <action>	*Plain*	<boolean>	Returns TRUE when users indicated they want to run the action by accepting the offer presented by the BES Client UI. When an offer has been accepted, the Client evaluates its constraints and runs as soon as conditions allow. Win:7.0, Lin:7.0, Sol:7.0, HPUX:7.0, AIX:7.0, Mac:7.1, WM:7.2
offer of <action>	*Plain*	<boolean>	Returns TRUE when the Action is presented as an offer (as indicated by the header "x-offer: 1"). Win:7.0, Lin:7.0, Sol:7.0, HPUX:7.0, AIX:7.0, Mac:7.1, WM:7.2
origin fixlet id of <action>	*Plain*	<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, WM:7.2
parameter <string> of <action>	*Named*	<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, WM:7.2
pending login of <action>	*Plain*	<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, WM:7.2
pending of <action>	*Plain*	<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, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
pending restart of <action>	*Plain*	<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, WM:7.2
pending time of <action>	*Plain*	<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, WM:7.2
status of <action>	*Plain*	<string>	Returns one of the following strings: • Running = when the action is currently active. • Executed = no longer relevant and action has completed. • Not Relevant = action was not relevant. • Waiting = action is relevant, but waiting to run. • Not Executed = action is relevant, unconstrained, but has not yet started. • Failed = action is relevant, unconstrained, has completed, but is still relevant. Win:1.2, Lin:3.1, Sol:3.1, HPUX:4.0, AIX:4.1, Mac:4.1, WM:7.2
waiting for download of <action>	*Plain*	<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, WM:7.2

Copyright © 2009 BigFix, Inc.

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	*PlainGlobal*	Creates an object containing properties of the network. Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
adapter of <network>	*Plain*	<network adapter>	Returns the one or more network adapter objects of the network. Win:2.0, Mac:7.1, WM:7.2
dns server of <network>	*Plain*	<network address list>	Returns a list of DNS servers used by the local computer. Win:4.1, WM:7.2
interface <integer> of <network>	*Numbered*	<network interface>	Returns the particular interface of the network. Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1, WM:7.2
interface of <network>	*Plain*	<network interface>	Returns all the interfaces of the network. Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1, WM:7.2
ip interface <integer> of <network>	*Numbered*	<network ip interface>	Returns the particular ip interface of the network. Win:1.2, Lin:4.1, Sol:4.1, HPUX:4.1, AIX:4.1, Mac:5.1, WM:7.2
ip interface of <network>	*Plain*	<network ip interface>	Returns 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, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
winsock2 supported of <network>	*Plain*	<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, WM:7.2

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>	*Numbered*	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, WM:7.2
interface of <network>	*Plain*	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, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
family of <network interface>	*Plain*	<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, WM:7.2

Examples

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

Copyright © 2009 BigFix, Inc.

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>	*Numbered*	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, WM:7.2
ip interface of <network>	*Plain*	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, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
address of <network ip interface>	*Plain*	<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, WM:7.2
broadcast address of <network ip interface>	*Plain*	<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, WM:7.2
broadcast support of <network ip interface>	*Plain*	<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, WM:7.2
cidr string of <network ip interface>	*Plain*	<string>	Returns the Classless Inter-Domain Routing value for the specified network ip interface as a string value. Win:7.1, Lin:7.1, Sol:7.1, HPUX:7.1, AIX:7.1, Mac:7.1, WM:7.2
loopback of <network ip interface>	*Plain*	<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, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
multicast support of <network ip interface>	*Plain*	<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, WM:7.2
point to point of <network ip interface>	*Plain*	<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, WM:7.2
subnet address of <network ip interface>	*Plain*	<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, WM:7.2
subnet mask of <network ip interface>	*Plain*	<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, WM:7.2

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.

Copyright © 2009 BigFix, Inc.

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>	*Plain*	Returns one or more adapters of the network. Win:2.0, Mac:7.1, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
address list of <network adapter>	*Plain*	<network address list>	Returns the address list of the network adapter. Win:2.0, WM:7.2
address of <network adapter>	*Plain*	<ipv4 address>	Returns the ip address of the network adapter (returns the first address if it is a list). Win:2.0, Mac:7.1, WM:7.2
cidr string of <network adapter>	*Plain*	<string>	Returns the Classless Inter-Domain Routing value for the specified network adapter as a string value. Win:7.1, Mac:7.1, WM:7.2
description of <network adapter>	*Plain*	<string>	Returns the description of the network adapter. Win:2.0, WM:7.2
dhcp enabled of <network adapter>	*Plain*	<boolean>	Returns TRUE if dhcp is enabled on the network adapter. Win:2.0, WM:7.2
dhcp server of <network adapter>	*Plain*	<ipv4 address>	Returns the ip address of the dhcp server of the network adapter. Win:2.0, WM:7.2
dns server of <network adapter>	*Plain*	<network address list>	Returns a list of DNS servers used by the specified adapter. Win:4.1, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
dns suffix of <network adapter>	*Plain*	<string>	Returns the Domain Name System (DNS) suffix associated with the specified adapter. Win:7.0, WM:7.2
friendly name of <network adapter>	*Plain*	<string>	Returns a user-friendly name for the adapter, for example "Local Area Connection 1". Win:7.0, WM:7.2
gateway list of <network adapter>	*Plain*	<network address list>	Returns the gateway network address list of the network adapter. Win:2.0, WM:7.2
gateway of <network adapter>	*Plain*	<ipv4 address>	Returns the ip address of the gateway of the network adapter. Win:2.0, WM:7.2
ipv6 address of <network adapter>	*Plain*	<ipv6 address>	Returns the local IPv6 address of the adapter. Only for XP/Server 2003 and later. Win:7.0, WM:7.2
ipv6 dns server of <network adapter>	*Plain*	<ipv6 address>	Returns the DNS server IPv6 address of the adapter. Only for XP/Server 2003 and later. Win:7.0, WM:7.2
lease expires of <network adapter>	*Plain*	<time>	Returns the time that the dhcp lease will expire of the network adapter. Win:2.0, WM:7.2
lease obtained of <network adapter>	*Plain*	<time>	Returns the time that the dhcp lease was obtained of the network adapter. Win:2.0, WM:7.2
link speed of <network adapter>	*Plain*	<integer>	This is a property of a network adapter. It returns the maximum speed of the NIC card in bits per second. Win:6.0, WM:7.2
mac address of <network adapter>	*Plain*	<string>	Returns the mac address of the network adapter. Win:2.0, Mac:7.1, WM:7.2
maximum transmission unit of <network adapter>	*Plain*	<integer>	The maximum transmission unit (MTU) size, in bytes, of the specified adapter. Win:7.0, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
name of <network adapter>	*Plain*	<string>	Returns the name of the network adapter. Win:2.0, Mac:7.1, WM:7.2
primary wins server of <network adapter>	*Plain*	<ipv4 address>	Returns the ip address of the primary wins server of the network adapter. Win:2.0, WM:7.2
secondary wins server of <network adapter>	*Plain*	<ipv4 address>	Returns the ip address of the secondary wins server of the network adapter. Win:2.0, WM:7.2
status of <network adapter>	*Plain*	<integer>	The operational status for the interface as defined in RFC 2863. It can be one of the values from the IF_OPER_STATUS enumeration type defined in the Iftypes.h header file. On Windows Vista and later, the header files were reorganized and this enumeration is defined in the Ifdef.h header file. Win:7.0, WM:7.2
subnet address of <network adapter>	*Plain*	<ipv4 address>	Returns the subnet address of the network adapter. Win:2.0, Mac:7.1, WM:7.2
subnet mask of <network adapter>	*Plain*	<ipv4 address>	Returns the subnet mask of the network adapter. Win:2.0, Mac:7.1, WM:7.2
type of <network adapter>	*Plain*	<integer>	Returns the interface type of the specified adapter as defined by the Internet Assigned Names Authority (IANA). Possible values for the interface type are listed in the Ipifcons.h header file. Win:7.0, WM:7.2
wakeonlan enabled of <network adapter>	*Plain*	<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, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
wins enabled of <network adapter>	*Plain*	<boolean>	Returns TRUE if WINS is enabled on the network adapter. Win:2.0, WM:7.2

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>	*Plain*	Returns the address list of the network adapter. Win:2.0, WM:7.2
dns server of <network adapter>	*Plain*	Returns a list of DNS servers used by the specified adapter. Win:4.1, WM:7.2
dns server of <network>	*Plain*	Returns a list of DNS servers used by the local computer. Win:4.1, WM:7.2
gateway list of <network adapter>	*Plain*	Returns the gateway network address list of the network adapter. Win:2.0, WM:7.2

Properties

Key Phrase	*Form*	Return Type	Description
address of <network address list>	*Plain*	<ipv4 address>	Returns the address of the address list. Win:2.0, WM:7.2
cidr string of <network address list>	*Plain*	<string>	Returns the Classless Inter-Domain Routing value for the specified network address list as a string value. Win:7.1, WM:7.2

Copyright © 2009 BigFix, Inc.

Key Phrase	*Form*	Return Type	Description
subnet address of <network address list>	*Plain*	<ipv4 address>	Returns the subnet address of the network address list. Win:2.0, WM:7.2
subnet mask of <network address list>	*Plain*	<ipv4 address>	Returns the subnet mask of the network address list. Win:2.0, WM:7.2

Ipv4 Address

Operators

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.

Copyright © 2009 BigFix, Inc.

Appendix

Folders on Windows Mobile Devices

On Windows machines, including mobile devices, numeric identifiers can be used to locate a wide range of 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 "ICMPath"
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.

Copyright © 2009 BigFix, Inc.

Processors

On Windows machines, including mobile devices, the Vendor Name, Family, Type, Model, Extended Family, Extended Model and stepping are calculated using the CPUID instruction. The results depend upon the processor and the vendor of the processor. The Inspectors return values based upon the Intel specification for the CPUID 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 CPUID 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 CPUID 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

Copyright © 2009 BigFix, Inc.

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		

Copyright © 2009 BigFix, Inc.

Resources

Knowledge Base

For more detailed information on how to use the DSS SAM application, check the DSS SAM documentation page at www.bigfix.com. For questions or troubleshooting issues, search the list of available articles in the BigFix Knowledge Base at http://support.bigfix.com.

Community Support

Search the BigFix User Forums for discussion threads and community-based support on a variety of topics.

Contact Us

Get technical assistance from BigFix's support team from anywhere in the world through a variety of methods:

By Phone	**United States** (866) 752-6208	**International** (661) 367-2202	
By Email	**US** **enterprisesupport@bigfix.com**	**APAC and ANZ** **bigfixasia@bigfix.com**	**Europe** **europe@bigfix.com**
By Mail	**BigFix Headquarters** 1480 64th Street Suite 200 Emeryville, California 94608	**BigFix APAC and ANZ** 8 Temasek Boulevard, #42-01 Suntec Tower 3 Singapore 038988	**BigFix Europe** Building 3, Chiswick Park 566 Chiswick High Road Chiswick, London W4 5YA UK
By Web	**www.bigfix.com**		

Copyright © 2009 BigFix, Inc.

Index

Copyright © 2009 BigFix, Inc.

C

D

E

Copyright © 2009 BigFix, Inc.

F

Copyright © 2009 BigFix, Inc.

G

H

I

K

L

Copyright © 2009 BigFix, Inc.

M

N

Copyright © 2009 BigFix, Inc.

O

P

Copyright © 2009 BigFix, Inc.

R

S

Copyright © 2009 BigFix, Inc.

T

Copyright © 2009 BigFix, Inc.

U

V

W

X

Y

Copyright © 2009 BigFix, Inc.